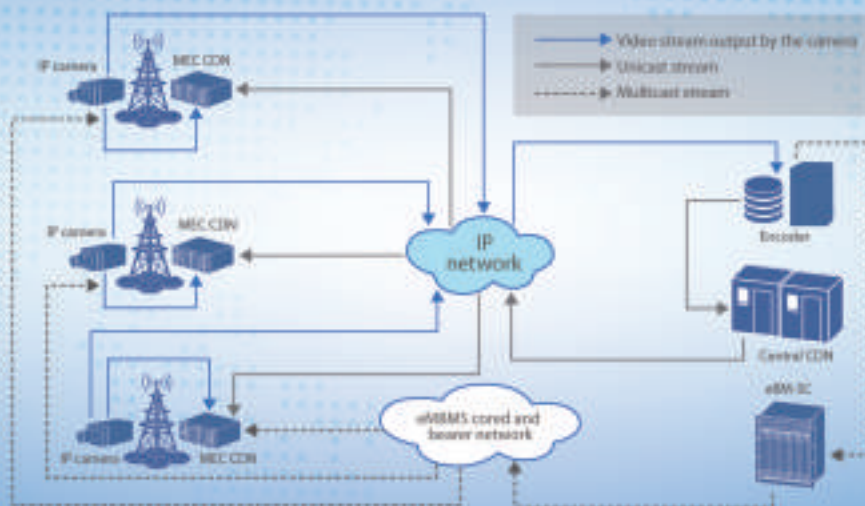


ZTE COMMUNICATIONS

tech.zte.com.cn

April 2017, Vol. 15 No. 2

SPECIAL TOPIC: Edge Computing



ZTE Communications Editorial Board

Chairman

ZHAO Houlin: International Telecommunication Union (Switzerland)

Vice Chairmen

ZHAO Xianming: ZTE Corporation (China) **XU Chengzhong:** Wayne State University (USA)

Members (in Alphabetical Order):

CAO Jiannong	Hong Kong Polytechnic University (Hong Kong, China)
CHEN Chang Wen	University at Buffalo, The State University of New York (USA)
CHEN Jie	ZTE Corporation (China)
CHEN Shigang	University of Florida (USA)
CHEN Yan	Northwestern University (USA)
Connie Chang-Hasnain	University of California, Berkeley (USA)
CUI Shuguang	University of California, Davis (USA)
DONG Yingfei	University of Hawaii (USA)
GAO Wen	Peking University (China)
HWANG Jenq-Neng	University of Washington (USA)
LI Guifang	University of Central Florida (USA)
LUO Fa-Long	Element CXI (USA)
MA Jianhua	Hosei University (Japan)
PAN Yi	Georgia State University (USA)
REN Fuji	The University of Tokushima (Japan)
SONG Wenzhan	University of Georgia (USA)
SUN Huifang	Mitsubishi Electric Research Laboratories (USA)
SUN Zhili	University of Surrey (UK)
Victor C. M. Leung	The University of British Columbia (Canada)
WANG Xiaodong	Columbia University (USA)
WANG Zhengdao	Iowa State University (USA)
WU Keli	The Chinese University of Hong Kong (Hong Kong, China)
XU Cheng-Zhong	Wayne State University (USA)
YANG Kun	University of Essex (UK)
YUAN Jinhong	University of New South Wales (Australia)
ZENG Wenjun	Microsoft Research Asia (USA)
ZHANG Chengqi	University of Technology Sydney (Australia)
ZHANG Honggang	Zhejiang University (China)
ZHANG Yueping	Nanyang Technological University (Singapore)
ZHAO Houlin	International Telecommunication Union (Switzerland)
ZHAO Xianming	ZTE Corporation (China)
ZHOU Wanlei	Deakin University (Australia)
ZHUANG Weihua	University of Waterloo (Canada)

► CONTENTS



Submission of a manuscript implies that the submitted work has not been published before (except as part of a thesis or lecture note or report or in the form of an abstract); that it is not under consideration for publication elsewhere; that its publication has been approved by all co-authors as well as by the authorities at the institute where the work has been carried out; that, if and when the manuscript is accepted for publication, the authors hand over the transferable copyrights of the accepted manuscript to *ZTE Communications*; and that the manuscript or parts thereof will not be published elsewhere in any language without the consent of the copyright holder. Copyrights include, without spatial or timely limitation, the mechanical, electronic and visual reproduction and distribution; electronic storage and retrieval; and all other forms of electronic publication or any other types of publication including all subsidiary rights.

Responsibility for content rests on authors of signed articles and not on the editorial board of *ZTE Communications* or its sponsors.

All rights reserved.

Special Topic: Edge Computing

Guest Editorial

01

CAO Jiannong and RONG Chunming

Adaptive Service Provisioning for Mobile Edge Cloud

02

HUANG Huawei and GUO Song

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

11

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie

An MEC and NFV Integrated Network Architecture

19

LI Bing, ZHANG Yunyong, and XU Lei

Key Technologies and Application of Edge Computing

26

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

35

Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritas, Thanasis Loukopoulos, and XU Cheng-Zhong

▶ CONTENTS

ZTE COMMUNICATIONS

Vol. 15 No. 2 (Issue 56)

Quarterly

First English Issue Published in 2003

Supervised by:

Anhui Science and Technology Department

Sponsored by:

Anhui Science and Technology Information
Research Institute and ZTE Corporation

Staff Members:

Editor-in-Chief: CHEN Jie

Executive Associate

Editor-in-Chief: HUANG Xinming

Editor-in-Charge: ZHU Li

Editors: XU Ye, LU Dan, ZHAO Lu

Producer: YU Gang

Circulation Executive: WANG Pingping

Assistant: WANG Kun

Editorial Correspondence:

Add: 12F Kaixuan Building,

329 Jinzhai Road,

Hefei 230061, China

Tel: +86-551-65533356

Fax: +86-551-65850139

Email: magazine@zte.com.cn

Published and Circulated

(Home and Abroad) by:

Editorial Office of

ZTE Communications

Printed by:

Hefei Tiancai Color Printing Company

Publication Date:

April 25, 2017

Publication Licenses:

ISSN 1673-5188

CN 34-1294/ TN

Annual Subscription:

RMB 80

Review

A Survey on Cloud Security

42

WU Chunming, LIU Qianjun, LI Yuwei, CHENG Qiumei, and ZHOU Haifeng

Research Paper

Network Coding-Based Interference Management Scheme in D2D Communications

48

WANG Shuang and HOU Ronghui

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

55

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

Roundup

Call for Papers: Special Issue on Motion and Emotion Sensing Driven by Big Data

10

Introduction to *ZTE Communications*

47

Call for Papers: Special Issue on Wireless Data and Energy Integrated Communication Networks

I

Edge Computing

► CAO Jiannong



CAO Jiannong is currently a chair professor and head of the Department of Computing at Hong Kong Polytechnic University, Hung Hom, Hong Kong, China. His research interests include parallel and distributed computing, wireless networks and mobile computing, big data and cloud computing, pervasive computing, and fault tolerant computing. He has co-authored three books, co-edited nine books, and published over 300 papers in major international journals and conference proceedings. He is a fellow of IEEE, a senior member of China Computer Federation, and a member of ACM. He was the Chair of the Technical Committee on Distributed Computing of IEEE Computer Society from 2012–2014. Dr. CAO has served as an associate editor and a member of the editorial boards of many international journals, including *ACM Transactions on Sensor Networks*, *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Networks*, *Pervasive and Mobile Computing Journal*, and *Peer-to-Peer Networking and Applications*. He has also served as a chair and member of organizing/program committees for many international conferences, including PERCOM, INFOCOM, ICDCS, IPDPS, ICPP, RTSS, DSN, ICNP, SRDS, MASS, PRDC, ICC, GLOBECOM, and WCNC. Dr. CAO received the B.Sc. degree in computer science from Nanjing University, Nanjing, China, and the M.Sc. and the Ph.D. degrees in computer science from Washington State University, Pullman, WA, USA.

► RONG Chunming



RONG Chunming is the chair of IEEE Cloud Computing. He works as the head of the Center for IP-based Service Innovation (CIPSI) at the University of Stavanger, Norway and also as adjunct Senior Scientist leading Big - Data Initiative at IRIS. He is co-founder and the chair of the Board of Dataunitor.com—a new start-up in Norway. He was the vice president of CSA Norway Chapter (2016–2017). His research work focuses on data science, cloud computing, security and privacy. He is an IEEE senior member and is honoured as member of the Norwegian Academy of Technological Sciences (NTVA) since 2011. He has extensive contact network and projects in both the industry and academic. He is also the founder and steering chair of IEEE CloudCom conference and workshop series. He is the steering chair and associate editor of *IEEE Transactions on Cloud Computing (TCC)*, and co-Editors-in-Chief of the *Journal of Cloud Computing* (ISSN: 2192-113X) by Springer. Prof. RONG has extensive experience in managing large-scale R&D projects funded by both industry and funding agencies, both in Norway and EU.

Edge computing refers to the computing paradigm in which the processing power, communication capabilities and intelligence are pushed down to the edge of the networking system like gateways and devices, where the data originates. In doing so, edge computing enables an infrastructure for processing the data directly from devices with low latency, battery consumption and bandwidth cost. With opportunities for research and advanced applications such as augmented reality and wearable cognitive assistance come new challenges. This special issue reports the current research on various topics related to edge computing, addressing the challenges in the enabling technologies and practical implementations.

The first paper, “Adaptive Service Provisioning for Mobile Edge Cloud”, by HUANG and GUO studies how to efficiently provide the services from the edge cloud to a given group of mobile users. The authors consider the challenge arising from the user’s mobility, and develop an adaptive method to decide when to update the service provision solution with the objective of maximizing the profit for network operators.

In the paper “Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks”, DONG et al. propose to implement SDN in VANETs, and develop an SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol (SVAO) to enhance the data transmission efficiency with VANETs. Through comprehensive simulations, the authors demonstrate that SVAO outperforms traditional ad-hoc routing protocols in terms of packet reception rate and average packet delay.

The paper “An MEC and NFV Integrated Network Architecture” by BING et al. explores the benefits of Mobile Edge Computing (MEC) at the radio access network and extends the NFV framework, and then proposes a new MEC/NFV fusion based architecture for 5G network. The authors further discuss several application scenarios of the new architecture.

In the paper “Key Technologies and Application of Edge Computing”, TU et al. present an overview of edge computing including its definition and models, applications, benefits and values, and research issues respectively related with computation, storage and networks. The authors then introduce ZTE’s edge computing solutions to 5G communications (5G MEC) and content delivery network (CDN MEC).

The paper “Scheduling Heuristics for Live Video Transcoding on Cloud Edges” by Oikonomou et al. studies the task scheduling problem in the video delivery. In the system model, the video coding and transcoding are performed at the network edges to decrease both the workload and network traffic towards the data centers. Several heuristics are designed to decide on which tasks should be assigned to an edge mini-datacenter, and which to backend datacenter.

With these articles, we wish to inform the readers of the state-of-the-art research and technologies on various topics in edge computing, and meanwhile attract the researchers and engineers to further investigate the challenges and issues that remain to be solved in the area.

The special issue would not be possible without the help from many people. We thank all the authors and reviewers for their contributions and efforts.

Adaptive Service Provisioning for Mobile Edge Cloud

HUANG Huawei¹ and GUO Song²

(1. School of Computer and Engineering, The University of Aizu, Aizu-wakamatsu 965-0006, Japan;

2. Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR 852, China)

Abstract

A mobile edge cloud provides a platform to accommodate the offloaded traffic workload generated by mobile devices. It can significantly reduce the access delay for mobile application users. However, the high user mobility brings significant challenges to the service provisioning for mobile users, especially to delay-sensitive mobile applications. With the objective to maximize a profit, which positively associates with the overall admitted traffic served by the local edge cloud, and negatively associates with the access delay as well as virtual machine migration delay, we study a fundamental problem in this paper: how to update the service provisioning solution for a given group of mobile users. Such a profit-maximization problem is formulated as a nonlinear integer linear programming and linearized by absolute value manipulation techniques. Then, we propose a framework of heuristic algorithms to solve this Nondeterministic Polynomial (NP)-hard problem. The numerical simulation results demonstrate the efficiency of the devised algorithms. Some useful summaries are concluded via the analysis of evaluation results.

Keywords

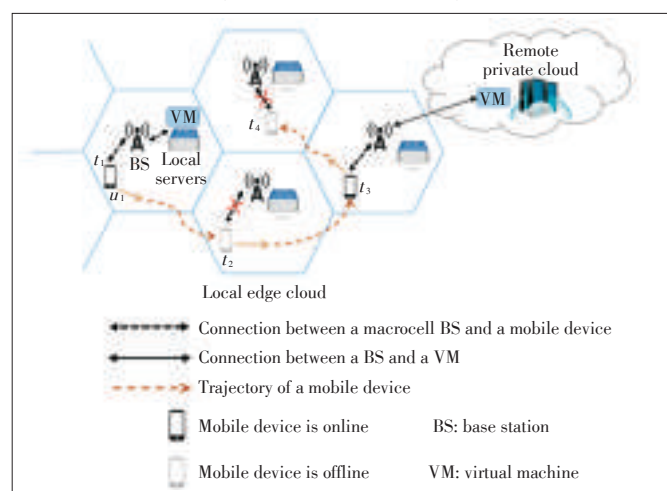
edge cloud; mobile computing; service provisioning

1 Introduction

In recent years, the fast development of mobile cloud technologies [1]–[3] has incubated large varieties of mobile online applications to facilitate our daily life, e.g., mobile online games, big data applications [4], [5]. More importantly, most of them are normally highly delay-sensitive when executed in smartphones [6]. Nowadays the mobile devices are facing numbers of challenges such as suffering the shortage of computing capacity [4] and the battery poverty [7]. Therefore, the computational-intensive workload generated from the mobile devices is suggested to offload to a remote private cloud [8]–[11] for execution.

To alleviate these challenges, recent studies [9], [12]–[19] pay particular attentions to the cluster of distributed servers in the intermediate layered edge cloud network, called cloudlet. However, in a cloudlet based network such as a metropolitan area network [18], a certain group of mobile users normally join in (or become online) and leave (or become offline) the network randomly when they are using a particular mobile application, as shown in **Fig. 1**. Therefore, the disruption of connection between the mobile device and the server under a mobile application frequently occurs at different locations and differ-

ent time frames. This brings a frequent churn to the service provisioning in cloudlet based network. Furthermore, in a real world, the access delay between each mobile device and the base station often dynamically changes in different locations even in a same cell (macrocell or smallcell).



▲ **Figure 1.** An example of service provisioning for mobile users under a cloudlet based network. The workload generated from a mobile device can be offloaded to a VM, which resides in the local edge cloud or in a remote private cloud. Meanwhile, this figure also demonstrates the dynamic characteristics of an edge network, e.g., a mobile user alternates in on-line and offline status frequently.

This work was partially supported by JSPS KAKENHI under Grant Number JP16J07062.

Via an extensive survey in the next section over the existing related studies, we find out that the challenge to deal with the dynamic characteristics of the mobile cloudlet based networks has not been well addressed so far. Therefore, we are motivated to study a fundamental problem in this paper: how to update (partially or entirely) the service provisioning solution for a certain group of online mobile - application users in a cloudlet based network, supposed that the trajectory of each mobile device can be obtained according to the daily routine of each user. We try to answer the following two questions: 1) when to update the service provisioning solution for each mobile user, and 2) how to make a trade-off between the admitted traffic rate offloaded by the local edge cloud and the induced access delay and VM-migration delay while updating the current configuration.

Our study leads to the major contributions as follows.

- We study a service provisioning problem in the cloudlet based network, and try to find a near optimal update scheme for updating the service provisioning solution for each mobile user at each time-frame if the trajectory of each mobile user is provided.
- With the objective to maximize a weighted profit for network operators, we first formulate this problem to a nonlinear programming problem, which is then transformed to a solvable integer linear programming using the absolute value manipulation techniques.
- Because of the NP-hardness of the formulated problem, we have designed a series of heuristic Algorithms to solve the problem. Extensive numerical simulation results show that the devised algorithms can yield a near optimal solution. We also conclude some useful findings via the discussion of evaluation results.

The remaining paper is organized as follows. Section 2 reviews the related work. Section 3 presents the system model and gives the problem statement. The heuristic algorithms are elaborated in Section 4. Section 5 demonstrates the numerical evaluation. Finally, Section 6 concludes this paper.

2 Related Work

2.1 Cloudlet Based Edge Computing

Recently, edge computing has attracted wide - spread research efforts [9], [12]–[20] for the mobile computing. For instance, Xia et al. [9], [12] explored a location-based offloading problem, aiming to permit requests offloaded to a cloudlet network. Then authors proposed several efficient online algorithms that can dynamically handle the requests from users. A novel hierarchical edge cloud architecture constituted with multiple cloudlets has been proposed in [17] to efficiently serve the peak loads originating from mobile users. Then, to adaptively balance the tradeoff between response delay of mobile applications and energy efficiency, Tong et al. [20] pro-

posed both offline and online algorithms to schedule the transmission in mobile cloud computing.

In wireless networks, the cloudlet placement problem also has been studied in [13], [14], [16], [18]. For example, in a wireless metropolitan area network (WMAN), in order to solve the problem of cloudlet placement, Jia et al. [14] proposed a placement scheme for a number of limited cloudlets. This approach is proved to greatly improve the mobile cloud performance. Similarly, Xu et al. [13], [16] also focused on the cloudlet placement problem, in which capacitated cloudlets need to find the best deployment locations within a given set of candidate locations. The objective is to minimize the average access delay between these activated cloudlets and mobile devices. To this end, some approximate algorithms have been devised with approximation ratios proved if all the cloudlet servers own the identical computing capability.

2.2 Task Offloading Using Edge Cloud

Wang et al. [21] studied a cost reduction problem in mobile edge clouds by deciding the assignment of mobile offloaded tasks. The authors formulated such a problem as a mixed integer program at first. Then, by introducing admission control, the problem is simplified and solved by the proposed efficient two-phase scheduling algorithm. To solve the decision making problem of computation offloading among multiple mobile users, Chen et al. [22] first formulated the problem as a multi-user computation offloading game, and proved that the game always assures a Nash equilibrium. Then, a game theoretic distributed algorithm is proposed to offload computation intensive tasks over the mobile edge cloud.

2.3 Comparison

Different from all efforts made by existing work mentioned above, this paper particularly studies the service provisioning update problem while considering the online and offline status of mobile users during their trajectories, as well as the highly dynamic characteristics of edge cloud networks. We find that this problem has not been well studied yet. To fill this gap, in this paper, we strive to design highly effective update schemes of service provisioning for edge cloud network operators.

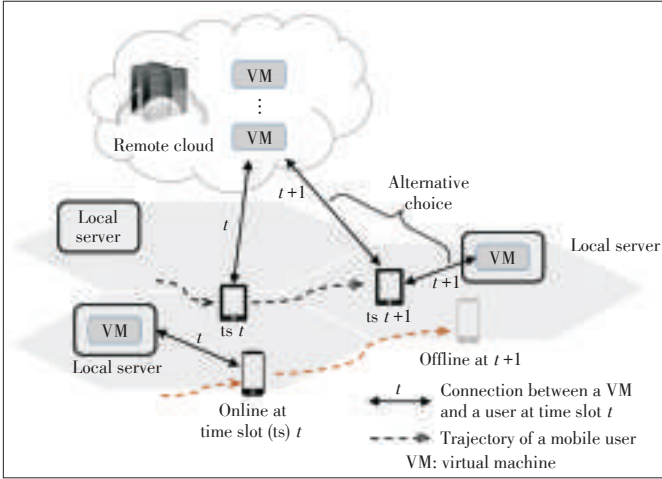
3 Network Model and Problem Statement

3.1 System Model

The network that we focus on includes a cloudlet - based edge cloud and a remote private cloud. The former network consists of a set S of local edge servers. Without loss of generality, as shown in **Fig. 2**, we assume that a powerful edge server locates at each macrocell. Therefore, a mobile user connecting with a macrocell base station is equivalent to connecting with the corresponding local edge server. In such a cloudlet based network, a set U of mobile application users traverse at differ-

Adaptive Service Provisioning for Mobile Edge Cloud

HUANG Huawei and GUO Song



▲ Figure 2. System model.

ent places in different time slots. Meanwhile, each of them becomes online and offline randomly while using the application on their mobile devices such as smartphone, tablet, etc. Suppose that the given trajectory of each mobile user is traced with the ID of its associated macrocell and online/offline status at each time slot. As a result, a timeslot labeled trajectory of a mobile user is constituted of a consecutive list of macrocell IDs. For example, a mobile user's trajectory looks like $\langle t_1, cell_a \rangle, \langle t_2, cell_b \rangle, \dots, \langle t_n, 0 \rangle, \langle t_{n+1}, cell_p \rangle, \langle t_{n+2}, cell_q \rangle, \dots$, where $\langle t_n, 0 \rangle$ particularly represents that this user is offline at time-slot t_n . When the granularity of trace is quite fine, a same macrocell ID may continually appears many times if the mobile user keeps online in the macrocell area.

With the provided trajectories of all mobile users, the network operator needs to make a decision on where to deploy the required VM for each user at each time slot only when the user online. There are generally three categories of optimization models [15] when planning a service provisioning solution in the cloudlet based networks: 1) static planning, in which both the user mobility and VM mobility are not taken into account; 2) planning with non-real-time VM migrations, in which both user mobility and M-migrations are considered; 3) planning with delay-sensitive live VM migrations, in which the difference from the previous category is that the live VM-migrations are taken into account. In this paper, the mobile applications are assumed as highly delay sensitive ones. Therefore, we adopt the optimization scenario under the third category, i.e., considering the live VM-migrations. However, according to practice, we only concern the live VM migrations between the remote cloud and the local cloudlet network, and ignore the delay of intra-cloudlet VM migrations. Table 1 shows the symbols and variables used in this paper.

3.2 Problem Statement and Formulation

We first define a binary variable x_u^t to denote the location to deploy the VM for an online mobile user $u \in U$ at the time-slot

▼ Table 1. Symbols and variables

Notation	Description
U	the set of mobile users in network
S	the set of servers in the local cloudlet based network
T	the set of candidate time slots when to update the provisioning solution for each online mobile user
D_u	the demanded traffic rate of user $u \in U$
C_s	the traffic processing capacity of server $s \in S$
$F(u)$	a set of time-slots, in each of which user u becomes online from offline status, according to its given trajectory
R_u^t	the access delay from user u to the remote private cloud at time slot t
C_u^t	the access delay from user u to the local edge server at time slot t
Δ^t	total access delay of all mobile users at time-slot t
ζ	the normalized VM-migration delay between the private cloud and a local edge server
Γ_u^t	total VM-migration delay of all mobile users at time-slot t
x_u^t	binary variable indicating the location where to deploy a VM for an online user $u \in U$ at time-slot $t \in T$
z_u^t	binary variable denoting whether to migrate a VM between the remote private cloud and the local cloudlet network for an online user $u \in U$ at time-slot $t \in T$

$t \in T$ during its trajectory:

$$x_u^t = \begin{cases} 1, & \text{if a VM is deployed for an online user } u \\ & \text{in a local edge server at the time slot } t; \\ 0, & \text{if a VM is deployed for an online user } u \\ & \text{in the remote cloud at the time slot } t. \end{cases}$$

It can be seen that, different VM deployments for an online user indicate different access delays and VM-migration delays. To represent such two terms of delays, we then define an event named inter-cloud VM-migration, in which the VM serving an online user $u \in U$ is migrated between the remote private cloud and the local edge cloud. Then, another binary variable z_u^t is defined to denote whether the inter-cloud VM-migration event occurs at the time-slot $t \in T$:

$$z_u^t = \begin{cases} 1, & \text{if an inter-cloud VM-migration event occurs} \\ & \text{for an online mobile user } u \text{ at the time slot } t; \\ 0, & \text{otherwise.} \end{cases}$$

By analyzing the given trajectory of each mobile user $u \in U$, we find that in some time slots, u becomes online from the offline status. Such a set of the online-activating time slots is denoted by $F(u)$. Naturally, we consider there is no inter-cloud VM-migration event occurring in each time slot $t \in F(u)$.

The objective is to maximize a weighted profit, which positively associates with the overall admitted traffic rate that is served by the local cloudlet network and negatively associates with the total access delay and the migration delay. In particular, letting ϕ_u^t denote the access delay of user $u \in U$ at the time-slot $t \in T$, we can calculate it as:

$$\phi_u^t = x_u^t \cdot C_u^t + (1 - x_u^t) \cdot R_u^t, \quad \forall t \in T, u \in U, \quad (1)$$

where C_u^t and R_u^t represents the access delay from user u to the local edge server and to the remote private cloud, respectively.

Then, we compute the access delay, which is denoted by Δ^t , at the time slot t in the following manner:

$$\Delta^t = \sum_{u \in U} \phi_u^t, \quad \forall t \in T. \quad (2)$$

On the other hand, we let Γ^t indicate the total VM-migration delay of all mobile users at the time slot t , and it can be calculated as:

$$\Gamma^t = \sum_{i \in T} z_u^t \cdot \zeta, \quad \forall t \in T, \quad (3)$$

where ζ denotes the normalized VM-migration delay between the private cloud and a local edge server.

Then, a profit-maximization is formulated as the following nonlinear programming:

$$\max P = \sum_{t \in T} \sum_{u \in U} D_u x_u^t - \sum_{t \in T} (w_1 \Delta^t + w_2 \Gamma^t), \quad (4a)$$

$$s.t. \sum_{u \in U} x_u^t \cdot D_u \cdot 1 \Big|_{(s=L(u,t))} \leq C_s, \quad \forall t \in T, s \in S, \quad (4b)$$

$$z_u^t = |x_u^t - x_u^{t-1}|, \quad \forall u \in U, \forall t, t-1 \in T \setminus F(u), \quad (4c)$$

$$z_u^t = 0, \quad \forall t \in F(u), \forall u \in U, \quad (4d)$$

$$x_u^t, z_u^t \in \{0, 1\}, \quad u \in U, \forall t \in T. \quad (4e)$$

In the objective function (4a), the first term $\sum_{t \in T} \sum_{u \in U} D_u x_u^t$ calculates the total admitted traffic rate that is served by the local cloudlet network, and w_1 and w_2 in the second term indicate the weight coefficients of the overall access delay and migration delay, respectively. Constraint (4b) expresses that the capacity of each server should not be expired. Note that $1 \Big|_{(s=L(u,t))}$ is a binary indicator, which returns 1 if and only if the given condition is satisfied, and $L(u, t)$ is a location function that returns the cell where user u locates. Equation (4c) describes the relationship between variables z_u^t and x_u^t . As shown in this constraint, in any two successive time slots that user u is active in both, the case under $|x_u^t - x_u^{t-1}| = 0$ indicates that both x_u^t and x_u^{t-1} have the same binary value, meaning that there is no inter-cloud VM-migration event occurring at the time slot t for user u . On the other hand, once the inter-cloud VM-migration event occurs at the time slot t , we have the situation $|x_u^t - x_u^{t-1}| = 1$, which implies x_u^t and x_u^{t-1} must take different binary values, enforcing $z_u^t = 1$. Furthermore, (4d) imposes the aforementioned special rule for variable z_u^t when user u is in each time-slot of set $F(u)$.

It is worth noting that the objective function of (4) contains z_u^t , which is decided by the constraints (4c) and (4d). However, (4c) involves the absolute value functions, making (4) become nonlinear and not able to be solved using linear programming methods. Therefore, we particularly transform (4c) to two linear constraints through the following manipulation of the absolute value expression:

$$|x_u^t - x_u^{t-1}| = 0. \quad (5)$$

Finally, the nonlinear profit-maximization (4) can be reformulated as the following linear programming:

$$\begin{aligned} & \max P \\ & s.t. (4b), (5) \text{ and } (4d), \\ & x_u^t, z_u^t \in \{0, 1\}, u \in U, \forall t \in T. \end{aligned} \quad (6)$$

4 Heuristic Algorithms

Conventionally, the service provisioning problem under the constraints of resource capacity is known as NP-hard [23]–[26]. To solve the aforementioned profit-maximization problem, in this section, we present two types of fast heuristic algorithms and their variants, aiming to yield the service provisioning solutions in each time frame for each mobile user. The major contribution of this section is the proposal of the framework of heuristic algorithms, i.e., **Algorithm 1**, using which many variants of heuristic algorithms can be devised.

4.1 The Framework of Heuristic Algorithms

We first present a framework of the heuristic algorithms in Algorithm 1, based on which we are going to devise several heuristic algorithms in the third subsection.

In line 1, the empty solution x_u^t, z_u^t is generated at first. Then, it is initialized in line 3 according to a feasibility specification, which is going to be presented afterwards. Line 4 is to find the set of mobile users who locate at each macrocell where the local server $s \in S$ is deployed. Then, in line 5, algorithms sort all the mobile users decreasingly/increasingly by their demanded rates, and decide the priority to use the local edge server. After that, a priority set \hat{U}_s^t is obtained in line 6 to denote the priority of users at each time slot $t \in T$. Next, the VM deployment for each server at each time slot can be decided as follows. Lines 9–15 show the operation under the case that a local server s is still capable to serve the traffic demanded by user u , while lines 16–22 demonstrate the opposite situation. Finally, algorithms deploy traffic demands in each local cloudlet-server, until the capacity of the server expires, and then deploy the remaining users to the remote cloud.

4.2 Structure and Feasibility Specification of a Solution

As mentioned, we have to specify a special feasibility specification to judge the feasibility of any element in a solution. Such a feasibility specification is elaborated with the explana-

Algorithm 1: Framework of Heuristic Algorithms

Input : U, T, S and trajectory traces
Output: $x_u^t, z_u^t \in \{0, 1\}, u \in U, \forall t \in T$

```

1 for  $t \in T, u \in U$  do
2    $x_u^t, z_u^t \leftarrow \emptyset$ 
3 Initialize  $x_u^t, z_u^t$  according to the given trajectory trace
4 Find the set of mobile users located at each macrocell
  where  $\forall s \in S$  is deployed
5 Check the priority to use the local edge server of each user;
  sort them decreasingly/increasingly by their demanded rates
6 Obtain a sequential set  $\hat{U}_s^t$  of mobile users by their
  priorities for each server  $s \in S$  at each time slot  $t \in T$ 
7 /*Decide the VM deployment for each mobile user at each
  time slot:*/
8 for  $t \in T, s \in S, u' \in \hat{U}_s^t$  do
9   if  $s$  is feasible to serve the traffic demanded by  $u'$ 
  then
10    /*Deploy a VM locally at  $s$  for  $u'$  */
11     $x_{u'}^t \leftarrow 1$ 
12    if  $t \geq 1$  and  $1 = x_{u'}^{t-1}$  then
13       $z_{u'}^t \leftarrow 0$ 
14    else if  $t \geq 1$  and  $0 = x_{u'}^{t-1}$  then
15       $z_{u'}^t \leftarrow 1$ 
16  else
17    /*Deploy a VM remotely for  $u'$  */
18     $x_{u'}^t \leftarrow 0$ 
19    if  $t \geq 1$  and  $1 = x_{u'}^{t-1}$  then
20       $z_{u'}^t \leftarrow 1$ 
21    else if  $t \geq 1$  and  $0 = x_{u'}^{t-1}$  then
22       $z_{u'}^t \leftarrow 0$ 

```

tion of solution structure in the following.

An example of the structure in a solution is shown in **Fig. 3a**. We can see that each solution particularly includes two row of binary codes. The intention of each row is illustrated in **Fig. 3b**. The first row indicates the variable $x_u^t (\forall u \in U, \forall t \in T)$, while the second row represents the offline/online status in each time slot. Only the bits in the first row labeled with an online indicator in the second row are valid bits, which are highlighted with shadow in Fig. 3a. The bit labeled with * denotes a “do-not-care” invalid bit, which will not be included in solution x . A valid binary bit in the first row implies that a VM is deployed in the local edge server for the current time slot, if it is equal to 1. Otherwise, it indicates that the VM serving a mobile user is deployed to the remote cloud. According to the given trajectory of each mobile user, the second row of a solution can be retrieved quickly. In the next step, each valid bit in the first row can be initialized randomly. After the initial-

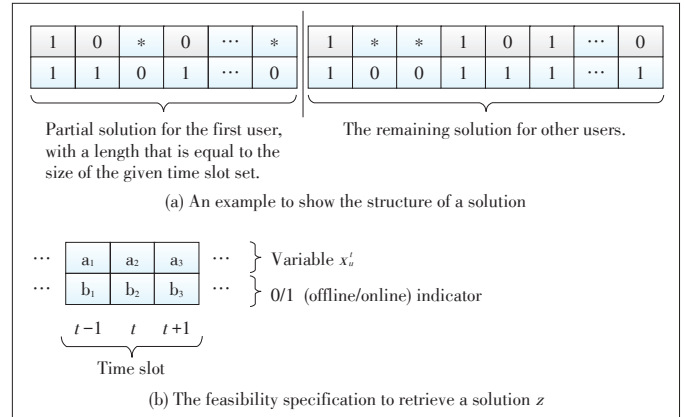


Figure 3. The structure and the feasibility specification of a solution.

ization of solutions x and z , only the valid bits in the first row are need to be decided according to the chosen algorithm.

We then explain how to retrieve the solution of inter-cloud VM-migration event, i.e., variables $z_u^t (\forall u \in U, \forall t \in T)$, when a solution x is provided. According to the definition of z_u^t and constraints (4d) and (4c), the rules are as follows: 1) to an invalid bit in the first row, we consider no inter-cloud VM-migration event occurs at this current corresponding time slot; 2) to any two adjacent valid bits in the first row, if the bit corresponding to the second time slot is labeled with 1 while the bit corresponding to the first time slot is labeled with 0, we still consider that there is no inter-cloud VM-migration event occurring at the second time slot; 3) if any two adjacent valid bits in the first row are labeled with different binary values, we consider the inter-cloud VM-migration event occurs at the second time slot. For the example shown in Fig. 3b, once $b_1 = 0$, we definitely have $z_u^{t-1} = 0$. On one hand, if $b_1 = 0$, both b_2 and b_3 are 1-labeled with 1, the cases under $a_2 = 0, a_3 = 1$ and $a_2 = 1, a_3 = 0$ both yield $z_u^t = 0$ and $z_u^{t+1} = 1$. On the other hand, when b_1, b_2 and b_3 are all equal to 1, the same cases under $a_2 = 0, a_3 = 1$ and $a_2 = 1, a_3 = 0$ will both yield $z_u^{t+1} = 1$ for sure, and the value of z_u^t depends on a_1 .

4.3 Heuristic Algorithms and Variants

Based on the algorithm framework, we now present two types of heuristic algorithms and their variants. The first one is called Online-First algorithm, the basic idea of which is to try to assign higher priority to the set of mobile users who are still in online status at the previous one time-slot. As a result, a mobile user who just becomes online at the current time slot has a lower priority than other local online mobile users. Finally, all the mobile users located at a local cell are classified into two groups by their priorities. We further get the final sequential set of users according to their demanded traffic rates. By sorting them decreasingly or increasingly by the demanded traffic rates, we finally receive the variants of such Online-First algorithm, which are named as Online-First-Decreasing and Online

-First-Increasing, respectively.

Another heuristic algorithm is called First - Fit, which is widely adopted to solve the bin-packing problem [24]. Similarly, according to the decreasingly/increasingly sorting manner towards the demanded traffic rate of each mobile user, the variants of First-Fit are labeled as First-Fit-Decreasing and First-Fit-Increasing, respectively.

5 Performance Evaluation

In this section, we conduct extensive numerical simulations to evaluate the presented 4 heuristic algorithms: First-Fit-Decreasing (FFD), First-Fit-Increasing (FFI), Online-First-Decreasing (OFD), and Online-First-Increasing (OFI).

The basic ideas of these 4 heuristic algorithms have been widely used by existing studies related to the resource allocation in cloud. Here we mainly compare the performance differences of the 4 heuristic algorithms designed under our proposed algorithm-framework. Furthermore, we are also interested in the performance gaps between such 4 algorithms and the Optimal one under different system settings. Finally, we would like to draw some useful conclusions over their performance by analyzing the simulation results, and try to suggest the service providers which heuristic algorithm is the best choice under a network configuration.

5.1 Simulation Settings

The network topology adopted in our simulations is a cloudlet based urban access network with 10 adjacent macrocells, each of which has an isolated local server that can only serve the mobile users located in the current macrocell. We randomly generate a traffic demand trace for each mobile user within [10, 100] Mb/s. The access delay to the remote cloud is fixed to 10 ms, while the local access delay of any mobile user to its local edge server is randomly generated within [1, 3] ms. Furthermore, the inter-cloud VM-migration delay is normalized to 10 ms.

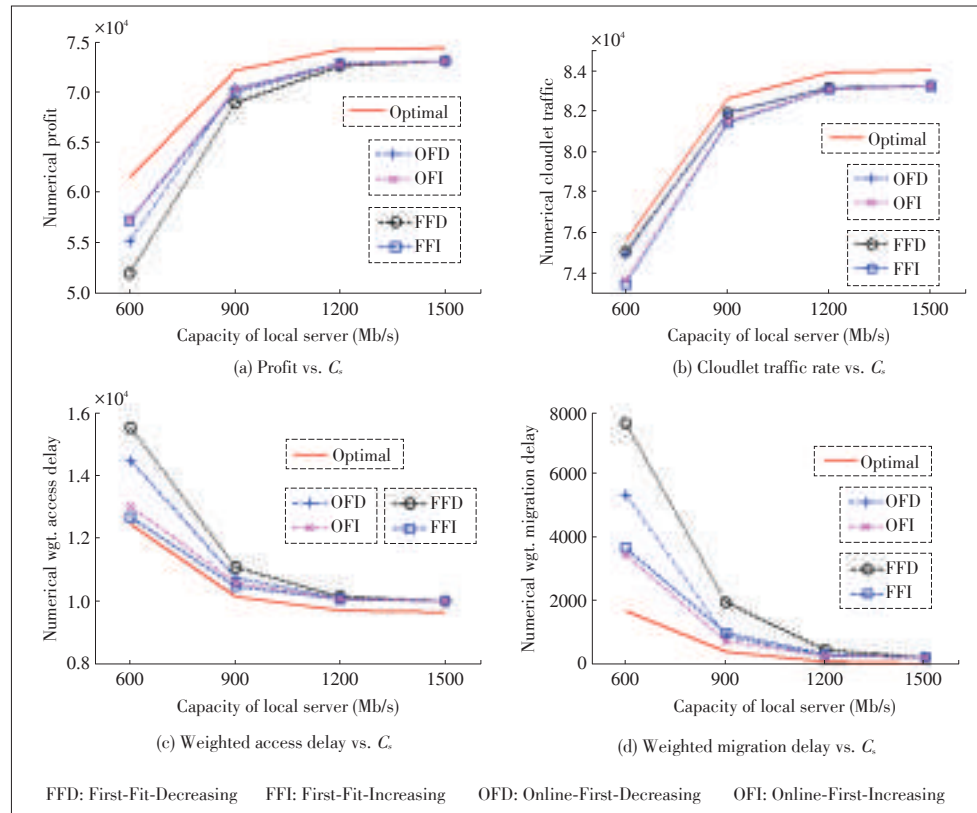
We then generate a sequential trajectory for each mobile user within 20 time slots. At each time slot, we first decide the online status of any mobile user using a predefined probability, which is fixed to 0.8 in this paper. If a user is offline in a time slot, we mark its traversed cell ID to 0. Otherwise, we find a cell location following a twofold rule: 1) when a

mobile user becomes online from an offline status, we randomly find a cell that it appears at; 2) when a mobile user keeps online from the previous one time slot, we find a cell for the current time slot within its located cell and the neighboring cells as well. On the other hand, as a benchmark to compare performance with our devised heuristic algorithms, we also solve (6) to retrieve the Optimal solution using Gurobi 6.0 [27], under each simulation setting. We compare heuristic algorithms and the optimal solution in terms of 4 metrics, i.e., total numerical profit, total traffic rate allocated to the local edge cloud, the weighted access delay and the weighted migration delay.

5.2 Effect of Traffic Processing Capacity of Edge Servers

In the first group of the simulations, we study the effect of server's traffic processing capacity by varying $C_s \in \{600, 900, 1200, 1500\}$ Mb/s, and fixing both w_1 and w_2 to 3. From **Figs.4a** and **4b**, we can observe that the profit and total numerical cloudlet traffic rate are increasing functions over the capacity of servers. When the capacity is insufficient, e.g., when $C_s = 600$ Mb/s, algorithms FFI and OFI perform better than the other two heuristics. This is because in the previous two algorithms, more mobile users who request traffic demands with small rates can be served in the local cloudlet servers resulting in smaller total access delay and migration delay.

Furthermore, in **Figs. 4c** and **4d**, we can see that the access



▲ **Figure 4.** Performance of algorithms when the serving capacity of a local server (i.e., C_s) varies in a range 600 Mb/s–1500 Mb/s.

Adaptive Service Provisioning for Mobile Edge Cloud

HUANG Huawei and GUO Song

and migration delays decrease as the traffic processing capacity grows. As expected, the algorithms considering the demands with small traffic rates to be first served, i.e., FFI and OFI, have the lower delays than FFD and OFD algorithms.

Finally, once the processing capacity of local edge servers grows sufficiently, the performance of all algorithms becomes same. This can be explained by the reason that every algorithm yields a similar solution and performs close to the optimal solution, when the processing capacity of edge servers is not the bottleneck resource any more.

5.3 Effect of w_1

Using the same traces, we evaluate the effect of the weight of access delay, by varying $w_1 \in \{1, 2, 3, 4, 5\}$ and fixing $w_2 = 1$ and $C_i = 500$ Mb/s. **Fig. 5** illustrates the same four metrics of the previous group of simulations. Because the access delay contributes negatively to the objective function, we observe the decreasing profits in Fig. 5a and the increasing numerical weighted (shorten as wgt.) access delay in Fig 5c, while enlarging the weight of access delay from 1 to 5. FFI and OFI show the larger profits than that of the other two algorithms. The reason is same with the previous simulation.

Interestingly, Figs. 5b and 5d demonstrate that improving the weight of access delay has no effect to the total cloudlet traffic and the weighted migrations delay. This is because changing w_1 will not significantly affect the task allocation to

the local edge cloud or to the remote cloud. This is a useful finding to network operators.

5.4 Effect of w_2

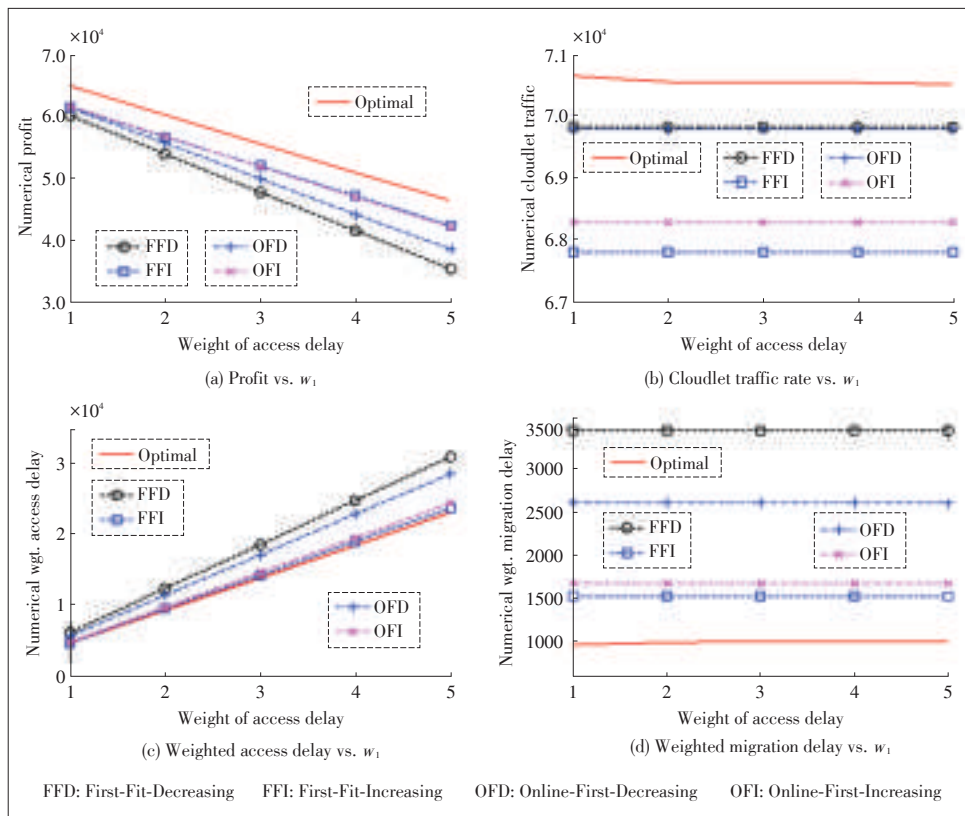
By varying $w_2 \in \{1, 2, 3, 4, 5\}$ and setting w_1 to 1, we then study the effect of the weight of migration delay in this group of simulations. **Fig. 6** presents the 4 metrics of the four heuristic algorithms and the optimal solution as well. In Figs. 6a and 6b, we have similar observations on both the total profit and the total cloudlet traffic rate, compared with the previous group of simulations. This is because w_2 plays a similar role with w_1 to the system objective.

Although w_2 in all heuristic algorithms has no effect on the weighted access delay from Fig.6c, the increasing weight of migration delay makes the weighted migration delay higher. Thus, the total profit is reduced significantly. Especially under FFD, more traffic demands with small traffic rates have to experience the inter-cloud VM-migration, than that under other algorithms. This is because when the server capacity is limited, only a small number of requests can be provisioned in the local edge cloud. The VMs serving other users with tiny-rate demands have to be migrated to the remote cloud, thus incurring higher migration delay when performing the FFD and OFD algorithms.

In a summary, via all the simulation results, we can always observe that the FFI and OFI have a similar performance and outperform the other two heuristics in terms of total profit, the weighted access delay, and the weighted migration delay.

6 Conclusions

In this paper, we study the update problem of service provisioning in the cloudlet based mobile edge network. We try to find an adaptive update scheme to decide when to update the service provisioning solution for each mobile user at each time-frame, if the trajectory of each mobile user is known. With the objective of maximizing a weighted profit for network operators, we first formulate this problem as nonlinear programming problem. Then, it is transformed to solvable integer linear programming using the absolute value manipulation technique. Next, to solve this problem, we devise a series of heuristic algorithms. Extensive numerical simulation results demonstrate that



▲ Figure 5. Performance of algorithms when the weight of access delay (i.e., w_1) varies in a range 1–5.

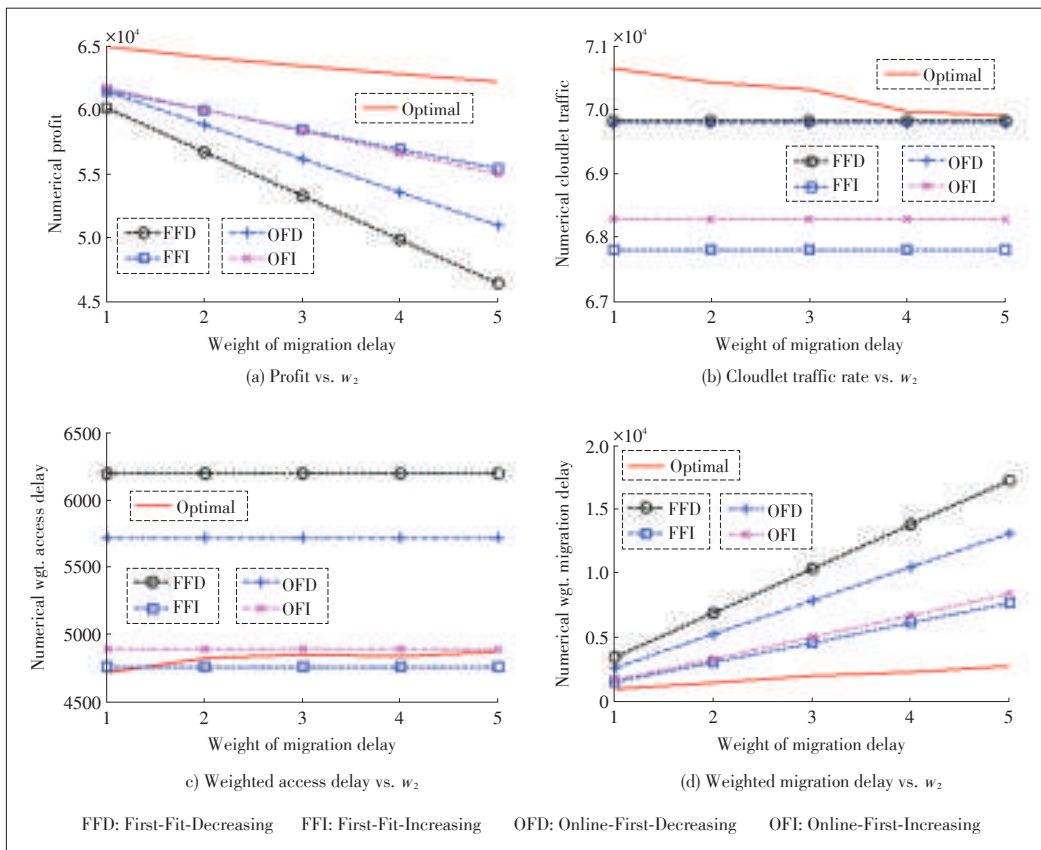


Figure 6. Performance of algorithms when the weight of migration delay (i.e., w_2) varies in a range 1–5.

the devised algorithms could yield near optimal solutions. Some useful findings have been also revealed through the evaluation of simulation results.

References

- [1] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: bringing the cloud to the mobile user," in *Third ACM Workshop on Mobile Cloud Computing and Services*, Low Wood Bay, Lake District, UK, 2012, pp. 29–36. doi: 10.1145/2307849.2307858.
- [2] D. Meilander, F. Glinka, S. Gorlatch, et al., "Using mobile cloud computing for real-time online applications," in *IEEE International Conference on Mobile Cloud Computing, Services, and Engineering*, Oxford, UK, 2014, pp. 48–56. doi: 10.1109/MobileCloud.2014.19.
- [3] N. Fernando, W. L. Seng, and W. Rahayu, "Mobile cloud computing: A survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, 2016.
- [4] A. T. Lo' ai, W. Bakheder, and H. Song, "A mobile cloud computing model using the cloudlet scheme for big data applications," in *IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies*, Washington, DC, USA, 2016, pp. 73–77. doi: 10.1109/CHASE.2016.40.
- [5] A. T. Lo' ai, R. Mehmood, E. Benkhelifa, and H. Song, "Mobile cloud computing model and big data analysis for healthcare applications," *IEEE Access*, vol. 4, pp. 6171–6180, 2016. doi: 10.1109/ACCESS.2016.2613278.
- [6] K. Ha, Z. Chen, W. Hu, et al., "Towards wearable cognitive assistance," in *International Conference on Mobile Systems*, Bretton Woods, USA, 2014, pp. 68–81. doi: 10.1145/2594368.2594383.
- [7] K. Yang, S. Ou, and H. H. Chen, "On effective offloading services for resource-constrained mobile devices running heavier mobile internet applications," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 56–63, 2008. doi: 10.1109/MCOM.2008.4427231.
- [8] D. Kovachev, T. Yu, and R. Klamka, "Adaptive computation offloading from mobile devices into the cloud," in *IEEE International Symposium on Parallel and Distributed Processing with Applications*, Madrid, Spain, 2012, pp. 784–791. doi: 10.1109/ISPA.2012.115.
- [9] Q. Xia, W. Liang, and W. Xu, "Throughput maximization for online re-request admissions in mobile cloudlets," in *IEEE 38th Conference on Local Computer Networks*, Sydney, Australia, 2013, pp. 589–596. doi: 10.1109/LCN.2013.6761295.
- [10] W. Gao, Y. Li, H. Lu, T. Wang, and C. Liu, "On exploiting dynamic execution patterns for workload offloading in mobile cloud applications," in *IEEE 22nd International Conference on Network Protocols (ICNP)*, Raleigh, USA, 2014, pp. 1–12. doi: 10.1109/ICNP.2014.22.
- [11] E. J. Haughn, "Mobile device management through an offloading network," U. S. Patent 8,626,143, Jan. 7, 2014.
- [12] Q. Xia, W. Liang, Z. Xu, and B. Zhou, "Online algorithms for location-aware task offloading in two-tiered mobile cloud environments," in *IEEE/ACM 7th International Conference on Utility and Cloud Computing*, London, UK, 2014, pp. 109–116. doi: 10.1109/UCC.2014.19.
- [13] Z. Xu, W. Liang, W. Xu, M. Jia, and S. Guo, "Capacitated cloudlet placements in wireless metropolitan area networks," in *IEEE 40th Conference on Local Computer Networks*, Clearwater Beach, USA, 2015, pp. 570–578. doi: 10.1109/LCN.2015.7366372.
- [14] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Transactions on Cloud Computing*, vol. 6, no. 25, pp. 1–14, 2015. doi: 10.1109/TCC.2015.2449834.
- [15] A. Ceselli, M. Premoli, and S. Secci, "Cloudlet network design optimization," in *IFIP Networking Conference*, Toulouse, France, 2015, pp. 1–9. doi: 10.1109/IFIPNetworking.2015.7145315.
- [16] Z. Xu, W. Liang, W. Xu, M. Jia, and S. Guo, "Efficient algorithms for capacitated cloudlet placements," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 10, pp. 2866–2880, 2016.
- [17] L. Tong, Y. Li, and W. Gao, "A hierarchical edge cloud architecture for mobile computing," in *IEEE International Conference on Computer Communications*, San Francisco, USA, 2016, pp. 1–9. doi: 10.1109/INFOCOM.2016.7524340.
- [18] M. Jia, W. Liang, Z. Xu, and M. Huang, "Cloudlet load balancing in wireless metropolitan area networks," in *IEEE International Conference on Computer Communications*, San Francisco, USA, 2016, pp. 1–9. doi: 10.1109/INFOCOM.2016.7524411.
- [19] X. Sun and N. Ansari, (2016). *Green cloudlet network: A distributed green mo-*

Adaptive Service Provisioning for Mobile Edge Cloud

HUANG Huawei and GUO Song

- mobile cloud network [Online]. Available: <https://arxiv.org/abs/1605.07512>
- [20] L. Tong and W. Gao, "Application - aware traffic scheduling for workload offloading in mobile clouds," in *IEEE International Conference on Computer Communications*, San Francisco, USA, 2016, pp. 1–9. doi: 10.1109/INFOCOM.2016.7524520.
- [21] L. Wang, L. Jiao, D. Kliazovich, and P. Bouvry, "Reconciling task assignment and scheduling in mobile edge clouds," in *IEEE 24th International Conference on Network Protocols*, Singapore, 2016, pp. 1–6.
- [22] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2016. doi: 10.1109/TNET.2015.2487344.
- [23] N. M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual network embedding with coordinated node and link mapping," in *IEEE International Conference on Computer Communications*, San Francisco, 2009, pp. 783–791. doi: 10.1109/INFOCOM.2009.5061987.
- [24] H. Huang, D. Zeng, S. Guo, and H. Yao, "Joint optimization of task mapping and routing for service provisioning in distributed datacenters," in *IEEE International Conference on Communications*, Sydney, Australia, Jun. 2014, pp. 4196–4201.
- [25] H. Huang, P. Li, S. Guo, and B. Ye, "The joint optimization of rules allocation and traffic engineering in software defined network," in *IEEE 22nd International Symposium of Quality of Service*, Hong Kong, China, 2014, pp. 141–146. doi: 10.1109/IWQoS.2014.6914313.
- [26] H. Huang, S. Guo, P. Li, B. Ye, and I. Stojmenovic, "Joint optimization of rule placement and traffic engineering for QoS provisioning in software defined network," *IEEE Transactions on Computers*, vol. 64, no. 12, pp. 3488–3499, 2015. doi: 10.1109/TC.2015.2401031.

[27] Gurobi Optimization. (2016). *Gurobi optimizer reference manual* [Online]. Available: <http://www.gurobi.com>

Manuscript received: 2017-01-15

Biographies

HUANG Huawei (davyhwang.cug@gmail.com) received his Ph.D. in computer science from the University of Aizu, Japan. His research interests mainly include network optimization and algorithm design/analysis for wired/wireless networks. He is a member of IEEE and a JSPS Research Fellow.

GUO Song (song.guo@polyu.edu.hk) received his Ph.D. in computer science from University of Ottawa, Canada. He is currently a full professor at Department of Computing, The Hong Kong Polytechnic University (PolyU), China. Prior to joining PolyU, he was a full professor with the University of Aizu, Japan. His research interests are mainly in the areas of cloud and green computing, big data, wireless networks, and cyber-physical systems. He has published over 300 conference and journal papers in these areas and received multiple best paper awards from IEEE/ACM conferences. His research has been sponsored by JSPS, JST, MIC, NSF, NSFC, and industrial companies. Dr. GUO has served as an editor of several journals, including *IEEE TPDS*, *IEEE TETC*, *IEEE TGCN*, *IEEE Communications Magazine*, and *Wireless Networks*. He has been actively participating in international conferences serving as general chairs and TPC chairs. He is a senior member of IEEE, a senior member of ACM, and an IEEE Communications Society Distinguished Lecturer.

Call for Papers

ZTE Communications Special Issue on

Motion and Emotion Sensing Driven by Big Data

Motion and emotions are two critical features of human presence and activities. Recent developments in the field of indoor motion and emotion sensing have revealed their potentials in enhancing our living experiences through applications like public safety and smart health. However, existing solutions still face several critical downsides such as the availability (specialized hardware), reliability (illumination and line - of - sight constraints) and privacy issues (being watched). To this end, this special issue seeks original articles describing development, relevant trends, challenges, and current practices in the field of applying Artificial Intelligence to address various issues of motion and emotion sensing brought by the "Big data". Position papers, technology overviews, and case studies are also welcome.

Appropriate topics include, but not limited to

- Motion and Emotion Model/Theory driven by Big Data
- Motion and Emotion Sensing Algorithms driven by Big Data
- Multi-modality Data Processing for Motion and Emotion Sensing
- Multi - modality Data Mining for Motion and Emotion Sensing
- Novel Motion and Emotion Applications/Systems Sup-

ported by Big Data

- Evaluation Metrics and Empirical Studies of Motion and Emotion Sensing Systems
- Quality-enhanced and adaptive sensing models driven by Big Data
- Inherent Relationship Modeling between Motion and Emotions driven by Big Data

First submission due: August 15, 2017

Peer review: September 30, 2017

Final submission: October 15, 2017

Publication date: December 25, 2017

Submission Guideline:

Submission should be made electronically by email in WORD format.

Guest Editors:

Prof. Fuji Ren (ren@is.tokushima-u.ac.jp), University of Tokushima, Japan

Prof. Yixin Zhong (zyx@bupt.edu.cn), Beijing University Of Posts and Telecommunications, China

Prof. Yu Gu (yugu.bruce@ieee.org), Hefei University of Technology, China

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie

(Guangdong Province Key Lab. of Big Data Analysis and Processing, School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China)

Abstract

This paper comes up with a SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol (SVAO), which separates the data forwarding layer and network control layer, as in software defined networking (SDN), to enhance data transmission efficiency within vehicle ad-hoc networks (VANETs). The roadside service unit plays the role of local controller and is in charge of selecting vehicles to forward packets within a road segment. All the vehicles state in the road. Correspondingly, a two-level design is used. The global level is distributed and adopts a ranked query scheme to collect vehicle information and determine the road segments along which a message should be forwarded. On the other hand, the local level is in charge of selecting forwarding vehicles in each road segment determined by the global level. We implement two routing algorithms of SVAO, and compare their performance in our simulation. We compare SVAO with popular ad-hoc network routing protocols, including Optimized Link State Routing (OLSR), Dynamic Source Routing (DSR), Destination Sequence Distance Vector (DSDV), and distance-based routing protocol (DB) via simulations. We consider the impact of vehicle density, speed on data transmission rate and average packet delay. The simulation results show that SVAO performs better than the others in large-scale networks or with high vehicle speeds.

Keywords

VANETs; SDN; routing protocol; ad-hoc network; Internet of Vehicle

1 Introduction

Nowadays, with the popularity of personal vehicles, transportation systems are facing many problems, including traffic congestion, environment pollution, increasing energy consumption, etc. [1]. Therefore, the research of intelligent transportation system (ITS) has become a magnet for researchers in recent years, in which vehicle ad-hoc networks (VANETs) play a key role. In VANETs, a vehicle can either communicate with another vehicle directly via Vehicle-to-Vehicle (V2V) communications, or communicate to infrastructure such as a road side unit (RSU) via Vehicle-to-Infrastructure (V2I) links [2]. As an intersection of traffic network and traditional ad-hoc network, VANETs hold some of their features, for example, it has decentralized, self-organizing and dynamic topology. VANETs also have unique features in aspect of structure, support high-mobility scenarios and implement special traffic applications.

Message/packet routing and forwarding have been always a core issue for VANETs. How to find, build and choose an appropriate route in a highly dynamic vehicle network can be a tough problem, especially with stability, packet delay, computation overhead, and bandwidth taken into account.

Quite a lot of work has been conducted on designing routing protocols and forwarding mechanisms for VANETs. Considering the information required for routing protocols, the current routing work in VANETs can be concluded as follow: topology based protocols, position based protocols, map based protocols, and road based protocols. Topology based protocols route and forward according to the topology of road segments, no matter whether there exists a global route table such as Destination Sequence Distance Vector (DSDV) [3], or a local one such as Dynamic Source Routing (DSR) [4] and Ad-hoc On-demand Distance Vector Routing (AODV) [5]. Position based protocols only care for the position of vehicles such as Greedy Perimeter Coordinator Routing (GPCR) [6], or the position of RSUs such as Intersection-based Geographical Routing Protocol (IGRP) [7]. Map based protocols attempt to take some road segment states into account. Geographic Source Routing (GSR) [8] considers about the crossroad, while Shortest-Path-Based Traffic-Light-Aware Routing (STAR) [9] concerns the impact of the

This research is partially supported by National Key Research and Development Program of China (2016YFB0200400), National Natural Science Foundation of China (No. 61379157), Program of Science and Technology of Guangdong (No. 2015B010111001), and MOE-CMCC Joint Research Fund of China ((No. MCM20160104).

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie

traffic light. Road based protocols focus on the communication among road segments such as Vehicle-Assisted Data Delivery (VADD) [10]. However, most of the researches above mainly focus on distributed routing discovery and selection process. In reality, due to the difficulty of coordinating among the nodes, the data transmission efficiency is limited.

Software defined networking (SDN) is a promising technology aiming to reinvent the Internet and improve the network transmission efficiency. It separates the data forward layer and the network control layer, realizing a more effectively centralized network resource allocation and scheduling. It hammers at making the network more intelligent and realizing reasonable network resource utilization. In a traditional network, the network control layer and data forward layer are tight coupling on the routing unit, which makes it hard for applying the latest network control technology on it. So the traditional routing unit is so-called “simple” and “dumb” [11]. It is also referred to as “Internet ossification” [12]. On the one hand, researchers have to take various hardware differences between routers into account when they want to apply a new routing protocol. On the other hand, when hardware needs to be updated, whether to support the operation of the existing routing protocol should be considered as well. That is what SDN focuses on. SDN separates the logic control layer and data forward layer, so as to separate the route unit and control unit. In detail, the routing unit is responsible for forwarding according to the flow table. The generation, maintenance, configuration of flow table is managed by the control unit. The separation of control layer and data forward layer realizes flexible forward strategy management of SDN system.

However, as SDN is originally designed for general cable wide area networks (WANs) and local area networks (LANs), the routing and forwarding mechanism need to be redesigned when we try to apply SDN into VANETs. A VANET involves wireless sensor network, wireless cellular network, mobile self-organizing network, and more. Furthermore, the node mobility, architecture, and traffic rule effect of the VANET make it more complicated than the traditional wireless network.

This paper comes up with a SDN based on-demand routing mechanism under the on-demand VANETs forwarding scenario, aiming at improving the routing and forwarding efficiency. The largest challenge of this paper is redesigning routing and forwarding mechanism in VANETs. This paper proposes a two-level structure protocol, SDN Based Vehicle Ad-Hoc On-demand Routing Protocol (SVAO), including the distributed global level consisting of local controllers (LCs) and the centralized local level consisting of vehicles. The global level is responsible for calculating and distributing the route among road segments/LCs, while the local level in charge of calculating the route for each vehicle in each section. We implement two routing algorithms of SVAO in this paper, Bellman-Ford algorithm and Floyd algorithm, to explore the difference of their performance. Finally, we compare the performance of the new proto-

col with four traditional self-organizing network routing protocols (Optimized Link State Routing (OLSR), DSR, DSDV and distance-based routing protocol (DB)). The simulation result shows that the proposed protocol performs better.

2 Related Work

The routing protocol is vital for VANETs, whose features include highly dynamic topology, wide coverage and low robustness. A suitable routing protocol can keep up a good transmission quality in VANETs. According to different types of needed information, the current routing protocols in VANETs can be divided into the following four categories: topology based protocols, position based protocols, map based protocols, and road based protocols.

2.1 Topology Based Protocols

DSRV [3] is a table-driven scheme based on Bellman-Ford algorithm. Each entry in the route table contains a sequence number, which identifies whether this entry is valid. The sequence number is generated by the destination node, and periodically updated by sending packets. Routing information is updated by sending packets among nodes periodically and triggered incremental updates packets more frequently.

DSR [4] is a source routing type on-demand routing protocol. DSR is beacons-less, which means that the update of the network topology information will not rely on periodically sending Hello packet (beacons). The main process of establishing a route described as follow: at the very beginning, the source node floods the RouteRequest packet to the whole network; once the destination node receives a RouteRequest packet, it will return a RouteReply packet along the reverse path; lastly the route tables alongside the path are updated by adding the route from the source node to the destination node.

AODV [5] floods route messages to conduct route discovery, similar to DSR. The difference between them is that AODV is a node routing scheme, that is, when a node (whether it is the destination node or not) receives a RouteRequest packet, and this node has cached the latest route towards the destination node, it will not forward this packet but return a RouteReply packet along the reverse path and update the route table alongside the path.

Due to the high-mobility and highly dynamic network topology in VANETs, the main difficulty of topology-based protocols is how to reduce the route discovery cost, time cost and resource cost. If the time cost is high, the latest route will lose efficacy frequently.

2.2 Position Based Protocols

GPCR [6] is a greedy routing protocol without link state. Different from traditional routing protocols, nodes directly take use of the position information of their neighbors to establish a route instead of finding the shortest path towards destination

node or judging the reachability of the route. The source node always greedily sends packets towards the node that is most close to the destination node, until it could not reach the destination node. Then it will begin a new round of forwarding and the greedy strategy will be adopted in a new node again, until the packet reaches a closer node towards the destination node.

IGRP [7] is designed to solve the QoS routing problems under the city traffic scenario. Since it may in fact not be feasible and economical to absolutely deploy RSUs, IGRP attempts to ensure the maximum packet received rate by changing the point-to-point routing strategy to check whether the RSUs exist.

2.3 Map Based Protocols

GSR [8] is a routing protocol based on map and vehicle position. It uses Dijkstra algorithm to compute the shortest path from the source node towards the destination node. Each cross-road acts as an anchor node, and a greedy forwarding strategy is adopted between each two anchor nodes. With a high data packet reception rate, GSR is adaptive to the situation that contains more vehicles and larger traffic density. It does not suitable for the condition with less car and poor connectivity.

STAR [9] takes the impact of the traffic light into account. It regards the traffic light as a main factor in the dynamic network topology in the practical city traffic scenario. The state of the traffic light in crossroad will be treated as an influence factor in the packet forwarding strategy. In practice, STAR has lower average delay, higher packet reception rate and higher TCP throughput capacity in unit time.

2.4 Road Based Protocols

VADD [10] adopts a store-and-forward strategy which is adaptive to the scenario with low vehicle density and poor connectivity especially. Predicting the future position of the vehicle, every packet will be marked with a packet transfer delay, which ensures the vehicle can store the packet until the vehicle reaches next road.

The distributed routing protocols in current vehicle networks aim at taking the balance among route computation overhead, link hops, link quality and link stability. Due to the dynamic nature of VANETs, a distributed protocol may cause large link state delay to discover a route. To settle the problems above, this paper proposes an SDN based on-demand forwarding routing protocol which adds local controller as a roadside unit in current VANETs and gets use of the advantages of centralized method to get an optimized routing mechanism.

Recently, there are already some works on SDN in VANETs. S.-A. Lazar and C.-E. Stefan concluded that SDN and fog working together can be the best solution to VANETs, after over-viewing both side of works [13]. K. Liu and J. K. Y. Ng were the first focusing on scheduling cooperative data dissemination in hybrid infrastructure-to-vehicle (I2V) and vehicle-to-vehicle (V2V) communication environments [14]. A. Kazmi and M. A. Khan tried to deal with problems in topology dynamics and

connectivity losses. They exploited SDN planes by partitioning VANETs to work in distributed manner, which is better than traditional VANETs architectures [15]. X. Xiao and X. Kui analyzed the traces of taxis in Shanghai to find out the regular pattern of the inter-contact time between moving vehicles and intersections, and the duration of each contact, making way for the future study of SDN VANETs [16]. A. Di Maio and M. R. Palattella focused on the security issues for enabling SDN in VANETS in real use cases (smart parking, smart grid of electric vehicles, platooning, and emergency services) [17]. A. KazmiEmail and M. A. Khan kept their eyes on single point of failure (SPOF) in SDN, coming up with an abstracted VANET model but also complying with SDN principals [18].

3 System Model and Introduced Coefficient

In this section, we firstly describe the system model and then introduce the coefficient used in proposed protocol.

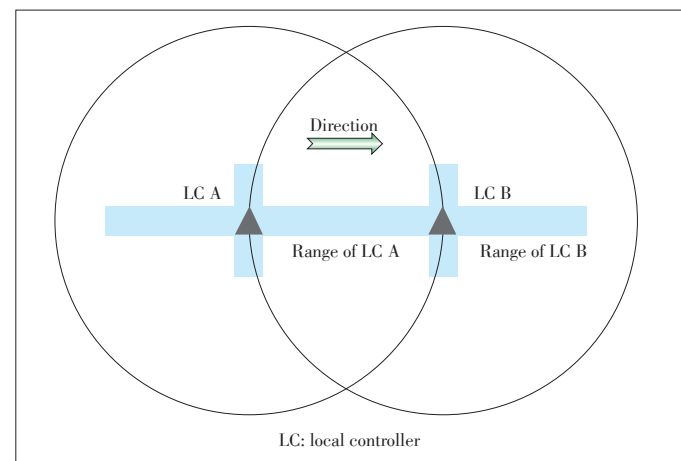
3.1 Sytem Model

Firstly, the city road is simplified into a neat network diagram. We assume that every vehicle moves towards the same direction in each road. That is, this paper only considers one-way road. A roadside control unit named LC will be placed in every crossroad (Fig. 1), which is responsible for collecting all the information of the vehicles on this road and computing route. LCs can communicate with each other to exchange information about road segment states.

3.2 Link Stability Coefficient

In VANETs, the transmission quality and efficiency are influenced by many factors, such as the hop number of the link, the relative speed of the vehicles in the link, etc. For the sake of quantitative evaluation, we propose a link stability coefficient (CV) to evaluate the stability of different link. The following notions are used in our formulas.

- n is the hop count of link between two adjacent LC,



▲ Figure 1. The positions of LC.

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie

- sd is the standard variance of the vehicle speed on the link,
- av is the average velocity of all the car in the road,
- V_i is the velocity of the car i .

The formula of CV is shown as follow. (1) and (2) take the relative velocity and node number of link into account. (3) refers to the coefficient of variation in math, ensuring the CVs of different road links with different node numbers can be compared immediately.

$$sd = \sqrt{\frac{1}{n} \sum_{i=1}^n (V_i - aV)^2}, \quad (1)$$

$$av = \frac{\sum V_i}{n}, \quad (2)$$

$$CV = \frac{sd}{av} * 100\%. \quad (3)$$

4 Two-Level Structure Protocol Design

Based on the system model, we propose a two-level SDN based Vehicle Ad-Hoc On-Demand Routing Protocol (SVAO), with a global level and a local level. The global level is distributed, using the ranked query scheme to query the objective vehicle information and the improved AODV method to calculate the route among multiple road segments/LCs. The local level is centralized, using Bellman-Ford algorithm and Floyd algorithm to maintain a stable route between two adjacent LCs. That is, to explore the different impacts of the routing algorithms, this paper implements two type of shortest path algorithms for computing route.

4.1 Local Level

The local level is responsible for computing the route for every vehicle on each road. Firstly, we need to maintain a stable vehicle link between two adjacent LCs. Every vehicle will flood Hello Message (IDv, GPS, Speed) to LC periodically. When a vehicle enters a road segment, its Hello Message will be collected by the LC. Within a certain time, LC will collect all the hello information from the vehicles on the road, and try to build a network topology for the road. Since the Hello Message contains the velocity information of the vehicle, LC can predict the vehicle movement trajectory. Furthermore, LC will simulate and predict the topology changes according to the network topology we build above. At last, LC try to maintain a shortest transmission path from this LC towards the next adjacent LC, using Bellman - Ford algorithm (**Algorithm 1**) and Floyd algorithm (**Algorithm 2**).

Algorithm 1. The Bellman-Ford Algorithm

- T := road network topology

- **for each** vehicle i **do**//initialization
- $Distant[i] = 10000$;
- $Distant[s] = 0$;
- **end for**
- **for each** edge(u, v) in T **do**
- **if** edge(u, v) == 1 **then**
- $w(u, v) = 1$;
- **else** $w(u, v) = 0$;
- **end for**
- **boot** neg = false;
- **for each** nodes u in T **do** //relaxation
- **for each** nodes v in T **do**
- **if** $Distant[u] + w(u, v) < Distant[v]$ **then**
- $Distant[v] = Distant[u] + w(u, v)$
- neg = true;
- **end for**
- **end for**
- **if** neg == false **then**
- **end**;
- **else** send error;

Algorithm 2. The Floyd Algorithm

- T := road network topology
- D := the minimum distances between any two vehicle
- **for each** vehicle i **do**//initialization
- **for each** vehicle j **do**
- $D[i][j] = \text{infinity}$;
- **end for**
- **end for**
- **for each** vertex v
- $D[v][v] = 0$;
- **end for**
- **for each** edge(u, v) in T **do**
- **if** edge(u, v) exist **then**
- $D[u][v] = 1$;
- **end for**
- **for** k from 1 to size(T) **do** //floyd
- **for** k from 1 to size(T) **do**
- **for** k from 1 to size(T) **do**
- **if** $D[u][v] > D[u][k] + D[k][v]$ **then**
- $D[u][v] = D[u][k] + D[k][v]$;
- **end for**
- **end for**
- **end for**
- **end for**

4.2 Global Level

The global level is responsible for finding the position of the objective vehicle and for calculating the global route among road segments/LCs. We assume that vehicle S needs to send a message to the vehicle D. Firstly, vehicle S sends a RouteRe-

quest message towards LC, asking for the position of D. If LC finds the location of D, it will return a route table to S, indicating the next hop.

4.2.1 Ranked Query Strategy

Once a LC receives a RouteRequest packet, it will begin to query for the position of vehicle D. In order to reduce the query cost, LC uses a ranked query strategy like cellular network. That is, LC is divided into two levels: the first level LC caches all the vehicle information in the road segment under its jurisdiction (Vehicle_ID, Direction, Speed), and then the second level LC only caches the vector of LC ID and vehicle ID instead (Vehicle_ID, LcID). The ratio between the first level LC and second level LC should be 20:1.

The ranked query strategy process is as follows:

- 1) Firstly, LC queries whether vehicle D is within the scope of its signal coverage.
- 2) If not, LC will send the query request to its second level LC. The second level LC then queries for the information of vehicle D among all the first level LCs within its signal coverage.
- 3) If not, the second level LC will flood the query request within all the second level LCs.
- 4) The process above is repeated until the position of vehicle D is found.

4.2.2 Computing Global Route Using Improved AODV Algorithm

Until now, each LC has maintained a network topology for corresponding road segment. Then, we need to compute a global route among road segments/LCs, connecting source vehicle S and destination vehicle D. We use the optimized AODV algorithm to get the route running in LC. The algorithm is shown as follow:

- 1) LC broadcasts a RREQ (Route Request) packet to other LCs.
- 2) To reduce broadcast scale and constrain RREQ broadcast direction, LC checks whether the next hop is within the rectangle constituted by source S and destination D (Fig. 2). If

not, LC drops the packet.

- 3) LC counts the passed LC number N from source vehicle S to itself. N will be compared and the best one in smaller CV_max will be selected; If $N_{new} > N_{his}$, we just drop the packet.
- 4) Within a certain time, the destination LC has received a series of RREQ packets. To make a best decision, LC tends to select the route with the smallest N, and the smallest CV_max takes the second place in the case of the same N. Then the destination LC sends a RREP (Route Request Respond) packet following in reverse route taken by the RREQ packet, updating the routing table of all the passed LCs.
- 5) LC adds the first vehicle information of the next road segment/LC into the route table, and broadcast the route table to all the vehicles within its jurisdiction.

5 Simulation Results and Analysis

For the sake of cost and safety, it is impossible to setup experiments with a practical network topology, which needs a large number of cars. So researchers usually test their solutions via simulation. We also setup experiments in NS-3, a famous simulation platform, and use Simulation of Urban MObility (SUMO) to generate vehicle trajectory files.

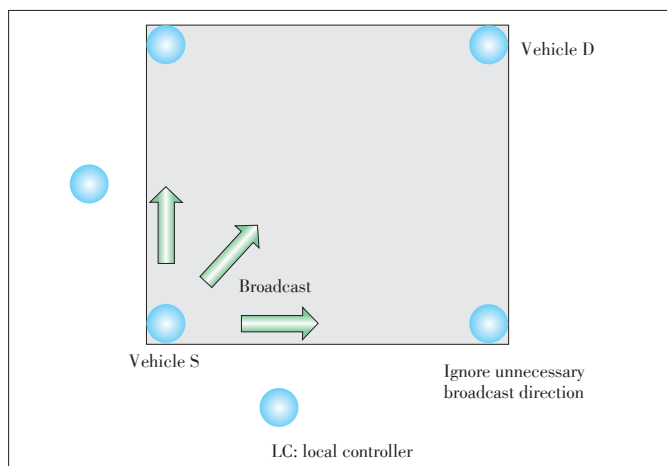
5.1 Simulation Setup

In order to evaluate the performance of the proposed routing protocol, we compare SVAO with OLSR, DSR, DSDV and DB within different vehicle density, at different vehicle speeds and in different communication ranges. Moreover, as this paper implement two different route computing algorithm, Bellman - Ford and Floyd, there are two different versions of SVAO in the following figures, while SVAO - BF represents the SVAO version using Bellman - Ford and SVAO - Flo represents the Floyd version. Data transmission and average packet delay are measured to get the contrast ratio. The simulation specifications with NS-3 are shown in Table 1.

We use map software SUMO to generate a three-lane road, whose length is 2 km. The road is uniformly divided into two sections. One section ranges from 0 km to 1 km and the other from 1 km to 2 km. At the very beginning of the simulation, the road is empty. Then the vehicles begin to appear in a random starting point at the beginning of the first section, moving forward to the second section at a certain speed. This paper sets vehicle S on the roadside that is 5.1 m away from the beginning of the road as the source vehicle and vehicle D on the end of the road as the destination vehicle. Vehicle S will send packets to vehicle D constantly.

5.2 Impact of Vehicle Density on Protocol Performance

The data transmission rate and average packet delay are important indicators to evaluate the performance of a routing protocol. We compare all the routing protocols quantitatively by



▲ Figure 2. LC ignores unnecessary broadcast directions.

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie

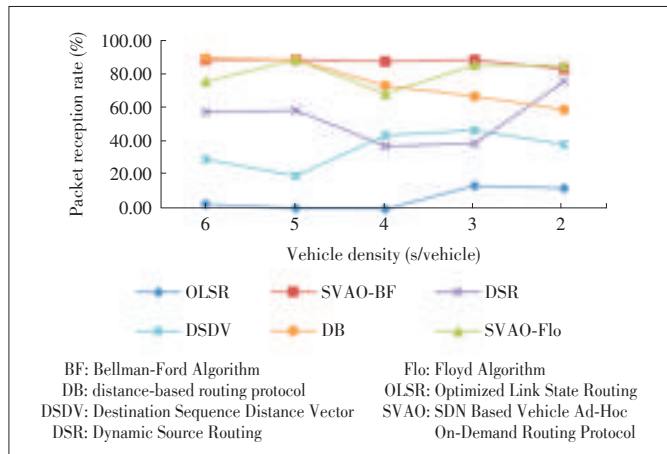
▼ **Table 1. Simulation setup**

Parameter	Simulation value
SCH transmission radius (m)	400
CCH transmission radius (m)	1000
Vehicle speed (m/s)	10, 15, 20, 25, 30
Node movement mode	Random
Packet size (byte)	1000
Routing protocol	SVAO-BF, OLSR, SVAO-Flo, DSR, DSDV, DB
Road number	2
Road length (m)	1000

BF: Bellman-Ford Algorithm Flo: Floyd Algorithm
 CCH: control channel OLSR: Optimized Link State Routing
 DB: distance-based routing protocol SCH: service channel
 DSDV: Destination Sequence Distance Vector SVAO: SDN Based Vehicle Ad-Hoc
 DSR: Dynamic Source Routing On-Demand Routing Protocol

changing the density of nodes in the network. **Fig. 3** shows the packet reception rate under different vehicle densities. The average speed of each vehicle is 15 m/s. The horizontal axis shows that there will be a new vehicle appear in the very beginning of the road every n seconds. Under the same conditions, SVAO holds a better data transmission rate than OLSR, DSDV, DSR and DB. Along with the increasing number of nodes, i.e. the network scales larger, OLSR performs stably with a low value, DSDV and DSR performs undulated, and DB performs worse and worse. On the other hand, with the network scale expanding, SVAO maintains a stable lower value. SVAO-BF performs more stable than SVAO-Flo. It is because the time complexity of Floyd algorithm is higher than Bellman-Ford. That means it will cost more time to compute route when the density is getting higher. Both of them hold a good performance in average packet delay with a high vehicle density.

As for average packet delay illustrated in **Fig. 4**, the superiority of SVAO is remarkable and intuitive. In general, simulation results show that, as the number of nodes grows, SVAO obtains better performance and is more adaptive to the large-scale network. When it comes to the average packet delay, the

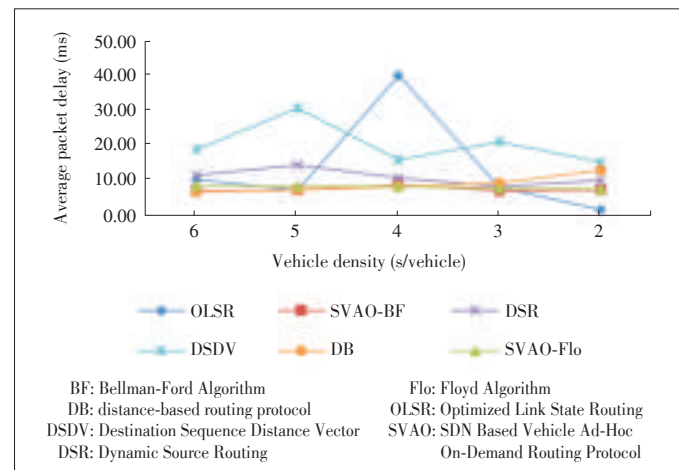


▲ **Figure 3. Packet reception rates under different vehicle densities.**

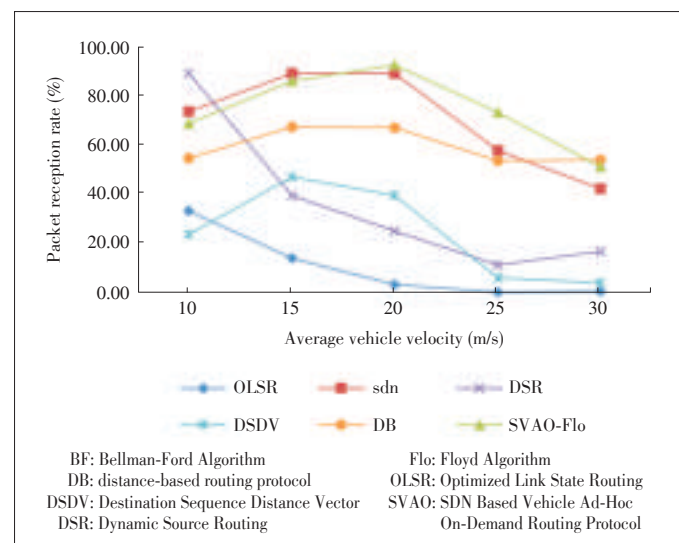
two algorithms of SVAO perform well and similar, because both of them can find a relatively shortest road for routing.

5.3 Impact of Vehicle Velocity on Protocol Performance

It can be seen from **Fig. 5** that at the same vehicle velocity, SVAO has an observably better packet reception rate. Along with the increase of speed, i.e. the road topology changes more frequently, all the protocols are observed a decline. The DSDV and DSR performance drop shapely, while the decline of SVAO and DB are unapparent. From the perspective of average packet delay (**Fig. 6**), with the vehicle speed gradually increasing, SVAO always keeps the best average packet delay (a lower value), while DSDV and OLSR perform worse. When it comes to the comparison between SVAO-BF and SVAO-Flo, we can see that they have the similar trend. When the vehicle goes faster, SVAO-Flo performs a little better. In short, simulation results show that with the vehicle speed increasing, i.e. the network topology of the road changes drastically, SVAO



▲ **Figure 4. Average packet delay under different vehicle densities.**



▲ **Figure 5. Packet reception rates under different vehicle velocities.**

has a more stable and excellent packet reception rate and average packet delay.

5.4 Impact of Communication Range on Protocol Performance

Fig. 7 shows that SVAO has a better performance on packet reception rate under the same vehicle communication range. Along with the communication range increasing, the road topology becomes more complex, but the connectivity of road topology improves at the same time. In view of packet reception rate, the order is $SVAO > DB > DSDV > DSR > OLSR$. However, when the vehicle communication range becomes larger (larger than 400 m), DB performs better than SVAO. That is because DB can find a shorter path towards the destination easier although the topology becomes more complex. The average packet delay illustrated in **Fig. 8** is similar to sections 5.2 and 5.3, i. e. SVAO can obtain a stable and better performance. In general, along with the increase of the vehicle communication range, the road topology becomes more complex and SVAO performs better in both packet reception rate and average packet delay. Moreover, as shown in Fig. 7, the SVAO-Flo has a higher time complexity than SVAO-BF and the performance of SVAO-Flo can be better within a certain density and different communication ranges. That is, SVAO-Flo is more adaptable under different communication ranges.

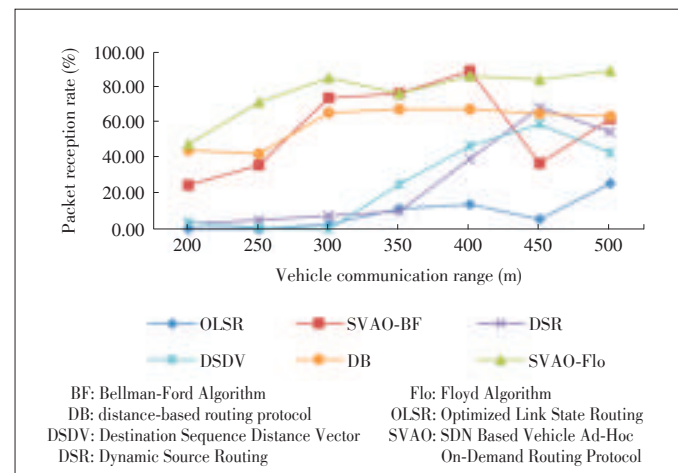
5.5 Impact of Route Computation Interval on SVAO Performance

In this subsection, we collect data information from SVAO-BF with different route computing intervals. SVAO-BF adopts centralized network architecture, and routing discovery and computation process are conducted on the LC, which effectively reduces the route computing overhead. Therefore, how to achieve balance on the route computation cost and time cost is an important research direction. We quantitatively evaluate the impact of different route computation intervals on SVAO-

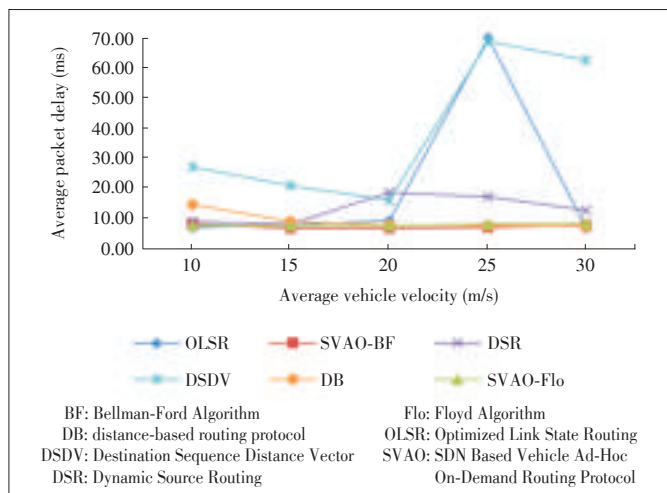
BF performance. It can be seen from **Fig. 9** that a 10 s–14 s route computation interval can be a balanced choice. On the one hand, reducing the interval cannot improve the packet reception rate but can increase the route computing overhead. On the other hand, increasing the route computing interval will significantly reduces the SVAO-BF packet reception rate.

6 Conclusions and Future Work

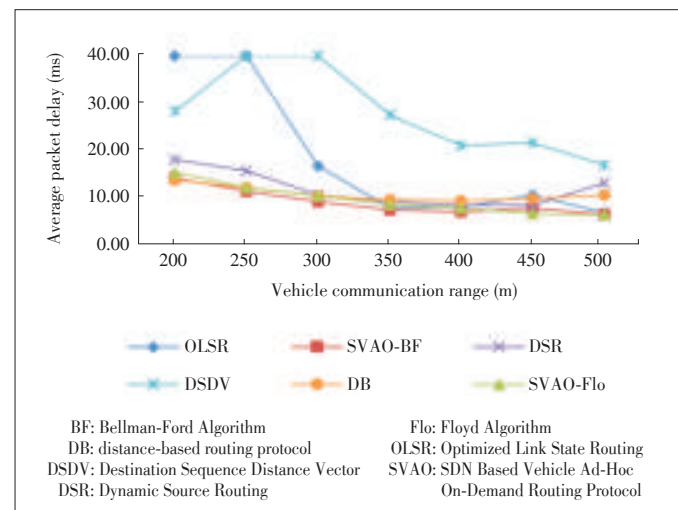
In this paper, we propose a SDN based on-demand routing protocol (SVAO), utilizing the separation of the data transfer layer and network control layer of SDN to enhance the data transmission efficiency within VANETs. Different from similar works on VANETs, our main work focuses on redesigning the network control layer and data transfer layer in VANETs, making SDN implemented in VANETs. This paper comes up with a two-level structure, including a distributed global level and a



▲ **Figure 7.** Packet reception rates under different vehicle communication ranges.



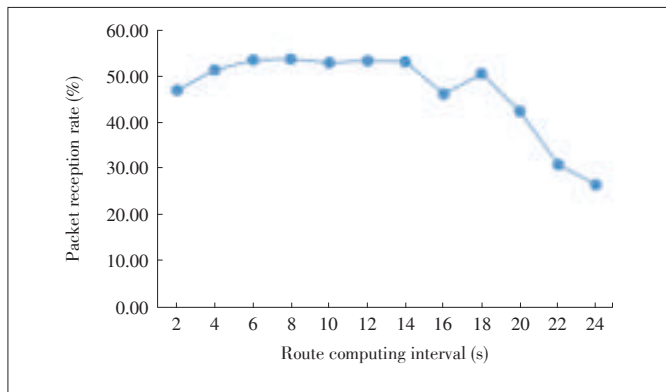
▲ **Figure 6.** Average packet delay under different vehicle velocities.



▲ **Figure 8.** Average packet delay under different vehicle communication ranges.

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie



▲ Figure 9. Packet reception rates under different route computing intervals.

centralized local level. The simulation results show that, in view of packet reception rate and average packet delay, SVAO performs better and more stable than traditional ad-hoc routing protocols (DSR, DSDV, OLSR and DB). In more detailed, SVAO-Flo has a similar performance with SVAO-BF no matter in average packet delay and packet reception rate, and SVAO-Flo is more adaptable under different vehicle communication ranges.

In future work, we may focus on the connection between two adjacent links. How to obtain the balance between the node number in a link and the stability of a link will be a promising direction. What's more, the simulation results show that SVAO performance will decline sharply under high-speed conditions. This needs further optimization.

References

- [1] G. Samara, W. A. H. Al-Salihy, and R. Sures, "Security analysis of vehicular ad hoc networks (VANET)," in *2010 Second International Conference on Network Applications, Protocols and Services*, Alor Setar, Malaysia, 2010, pp. 55–60. doi: 10.1109/NETAPPS.2010.17.
- [2] S. U. Rehman, M. A. Khan, T. A. Zia, and L. Zheng, "Vehicular ad-hoc networks (VANETs)—an overview and challenges," *EURASIP Journal on Wireless Communications and Networking*, vol. 3, no. 3, pp. 29–38, 2013. doi: 10.5923/j.jwnc.20130303.02.
- [3] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994. doi: 10.1145/190809.190336.
- [4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996. doi: 10.1007/978-0-585-29603-6_5.
- [5] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999, New Orleans, USA, pp. 90–100. doi: 10.1109/MCSA.1999.749281.
- [6] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 1, p. 69, 2005. doi: 10.1145/1055959.1055970.
- [7] H. Saleet, R. Langar, K. Naik, et al., "Intersection-based geographical routing protocol for VANETs: a proposal and analysis," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4560–4574, 2011. doi: 10.1109/TVT.2011.2173510.
- [8] C. Lochert, H. Hartenstein, J. Tian, et al., "A routing strategy for vehicular ad hoc networks in city environments," in *Proc. IEEE IV2003 Intelligent Vehicles Symposium, Columbus, USA*, pp. 156–161, 2003. doi: 10.1109/IVS.2003.1212901.

- [9] J. J. Chang, Y. H. Li, W. Liao, and I. C. Chang, "Intersection-based routing for urban vehicular communications with traffic-light considerations," *IEEE Wireless Communications*, vol. 19, no. 1, pp. 82–88, 2012. doi: 10.1109/MWC.2012.6155880.
- [10] J. Zhao and G. Cao, "VADD: vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1910–1922, 2008. doi: 10.1109/TVT.2007.901869.
- [11] Q.-Y. Zuo, M. Chen, G.-S. Zhao, et al., "Research on openflow-based SDN technologies," *Journal of Software*, vol. 24, no. 5, pp. 1078–1097, 2013. doi: 10.3724/SP.J.1001.2013.04390.
- [12] B. N. Astuto, X. N. Nguyen, K. Obraczka, T. Turletti, and M. Mendonca, "A survey of software defined networking: past, present, and future of programmable networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014. doi: 10.5923/j.jwnc.20130303.02.
- [13] S.-A. L. Lazar and C.-E. Stefan, "Future vehicular networks: what control technologies?" in *International Conference on Communications*, Bucharest, Romania, 2016, pp. 337–340, 2016. doi: 10.1109/ICComm.2016.7528203.
- [14] K. Liu, J. K. Y. Ng, V. C. S. Lee, S. H. Son, and I. Stojmenovic, "Cooperative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1759–1773, 2016. doi: 10.1109/SURV.2014.012214.00180.
- [15] A. Kazmi, M. A. Khan, and M. U. Akram, "DeVANET: decentralized software-defined VANET architecture," in *IEEE International Conference on Cloud Engineering*, Berlin, Germany, 2016, pp. 42–47. doi: 10.1109/IC2EW.2016.1662.
- [16] X. F. Xiao and X. Kui, "The characterizes of communication contacts between vehicles and intersections for software-defined vehicular networks," *Mobile Networks and Applications*, vol. 20, no. 1, pp. 98–104, 2015. doi: 10.1007/s11036-014-0535-6.
- [17] A. Di Maio, M. R. Palattella, R. Soua, et al., "Enabling SDN in VANETs: what is the impact on security?" *Sensors*, vol. 16, no. 12, p. 2077, 2016. doi: 10.3390/s16122077.
- [18] A. Kazmi, M. A. Khan, F. Bashir, et al., "Model driven architecture for decentralized software defined VANETs," in *Future Intelligent Vehicular Technologies*, LNCS, vol. 185. Cham, Switzerland: Springer, 2017, pp. 46–56. doi: 10.1007/978-3-319-51207-5_5.

Manuscript received: 2017-01-20

Biographies

DONG Baihong (bh.dong@foxmail.com) received the B.Sc. degree in computer science from Sun Yat-sen University, China in 2012. He is studying for a M.Sc. degree in Sun Yat-sen University. His research interest is the SDN-based vehicular ad-hoc networks.

WU Weigang (wuweig@mail.sysu.edu.cn) received the B.Sc. degree in 1998 and the M.Sc. degree in 2003, both from Xi'an Jiaotong University, China. He received the Ph.D. degree in computer science in 2007 from Hong Kong Polytechnic University, China. He is currently an associate professor at the Department of Computer Science, Sun Yat-sen University, China. His research interests include distributed systems and wireless networks, especially cloud computing platforms and ad-hoc networks. He has published more than 50 papers in conferences and journals. He has served as a member of editorial board of two international journals, *Frontiers of Computer Science* and *Ad-Hoc & Sensor Wireless Networks*. He is also an organizing/program committee member for many international conferences. He is a member of the IEEE and ACM.

YANG Zhiwei (zhiwei200654@163.com) received the B.Sc. degree in 2006 and the M.Sc. degree in 2008, both from Sun Yat-sen University, China. He is currently a Ph.D. student majored in software and computer theory in Sun Yat-sen University. His research interests include vehicular ad-hoc networks and distributed systems, especially dynamic networks. He mainly focuses on information dissemination, counting, consensus and dynamic models in dynamic network, and has published related papers.

LI Junjie (goals.lee@qq.com) received the B.Sc. degree in automation science from South China University of Technology, China in 2012. He is studying for a M.Sc. degree in Sun Yat-sen University. His research interest is the SDN-based vehicular ad-hoc networks.

An MEC and NFV Integrated Network Architecture

LI Bing, ZHANG Yunyong, and XU Lei

(China Unicom Research Institute, Beijing 100032, China)

Abstract

The demand for 5G services and applications is driving the change of network architecture. The mobile edge computing (MEC) technology combines the mobile network technology with cloud computing and virtualization, and is one of the key technologies for 5G networks. Compared to network function virtualization (NFV), another critical enabler of 5G networks, MEC reduces latency and enhances the offered capacity. In this paper, we discuss the combination of the two technologies and propose a new architecture. Moreover, we list the application scenarios using the proposed architecture.

Keywords

MEC; 5G; NFV; cloud computing; network architecture evolution

1 Introduction

According to the IMT-2020 promotion group's description, the development of 5G networks is mainly motivated by two future visions: one is to increasingly enhance user experience of "people-oriented" mobile Internet, and the other is to support the Internet of Things (IoT) services that will create an "Internet of Everything". To meet the requirements of different scenarios, the 5G network should have the following capabilities: high-speed and high-bandwidth access capabilities for upgraded mobile applications such as voice calls, Internet access and ultra-high-definition video; ultra-dense device access and management capabilities for IoT device interconnections such as digital medical, open-air gatherings and concerts scenarios; and low-latency and highly-reliable end-to-end communication capabilities for the Internet of Vehicles (IoV), industrial automation, and other vertical industry applications [1]. However, the current mobile communication networks are far from these capacities.

With the mutual penetration and influence of the mobile network and Internet technologies, the software defined networking/network function virtualization (SDN/NFV) technology that uses such IT technologies as cloud computing, virtualization and software has become an important enabler for 5G networks, and has been applied in wide area networks (WANs) and metropolitan area networks (MANs). SDN/NFV enables resource management and interconnection across the data center at the WAN level, and achieves hardware resource virtualization and resource scheduling within the data center at the MAN level. However, the application of the NFV technology at

the radio access network level is still under research in the industry [2]–[4].

Mobile edge computing (MEC), a new network architecture technology for the fusion of mobile access network and IT technology, has recently been proposed. MEC uses the wireless access network to provide services and cloud computing functions required by telecom users, and to construct a carrier-class service environment with high performance, low latency and high bandwidth to improve communication experience of mobile users. In 2014, the ETSI formally included MEC into the standards for discussion [5], signifying that MEC had become one of the key 5G technologies [6]–[7]. Network equipment manufacturers have worked together with telecom operators to develop the MEC platform and solutions, and made a public demonstration [8]–[11]. MEC has also been tested in the first experimental phase of 5G network in China. The expected performance has been achieved in the test [12].

This paper introduces the MEC technology and network architecture, compares MEC with NFV, integrates these two technologies, and enumerates the applications of MEC and NFV integrated architecture. Finally, this paper summarizes and forecasts the development of the MEC technology.

2 MEC Technology

2.1 Basic Concepts

According to the ETSI definition, the MEC technology is to provide wireless access networks with IT and cloud computing capabilities by deploying Commercial Off-The-Shelf (COTS)

An MEC and NFV Integrated Network Architecture

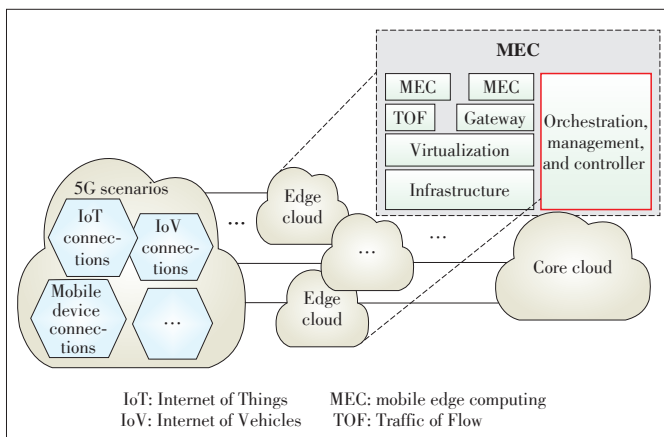
LI Bing, ZHANG Yunyong, and XU Lei

on the wireless access side [13]. According to IMT-2020, MEC is to place the service platform on the edge of the network and provide nearby service computing and data caching for mobile users. **Fig. 1** gives out the location of MEC in the 5G network architecture.

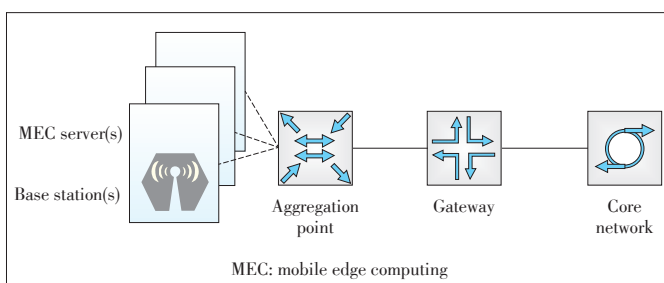
MEC can be flexibly deployed on the mobile wireless base station or the convergence point of a variety of access technologies. If deployed on the base station side, the operator can go deeper to analyze user scenarios such as mobile data traffic, wireless network environment, and the user's specific location to provide a better user experience and enable a smart base station (**Fig. 2**). If deployed on a converged point, the operator can implement operation support for multiple services in the region to provide customized and differentiated services for different services, thus improving network utilization efficiency and adding values (**Fig. 3**).

2.2 MEC Architecture

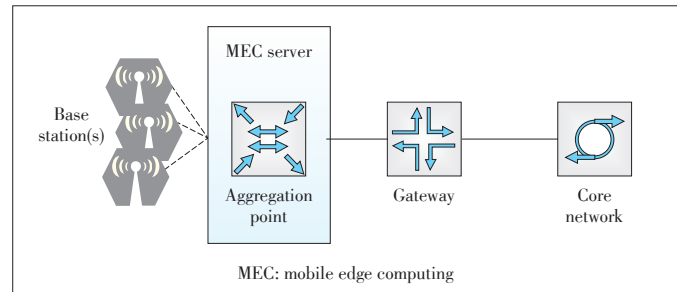
The MEC system architecture (**Fig. 4**) consists of a mobile edge host and associated management components. The mobile edge host includes the mobile edge platform and the virtualization infrastructure. The mobile edge platform has all necessary functions to support various types of mobile edge applications on the mobile edge host. It also provides a variety of mobile edge services to other mobile edge hosts and itself. The mobile edge platform is responsible for providing services including



▲ Figure 1. The location of MEC in 5G scenarios.



▲ Figure 2. MEC network architecture solution 1: MEC deployed on the base station [13].



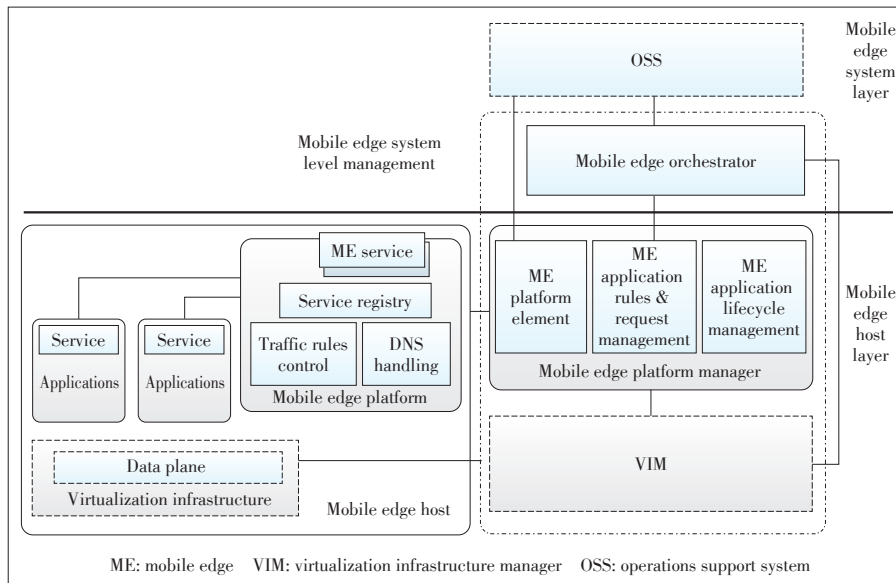
▲ Figure 3. MEC network architecture solution 2: MEC deployed on the aggregation point [13].

service discovery, traffic forwarding control and domain name system (DNS) management. Service discovery ensures that mobile edge applications discover or utilize mobile edge services; the traffic forwarding control service is responsible for forwarding traffic data packets; and DNS management ensures that applications are discovered in the network [14]. Mobile edge applications run on a mobile edge host in the form of a virtual machine instance to utilize or provide mobile edge services.

The mobile edge management system consists of the mobile edge orchestrator (MEO), mobile edge platform manager (MEPM), and virtualization infrastructure manager (VIM). The MEO has a global view of the MEC system, including information on all deployed mobile edge hosts, information on available services and resources in each host, and information on instantiated mobile edge applications and network topologies. In addition, it is responsible for the ME application installation, application integrity check and authentication. The MEPM corresponds to a single mobile edge host. It is responsible for mobile edge platform element management, ME application lifecycle management and ME application rules and requirements management. The application rules and requirements refer to the rules and requirements associated with various types of mobile edge applications, including the required resources, maximum latency/delay, required or useful services, traffic rules, DNS rules, and mobility support. The MEPM obtains the latest information on the services available in the system from the MEO and interacts with the OSS to implement fault configuration and performance management functions.

The VIM manages the virtualized resources assigned to mobile edge applications. It can obtain information from the MEO for managing application images and virtualization resources and for monitoring resource availability. In addition, it interacts with the MEPM to manage the virtualization resources associated with the mobile edge application lifecycle.

In addition to the functional modules described above, the MEC architecture also includes modules for user portals, UE applications and user application lifecycle agents, as well as interfaces for communication between the modules. Wherein an UE application is an application run by the user terminal and has the capability of interacting with the mobile edge system through the user application lifecycle management agent. An



▲ Figure 4. MEC system architecture.

user application lifecycle management agent allows initiation and instantiation of the terminal application request, termination of terminal application requests, and relocation or removal of user applications in the mobile edge system. It also informs an terminal application of the state of a user application. The specific functions of the interface in MEC can be found in [14].

2.3 Work Related with MEC Standardization

The MEC concept was originated in the 3G era. After the 3G smart phone became popular, the MEC idea was embodied in the mobile CDN specifically developed for the mobile network. Some operators and manufacturers proposed that, based on 4G base stations, the computing capability can be integrated in the 4G base stations. This is the “prototype” of the MEC technology [16]. In 2013, Huawei, IBM, Intel, Nokia, NTT Docomo and Vodafone formally put forward the MEC technology [13]. In October 2014, the ETSI ISG introduced the MEC into the standards, and conducted in-depth studies on service scenarios and technical requirements for the MEC technology. ETSI has published five MEC-related standards, including terminology, service scenarios, technical requirements, framework and reference architecture, and Proof of Concept framework, as shown in **Table 1**. The MEC application interface standardization and what services are to be provided with the MEC are still under discussion.

The MEC Proof of Concept (POC) is another key work of the ETSI, which solicits MEC prototypes and verification tests to promote the development of the MEC industry. The MEC POC has eight projects (**Table 2**), the project participants are from all the network service sectors, including network equipment manufacturers (Intel and Nokia), telecom operators (China Mobile and Deutsche Telekom), and Internet companies (iQIYI and SeeTec) [17]. In 2016, Nokia used the MEC networking so-

lution to build a live broadcast platform in Shanghai International Circuit to provide live broadcast services to the audience. With this MEC solution, the live video had only a delay of as short as 0.5 second [11].

In the first test phase of the IMT-2020 (5G) promotion group, three Chinese operators completed the performance and functional test of mobile edge computing with Huawei, Nokia, Shanghai Bell, Datang and Intel. In the next phase of the test, Intel will explore the commercial value brought by the MEC technology for the mobile edge entrance; Nokia will use the AirScale cloud base station server to deliver the MEC access network architecture to meet the needs of a variety of 5G application scenarios; and China Telecom will set up pilot MEC sites in multiple Chinese provinces to verify main 5G service scenes

[12], [18].

3 NFV Technology

The NFV technology uses cloud computing and virtualiza-

▼ **Table 1.** MEC standards released by ETSI Industry Specifications Group (ISG) [15]

Number	Title
GS MEC 002	MEC, technical requirements
GS MEC 003	MEC, framework and reference architecture
GS MEC 001	MEC, terminology
GS MEC-IEG 004	MEC, service scenarios
GS MEC-IEG 005	MEC, proof of concept framework

MEC: mobile edge computing

▼ **Table 2.** MEC POC projects completed by ETSI [17]

Application case	Participants
Video user experience optimization	Intel, China Mobile, iQIYI
Edge video orchestration and video clip playback	Nokia, EE (a British telecom operator), Smart Mobile Labs
Radio aware video optimization in a fully virtualized network	Telecom Italia, Intel UK, Eurecom, Politecnico di Torino
Video analytics	Nokia, Vodafone Hutchison Australia, SeeTec
FLIPS-IP-based flexible service	InterDigital Communications, UK “Bristol Is Open”, Intracom (Greece), CVTC, Essex University (UK)
Enterprise services	Saguna, Adva Optical Network (Germany), Bezeq International (Israel)
Healthcare—dynamic hospital user, IoT and alert status management	Quortus Ltd.(UK), Argela (Turkey), Turk Telekom
Multiservice MEC platform for advanced service delivery	Brocade, Gigaspaces, Advantech, Saguna, Vasona, Vodafone

An MEC and NFV Integrated Network Architecture

LI Bing, ZHANG Yunyong, and XU Lei

tion technology to strip out the network functions from dedicated hardware devices so that network functions can be delivered with software in an independent way. In the NFV technology, the independent network element function is applied to the cloud computing platform built on the standard server to realize the rapid expansion/contraction of the network, enhancing the resilience and flexibility of the network.

3.1 NFV Architecture

According to the NFV reference architecture developed by the ETSI (Fig. 5), the NFV has two horizontal layers: the network service (NS) domain and management and orchestration (MANO) domain. The NS domain consists of three vertical layers: the NFV infrastructure (NFVI), virtual network function (VNF), and operational support layer. The MANO domain consists of three entities: the virtualized infrastructure manager (VIM), VNF manager (VNFM), and NFV orchestrator (NFVO), which are responsible for the whole lifecycle and scheduling strategy management of NFVI, VNF and network services [20].

The NFVI is similar to a cloud data center for hosting and connecting virtual functions. It is responsible for virtualizing the underlying physical resources and realizing the physical bearing of all the elements required by the VNF. The VNF is to virtualize all existing physical network infrastructure in the communication network on the basis of the NFVI, such as the residential gateway in a home network, DHCP server, and firewall. A single VNF may have multiple internal components, so it can be deployed in multiple virtual machines, each carrying

one component of the VNF. One or more VNFs can serve the customer as a whole. The operational support layer does virtualization adjustments based on the current OSS/BSS. The MANO delivers orchestration and management for the NFVI, VNF, and NS through interactions of the VIM, VNFM, and NFVO, to achieve management and orchestration of entire network services [21].

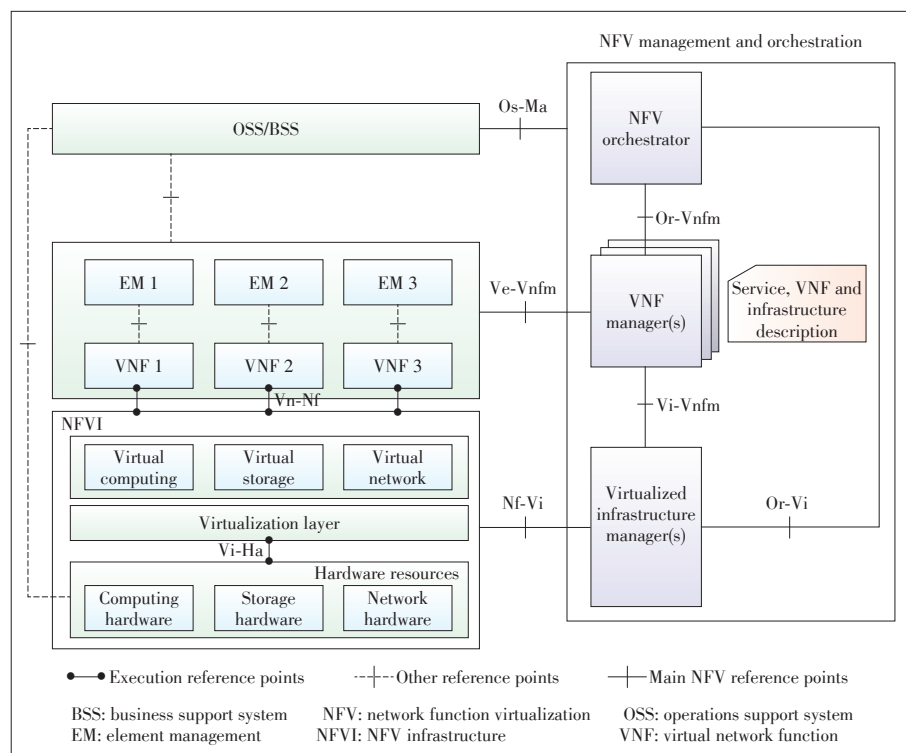
3.2 NFV-Related Studies

In 2012, seven operators, including AT&T, Deutsche Telekom, Orange and Telefonica, co-established the NFV ISG at ETSI to develop the requirements and architecture specifications supporting the NFV hardware and software infrastructure, and the virtual network function guide as well. They also worked with other standard organizations to integrate existing virtualization technologies and related standards when necessary [4].

While the conventional standardization works were being carried out, the open source organizations were actively promoting the NFV technology. In 2014, telecom operators including AT&T and China Mobile and telecom vendors including Dell and Hewlett-Packard cooperated with the Linux Foundation on establishing an open source project, the Open Platform for NFV (OPNFV), aiming to build a complete NFV implementation standard. The OPNFV integrates existing open source components with new components and tests to ensure consistency, performance and interoperability across multiple open source components, thereby speeding up the development and deployment of the NFV technology. In June 2015, the organization officially released the first version of NFV open source framework [3].

4 MEC/NFV Network

The MEC utilizes virtualization and cloud computing technologies to run and manage applications and services at the edge of the mobile network using virtualized platforms, providing a low-latency, high-bandwidth service experience. The NFV uses general-purpose open hardware and virtualization technology instead of traditional dedicated network equipment, to provide hardware and software decoupling for network functions and flexible management of network services and functions. The MEC and NFV have similar architectures and both require a network infrastructure virtualization platform. In addition, the management function of the two have components that can be combined and re-divided. Therefore, an MEC/NFV



▲ Figure 5. NFV architecture [19].

architecture is proposed in this paper, in a bid for achieving the goals of the MEC technology.

4.1 MEC/NFV Network Architecture

Fig. 6 shows the proposed MEC/NFV architecture. This MEC/NFV architecture has NFV system as the main body. It divides the MEC management modules in a new way and puts forward a new system architecture that features two vertical domains and three horizontal layers. The two vertical domains are the MEC domain supporting the upper layer application and the NFV domain supporting the underlying network resources. The three horizontal layers are the orchestration layer, the management layer and the basic resource layer from top down. The orchestration layer and the management layer span the MEC and the NFV domains, and the basic resource layer is provided by the NFV domain.

First of all, the basic resource layer comprises virtual infrastructure and the VNF, which are implemented by the modules in the NFV domain. The NFVI implements a virtualized infrastructure for MEC to use and share with other network functions or applications. The VNF implements the ME application and service components in the MEC domain, and the network element functions of the NFV domain.

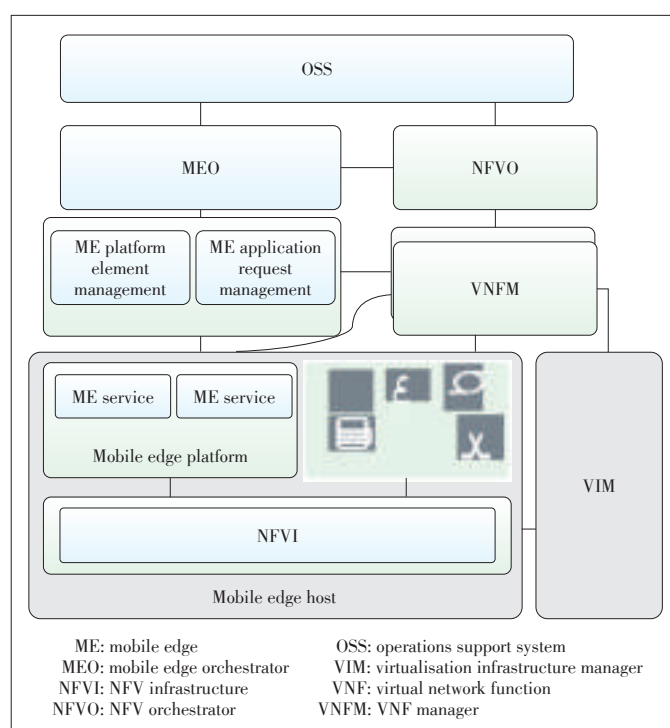
The VIM manages virtualized infrastructure. The VNFM controls the creation and release, expansion and shrinkage of ME applications and service instances, and interacts with the VIM to apply for VM resources based on the resource usage of MEC applications and services, monitor resource usage and warn for faults. The NFVO in the NFV domain is responsible

for the orchestration of the NFV-related services. It determines the number of VNFs, the VNF type, and the VNF topology to be deployed, creates the VNFM instance, and interacts with the VNFM and MEO.

The key to the MEC technology is to achieve an open framework, to support the development of user applications, and to simplify the deployment of related services. Therefore, outside the NFV domain, the MEC domain is responsible for the application service-related computing services, service content storage, content delivery and download acceleration, and the definition and management modules for the application service open interface. In the converged architecture, the MEC domain consists of three modules from bottom to the top: mobile edge platform, MEPM, and MEO. The mobile edge platform provides services such as application-level traffic forwarding, processing and calculation of application service content data, and real-time access network state awareness, which can be implemented by the VNF. The MEPM provides application service communication interface management, application-level traffic forwarding rule management, and DNS configuration and IP address conflict management. The MEO implements application and service orchestration based on the requirements of the operational system and service applications.

The interface between the MEC domain and the NFV modules in the converged architecture is responsible for the interaction between the service application information and the network service information in the MEC domain and the NFV domain to ensure the modules' awareness of application and the network system status. In this way, the modules in different domains at the orchestration and management layers can jointly meet the requirements for application services. For example, the MEO informs the NFVO of the update of the UE application instance and the mobile edge system status through the interface between the MEO and the NFVO, and the NFVO performs system orchestration according to the update information. The MEPM informs the VNFM of conflicts and location change state caused by network resources between different applications through the interface between the MEPM and the VNFM, and VNFM manages the network element instance life-cycle according to the message.

In the MEC/NFV architecture, NFVO, VNFM, VNF, NFVI and VIM function modules in the NFV domain corporately provide applications with network services and infrastructure services from top to bottom. The MEO, MEPM and mobile edge platform in the MEC domain collaborates and responsible for the application demand information and application instance management service. The orchestration layer and the management layer ensure the communication and status consistency of modules in the NFV domain and the MEC domain via the interfaces and jointly meet the performance requirements of applications for low delay and high bandwidth. In addition, the MEC/NFV architecture demonstrates the MEC benefits and makes as much use as possible of the NFV module to effective-



▲ **Figure 6.** MEC/NFV architecture.

An MEC and NFV Integrated Network Architecture

LI Bing, ZHANG Yunyong, and XU Lei

ly enhance the resource utilization and reduce operating costs.

4.2 Application Scenarios of the MEC/NFV Network

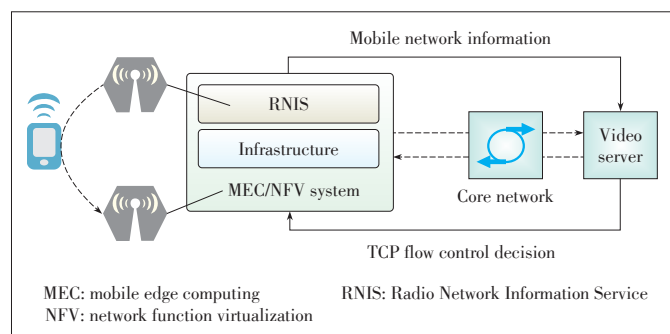
4.2.1 Optimized High-Speed and High-Definition Video Experience

Fig. 7 illustrates the use scenarios of high-speed mobile high-definition video experience optimized by the MEC/NFV system. The Radio Network Information Service (RNIS) provided by the MEC/NFV system provides applications with information such as the load capacity of the real-time mobile network of the wireless downlink interface. In the cellular network, the rapid movement of the user terminal causes changes to the underlying wireless channel environment. In a traditional network architecture, the information of the mobile network is hard to be perceived by the TCP network. The TCP protocol needs several handshakes to adapt to the change of the wireless network, and the delay is big (the dotted line in Fig. 7).

When a mobile terminal of a mobile high-definition video application is moving fast, the RNIS service provided by the MEC/NFV system assists the video server in making a more reasonable TCP congestion control decision and ensures that the application layer code can match the estimated capacity of the wireless downlink (the solid arrow in Fig. 7). Among them, the TCP will no longer need to proactively detect the available wireless network resources, or reduce the data transmission rate according to relevant detection results. The MEC/NFV system in Fig. 7 is deployed on the base station. The infrastructure is provided by the cloud platform and is distributed inside base stations at different geographical locations and communicates through standard interfaces.

4.2.2 Optimized Mobile Game Experience

For applications requiring high data processing latency, such as gaming and augmented reality/virtual reality (VR/AR), the MEC offers lower latency and an optimized user experience. After the deployment of the MEC technology, the calculation of computing-intensive applications can be shunted to the wireless access network side. The MEC server with high computing performance can issue decisive orders in a short time to improve overall performance.



▲ Figure 7. Mobile high-definition video experience optimization.

The mobile edge management system may consider relocating the mobile edge application (or its partial memory state) to better meet the latency requirement when the position of a game player moves into the area serviced by multiple mobile edge hosts. But if the relocation execution can take place in the key stages of the game, service degradation will still lead to a bad user experience. The MEPM allows the ME to pass the relevant parameters to the MEO. Thus, the ME application may suggest that the MEO perform relocation after the player reaches the checkpoint or when the game loads the next level, preventing a drastic downgrade of the service.

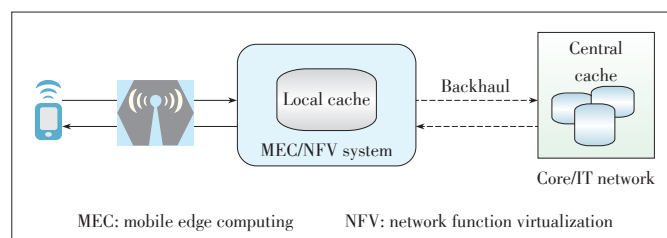
4.2.3 Local Content Caching of AR

The mobile edge application stores content that is frequently used by the user locally and provides content from the local cache when the user requests access to the content without the need to always transmit content over the core network, which reduces backhaul capacity and downloads. The network architecture of the local information cache is shown in **Fig. 8**, where the content can be sent to the device (see the solid line in Fig. 8) when the content is hit by the local cache of the MEC server, instead of being obtained through the backhaul network from the central cache of the core network/IT network, which greatly reduces the traffic pressure of the backhaul network.

The local content caching of the MEC technology brings great advantage to the AR service. Because the application services need to record the user location and camera perspective and such information is highly localized and needs to be updated in real time, so the information should better be stored locally rather than in the cloud. Otherwise, when AR services are used in a user-intensive scenario, frequent and large content access requests will cause greater bandwidth pressure on the backhaul network. Once the network congestion occurs, AR application latency will increase, and the user experience will be very bad. In Fig. 8, the MEC/NFV system is deployed on the base station, the local cache service is provided to the user by the MEC/NFV system in the form of services, and the infrastructure is provided by the cloud platform and located inside base stations at different geographical locations, and they communicate through standard interfaces.

5 Conclusions

This paper describes the MEC technology that closely inte-



▲ Figure 8. Local content cache network architecture.

An MEC and NFV Integrated Network Architecture

LI Bing, ZHANG Yunyong, and XU Lei

grates mobile access networks with IT technologies. To explore the benefits of MEC at the radio access network and extends the NFV framework, we propose a new MEC and NFV integrated network architecture with two vertical and three horizontal domains. Finally, the application scenarios of the proposed MEC/NFV architecture are also illustrated.

The MEC has the characteristics of localization, proximity, low latency, location awareness, and access to network context information. With MEC technology, network operators is capable of making use of existing network infrastructure to achieve low latency and real-time processing at network edge, thus effectively reducing operating costs while improving service levels. The MEC can create a new value chain and ecosystem. Mobile network operators should work with upstream and downstream vendors to jointly promote MEC technology research and development and commercialization to accelerate the development and deployment of 5G.

References

- [1] IMT-2020 (5G) Promotion Group, "5G vision and demand white paper", China, May 2016.
- [2] F. Shi and C. Zhang, "Analysis of the reconstruction of the SDN/NFV-based telecommunications network architecture," *Telecommunications Technology*, no. 6, pp. 30–32, Jun. 2016.
- [3] L. Xu, Y. Zhang, and W. Xiong, "Research on the architecture of the open source-based NFV platform," *Information and Communications Technologies*, vol.10, no.50, pp. 22–27, Feb. 2016.
- [4] H. Zhu, K. Chen, and X. Liu, "Research on the NFV-based new architecture and key technologies of the mobile networks," *Telecommunications Network Technology*, no. 8, pp. 7–12, Aug. 2014.
- [5] H. Li, G. Shou, Y. Hu, and Z. Guo, "Mobile edge computing: progress and challenges," in *4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, Oxford, UK, 2016. doi: 10.1109/MobileCloud.2016.16.
- [6] IMT-2020 (5G) Promotion Group, "White Paper on 5G Network Architecture Design," China, May 2016.
- [7] B. Blanco, J. O. Fajardo, I. Giannoulakis, et al., "Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN," *Computer Standards & Interfaces*, accepted. doi: 10.1016/j.csi.2016.12.007.
- [8] H. Wei, "Intel: mobile edge computing, an important pillar of 5G," *Communications World*, no. 17, p. 25, May 2016. doi: 10.13571/j.cnki.cww.2016.17.015.
- [9] Y. Sun, "Intel: multi-faceted, differentiated advancement of MEC," *Communications World*, no. 12, p. 39, Apr. 2016. doi: 10.13571/j.cnki.cww.2016.12.026.
- [10] H. Huang, "MEC to help operators in digitalization transformation and huawei has made positive practice in indoor scenes," *Communications World*, no. 17, pp. 20–21, May 2016. doi: 10.13571/j.cnki.cww.2016.17.011.
- [11] X. Wang, "Different competition event video experience: nokia's first commercial network MEC," *Communications World*, no. 12, p. 34, Apr. 2016. doi: 10.13571/j.cnki.cww.2016.12.026.
- [12] IMT-2020 (5G) Promotion Group, "5G wireless technology test progress and follow-up plan," China, Sept. 2016.
- [13] M. Patel and Y. Hu, "Mobile-edge computing—introductory technical white paper," Mobile-Edge Computing (MEC) Industry Initiative, China, Sept. 2014.
- [14] *Mobile Edge Computing (MEC); Framework and Reference Architecture*, ETSI GS MEC 003 V1.1.1, Mar. 2016.
- [15] S. Dahmen - Lhuissier. (2017, Jan.). *Mobile edge computing*, ETSI. [Online] Available: <http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing>
- [16] G. Brown. (2016, Jul.). Mobile edge computing use cases & deployment options [Online]. Available: <https://www.juniper.net/assets/us/en/local/pdf/whitepapers/2000642-en.pdf>
- [17] ETSI. (2017, Jan. 9). *Ongoing PoCs—MECwiki* [Online]. Available: http://mec-wiki.etsi.org/index.php?title=Ongoing_PoCs
- [18] CCID. (2017, Jan. 22). *Standard release for four tests, five unifications, five major focuses, and the 5G second phase of test_5G* [Online]. Available: <http://www.ccidcom.com/html/chanpinjishu/wuxiantongxin/5G/20161128.html>
- [19] *Network Functions Virtualization (NFV): Management and Orchestration*, ETSI GS NFV-MAN 001, Dec. 2014.
- [20] W. Chen, X. Han, and L. Ni, "Mobile core network application of NFV: key issues and practice," *ZTE Technology Journal*, vol. 20, no. 3, pp. 12–15, Jun. 2014.
- [21] H. Xue and H. Zhang, "Virtualization and standardization of network functions," *ZTE Technology Journal*, vol. 21, no. 2, pp. 30–34, 2015.

Manuscript received: 2017-01-24

Biographies

LI Bing (lib78@chinaunicom.cn) received her Ph.D. from the University of Chinese Academy of Sciences, China. She is now an engineer at the China Unicom Research Institute with specialist research in cloud computing, container, the future network and computer system structure. She received the Outstanding Student Award from the University of Chinese Academy of Sciences.

ZHANG Yunyong (zhangyy@chinaunicom.cn) graduated from the postdoctoral program of University of Electronic Science and Technology, China. He is the head of the China Unicom Research Institute and the director of the Technical Committee of the Institute. His research interests include next-generation open networks, fixed mobile convergence core network, mobile Internet and services, and public computing. He won the MIIT Excellent ITU Contribution Award twice, Outstanding Individual Award twice, one international award, and 15 provincial and ministerial awards.

XU Lei (xulei56@chinaunicom.cn) received his Ph.D. from Beijing Institute of Technology, China. He is now a senior engineer and the cloud management research group leader with the China Unicom Research Institute. His research interests include cloud computing, SDN/NFV, and the future network. He won the Top-Notch Talent Award from the China Unicom Research Institute.

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

(Nanjing R&D Center, ZTE Corporation, Nanjing 210021, China)

Abstract

Cloud computing faces a series of challenges, such as insufficient bandwidth, unsatisfactory real-time, privacy protection, and energy consumption. To overcome the challenges, edge computing emerges. Edge computing refers to a process where the open platform that converges the core capabilities of networks, computing, storage, and applications provides intelligent services at the network edge near the source of the objects or data to meet the critical requirements for agile connection, real-time services, data optimization, application intelligence, security and privacy protection of industry digitization. Edge computing consists of three elements: edge, computing, and intelligence. Edge computing and the Internet of Things (IoT) mutually create, and edge computing and cloud computing complement each other. In the architecture of edge computing, resources are distributed to the edge nodes, and therefore the storage system is near users while the computation function is near data. In this way, the stress on the backbone network can be lessened. With this architecture, the existing key technologies for computation, networks, and storage will change significantly. ZTE's edge computing solutions can ensure the service quality of operators and greatly enhance the experience of mobile users.

Keywords

edge computing; cloud computing; IoT

The era of the Internet of Things and big data is coming and many industries are moving toward digital transformation. Storage and computing are increasingly dependent on the cloud, which makes the cloud become a hub of constructing the information society. However, the cloud cannot solve all the problems. The network edge side of intelligent Internet faces the following challenges:

- 1) Insufficient bandwidth for the connection and analysis of massive data: The explosive growth of IoT terminals and data densely occupies the bandwidth at the network edge, and the continuous development of the communication technologies and the increasingly rapid growth of network construction still cannot afford the massive transmission caused by the connection between man and things, man and man, or things and things.
- 2) Steady increase of the demand of real-time experience: Many data streams are generated by edge equipment. However, it is difficult to make real-time decisions through "remote" cloud computing and analysis. For example, the response time of the visual service of a wearable camera must be within 50 ms, which cannot be met if cloud computing is used because the delay is quite long.
- 3) Security and privacy protection: Security is a basic requirement for both cloud computing and edge computing and therefore end-to-end protection is required. It is difficult to

greatly and extensively improve the access control and threat protection of the network edge because the network edge is closer to the IoT devices.

- 4) Constant energy consumption: As more and more applications are transferred to the cloud, energy demands are growing. Therefore, the computing strategy for maximizing the energy efficiency becomes a particularly urgent demand. It is not necessary to transmit each piece of original data to the cloud during the collection and processing of basic information about some embedded small devices, saving a lot of energy costs.

To overcome the network-edge challenges to the cloud, the industry and the academia attempt to build a new architecture called edge computing.

1 Definition of Edge Computing

Edge computing [1] refers to a process where the open platform that converges the core capabilities of networks, computing, storage, and applications provides intelligent services at the network edge near the source of the objects or data to meet the critical requirements for agile connection, real-time services, data optimization, application intelligence, security and privacy protection of industry digitization.

To implement edge computing, the following three elements

are required: edge, computing, and intelligence [2].

- 1) Edge: For cloud computing, all data is gathered to the back-end data center for processing, which is a cloud-focused process. For edge computing, however, the focus is the physical area of the “edge” (or “end”). It is obvious that the timeliness requirement of the real-time services is more likely to be satisfied if the network, computation, and storage resources can be provided nearby for the “edge”.
- 2) Computation: Computation is of great importance during edge computing. Because the computation is implemented on the edge, the bottleneck problem concentrating the computing power to the cloud can be avoided.
- 3) Intelligence: Edge computing covers the operation technology (OT), information technology (IT), and communication technology (CT) fields, and involves network connection, data collection, sensing, chips, and industry applications. The autonomous management and orderly collaboration of various heterogeneous platforms, and the collection, optimization, and unified presentation of heterogeneous data can be implemented only through edge intelligence. This is another difference from cloud computing.

In summary, compared with the large and comprehensive functions implemented by cloud computing, those implemented by edge computing are said to be smaller and more beautiful. Based on the data source, edge computing is complementary with cloud computing applications in a real-time and quick way.

2 Value of Edge Computing

2.1 Mutual Creation of Edge Computing and IoT

Edge computing makes the IoT smarter. The core of the IoT is to implement smart connection and operation [3] of each object while that of edge computing is to implement the sensing, interaction, and control between objects through data collection and analysis. Edge computing is the key to the application of the IoT and can implement the comprehensive perception, seamless interconnection, and high intelligence of the smart terminals in the autonomous areas.

Edge computing is one of the key technologies used in the IoT and big data. The essential changes occurring during the evolution from the Internet, mobile Internet, to IoT are the amount of data: the rapid increase of the number of devices connected to the Internet and the data generated per unit. With the mass deployment of IoT terminals, the data generated by the IoT will experience an exponential growth. Data filtering and intelligent analysis on the network edge through scattered terminals and IoT is an important development direction of IoT applications.

Edge computing can enhance the efficiency and security of the IoT. Edge computing is already successfully used in efficient energy and intelligent manufacturing. After the intelli-

gent transformation based on the IoT technologies, the data analysis, processing, and monitoring on the network edge can improve both the efficiency and the response speed. In addition, data gradually becomes an important asset of enterprises. In the case of network or data center failures, the intelligent network edge can ensure data security to help enterprises avoid risks.

2.2 Edge Computing and Cloud Computing Complement Each Other

Not all computations need the cloud. Cloud computing is a basic technology, and almost all data must be connected to the cloud for storage, computation, and analysis. However, cloud computing is not the only solution. The types of computation typically not suitable for the cloud are as follows:

- 1) Time-delay sensitive computation: With traditional cloud computing, after you click a button, the back-end system makes a computation and then responds with the result. However, some services with real-time responses (such as online videos, augmented reality, and virtual reality) have high requirements for the delay, caching, and security. If the service processing completely depends on the heavy-weight cloud computing far away from customers, bottlenecks are inevitable.
- 2) Low value density: In a traditional cloud computing model, terminals can only collect data. All big data is analyzed and processed after being transmitted to the cloud through the network, which puts a huge interaction pressure on the network. If there is only a small amount of valuable data, the analysis and processing of data are a waste of bandwidth. Therefore, it is reasonable to transmit only valuable data.
- 3) Emergency power outage: In the case of power outage, the advantages of the autonomous system of edge computing can be developed. In the traditional cloud computing architecture, a terminal can hardly cope with a disaster because it totally depends on the cloud for data processing.

Moreover, the advantages of edge computing and cloud computing can be combined, and the cloud and fog can collaborate with each other.

The processing and storage of massive data must rely on a solid cloud platform. Focusing on the big data analysis of non real-time and long-period data, cloud computing can take advantage of “logic concentration of resources” in periodic maintenance and service decision-making support. Compared with cloud computing, edge computing is secure, fast, easy to manage, and more applicable to the intelligent processing and execution of local services in real time.

The two solutions can be highly collaborative based on the complementary advantages. On the one hand, close to execution units, edge computing can act as the collector of the high-value data required by the cloud to better support the big data analysis of cloud applications. On the other hand, the service rules output through big-data analysis and optimization of

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

cloud computing can be delivered to the edge for the optimization and processing of service execution by edge computing based on new service rules. The deployment of lightweight edge computing on the edge of networks will undoubtedly reduce the burden of the upper cloud computing center.

In most cases, edge computing needs to interact with cloud computing (**Fig. 1**) [4]. Analysis on the edge combines in-depth analysis on the cloud.

2.3 Edge Computing and Cloud Computing Jointly Promote the Collaboration of Cloud and Fog

The concept of fog computing [5] was put forward by Cisco in 2011. The fog mainly uses the equipment on the network edge. The equipment can be any traditional network device, such as routers, switches, and gateways, or the local servers that are specially deployed. In general, the deployment of special equipment needs more resources, while the sharing of abundant traditional network equipment can greatly reduce the cost. The resource capacity of a single specially-deployed device and a single traditional network device is far less than that of a data center; however, a huge number of the two devices can produce enormous resource capacity. A fog platform consists of a vast number of fog nodes. In contrast to the data center where resources are gathered, the fog nodes can be geographically distributed in different locations.

Ginny Nichols, who proposed the concept of fog computing, has an interesting view: the fog is the cloud close to the ground. In other words, the common point of fog computing and cloud computing is that they provide the resources from the shared resource pool for multiple users based on the virtu-

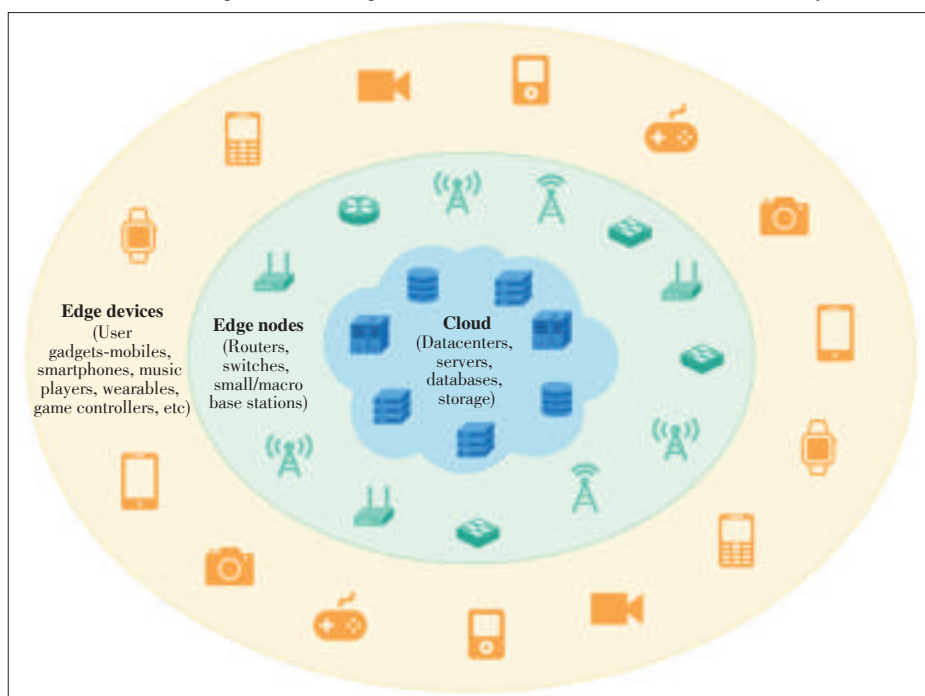
alization technology. The difference between fog computing and cloud computing is their positions in the network topology.

However, edge computing is essentially different from fog computing and cloud computing because it does not target resource virtualization. Edge computing is a solution to processing real-time big data and aims to reduce the great pressure on the data center that processes massive data alone in the Internet of Everything (IoE) era, the advanced stage of the IoT.

Sharing similar ideas, both edge computing and fog computing provide computation near the on-site applications. In terms of the nature, both edge computing and fog computing are the counterpart of cloud computing and will be definitely integrated into the architecture of “collaborative cloud-fog development.”

3 Key Technologies of Edge Computing

In the architecture of cloud computing, the data center is the controller of resources and users apply for resources on demand. User data is uploaded to the data center for computation and processing and the data center finally sends a feedback of the processing result to the user through the network. However, in the architecture of edge computing, users do not need to entirely rely on the data center because resources are distributed to the edge nodes. Therefore, the storage system is near users while the computation function is near data, meeting the real-time requirements of the network and effectively using the computing resources. With this architecture, the existing key technologies for computation, networks, and storage will change significantly.



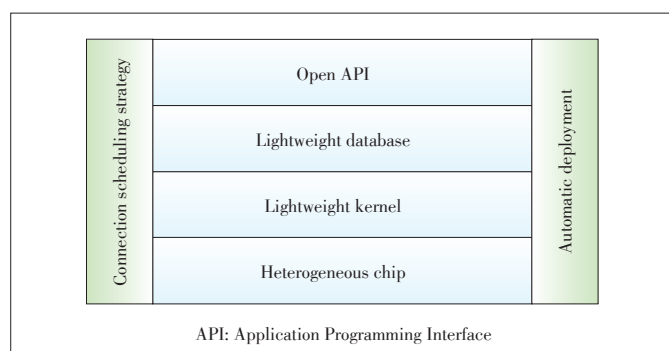
▲ Figure 1. Edge computing architecture.

3.1 Key Computing Technologies

In the resource-gathered cloud computing architecture, the computing model is relatively mature, and the computing resources are abundant. In the edge computing architecture, however, the edge nodes and the technology capabilities vary significantly, and some key technical challenges (**Fig. 2**) need to be overcome, such as intelligent connection scheduling, parallel processing, and automatic deployment.

1) Intelligent connection scheduling policy

For edge computing, the biggest difficulty lies in how to dynamically deploy the computation and storage capacity on a large scale, and how to implement efficient coordination and seamless connection between the cloud and equipment [6]. The continuous development of distributed computation leads to many tech-



▲ Figure 2. Key computing technologies.

nologies to drive split task execution in many geographic locations. Task splitting is usually clearly specified in the programming languages or management tools. However, the edge node and splitting-based computing poses some challenges, such as how to effectively split computing tasks, and how to automatically calculate the capabilities and locations of edge nodes without clearly specifying. Therefore, a new scheduling method is required to deploy the split tasks to every edge node. An ideal strategy is that the edge side can fully intelligently determine when to use edge nodes and which data is processed with edge nodes.

Another challenge is how to ensure that the edge nodes can still reliably operate when making additional computation. For example, if a base station is overloaded, the edge devices connected to the base station may be affected. Therefore, the operation of the edge nodes needs to be intelligently sensed and controlled and the tasks need to be flexibly split and scheduled. How to integrate and separate a complex algorithm between the cloud and edge devices can be realized only with the technologies that can simultaneously control the cloud, channels, and devices.

2) Computation of heterogeneous nodes

“Heterogeneity” is the most distinctive characteristic of massive edge devices and the edge devices manufactured by most of the vendors support general-purpose computation through software solutions. With the increase of the number of edge nodes supporting general-purpose computation, the demand for the development frameworks and toolkits will also rise. The programming model needs to support the parallel processing of tasks and data by the edge nodes, and simultaneously makes computations on the hardware at multiple levels. The programming language needs to consider the heterogeneity of hardware and the computing power of various resources in the workflow. This is more complicated than the existing cloud computing model. The container technology is becoming mature. The mobile containers that reuse hardware across multiple virtual devices can provide the performance equal to that of local hardware and can quickly deploy applications on heterogeneous platforms. The container- or virtualization-based solutions can overcome the challenges to the deployment of ap-

plications on the heterogeneous nodes.

3) Lightweight database and kernel

Unlike large servers, edge nodes may not support large software due to hardware limitations. For example, the small Intel T3K base station with the concurrent dual-mode SoC is designed with a four-core ARM CPU and limited memory, and cannot afford complex data processing. Apache Quarks, a lightweight database, can be used on small edge devices (for example, smart phones) for real-time data analysis. However, a single Quarks cannot perform advanced analysis tasks. The lightweight database that consumes less computation and storage resources is more suitable for the application edge computing.

4) Automatic deployment and service discovery

By 2020, about 50 billion terminals and devices will be connected to the Internet. The shorter product life cycle, higher individual demand, and more obvious trend of whole-life-cycle management and service need an automatic deployment mechanism to provide a powerful technical support to deal with routine operation and maintenance, such as rapid deployment in batches, intelligent configuration, automatic troubleshooting, and service recovery. In addition, how to find resources and services in an edge computing environment is also an area for further development.

3.2 Key Storage Technologies

Focusing on storage, the existing distributed storage architecture is suitable for centralized computing systems [7]. With the in-depth development of edge computing, the computation capability is shared by the edge. Storage in the future, especially the local file storage system on the edge, will focus on computation. The transformation from the storage-centric mechanism to the computation-centric mechanism is a reverse of the design idea of the existing storage systems.

1) Data distribution

This is a basic issue that must be considered for a distributed storage system at the beginning of the design. The purpose of typical data distribution algorithms, such as Controlled Replication under Scalable Hashing (CRUSH), Distributed Hash Table (DHT), and consistent Hash, is to evenly distribute data to every node of the system after splitting and fully breaking up the data. Such data distribution can equalize the computing power of the central computing node. However, for the architecture where edge nodes are involved in computation, data should be stored where it is needed rather than in a random place. In other words, data should be stored where it is computed. The focus should be the reduction of the computation delay instead of data equalization.

2) Data consistency in a distributed environment

In a distributed scenario, data typically has multiple copies and the copies may be read and written at the same time. Therefore, the resulting data consistency becomes a long standing problem for the distributed storage system. In the edge computing architecture, data is accessed on the edge instead of

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

by other clients, so the traditional consistency guarantee mechanism is not required. A new architecture will be used in the future.

3) New requirements for storage software from new storage hardware

Edge computing is a delay-sensitive application. Especially for the Internet and embedded applications, it is a general trend in storage devices to replace the mechanical disks with flash drives. However, the design of the existing storage systems depends too much on the characteristics of disks instead of flash drives. With the continuous development of edge computing, the high-speed, energy-saving, and small flash drives will be heavily deployed on the edge nodes. Whether a single disk or a full-flash server need matched storage software, flash-oriented software storage systems will become a key technology of edge computing in the future.

3.3 Key Network Technologies

With the shifting of the storage and computing resources from the cloud data center down to the edge nodes, the network bottlenecks migrate from the backbone networks to the edge nodes. In this case, the internal and external interaction on the servers greatly increases, which imposes a demanding requirement that cannot be met by traditional TCP/IP technologies. To respond to this challenge, InfiniBand (IB), Remote Direct Memory Access (RDMA), and Data Plane Development Kit (DPDK) become key network acceleration technologies of edge computing.

- 1) With RDMA, data is directly transferred to the storage area of a computer through the network, which means that data is quickly moved from one system to the memory of a remote system without affecting the operating system. Therefore, this technology does not have a high requirement for the processing capability of computers. It removes the text copy and exchange operations on external memories and therefore can improve the system performance because it can reduce the occupation of the memory bandwidth and CPU cycle time [8].
- 2) The InfiniBand is a cable conversion technology that supports concurrent links. With the high bandwidth, low delay, and high expandability, InfiniBand is applicable to the communications between a server and server, a server and storage device, and a server and the network.
- 3) DPDK is an application development kit provided by Intel to increase the data processing speed of data-plane packets. Supported by Userspace Input/Output (UIO), DPDK supports the drivers in the application space and the network adapter driver operates in the user space, reducing the number of duplications of the packets between the user space and application space. With the Linux affinity, DPDK binds the control plane threads and data plane threads to different CPU cores, reducing the scheduling of CPU cores among the threads. It provides memory pool and lock-free ring buffer

management to improve the memory access efficiency.

3.4 Key Security Technologies

Security is a basic requirement for both cloud computing and edge computing and therefore end-to-end protection is required. It is difficult to greatly and extensively improve the access control and threat protection of the network edge because the network edge is closer to the IoT devices. Edge security mainly involves the security of devices, networks, data, and applications. In addition, the integrity and confidentiality of key data are the focus of security.

If terminals act as shared edge nodes, the associated risks for the users and owners of the edge devices need to be defined. If the terminals act as edge computing nodes, the original functions of the devices cannot be damaged. For the multiple users on the edge nodes, security is the first priority. Therefore, the minimum service level should be guaranteed for the users on the edge nodes.

4 ZTE Edge Computing Solution

With the rapid evolution of mobile data networks from 2G to 4G and the wide spread of mobile smart terminals, man enters a booming mobile Internet era (**Fig. 3**). Traffic rise and abundant new services place an increasing pressure on the networks and services of the operators. From the perspective of both the revenues and user development, the mobile operators are more concerned about user experience while content providers want the operators to provide a more intelligent channel, which creates a good opportunity for the development of edge computing. ZTE's 5G Mobile Edge Computing (MEC) solution and MEC content delivery network (CDN) solution can bring superior service experience for users and safeguard the operators' networks. The two solutions were put into commercial use in some offices at home and abroad.

4.1 5G MEC Solution

The application of edge computing in mobile communications is called mobile edge computing [9]. MEC can provide IT services and cloud computing functions for telecom users through the nearby wireless access networks, creating a carrier-class service environment with a high bandwidth, low delay, and high performance, accelerating the download of the contents, services, and applications in the networks, and let users enjoy uninterrupted and quality network access.

As the leader and initiator of MEC, ZTE actively participates in and leads the establishment of the standards and specifications for MEC in ETSI and 3GPP. Based on the cloud architecture, the 5G MEC solution integrates computation, storage, networks, and hardware resources (for example, special functional units) into a natural, flexibly-scheduled, and easily-extended resource pool, implements the componentization of software units and Application Programming Interface (API)

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

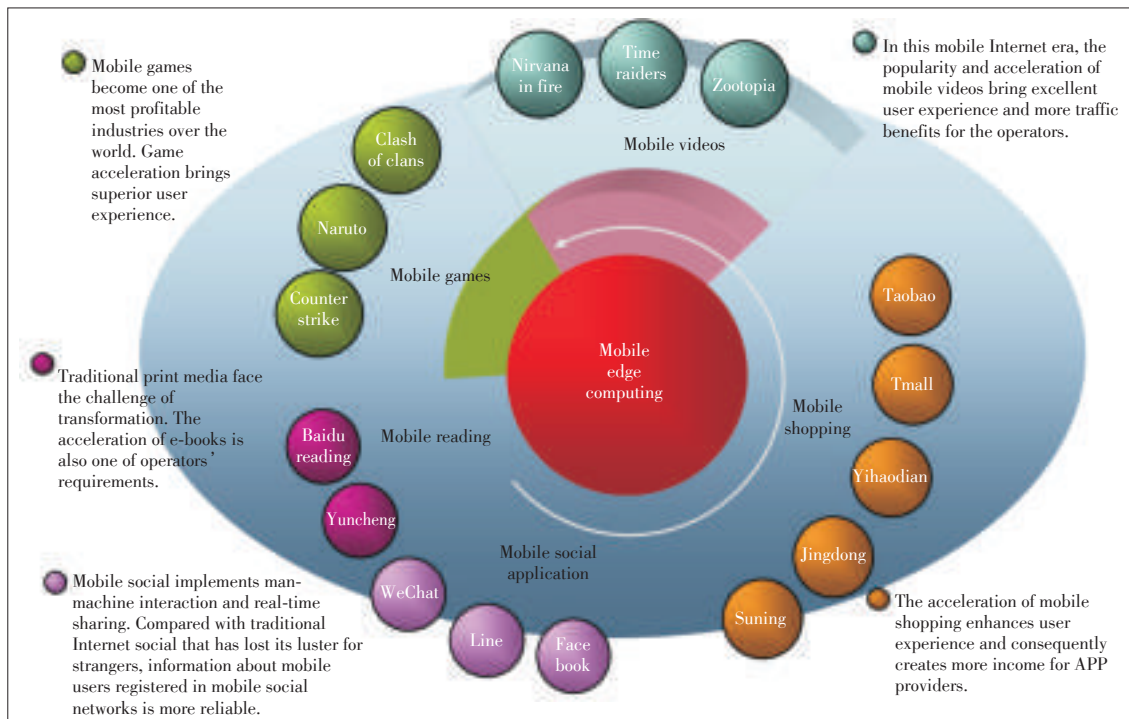


Figure 3.
Service application
requirements on mobile
networks.

based capability opening units on the central Information And Communications Technology-Platform as a Service (ICT-PaaS) platform [10], and realizes integrated deployment, monitoring, management, and operation. Based on the ICT-PaaS platform architecture, the edge data center, region data center, and core data center can be deployed at multiple levels through component collocation, implementing multi-level distributed cloud and fog deployment. Each component of the Virtualized Network Function (VNF) can be flexibly deployed in different data centers based on their service processing features, resource requirements, and user experience. Through component sharing, the common function modules of different VNFs are extracted as common components that are shared among various service flows, greatly improving the processing efficiency of the whole network. After the componentization, operators can flexibly combine independent virtual networks according to the capacity and function requirements of individual users, home users, and enterprise users, and automatically slice networks based on traffic volume (elastic scaling) [11]. In general, the 5G MEC solution realizes the convergence and development of network slicing, software-defined networking (SDN)/network function virtualization (NFV), cloud computing, and virtualization. It can meet the requirements for on-demand network slicing, quick service chains, dynamic orchestration, fast introduction of new features, maximum level of software and hardware decoupling, and strict separation between forwarding from control, finally implementing end-to-end service delivery and full-range network-wide service delivery.

The 5G MEC solution deeply integrates wireless networks and the Internet. The MEC nodes are deployed near the radio

access network (RAN) side and provide resident services, such as the LBS service and RAN cache service, meeting the strict requirements of the online video, augmented reality (AR), and virtual reality (VR) services in the era of big data for caching, delay, policy control, and security, saving bandwidth, and improving user experience [12].

Independent of the radio link technologies, the 5G MEC solution can also be implemented and deployed in 4G and 4.5G networks. MEC devices are loosely coupled with the existing 4G network devices in terms of the standards. In addition to customized applications, the standard use cases do not affect the network quality and versions of the existing 4G networks.

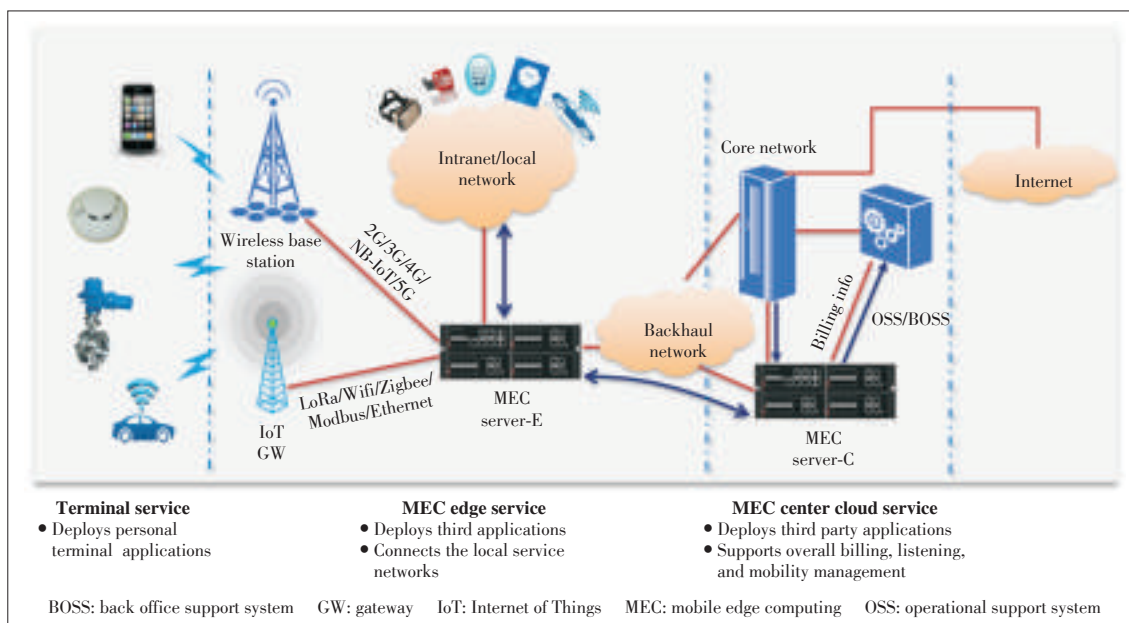
The 5G MEC solution (Fig. 4) provides an open platform that allows third parties to rapidly develop and deploy innovative applications and services, promoting the sound development of the MEC application ecosystem, facilitating the innovation of mobile edge service interfaces, and providing abundant applications.

The 5G MEC solution provides third-party applications with the capability engines for the integrated positioning, video, IoT services to accelerate the development and deployment of the third-party applications and improve user experience. It has the following characteristics:

- 1) **Positioning and tracing:** The base stations can provide the positioning function accurate to within five meters. This is a specific function of mobile networks. This function can help businesses to push advertisements for nearby business areas and stores to mobile users, bring operators and service providers backward advertising revenues. It can also be used for football match analysis and campus management.

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang



◀ **Figure 4.**
5G MEC solution.

- 2) Video analysis: The MEC server side implements image management (camera management and image capture) and image analysis. The users of the information are independent of MEC and can be a smart city, retailer, advertisement, and related APP or server. They can be public places such as airports and railway stations for managing lost luggage. They can also be the security personnel of public places for locating missing children and old men. Video analysis can also be used in other scenarios, such as high-speed monitoring and building security, to help the construction of smart cities.
- 3) Optimization of wireless network perception: Apps are deployed on the MEC server side for wireless network perception and content optimization servers are deployed in the core network. The apps transmit the information about user IDs, cell load, and link quality to the content optimization servers to help the servers to dynamically optimize contents based the cell load and link quality, improve the QoE and network efficiency, shorten the time delay, and provide optimal user experience.

4.2 MEC CDN Solutions

The CDN is used to clear the Internet bandwidth bottleneck. The basic idea of the CDN is to store the hot-spot content on the server near the access side of the network. When users access the hot-spot content, they no longer access the servers on backbone side, reducing the demand for the traffic from the backbone network and improving the QoS. Through the intelligent virtual network on the existing Internet formed by server nodes, the CDN system can re-direct the user requests to the nearest server node in real time based on the network traffic, node connections, load on the nodes, the distance to the users, and the response time to relieve network congestion and im-

prove the response to users [13].

Both the CDN and edge computing focus on the edge side. The difference lies in that the CDN focuses on storage while edge computing on computation based on storage. Therefore, it is extremely advantageous to introduce edge computing to the existing CDN service.

With the development of mobile Internet and HD videos, the content, users, and service scope of the CDN are greatly expanded. How to quickly and accurately delivering the content to the users who really need it becomes a bottleneck to the development of new services. Internet CDN operators (for example, ChinaCache) and traditional telecom equipment manufacturers (such as NSN and Ericsson) have started the technology and algorithm researches on mobile network content delivery, and piloted their research results in the positioning and advertising services. China Net Center puts forward the concept of community cloud that integrates edge computing with the existing CDN platform [14]. The MEC CDN solution of ZTE can overcome the traditional CDN difficulties and satisfy the needs for rapid development of new services.

Based on the hyper-converged infrastructure (HCI) and the introduction of edge computing to CDN nodes, the ZTE MEC CDN solution (**Fig. 5**) can change the storage nodes to edge computing nodes without massive node creation, implementing all-around flexible deployment and interconnection, centralized resource cloudification, nearby service deployment, and collaboration between edge computing and cloud computing. Through standard interfaces and service procedures, the solution focuses on the improvement in content introduction, storage, and delivery.

The MEC CDN solution supports the intelligent scheduling of application perception. For a traditional CDN edge server, video storage and deletion are passive and depend on the push

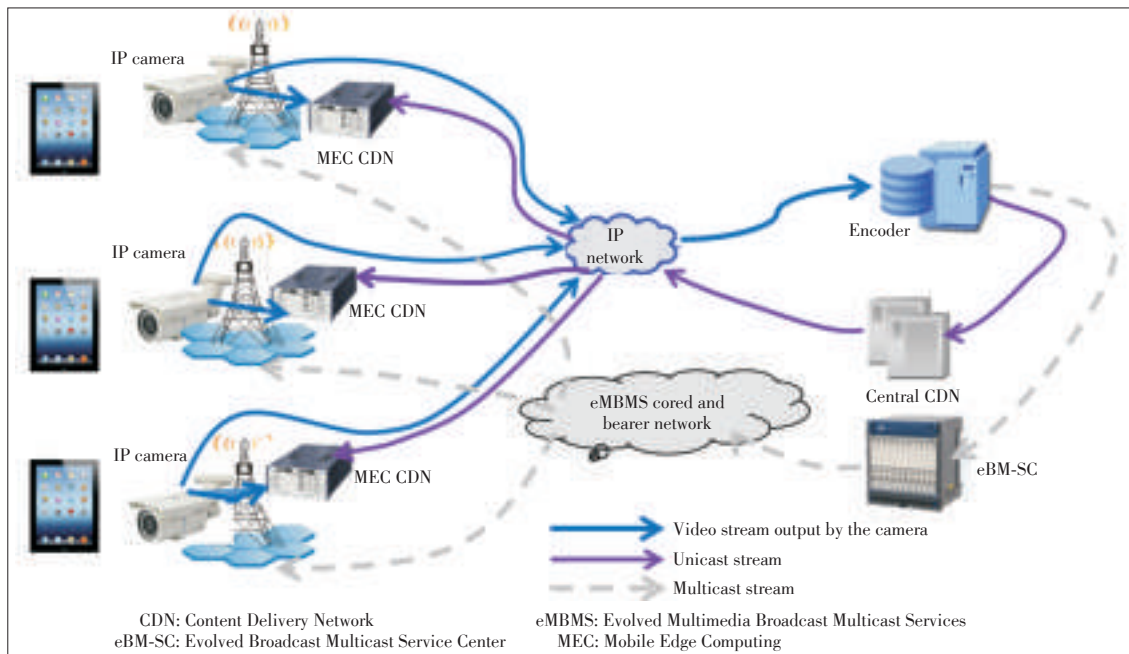


Figure 5.
MEC CDN Solution.

of the core data center. An MEC CDN edge node can monitor traffic in real time. When finding an access preference or hot spot in the local area, it actively obtains data from the core data center. It can also schedule services based on applications and requirements to provide users with optimal services. An MEC CDN edge node has some network management functions, such as node self-healing and intelligent load balancing, which can effectively prevent hot-spot bottlenecks, enhance system reliability, and ensure to provide users with quality and stable services.

Users access the MEC CDN through intelligent terminals, resulting in the requirements for the mobility of the content. Due to the limitation of network resources, the movement speed of the content does not match the mobility of users in real time; namely, the content cannot be moved in real time after the location of a user changes. The MEC CDN can set up a behavior track model of users, forecast and analyze the behavior patterns and tracks of users to improve the accuracy of the prediction about hot-spot locations and the hot content at the hot-spot locations. The delivery of content in advance to the next location or the hot-spot location of users can solve the real-time matching of movement between users and the content. The prediction about content placement can reduce the delay in user's access to the content and prevent too much resource occupation during the content delivery request to implement network load balancing and avoid network congestion.

The MEC CDN solution realizes integrated multi-service bearing and supports the unified access of various terminals. The wireless resources allocated to each user over the radio link vary in real time with the changes in the type and number of terminals. In addition, the bandwidth for mobile terminals is directly affected by the signal strength for radio network ac-

cess. Signal instability can lead to imperfect video playing and slow response when users watch videos. With the stream auto-adaptation technology, the MEC CDN solution dynamically detects the changes in user resources over the radio link, encapsulates transcoding in real time, and adjusts the format and code rate of the transmitted content in a timely manner to fully ensure smooth video playing and enhance user experience. For the same content, the MEC CDN system keeps only one copy of the bit rate file. The system converts the format and bit rate in real time based on the bandwidth and the type of terminals to ensure effective and continuous playing of videos, lowering the requirements for the storage space of the MEC CDN system and the bandwidth requirements for content transmission.

The MEC CDN solution realizes the combination of cloud technologies and mobile network technologies and supports distributed content delivery and DNS caching. With a top-down construction mode, the caches are gradually deployed to the base station side from the core network side, and further to the terminals based on apps. In this way, the backhaul network bandwidth can be reduced by about 35%. The system can provide better QoS and service experience and supports the content delivery of some big video services, for example, AR. When a mobile user who is downloading files or watching videos is handed over between base stations, the user identification session storage technology can ensure continuous file download and video playing without an interruption even though a handover occurs for base station access. In this way, the effect of the location change in the mobile network can be completely avoided.

The traditional TCP mechanism is used in the scenarios with a low error rate, such as wired networks and data packet networks. Assume that packet loss is caused by network con-

Key Technologies and Application of Edge Computing

TU Yaofeng, DONG Zhenjiang, and YANG Hongzhang

gestion and data packets are retransmitted. If the network environment is adverse, the retransmission further aggravates the condition. However, a wireless network is not suitable for the traditional TCP transmission mechanism due to its high bit error rate, low transmission bandwidth, and mobility. The MEC CDN solution optimizes TCP transmission for wireless networks. With the Westwood algorithm, the MEC CDN solution determines the threshold for slow start and the size of the congestion window based on the effective bandwidth in the case of congestion (estimated through the receiving rate of ACK packets), avoiding the effect of random packet loss on the bandwidth utilization. The Westwood optimization algorithm is more suitable for the characteristics of wireless networks and can make full use of the transmission bandwidth.

5 Conclusions

Edge computing is essentially "lightweight cloud computing." The edge computing solutions of ZTE implements a unified architecture through component collocation, multi-level distributed cloud and fog deployment, on-demand network slicing, quick service chains, dynamic orchestration, fast introduction of new features, and end-to-end fast service delivery.

Edge computing is a new concept and the Edge Computing Consortium [15] has been just formed. The rapid development of edge computing in the future depends on the participation of the academia, overseas enterprises, and Internet enterprises. As more members join the Edge Computing Consortium, unified industry norms and standards must be developed in the future to jointly and orderly promote the development of edge computing.

References

- [1] P. G. Lopez, A. Montresor, D. Epema, et al., "Edge-centric computing: vision and challenges," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 37–42, Oct. 2015. doi: 10.1145/2831347.2831354.
- [2] T. Guo, "Solution to edge computing for digital transformation," *China Information World*, no. 45, p. 3, Nov. 28, 2016.
- [3] Y. Ye, "Future applications and solutions to the IoT," *ZTE Technology Journal*, vol. 18, no. 2, pp. 15–17, Apr. 2012.
- [4] Iot101. (2017, Jan. 12). *Edge of computing will be popular in 2017, to count the value, the opportunities and challenges* [Online]. Available: <http://www.10101.com/kpwl/2017-01-12/12755.html>
- [5] Y. Shi, "Software architecture analysis and application research on the fog computing platform," M.S. thesis, Zhejiang Normal University, Hangzhou, China, 2016.
- [6] Iotworld. (2017, Jan. 19). *Value, opportunities, and challenges of edge computing that will be popular in 2017* [Online]. Available: <http://www.10101.com.cn/html/News/201701/8122ed7f43e02c1f.shtml>
- [7] L. Shen, Q. Wu, and S. Yang, "Research on distributed storage energy saving technology on ceph," *Computer Engineering*, vol. 41, no. 8, pp. 13–17, Aug. 2015.
- [8] S.-G. Wang, W.-X. Xu, Wu Dan, Z.-B. Pang, and J. Xia, "A fast RDMA offload method for unreliable interconnection networks," *Journal of Hunan University (Natural Sciences)*, vol. 42, no. 8, pp. 100–107, Aug. 2015.
- [9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief. (2017, Jan. 11). *Mobile edge computing: survey and research outlook* [Online]. Available: <https://arxiv.org/pdf/1701.01090.pdf>
- [10] Y. Wu and L. Yang, "Introduction and value exploration on NFV PaaS," *ZTE Technologies (Chinese Version)*, no. 2, pp. 20–22, Feb. 2016.
- [11] D. Liu. (2016, Apr. 20). *ZTE's NFV path: "three trends" based network transition from IaaS to PaaS* [Online]. Available: <http://www.c114.net/news/127/a950179.html>
- [12] Y. Lu. (2016, May 12). *ZTE 5G MEC test on the way* [Online]. Available: <http://www.cww.net.cn/zhuanli/html/2016/5/12/20165121547158041.htm>
- [13] J. Liang, G. Chen, and Y. Zhuang, *Content Delivery Network (CDN) Key Technologies, Architecture, and Applications*, Beijing, China: Posts & Telecom Press, 2013.
- [14] W. Shu, "ChinaNetCenter: community cloud based on the convergence of edge computing and CDN," *Communications World*, no. 33, pp. 37–38, Dec. 2016.
- [15] H. Huang, "Influence of the establishment of the edge computing consortium," *Communications World*, no. 33, pp. 52–53, Dec. 2016.

Manuscript received: 2017-02-10

Biographies

TU Yaofeng (tu.yaofeng@zte.com.cn) received his master's degree from Nanjing University of Aeronautics and Astronautics, China. He is a deputy director of the Cloud Computing & IT Institute of ZTE Corporation. His research interests include telecommunication services, cloud computing products, big data, and artificial intelligence.

Dong Zhenjiang (dong.zhenjiang@zte.com.cn) received his master's degree from Harbin Institute of Technology, China. He is a deputy director of the Cloud Computing & IT Institute of ZTE Corporation. His research interests include cloud computing, artificial intelligence and media processing, and mobile Internet and security.

YANG Hongzhang (yang.hongzhang@zte.com.cn) received his master's degree in computing technology from Chinese Academy of Sciences, China. He is a technology pre-research engineer with the Cloud Computing & IT Institute of ZTE Corporation. His research interests include cloud computing, big data, and distributed storage system.

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

Panagiotis Oikonomou¹, Maria G. Koziri¹, Nikos Tziritas², Thanasis Loukopoulos¹, and XU Cheng-Zhong²

(1. University of Thessaly, Lamia 35100, Greece;

2. Research Center for Cloud Computing, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China)

Abstract

Efficient video delivery involves the transcoding of the original sequence into various resolutions, bitrates and standards, in order to match viewers' capabilities. Since video coding and transcoding are computationally demanding, performing a portion of these tasks at the network edges promises to decrease both the workload and network traffic towards the data centers of media providers. Motivated by the increasing popularity of live casting on social media platforms, in this paper we focus on the case of live video transcoding. Specifically, we investigate scheduling heuristics that decide on which jobs should be assigned to an edge mini-datacenter and which to a backend datacenter. Through simulation experiments with different QoS requirements we conclude on the best alternative.

Keywords

video transcoding; edge computing; scheduling; heuristics; x264

1 Introduction

Modern applications built on top of an integrated Internet of Things (IoT) environment [1], together with Cyber Physical Systems (CPSs) [2], involve heavy video traffic, e.g., in smart vehicle traffic management. At the same time, the proliferation of smart mobile devices carrying cameras of continuously higher resolution, together with the explosive growth in the popularity of social media platforms, poses great challenges in cloud resource management. As an indication, Cisco reported in [3] that during 2015, mobile Internet traffic experienced a growth of 74%, the majority of which (>50%) was video transmissions. Therefore, minimizing video related network traffic becomes of paramount importance.

Video coding is the process of compressing a raw video sequence using some standards. Examples of such standards are H.264/AVC [4] which is the most popular (but aging) standard currently in use, High Efficiency Video Coding (HEVC) [5] and VP9 [6], which are newer standards achieving higher compression ratios compared to H.264/AVC. Although video coding is a computationally demanding task, it is usually performed at the point where the initial video is captured (camera, smart device etc.), often with the aid of specialized hardware. Thus, the initial coding of a video sequence does not hinder a cloud based social media platform (SMP) computationally wise

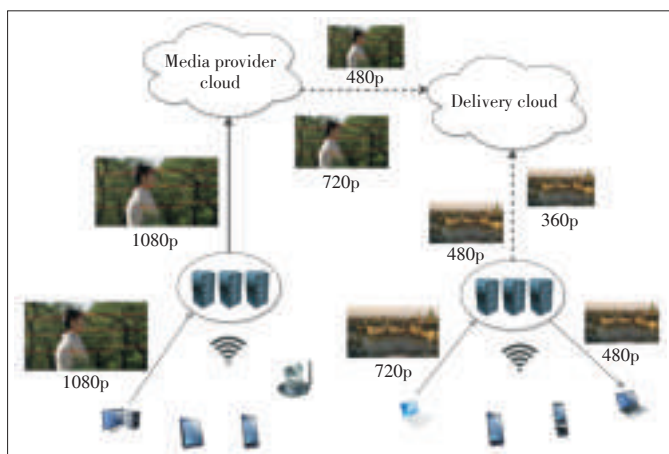
and the only overhead is the consumed bandwidth for uploading. However, in order to be able to deliver the video sequence to a variety of clients differing in screen resolutions, decoders and network capabilities, the originally uploaded sequence must be encoded into multiple output sequences of various resolutions, bitrates, quality levels and perhaps coding standards. This process is called transcoding and burdens computationally and network-wise the SMP's cloud. In particular, the case of live casting offers the most challenges since real time performance is a requirement.

Motivated by the above, we investigate the case where an SMP can take advantage of mini-data centers existing at network edges in order to offload live transcoding jobs, thus, saving resources and bandwidth. **Fig. 1** illustrates an example whereby two broadcasts are performed, one at 1080p and the other at 720p from two different edges. In the first case (1080p), the sequence is not transcoded at the edge but transmitted to one of the SMP's data centers for processing. Then two different outputs (720p and 480p) are sent to some Content Delivery Network (CDN). In contrast to this, the other input sequence (720p) is transcoded into two output sequences at the edge. Copies of the outputs are sent to the CDN and also used to satisfy local demands (480p). Clearly, the second alternative of using edge transcoding reduces both the processing and network resource consumption at the SMP's cloud.

In this paper, we tackle the associated scheduling problem

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritis, Thanasis Loukopoulos, and XU Cheng-Zhong



▲ Figure 1. Example system model with edge transcoding.

induced by the scenario of Fig. 1. Namely, given edge resources and the characteristics of arriving transcoding tasks, task-server assignment must be made so that the percentage of tasks not processed by the edge (satisfied with overhead by SMP's Cloud) is minimized. We evaluate different scheduling heuristics for the scheduling problem under the constraint that each assigned task must obtain the required processing power to exhibit real time behavior. We then examine the case where the aforementioned constraint is softened, allowing for some quality loss in order to increase the number of tasks assigned to the edge. All heuristics are evaluated using a dataset of Twitch broadcasts [7] and realistic values for transcoding job characteristics obtained by using x264 codec [8] over class B and class A common test video sequences [9].

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents the problem formulation. Heuristics are illustrated in Section 4 and evaluated in Section 5. Finally Section 6 concludes the paper.

2 Related Work

The number of transcoding tasks hosted by edges is dictated by their processing requirements. Related to this requirement is research concerning speeding up of video coding and transcoding. An avid research exists on parallelizing video coding with approaches varying from coarse grained parallelism, whereby parallelism is considered at the level of group of Macroblocks (H.264/AVC) or Coding Tree Units (CTUs in HEVC), to finer grained parallel approaches implementable within a block of pels. Examples of coarse grained parallelization include slices, tiles and wavefront in the HEVC standard. Efficient implementation of these parallel options are described in [10] for slices, [11] for tiles and [12] for wavefront. Fine grained techniques usually consist of applying the Single Instruction Multiple Data (SIMD) paradigm at various levels of the encoding [13] and decoding stages [14].

As far as transcoding is concerned, a straightforward method

is to first decode fully the input sequence, scale its resolution and then re-encode it. More efficient approaches target at utilizing the information already coded in the input, most noticeably the one concerning motion estimation, in order to reduce the search space when transcoding to another standard. Example works in the area include [15] where an H.264/AVC to HEVC transcoding architecture is presented that achieves a nominal speedup reaching 8x, when compared to re-encoding from scratch. If a bitrate change rather than a change in standard is needed, the process is often referred to as transrating. A survey on fast transrating methods can be found in [16]. In the experiments we obtained transcoding task weights by using the straightforward approach of re-encoding without using the information already coded. This was done both for reasons of simplicity and due to code availability (ffmpeg and x264 used). However, based on the aforementioned research we scaled the values obtained to depict the case where a more efficient transcoder is used.

Concerning cloud transcoding, most works focused on providing job scheduling techniques at the level of a server cluster or a data center. In [7], the authors considered the case of live video transcoding and proposed an integer linear program (ILP) formulation to tackle scheduling decisions. An online algorithm that schedules jobs among the servers of a datacenter with the target of satisfying delay requirements while using minimum energy was proposed in [17]. In [18] the scope was a single cluster and the optimization target was to keep the servers load balanced. In [19] an admission control algorithm was developed that differs or rejects requests that cannot be satisfied based on current workload. It is worth noting that this is the contrary approach to the one used in this paper for the case of edges, whereby it might be viable to reduce quality by over-assigning tasks to servers if the relevant benefits from edge processing are deemed sufficient. Finally, in [20] a combined caching and transcoding approach is discussed, whereby transcoding jobs are partially processed to allow for efficient caching. The target considered in this paper, i.e., live transcoding excludes partial transcoding as an option. Caching and replication techniques in the cloud are surveyed in [21], while [22] and [23] concern efficient video delivery.

Overall, compared to [7], [17], [18], [19] and [20], we differ in scope since we examine transcoding at edges while we view [21], [22] and [23] as orthogonal to our approach. Perhaps the closest work in the literature is [24], where system architecture for edge transcoding is described. Nevertheless, scheduling issues were not tackled in the manner done in this paper.

3 Problem Definition

We consider the case of a media provider receiving requests for live video casting, whereby the input stream must be transcoded into a set of output streams with different resolution, bitrate and quality demands. We consider two options for the set

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritas, Thanasis Loukopoulou, and XU Cheng-Zhong

of transcoding tasks associated with each input. Either they are all assigned to a mini-datacenter existing at the edge of the network or they are all assigned to the backend main datacenter of the media provider. Clearly, if the tasks are processed at the edge, the processing workload at the backend datacenter is reduced and the network overhead for transmitting the input sequence is avoided.

Let the mini-datacenter consist of S servers, with S_i denoting the i th of them, assuming a total ordering ($1 \leq i \leq S$). Each server has an associated processing capacity (let C_i), which denotes the number of baseline transcoding tasks that can be processed concurrently at real time. Baseline tasks are the ones requiring the minimum power to process. Let B_j be the j th broadcast, assuming an ordering of the B total broadcasting events ($1 \leq j \leq B$). Similarly, let s_j and d_j be the arrival time and duration of B_j , respectively. Each broadcast entails a set of transcoding tasks. Let T be the total number of transcoding tasks for all broadcasts, and T_k be the k th such task, assuming a total ordering of them ($1 \leq k \leq T$). We represent whether T_k is a task of B_j or not, using a Boolean matrix A of $B \times T$ size, whereby $A_{jk}=1$ if and only if (iff) B_j has task T_k and 0 otherwise. Moreover, W_k depicts the relevant weight of T_k in processing terms over the baseline task. Put in other terms, W_k shows how much more computationally demanding T_k is, compared to the baseline scenario. Last, let X be an $S \times T$ Boolean matrix used to encode task server assignments as follows: $X_{ik}=1$ iff T_k is assigned for processing at S_i , otherwise $X_{ik}=0$. We assume that once assigned, a task cannot be preempted and will remain for the whole duration $[s_j, \dots, s_j+d_j]$. We consider that we want to optimize the system starting from a clean state (no task assignments exist) over a time frame divided into E equally sized slots (s_j and d_j values are now measured in time slot terms). Let e_t be the t th such time slot, with a corresponding assignment matrix X^t . We typically formulate the problem as follows: Find all values in the E total matrices X^t , so that the objective function f given in (1) is maximized:

$$f = \sum_{t=1}^E \sum_{i=1}^S \sum_{k=1}^T X_{ik}^t (1 - X_{ik}^{t-1}), \quad (1)$$

subject to the following constraints:

$$\left(\sum_{i=1}^S \sum_{k=1}^T A_{jk} X_{ik}^t - \sum_{k=1}^T A_{jk} \right) \sum_{i=1}^S A_{jk} X_{ik}^t = 0, \quad \forall j, t = s_j, \quad (2)$$

$$X_{ik}^t = X_{ik}^{t+1}, \quad \forall i, k, t | s_j \leq t < s_j + d_j \wedge A_{jk} = 1, \quad (3)$$

$$\sum_{i=1}^S \sum_{k=1}^T A_{jk} X_{ik}^t = 0, \quad \forall j, t | t < s_j \vee t > s_j + d_j, \quad (4)$$

$$\sum_{k=1}^T X_{ik}^t W_k \leq C_i, \quad \forall i, t, \quad (5)$$

$$\sum_{i=1}^S X_{ik}^t \leq 1, \quad \forall k, t. \quad (6)$$

The objective function encodes the tasks that will be assigned to the edge. Eqs. (2)–(6) give the main constraints of the problem. Constraint (2) states that either all tasks of a broad-

cast B_j will be assigned to the edge at the time the broadcast arrives or none. Constraint (3) enforces that the decision taken for a transcoding task at the time of its broadcast arrival remains for the duration of the broadcast. Constraint (4) ensures that neither before a broadcast arrival, nor after its end time, can a corresponding task be scheduled for edge transcoding. Constraint (5) dictates that a server can exceed its capacity at no point in time. Finally, (6) states that a task can only be scheduled at one server.

Clearly, the fact that broadcasts are known in advance reduces the applicability of the presented problem formulation to cases of prescheduled event covering, e.g., sports. Nevertheless, the formulation provides a thorough definition of the optimization target and the related constraints. These remain the same both in the static problem variation presented and in the dynamic case. A last note concerns complexity. It can be shown that the relevant decision problem is NP-complete since the processing capacity constraint at the servers effectively introduces a (0, 1) Knapsack component. Next, we present heuristics for dynamic scheduling of transcoding tasks at the network edge.

4 Scheduling Heuristics

4.1 Scheduling with Tight Task QoS Requirements

The proposed heuristics tackle the dynamic version of the scheduling problem presented in the previous section. Specifically, upon the arrival of a broadcast request, the necessary transcoding tasks are defined. Then, they are sorted according to their weight and considered either in increasing order (MIN policy) or in decreasing (MAX policy). Each task is assigned to a server (using one of the policies described in the sequel) provided the task computational demands can be met by the server as per (5). If a suitable server is found for every transcoding task of the broadcast under consideration, the assignments are committed; otherwise, even if one task fails to find a hosting server, all the tasks are sent to the SMP's datacenter for processing. The assignment policies considered are based on the well-known bin-packing heuristics:

- Best Fit (BF): Select the server where the remaining capacity, left after task assignment, is the minimum possible.
- Worst Fit (WF): Similar to BF only that the server with the maximum remaining capacity will be selected.
- First Fit (FF): The first server where the task fits will be selected.

The corresponding heuristics are named after the order with which the task list is considered and the packing method followed. For instance MAX-BF refers to the heuristic that considers the heaviest task first and assigns it using Best Fit.

4.2 Scheduling with Relaxed Task QoS Requirements

The motivation for the relaxed QoS case is the following. As-

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritas, Thanasis Loukopoulos, and XU Cheng-Zhong

sume that all but one task of a broadcast could fit to the available servers of the edge. With strict QoS requirements, none of these tasks will be assigned. However, it might be possible to assign the remaining task to one server so that its processing capacity is exceeded by a very small margin. In practice, this means that all the tasks processed by this server will exhibit a small quality drop. For instance, if a broadcaster transmits at 30 fps (frames per second) then a 3.3% drop at the processing rate of one of its transcoding tasks means that roughly the output stream will be at 29 fps. Depending on decoder characteristics, such a drop might not even be noticeable by a human viewer. Assuming that p denotes the maximum percentage of allowable performance drop, (5) becomes:

$$\sum_{k=1}^T X_{ik} W_k \leq (1+p) C_i, \quad \forall i, t. \quad (7)$$

The heuristics first attempt to allocate all the tasks of a broadcast as per Section 4.1. In case a task does not fit, it is considered for assignment using (7) as server capacity constraint and one of the below described policies.

- **Min Quality Decrease (MQD):** Selects the server that incurs the minimum proportional capacity violation (equivalent to asking for the minimum quality penalty for its hosted tasks).
- **First Fit (FF):** The first server where the task fits as per (7) will be selected.
- **View Weighted Penalty (VWP):** Weights the quality penalty of each task by the number of its viewers. The server with the minimum aggregated weighted quality penalty value is selected.

5 Experiments

5.1 Setup

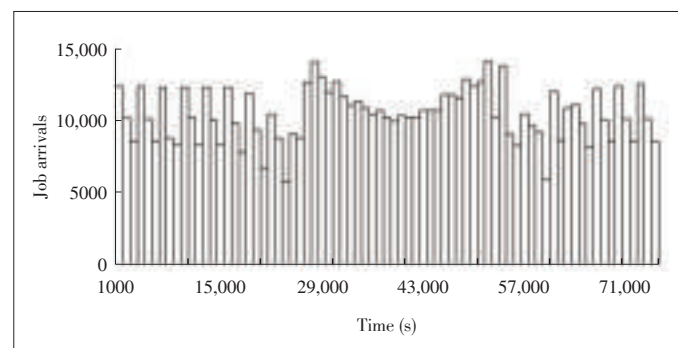
To simulate broadcasting activity, we used the same dataset from Twitch as the one described in [7]. We kept the portion of the dataset representing one day activity (Jan. 6th, 2014). We then filtered it by deleting entries with broadcasts having no viewers and the broadcasts of resolution less than 220p. To keep the simulation time manageable we considered the following 5 resolutions: 240p, 360p, 480p, 720p and 1080p. In case a broadcast in the trace did not follow one of the previously mentioned resolutions, we clustered it to its closest matching. We assumed that a broadcast must be transcoded to all the resolutions that were lower than the one it used. Clearly, with this setting the maximum number of transcoding tasks incurred by a broadcast is 4, corresponding to a 1080p stream that must be downscaled to 720p, 480p, 360p and 240p. Upscaling was not considered in the experiments. Finally, for simulation purposes we assumed that all videos used 30 fps. Furthermore, the recorded in the dataset viewing demand was split equally among the resolutions used by a broadcast, i.e., the input and all lower ones. **Table 1** summarizes some of the dataset characteristics,

while **Fig. 2** plots the broadcasting job arrival rates as a histogram of a 1000 seconds (s) step. As it can be seen, the arriving jobs do not exhibit sharp peaks (at least with the used interval), but the distribution is rather uniform. This favors job scheduling at edges since it makes sizing decisions for edges less demanding. However, duration of broadcasts does not follow a similar trend. As noted in Table 1, the difference between the average and maximum duration is two orders of magnitude, implying a heavy tailed distribution. This hinders scheduling decisions, since it means that duration estimation will be hard to achieve in the general case. For this reason, none of the scheduling heuristics described in Section 4 uses such estimates.

Next, we needed to characterize the weights of the transcoding tasks. For this reason we used class A and class B common test video sequences and transcoded them to the levels for which we wanted to obtain weight values. To do so, a sequence was first fully decoded, then scaled to the desired resolution using ffmpeg and then encoded using x264. The encoding settings followed the Peak Signal to Noise Ratio (PSNR) tailored scenario of [25], which aims at maximizing quality in PSNR terms. The exact parameters are given below (Kimono example): x264 --input-depth 8 --frames 0 --input-res 1920x1080 --fps 24 --input-csp i420 --log-level debug --tune psnr --psnr --profile high --preset placebo --keyint 96 --min-keyint 96 --me

▼ **Table 1. Dataset for broadcasters (general characteristics)**

Name	Value
Dataset duration	75,079 s
Average broadcast duration	23,263.4 s
Max broadcast duration	1.05E6 s
Number of broadcasts	786,100
Total transcoding tasks	1,244,450
Percentage of broadcasts at 1080p	26.21%
Percentage of broadcasts at 720p	53.24%
Percentage of broadcasts at 480p	11.18%
Percentage of broadcasts at 360p	7.49%
Percentage of broadcasts at 240p	1.86%



▲ **Figure 2. Histogram for broadcasting arrival rates.**

umh --merange 240 --ref 4 --partitions all --threads 1 --subme 9 --aq-mode 0 --aq-strength 0.0 --psy-rd 0.0 --output kimo-no_out.264 Kimono_in.yuv.

Table 2 summarizes the general characteristics of the test sequences, together with the coding time for each targeted resolution, measured as the number of frames per second processed by the codec. The case of 240p forms the baseline transcoding scenario, with the remaining resolutions assigned proportional weights.

Having defined the time of the baseline scenario and task weights accordingly, next we define server capacity in the following manner. In order to fully control the system environment we used a dedicated server for which we had full ownership. The server used for the x264 coding jobs carried two 6-core Intel Xeon E5-2630 CPUs running at 2.3 GHz. Since the coding speeds at Table 2 used one thread and the nominal rate considered for the simulation is 30 fps, each core of the server accounts for a processing capacity of 21.6/30 (the baseline scenario). The total server capacity is then calculated by multiplying with 12 (the total number of physical cores) and equals 8.64. Since our server setting is not of generic use, we translate it into one of the Amazon EC2 instances [26] to make our simulation setting more applicable. Specifically, we consider the C3 instances which are recommended for video coding. Comparing the processor passmark ratios between the CPU of our server and the one used in the C3 instances, i.e., Intel Xeon E5-2680 v2 (Ivy Bridge), it can be estimated that the instance c3.4xlarge will account for a speedup of 2.2x compared to our server. Since video coding is CPU-bound, the aforementioned methodology (i.e., comparing CPUs) provides a good estimation on relative performance. Last, we consider the case where specialized transcoding software is available, which accounts for a speedup of 8x (same as in [15]) compared to the simple methodology used to obtain the processing rates of Table 2.

5.2 Results for Tight QoS Requirements

Here we present results for the case where the assigned transcoding tasks at the edge must be satisfied at their nominal

▼ **Table 2.** Video sequences used for weight calculation

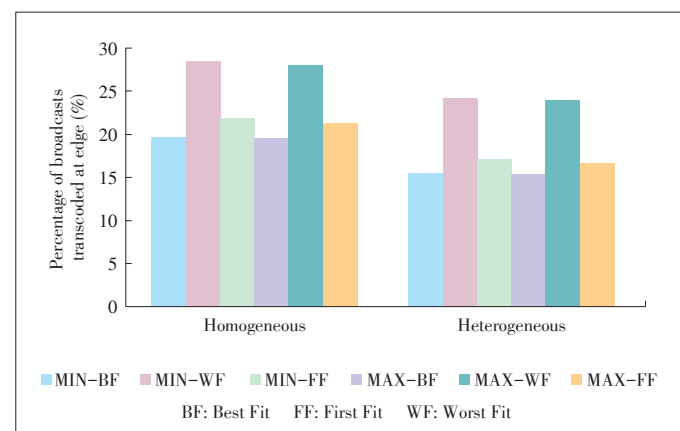
Name	Resolution	Frames	Time 240p (fps)	Time 360p (fps)	Time 480p (fps)	Time 720p (fps)
BasketballDrive	1920×1080	500	15.78	5.37	3.78	1.47
BQTerrace	1920×1080	600	29.10	9.11	6.42	2.37
Cactus	1920×1080	500	25.48	8.77	5.09	1.96
Kimono	1920×1080	240	18.54	6.05	4.33	1.62
ParkScene	1920×1080	240	24.95	7.62	5.28	1.94
PeopleOnStreet	2560×1600	150	10.01	2.82	2.00	0.74
Traffic	2560×1600	150	27.36	8.06	5.77	2.17
Average	-	-	21.60	6.82	4.66	1.75
Weights	-	-	1.00	3.16	4.63	12.34

rate (30 fps). We consider two scenarios. In the first (homogeneous), 1000 servers each of capacity described in Section 5.1 exist in the micro datacenter of the edge, while in the second scenario (heterogeneous) 500 servers have the aforementioned capacity and 500 half of it (presumably equivalent to a c3.2xlarge EC2 instance). **Fig. 3** plots the performance of the scheduling heuristics, measured in terms of the percentage of broadcasts that are assigned for edge transcoding. Results show that the same trends are exhibited both in the homogeneous and the heterogeneous cases. The later achieves lower performance since it accounts for smaller total capacity. Furthermore, sorting the transcoding tasks of a broadcast has marginal effect. This is presumably due to the fact that the tasks are scheduled as soon as a broadcast arrives and are rather small in number, compared to the available servers. Clearly the WF policy outperforms the other alternatives by a substantially large margin (an extra 5% roughly of the arriving requests can be accommodated by the edge).

Having identified WF to be the most promising heuristic, we evaluated the impact of the arrival rate on the achievable performance. To do so, we used the same trace with above, but sampled it every 1, 2 and 3 entries. Clearly, a sampling rate of 1 is equivalent to using the whole dataset (Fig. 3), while 2 and 3 effectively account for 1/2 and 1/3 of the arrival rate. **Fig. 4** plots the performance of the WF scheme for the three arrival rates and for both the homogeneous and heterogeneous cases. As expected, the percentage of jobs that can be satisfied by the edge increases as the arrival rate decreases. It is worth noting that with 1/3 arrival rate which accounts for roughly 262,000 daily requests, roughly 60% of them (homogeneous case) can be satisfied by the edge. This translates for great load reduction at the back end datacenters.

5.3 Results for Soft QoS Requirements

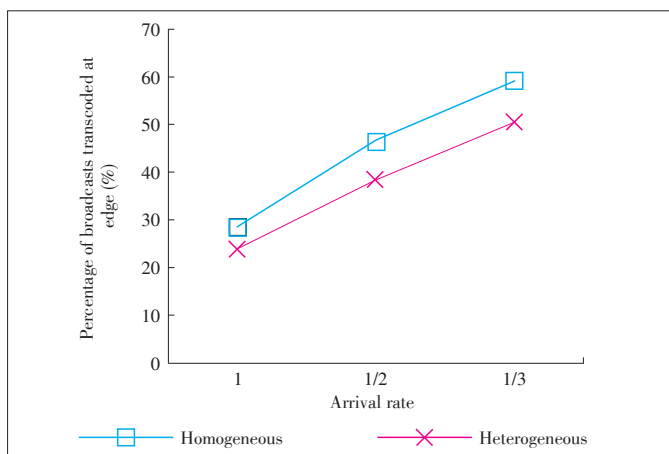
We consider that the processing requirement of real time performance is relaxed as per Section 4.2. We evaluated the performance of the algorithms of Section 4.2 when combined



▲ **Figure 3:** Percentage of broadcasts processed by the edge (1000 servers, full dataset). Two different cases: homogeneous and heterogeneous.

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

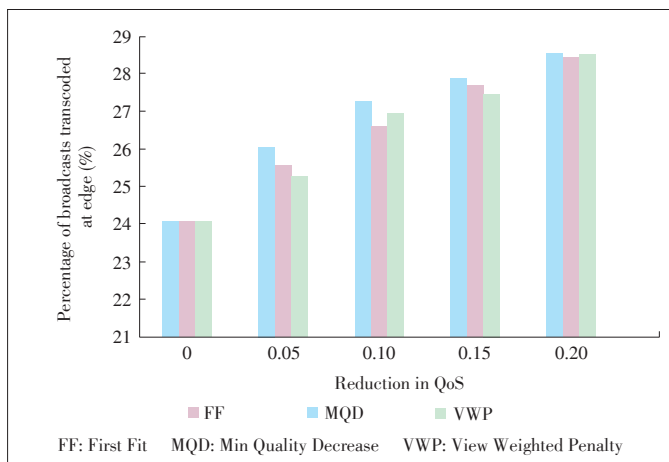
Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritas, Thanasis Loukopoulos, and XU Cheng-Zhong



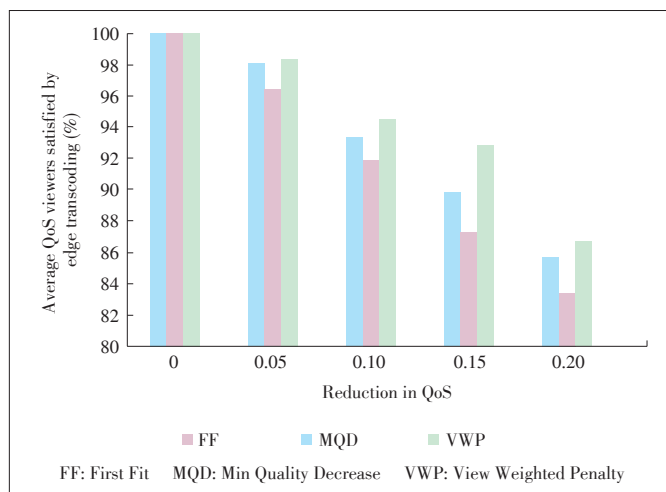
▲ Figure 4. Percentage of broadcasts processed by the edge for decreasing arrival rate (1000 servers).

with MAX-WF for various levels of QoS reduction, namely: 0% (no reduction - real time performance), 5%, 10%, 15% and 20%. Fig. 5 plots the percentage of broadcasts having their transcoding jobs assigned to the micro datacenter, for the heterogeneous case used in Figs. 3 and 4 and with the whole trace as input. All three heuristics assign to the edge an increasing number of jobs as QoS requirements are reduced. Among the algorithms, MQD achieves the best performance, followed by VWP and FF. As it can be observed, with a 5% decrease in processing rate, an extra 2% of jobs can be assigned to the edge, while with 20% an extra 4.5%. Although a 20% reduction seems impractical at first glance, it roughly means that instead of processing a stream at 30 fps, the stream might be processed at 24 fps. It is worth noting that this is the lowest rate for 1080p TV sets. Overall, by relaxing the nominal real time processing rate for transcoding jobs, significant extra load could be offset from back end datacenters.

In order to further quantify the impact of QoS reduction, in Fig. 6 we plot the average viewing quality (for the viewers sat-



▲ Figure 5. Percentage of broadcasts processed by the edge for varying QoS reduction percentages (full dataset, heterogeneous servers).



▲ Figure 6. Average QoS of viewers as the allowable reduction in QoS for edge transcoding jobs is increased (full dataset, heterogeneous servers).

fied by the edge) as a percentage of the achieved fps when compared to real time 30 fps. VWP achieves the best performance, which for a 15% allowable reduction (0.15 point in x-axis) leads to an average viewing quality of 93%. Put it in other terms the average processing rate will be almost 28 fps. For the same QoS reduction (15%), Fig. 5 depicts that VWP assigns an extra ~3.5% of jobs, or roughly 27,000 more broadcasts at the edge micro datacenter. Thus, an interesting tradeoff is present whereby VWP can offset substantial load towards the edge micro datacenter at only a small decrease on average viewing quality.

6 Conclusions

In this paper we examined scheduling heuristics for the problem of assigning live transcoding jobs at an edge micro datacenter. We considered two main cases. The first accounts for real time performance, while the second allows small quality degradation on the output video streams in order to increase the assigned jobs to the micro datacenter. Through simulation experiments using a realistic dataset, it is concluded that interesting tradeoffs can be obtained by a method (VWP) that takes into account viewer perceived QoS.

References

- [1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, Sept. 2013. doi: 10.1016/j.future.2013.01.010.

Scheduling Heuristics for Live Video Transcoding on Cloud Edges

Panagiotis Oikonomou, Maria G. Koziri, Nikos Tziritis, Thanasis Loukopoulou, and XU Cheng-Zhong

- [2] S. K. Khaitan and J. D. McCalley, "Design techniques and applications of cyber physical systems: a survey," *IEEE Systems Journal*, vol. 9, no. 2, pp. 350–365, Jun. 2015. doi: 10.1109/JSYST.2014.2322503.
- [3] Cisco Systems Inc. (2017, Jan. 30). *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020* [Online White Paper]. Available: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>
- [4] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, Jul. 2003. doi: 10.1109/TCSVT.2003.815165.
- [5] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012. doi: 10.1109/TCSVT.2012.2221191.
- [6] D. Grois, D. Marpe, A. Mulyoff, et al., "Performance comparison of H.265/ MPEG - HEVC, VP9, and H.264/MPEG - AVC encoders," in *Picture Coding Symposium (PCS)*, San Jose, USA, Dec. 2013, pp. 394–397. doi: 10.1109/PCS.2013.6737766.
- [7] R. A. Pardo, K. Pires, A. Blanc, and G. Simon, "Transcoding live adaptive video streams at a massive scale in the cloud," in *ACM SIGMM Conference on Multimedia Systems (MMSys)*, Portland, USA, Mar. 2015, pp. 49–60. doi:10.1145/2713168.2713177.
- [8] VideoLAN. (2017, Jan. 30). *x264 home page* [Online]. Available: <http://www.videolan.org/developers/x264.html>
- [9] F. Bossen, "Common test conditions and software reference configurations," JCTVC, San Jose, USA, Document: JCTVC-H1100, Feb. 2012.
- [10] M. G. Koziri, P. Papadopoulos, N. Tziritis, et al., "Slice-based parallelization in HEVC encoding: realizing the potential through efficient load balancing," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Montreal, Canada, Sept. 2016, pp. 1–6. doi: 10.1109/MMSP.2016.7813354.
- [11] M. Shafique, M. U. K. Khan, and J. Henkel, "Power efficient and workload balanced tiling for parallelized high efficiency video coding," in *IEEE International Conference on Image Processing (ICIP)*, Paris, France, Oct. 2014, pp. 1253–1257. doi: 10.1109/ICIP.2014.7025250.
- [12] C. C. Chi, M. A. Mesa, B. Juurlink, et al., "Parallel scalability and efficiency of HEVC parallelization approaches," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1827–1838, Dec. 2012. doi: 10.1109/TCSVT.2012.2223056.
- [13] Y.-J. Ahn, T.-J. Hwang, D.-G. Sim, and W.-J. Han, "Implementation of fast HEVC encoder based on SIMD and data-level parallelism," *EURASIP Journal Image and Video Processing*, vol. 16, Dec. 2014. doi: 10.1186/1687-5281-2014-16.
- [14] M. G. Koziri, D. Zacharis, I. Katsavounidis, and N. Bellas, "Implementation of the AVS video decoder on a heterogeneous dual-core SIMD processor," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 2, pp. 673–681, May 2011. doi: 10.1109/TCE.2011.5955207.
- [15] J. F. Franche and S. Coulombe, "Fast H.264 to HEVC transcoder based on post-order traversal of quadtree structure," in *IEEE International Conference on Image Processing (ICIP)*, Quebec, Canada, Sept. 2015, pp. 477–481. doi: 10.1109/ICIP.2015.7350844.
- [16] I. Ahmad, X. Wei, Y. Sun, and Y.-Q. Zhang, "Video transcoding: an overview of various techniques and research issues," *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 793–804, Oct. 2005. doi: 10.1109/TMM.2005.854472.
- [17] W. Zhang, Y. Wen, J. Cai, and D. O. Wu, "Toward transcoding as a service in a multimedia cloud: energy - efficient job - dispatching algorithm," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 5, pp. 2002–2012, Jun. 2014. doi: 10.1109/TVT.2014.2310394.
- [18] S. Lin, X. Zhang, Q. Yu, et al., "Parallelizing video transcoding with load balancing on cloud computing," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, Beijing, China, May 2013, pp. 2864–2867. doi: 10.1109/ISCAS.2013.6572476.
- [19] A. Ashraf, F. Jokhio, T. Deneke, et al., "Stream-based admission control and scheduling for video transcoding in cloud computing," in *IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, Delft, Netherlands, May 2013, pp. 482–489. doi: 10.1109/CCGrid.2013.21.
- [20] G. Gao, W. Zhang, Y. Wen, et al., "Towards cost-efficient video transcoding in media cloud: insights learned from user viewing patterns," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1286–1296, Aug. 2015. doi: 10.1109/TMM.2015.2438713.
- [21] S. U. R. Malik, S. U. Khan, S. J. Ewen, et al., "Performance analysis of data intensive cloud systems based on data management and replication: a survey," *Distributed and Parallel Databases*, vol. 34, no. 2, pp. 179–215, Jun. 2016. doi: 10.1007/s10619-015-7173-2.
- [22] W. Ji, Z. Li, and Y. Chen, "Joint source-channel coding and optimization for layered video broadcasting to heterogeneous devices," *IEEE Transactions on Multimedia*, vol. 14, no. 2, pp. 443–455, Apr. 2012. doi: 10.1109/TMM.2011.2177645.
- [23] W. Ji, Z. Li, and Y. Chen, "Content-aware utility-fair video streaming in wireless broadcasting networks," in *IEEE International Conference on Image Processing (ICIP)*, Brussels, Belgium, Sept. 2011, pp. 145–148. doi: 10.1109/ICIP.2011.6115717.
- [24] M. T. Beck, S. Feld, A. Fichtner, et al., "ME-VoLTE: network functions for energy-efficient video transcoding at the mobile edge," in *International Conference on Intelligence in Next Generation Networks (ICIN)*, Paris, France, Feb. 2015, pp. 38–44. doi: 10.1109/ICIN.2015.7073804.
- [25] J. De Cock, A. Mavlikar, A. Moorthy, and A. Aaron, "A large-scale video codec comparison of x264, x265 and libvpx for practical VOD applications," *SPIE Applications of Digital Image Processing XXXIX*, vol. 9971, 997116, Sept. 2016. doi: 10.1117/12.2238495.
- [26] Amazon Web Services. (2017, Jan. 30). *Amazon EC2 Instance Types* [Online]. Available: <https://aws.amazon.com/ec2/instance-types/>

Manuscript received: 2017-02-09

Biographies

Panagiotis Oikonomou (paikonom@uth.gr) received his Diploma degree (2008) and M.Sc. degree (2010) from the Department of Electrical and Computer Engineering, University of Thessaly, Greece. He is currently a Ph.D. candidate at the same Department. His research interests include optimization algorithms and fuzzy logic methods.

Maria G. Koziri (mkozi@uth.gr) received her Diploma degree in computer engineering from the Technical University of Crete, Greece in 2003 and Ph.D. degree in computer science from the University of Thessaly, Greece in 2007. She is currently a visiting lecturer in the Computer Science Department of the University of Thessaly. Her research interests include video compression, scalable video coding, rate-distortion optimization and computer architecture.

Nikos Tziritis (nikolaos@siat.ac.cn) received his B.Sc. degree from the Technological Educational Institute of Serres, Greece in 2004, and M.Sc. and Ph.D. degrees from the University of Thessaly, Greece in 2006 and 2011, respectively. He is currently an associate professor in Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China. His work has appeared in over 35 publications. He is the recipient of the Award for Excellence for Early Career Researchers in Scalable Computing from IEEE Technical Committee in Scalable Computing in 2016.

Thanasis Loukopoulou (luke@dib.uth.gr) received his Ph.D. degree in computer science from the Hong Kong University of Science and Technology, China. He is currently a lecturer at the Department of Computer Science and Biomedical Informatics of the University of Thessaly, Greece. His research interests are in green computing, cloud computing, WSNs, scheduling, load balancing and video coding parallelization. His work appeared in over 50 publications. He had the best paper award in ICPP 2001.

XU Cheng-Zhong (cz.xu@siat.ac.cn) received the Ph.D. degree in computer science from the University of Hong Kong, China in 1993. He is currently a professor in the Department of Electrical and Computer Engineering of Wayne State University, China and the director of Cloud and Internet Computing Laboratory (CIC) and Sun's Center of Excellence in Open Source Computing and Applications (OSCA). His research interest is mainly in scalable distributed and parallel systems and wireless embedded computing devices. He has published two books and more than 160 articles in peer-reviewed journals and conferences in these areas.

A Survey on Cloud Security

WU Chunming, LIU Qianjun, LI Yuwei,
CHENG Qiumei, and ZHOU Haifeng

(College of Computer Science and Technology,
Zhejiang University, Hangzhou 310058, China)

1 Introduction

Cloud computing has recently been experiencing fast development as a distributed model for performing utility computing. The cloud environment combined with virtualization techniques provides on-demand service, i.e., pay-per-use service, which ensures timely effective resource scheduling and solves the problem of resource shortage for cloud users. Currently, the most widely accepted concepts and features related to cloud computing are defined by the National Institute of Standards and Technology (NIST). NIST makes the following statements: "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources involving networks, servers, storage and applications that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]." In simple terms, the word "cloud" refers to resources (both hardware and software) stored in the Internet infrastructures. These infrastructures, also named as "data centers", are equipped with a large number of servers to store and compute user data. In 2016, Cisco predicted that traffic in data centers would enlarge three times from 2015 to 2010 and cloud traffic would account for more than 92% by 2020 [2]. These data reveal that IT industry is going to be heavily dependent on cloud computing.

Although it is easy to understand the advantages of cloud computing from a commercial view, its security issues are quite complex. As the promotion of cloud services, more and more enterprises start to adopt the cloud computing. Cloud Security Alliance (CSA) reported that outage is found more and more frequent in cloud computing area in recent years [3]. However, the cloud is encountering many security threats. The

This work was supported by the National Key Research and Development Program of China under Grant 2016YFB0800102, the National High Technology Research Program of China under Grant 2015AA016103, the Fundamental Research Funds for the Central Universities, and the ZTE Industry-Academia-Research Cooperation Funds.

Abstract

Cloud computing system packages infrastructures, applications and other resources as services, and delivers the services to market in an elastic and fast way. The significant advantages of cloud computing, e.g., scalability, elasticity, and pay-per-use, bring it considerable commercial values. Nevertheless, owing to the new application scenario, e.g., multi-tenant, cloud computing is encountering potential security risks. This paper reviews the state-of-art research in cloud security. According to the attack levels, it analyzes four kinds of attacks in the cloud, i.e., network-based attacks, VM-based attacks, storage-based attacks, and application-based attacks. The countermeasures and corresponding techniques are then introduced. Furthermore, this paper also discusses an innovative and promising solution for cloud security by dynamically changing system configuration.

Keywords

cloud computing; security; virtualization; storage

well-known American software company Symantec made a threat report in 2015 [4], in which a 91% increase of attacks targeted at certain victims was reported. Some research has been made to cope with the increasing security threats. The network-based attacks, e.g., botnet, is now able to be detected and prevented in time [5], [6]. To prevent the attacks targeting at data, the traditional techniques like encryption as well as authentication and authorization are utilized [7]–[9]. For precluding the attacks targeted at virtual machines (VMs) and hypervisors, a promising solution is proposed by setting access control policies [9]. Cloud computing inherits from the traditional network architecture to some extent, but it is more vulnerable to security compromise. Benefiting from the development of virtualization security techniques, trusted cloud computing, identity management and other key techniques, cloud security is improving gradually. However, potential users still hesitate to move their sensitive data off-premise. As a result, despite the efforts from the research community, the further development of cloud service also needs the assistance and progress from regulations and laws.

The rest of this paper is presented as follows. In Section 2, we analyze four kinds of attacks in the cloud, i.e., network-based attacks, VM-based attacks, storage-based attacks, and application-based attacks. The countermeasures and the corresponding core techniques are then introduced in Section 3. This is followed by a discussion of an innovative solution of cloud security by dynamically changing system configuration

in Section 4. In Section 5, we conclude the paper.

2 Cloud Security Categorization

Current cloud computing services could be categorized in to three main types: infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS), according to the system structure level. IaaS means that users can access services from well-constructed infrastructures through the Internet. PaaS packages the platform for software developing as a service and delivers it to the users in an SaaS model. SaaS enables users to rent web-based software from service providers.

As Fig. 1 shows, these three different service models have different components. The IaaS model provides users with servers, storage, network, virtualization and other fundamental resources; The PaaS model provides identity management, access control, work flow and other support for operating systems, databases and web servers; The SaaS model supplies a variety of applications that can be accessed through the Internet and that are charged by time or resource. In the cloud framework, any component is possible to expose system loopholes that can be utilized by attackers to conduct attacks. For instance, attacks based on network loopholes could bring about communication latency or connection failure; attacks based on storage loopholes could cause data exposure or destroy; and attacks based on VM, hypervisor and application loopholes are able to compromise cloud security in many ways. Generally, the cloud platform mainly includes high-efficiency networks, high-speed storage devices, high-powered servers, and applications.

According to attack targets, we classify attacks in cloud into four categories, i.e., network-based attacks, storage-based attacks, VM-based attacks, and application-based attacks. The

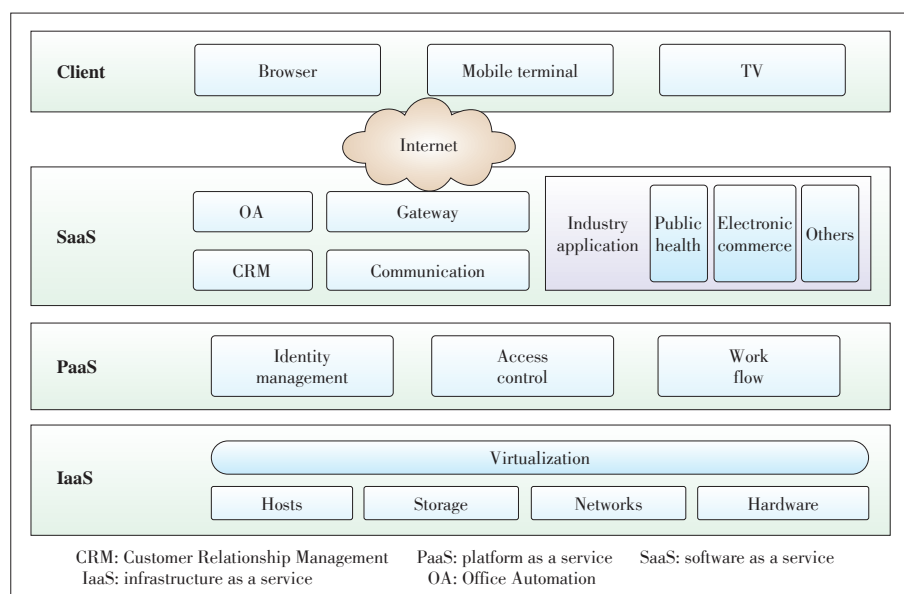
first category of attacks could bring about long communication latency or connection failure. The second one could induce data exposure or destroy. The third one is able to compromise cloud security in many ways. The last one is because of application vulnerabilities.

2.1 Network-Based Attacks

Network-based attacks in cloud are similar to the same kind of attack in traditional networks, but they are more destructive. In a traditional network, the system boundary can be clearly determined, and the infrastructures can be well-protected through physical and logical security domains. While in the cloud environment, multi-tenancy with scattered data storage makes it extremely difficult to fully provide safeguard for all the uses. This kind of attack, e.g., port scanning, botnets, spoofing, Denial of Service (DoS), could lead to deteriorative Quality of Service (QoS) and steal user data in cloud. For instance, botnets, e.g., Zeus, is able to utilize Amazon's Elastic Computing Cloud (EC2) to steal user passwords. Currently, some network-based attacks can be timely detected and prevented. Lin and Lee [5] proposed an approach to detect botnets by tracing the botmaster. This approach initially tries to find out the cryptographic keys used for botnet communication between bots and the botmaster. The attack traffic is first decrypted by identifying patterns of regions that may contain these keys. An entropy search is then performed to identify these keys. Subsequently, the communication between bots and the botmaster is decrypted. Finally, the botmaster's location is found by acquiring its IP address. To solve the abovementioned instance of attacks, EC2 is enforced by configuring an inner firewall for each user [6]. This inner firewall denies traffic in any mode by default. As a result, the users need to configure a port to allow traffic in. However, the boundary threshold of connections allowed by the firewall is a new problem that needs to be solved.

2.2 VM-Based Attacks

In cloud computing, virtualization technique enables creation, operation, shut-down, destruction and other functions for VMs, which brings convenient management for the computing resource. However, the VM technique also brings new security risks. Having multiple VMs in one system can lead to several serious security issues, i.e., wiretap from a malicious VM neighbor. Many attacks arise in different phases of VM management. These attacks are able to be roughly divided into four types, i.e., cross VM side channel attacks, VM creation attacks, VM migration and rollback attacks, VM scheduler-based attacks. The virtualization system in cloud



▲ Figure 1. Framework of cloud computing.

A Survey on Cloud Security

WU Chunming, LIU Qianjun, LI Yuwei, CHENG Qiumei, and ZHOU Haifeng

computing and its security threats are shown in **Fig. 2**.

- 1) Cross VM side channel attacks: Because of weak isolation mechanisms between VMs in the same physical machine, this type of attack is from a malicious VM, and aims to steal, falsify or destroy user data from neighbor VMs on the same machine by bypassing isolation mechanisms. It first tries to infer the functionality and activity of software by observing a variety of system hardware behavior. It then tries to obtain the physical machine's information, e.g., resource usage, secret keys, and other information [10]. This kind of attack happens in various hardware levels, e.g., CPU cache, memory, and access driver, which makes it hard to detect and defend. Moreover, potential security risks brought by this kind of attack are disconcerting, when considering that attackers can even control victim VMs by associating with other kinds of attacks.
- 2) VM creation attacks: An attacker that conducts such a kind of attack needs to inject malicious code, e.g., worms, into the VM image. As a result, the malicious code is proliferating in the VM creation and replication processes, which will induce serious problems. In addition, since VMs are copied and transferred as files, the attacker can copy a VM and get sensitive data from it in an easier way. Furthermore, since the survival time of a VM is usually very short, this VM may disappear before the malicious code is detected. What's worse, a virus infects a VM in the same way of infecting a file, while the antivirus in a VM needs to traverse each part of the guest operating system, which makes the detection very difficult.
- 3) VM migration and rollback attacks: Owing to the elastic and dynamic feature of cloud, VM is easy to migrate and roll back. Nevertheless, this makes sensitive data in VM exist for a relatively long time. When a VM image is copied during a VM migration, the data in this VM could be accessed by an attacker. For instance, in an S/key system, if the pass-

word has just been input for logging in and at the same time the VM is asked to roll back, attackers can easily get the password.

- 4) VM scheduler-based attacks: The time scheduling algorithm of VM that has some design loopholes can be utilized by attackers for initiating an attack. Given an unfair scheduler, an attacker is able to occupy a lot of other clients' resources with a little cost [11]. Wei et al. [9] proposed a mechanism to share VM images in a secure manner, which uses a filter to remove private information or malicious code from the image and traces those operations on these VM images. After the image is published, the framework can also be used to scan and repair infected software.

VM-based attacks usually come along with the steal or destroy of user data. Consequently, storage security is another important research aspect of cloud security, as we will discuss in the next section.

2.3 Storage-Based Attacks

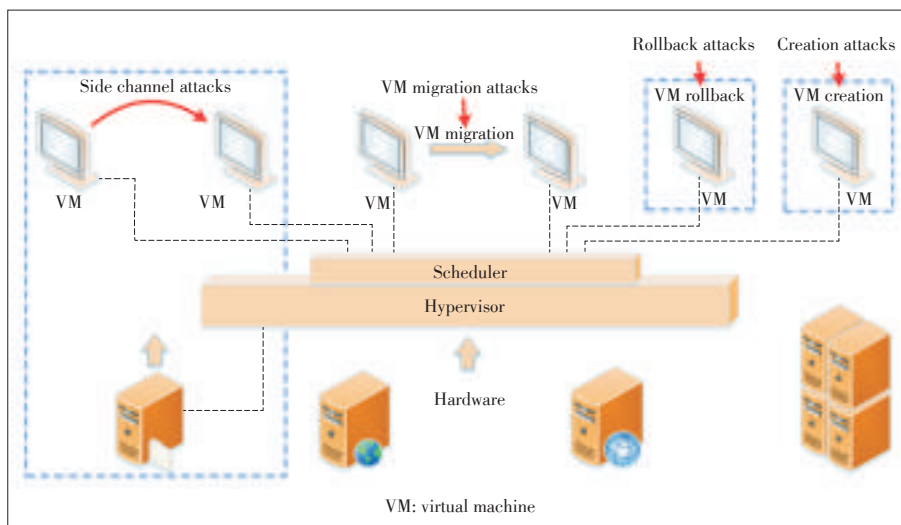
Cloud storage provides data storage and business access functions by gathering various types of storage devices. An external attacker usually steals privacy information from storage devices and attacks a series of vulnerabilities by manipulating the data. In the cloud, data need backing up, snapshot, or archiving, resulting in a significant increase in resource occupation and cost of cloud storage. In order to remove redundant and ineffective information, efficient data erasing technology and data deduplication technology are extremely important. However, due to multi-user data stored in a shared cloud environment, the manipulation of data is likely to cause information leakage or loss. Storage-based attacks thus can be divided mainly into two cases, i.e., data erase and data deduplication.

- 1) Data erase: When deleting data from storage devices, the file system in cloud will not remove them completely. Consequently, the remaining data can be found and utilized by attackers.

In elastic clouds, the data erasing usually occurs in the resource reallocation process. Since the data of previous users may still remain in the storage devices, it is possible that these data could be accessed by the new user or by an attacker.

- 2) Data deduplication: Data deduplication is used to keep a single copy of data, but it can also be utilized to identify files and file content. Currently, some attackers are even able to create a hidden channel to perform communication between the malicious software and the command server through data deduplication.

Several techniques, e.g., data encryption and identity management, are used to en-



▲ Figure 2. VM-based security threats in cloud.

sure data security and privacy to a certain extent. Kaaniche and Laurent [7] proposed an approach to use data deduplication for sharing data in public cloud. This approach encrypts data and encapsulates permissions in a single file, which is allowed to be decrypted only by authenticated users. Sanchez et al. [8] described an identity management system using the Security Assertion Markup Language (SAML), to provide users with access to cloud resources while protecting their privacy. The VM image access control mechanism is also proposed to ensure data cleanup security [9]. Wang et al. [12] introduced a privacy-preserving cloud data storage system which also enables public auditing/verification.

2.4 Application-Based Attacks

Applications running on cloud are also vulnerable to attacks, especially in the related protocols that serve these applications. We generally consider three types of application-based attacks, i.e., malware injection and steganography attacks, shared infrastructure based attacks, and network and protocol based attacks.

- 1) Malware injection and steganography attacks: Since common software usually has millions of lines of code and the code is usually written by numerous people, it is actually impossible to have fully reliable software. That is to say, the PaaS or SaaS provider is not always reliable. Moreover, if the cloud platform involves insecure interfaces, malicious code is possibly inserted into applications. Through a steganography attack [13], the attacker is able to add secret data within seemingly innocent carriers. Those secret data will be embedded in regular data. The secret data usually include malicious code that brings about unpredictable security risks.
- 2) Shared infrastructure based attacks: Multi-tenants' VMs are isolated by VM isolation mechanisms in the circumstance of shared infrastructures. Utilizing application loopholes or injecting malicious code into a SaaS system, an attacker is possibly able to break the isolation mechanisms. Further, the attacker can launch code injection or Cross-Site Scripting (XSS) to trace the victim application's execution path and activities [14].
- 3) Network and protocol based attacks: Network services involve a variety of protocols, e.g., Simple Object Access Protocol (SOAP). The packet header defined in SOAP can be replaced with an invalid request for conducting an attack [15]. If the related security policies and validation mechanisms fail to check the header, relevant services will not work normally.

3 Countermeasures and Key Techniques

To cope with the aforementioned security threats in cloud, many security countermeasures and key techniques for cloud computing have been proposed recently. We introduce four

key techniques in this section.

3.1 Virtualization Security

The cloud employs virtualization techniques to achieve flexible dynamic management of physical resources. The virtualization technique also enables the isolation of multi-tenant. The security of VMs and hypervisors directly determine the security of the whole cloud platform. IBM proposed a secure hypervisor architecture named as sHype [16]. It enforces access control for traffic between different VMs, which is capable of guaranteeing isolation. Wei et al. [17] put forward the image file management system to achieve access control, source tracking, filtering and scanning of VM images. Their method could ensure the integrity of VM image files.

3.2 Trusted Computing

Trusted computing in cloud is able to provide a safe and trusted execution environment, and to ensure the integrity of data and computing. Eguro and Venkatesan [18] proposed the idea of Field Programmable Gate Arrays (FPGAs), which can identify the computation implemented in the logic fabric. A symmetric encryption key is stored in FPGA memory. The FPGA is installed in a cloud server. A trusted authority in cloud can encrypt and sign applications with the keys of FPGAs. Consequently, the application can process data in a secure manner. Sadeghi et al. [19] designed a trustful software token and bound it with security verification module. Owing to their approaches, the leakage of sensitive outsourcing data can be largely avoided.

3.3 Data Security and Privacy Protection

Under the cloud computing architecture, data are usually stored in the data center which is usually away from users [20], [21]. As a result, users have no specific idea of where their data are stored and how their data are managed. Avoiding data loss, protecting data privacy and ensuring data isolation are important security requirements for cloud storage. Therefore, it is necessary to take effective measures to protect data, e.g., multiple copies, storage encryption, and trust mechanisms. To achieve reliable data storage, real-time data backup is essential. Maintaining a copy both in the cloud and in the enterprise is a considerable way. Jensen et al. [22] designed an encryption mechanism based on ring and group to achieve anonymous storage of user data. Mowbray et al. [23] proposed a client-based tool for privacy management while storing and using data. This tool provides a user-centric trust model to help users control their sensitive data stored and used in cloud.

3.4 Identification and Access Control

In a multi-tenant cloud environment, how to realize user identity management and access control and how to ensure the separation of data between different users are the key problems in cloud security. Yan et al. [24] combined federal identity

A Survey on Cloud Security

WU Chunming, LIU Qianjun, LI Yuwei, CHENG Qiumei, and ZHOU Haifeng

management and hierarchical cryptography in their identification system, which made the distribution and authentication easier and more safe. Yu et al. [25] defines and enforces access policies based on data attributes by exploiting and uniquely combining techniques of Attribute-Based Encryption (ABE), proxy re-encryption, and lazy re-encryption.

4 Dynamic Proactive Defense in Cloud

Current data centers usually adopt traditional security defense mechanisms, e.g., traditional firewall, Intrusion Detection System (IDS), and monitor. However, these defense mechanisms are increasingly becoming passive in cloud when encountering more and more advanced attacks, e.g., advanced persistent threat (APT). As a result, it is critical to explore completely new defense mechanisms to guard cloud in a more proactive way for coping with those potential threats.

Recently, some proactive defense mechanisms are proposed to enforce cloud security through dynamically changing the configuration of cloud system, e.g., IP addresses, network routing algorithms, communication encryption algorithms, and authentication methods. Evolving Defense Mechanism (EDM) [26] is such a new mechanism, which is designed to dynamically change the configuration of a network system to proactively defend potential attacks. Dynamic certificate mechanisms [27] are also proposed to remove the main obstruct for the further application of cloud, i.e., trustworthiness of cloud service providers. Traditional authentication methods, e.g., Cloud Service Certifications (CSCs), fail to guarantee a certificate valid all the time. The dynamic certificate mechanism uses a third-party authority to authenticate cloud services constantly in order to avoid illegal certificates or security vulnerabilities.

Such new research is able to solve some of cloud security issues. For instance, by dynamically changing VM IP addresses, internal and external malicious scanning can be largely avoided; by changing encryption algorithms, communication between VMs becomes more reliable; by changing authentication methods, the reliability of authentication between the users and administrators is able to be enforced. In practical applications, software defined networks (SDN) [28] is favorable for achieving dynamical configuration changes owing to its centralized control and programmable logic. Current SDN controllers, e.g., OpenDaylight, and ONOS, also provide network management services for cloud by opening its northbound interfaces for OpenStack.

5 Conclusions

As the cloud industry plays an increasingly important role in the information arena, it is encountering more and more security threats. This paper discusses several main issues and key techniques of cloud security associated with the existing research from both the academic and industry. Meanwhile, a safe

and reliable cloud environment relies not only on technological progress, but also on legal regulations. We look forward that the cloud industry community, academia community and government can work together to achieve a safer cloud environment.

References

- [1] P. Mell and T. Grance. (2011, Sept.). *The NIST definition of cloud computing* [Online]. Available: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>
- [2] Cisco. (2016). *Cisco global cloud index: forecast and methodology, 2015–2020* [online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns1175/Cloud_Index_White_Paper.pdf
- [3] R. Ko, S. Lee, and V. Rajan. (2013, Mar.). *Cloud computing vulnerability incidents: a statistical overview* [Online]. Available: <https://cloudsecurityalliance.org/download/cloud-computing-vulnerability-incidents-a-statistical-overview.March>
- [4] Symantec. (2015). *Internet security threat report* [Online]. Available: https://www.symantec.com/security_response/publications/monthlythreatreport.jsp
- [5] W. Lin and D. Lee. "Traceback attacks in cloud—pebbletrace botnet," in *IEEE International Conference on Distributed Computing Systems Workshops*, 2014, pp. 417–426.
- [6] A. Web, S. Overview, and S. P. May. (2009, Jun.). *Amazon web services: overview of storage processes* [Online]. Available: https://lenix.tecnico.ulisboa.pt/downloadFile/3779573805259/AWS_Security_Whitepaper.pdf
- [7] N. Kaaniche and M. Laurent. "A secure client side deduplication scheme in cloud storage environments," in *6th International Conference on New Technologies, Mobility and Security*, Dubai, United Arab Emirates, 2014, pp. 1–7. doi: 10.1109/NTMS.2014.6814002.
- [8] R. Sanchez, F. Almenares, P. Arias, et al., "Enhancing privacy and dynamic federation in IdM for consumer cloud computing," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 58, pp. 95–103, 2012.
- [9] J. Wei, X. Zhang, G. Ammons, et al., "Managing security of virtual machine images in a cloud environment," in *ACM Cloud Computing Security Workshop*, 2009, pp. 91–96.
- [10] Y. Zhang, A. Juels, M. K. Reiter, et al., "Cross-VM side channels and their use to extract private keys," in *ACM Conference on Computer and Communications Security*, Chicago, USA, 2012, pp. 91–96. doi: 10.1145/1655008.1655021.
- [11] F. Zhou, M. Goel, P. Desnoyers, et al., "Scheduler vulnerabilities and coordinated attacks in cloud computing," in *IEEE International Symposium on Network Computing and Applications*, Cambridge, USA, 2011, pp. 123–130. doi: 10.1109/NCA.2011.24.
- [12] Q. Wang, C. Wang, J. Li, et al., "Enabling public verifiability and data dynamics for storage security in cloud computing," in *14th European Conference on Research in Computer Security*, Saint-Malo, France, 2009, pp. 335–370.
- [13] J. Sen, "Security and privacy issues in cloud computing," in *Architectures and Protocols for Secure Information Technology Infrastructures*, Hershey, USA: IGI Global, 2013.
- [14] Y. Zhang, A. Juels, M. K. Reiter, et al., "Cross-tenant side-channel attacks in PaaS clouds," in *ACM SIGSAC Conference on Computer and Communications Security*, Scottsdale, USA, 2014, pp. 990–1003. doi: 10.1145/2660267.2660356.
- [15] N. Gruschka and L. L. Iacono, "Vulnerable cloud: SOAP message security validation revisited," in *IEEE International Conference on Web Services*, Los Angeles, USA, 2009, pp. 625–631. doi: 10.1109/ICWS.2009.70.
- [16] R. Sailer, E. Valdez, and T. Jaeger, "sHype: secure hypervisor approach to trusted virtualized systems," IBM, Yorktown Heights, USA, Research Report RC23511, 2005.
- [17] J. Wei, X. Zhang, G. Ammons, et al., "Managing security of virtual machine images in a cloud environment," in *ACM Workshop on Cloud Computing Security*, Chicago, USA, 2009, pp. 91–96. doi: 10.1145/1655008.1655021.
- [18] K. Eguro and R. Venkatesan, "FPGAs for trusted logic computing," in *IEEE International Conference on Field Programmable Logic and Applications (FPL)*, Oslo, Norway, 2012, pp. 63–70.
- [19] A. R. Sadeghi, T. Schneider, and M. Winandy, "Token-based cloud computing: secure outsourcing of data and arbitrary computations with lower latency," in *International Conference on Trust and Trustworthy Computing*, Berlin, Germany, 2010, pp. 417–429.

- [20] J. Tang, "Ensuring security and privacy preservation for cloud data services," *ACM Computing Surveys (CSUR)*, vol. 49, no. 1, article 13, Jul. 2016. doi: 10.1145/2906153.
- [21] C. Wang, K. Ren, W. Lou, et al., "Toward publicly auditable secure cloud data storage services," *IEEE Network*, vol. 24, no. 4, pp. 19–24, Jul./Aug. 2010.
- [22] M. Jensen, S. Schäge, and J. Schwenk, "Towards an anonymous access control and accountability scheme for cloud computing," in *IEEE International Conference on Cloud Computing*, Miami, USA, 2010, pp. 540–541. doi: 10.1109/CLOUD.2010.61.
- [23] K. D. Bowers, A. Juels, and A. Oprea, "Proofs of retrievability: theory and implementation," in *ACM Workshop on Cloud Computing Security*, Chicago, USA, 2009, pp. 43–54. doi: 10.1145/1655008.1655015.
- [24] L. Yan, C. Rong, and G. Zhao, "Strengthen cloud computing security with federal identity management using hierarchical identity-based cryptography," in *International Conference on Cloud Computing*, Bangalore, India, 2009, pp. 167–177.
- [25] S. Yu, C. Wang, K. Ren, et al., "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proc. IEEE INFOCOM*, San Diego, USA, 2010, pp. 534–542. doi: 10.1109/INFOCOM.2010.5462174.
- [26] H. Zhou, C. Wu, M. Jiang, et al., "Evolving defense mechanism for future network security," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 45–51, 2015. doi: 10.1109/MCOM.2015.7081074.
- [27] S. Lins, P. Grochol, S. Schneider, et al., "Dynamic certification of cloud services: trust, but verify," *IEEE Security & Privacy Magazine*, vol. 14, no. 2, pp. 66–71, 2016. doi: 10.1109/MSP.2016.26.
- [28] Open Networking Foundation. (2012, Apr. 13). *Software-defined networking: the new norm for networks* [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf>

Manuscript received: 2016-10-24

Biographies

WU Chunming (wuchunming@zju.edu.cn) received the Ph.D. degree in computer science from Zhejiang University in 1995. He is currently a professor with the College of Computer Science and Technology of Zhejiang University, and the Associate Director of the Research Institute of Computer System Architecture and Network Security. His research fields include Software-Defined Network, reconfigurable networks, proactive network defense, cloud security, network virtualization, and intelligent networks.

LIU Qianjun (liuqj0522@163.com) is now a Ph.D. candidate at the College of Computer Science, Zhejiang University, Hangzhou, China. Her research interests involve cloud security, big data security and smart security analytics.

LI Yuwei (liyuwei@zju.edu.cn) is now a Ph.D. candidate at the College of Computer Science, Zhejiang University, Hangzhou, China. Her research interests involve system security, code security and smart security.

CHENG Qiumei (chengqiumei@zju.edu.cn) is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China. Her research interests include cloud security, software-defined security, network virtualization, software-defined networks, and proactive network defense.

ZHOU Haifeng (zhouhaifeng@zju.edu.cn) is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China. His research interests include software-defined networks, software-defined network security, cloud security, proactive network defense, intelligent networks and security systems, network traffic engineering, and innovative network and security technologies.

Roundup

Introduction to ZTE Communications



ZTE Communications is a quarterly, peer-reviewed international technical journal (ISSN 1673–5188 and CODEN ZCTOAK) sponsored by ZTE Corporation, a major international provider of telecommunications, enterprise and consumer technology solutions for the Mobile Internet. The journal publishes original academic papers and research findings on the whole range of communications topics, including communications and information system design, optical fiber and electro-optical engineering, microwave technology, radio wave propagation, antenna engineering, electromagnetics, signal and image processing, and power engineering. The journal is designed to be an integrated forum for university academics and industry researchers from around the world. *ZTE Communications* was founded in 2003 and has a readership of 5500. The English version is distributed to universities, colleges, and research institutes in more than 140 countries. It is listed in Inspec, Cambridge Scientific Abstracts (CSA), Index of Copernicus (IC), Ulrich's Periodicals Directory, Abstract Journal, Norwegian Social Science Data Services (NSD), Chinese Journal Fulltext Databases, Wanfang Data — Digital Periodicals, and China Science and Technology Journal Database. Each issue of *ZTE Communications* is based around a Special Topic, and past issues have attracted contributions from leading international experts in their fields.

Network Coding-Based Interference Management Scheme in D2D Communications

WANG Shuang and HOU Ronghui

(Xidian University, Xi'an 710071, China)

Abstract

In this paper, we propose an interference management scheme for device-to-device (D2D) communications in cellular networks. Considering the underlay D2D communications, the signal quality of cellular users would be affected by D2D users. To solve this problem, we explore the application of network coding and relay-assistance to mitigate interference. In the proposed scheme, helper nodes overhear the signal from cellular users, encode the received packets, and send the encoded packets to the base station. We design the helper node selection scheme and the transmission policy of helper nodes. The performance of the proposed scheme for different positions of the cellular user and D2D users is then evaluated. The results suggest that the cellular transmission scheme should be adjusted dynamically when underlay D2D communications are active. Compared with the existing solutions, the proposed scheme can effectively increase system throughput.

Keywords

D2D; interference management; network coding; relay

1 Introduction

As an underlay to cellular network, device-to-device (D2D) technology has been intensively studied for fifth generation (5G) system. Resource reuse technology for D2D communications provides high data rate and improves spectrum efficiency [1], [2]. However, it also makes interference management more complicated. There-

fore, interference mitigation becomes a key problem for further development of D2D technology.

Power control and resource allocation are extensively discussed to guarantee the link quality of cellular users while D2D communications exist. On the other hand, network coding technology is a promising technique for improving network capacity. Applying network coding for multi-hop D2D communications is considered in [3] and [4]. Exploiting the inherent broadcasting nature of wireless medium, network coding can deliver multiple packets in a single transmission, and thus yield higher throughput.

Considering the situation where one cellular user device and two D2D pairs exploit the same resources, an interference coordination mechanism is proposed to enhance system capacity in [5], which can select proper D2D users without causing intensive interference. Channel allocation in a single cell system with D2D pairs is modeled in [6], and a scheme aiming to maximize the number of D2D pairs is proposed. An interference coordination scheme that does not allow different D2D pairs to share the same radio resources in a limited interference area is proposed in [7], and a power control scheme to obtain an upper bound of D2D transmitter power is also included to mitigate interference. In order to control the interference on cellular users, an algorithm for obtaining the upper bound of the number of D2D pairs is proposed in [8], which reuses the uplink resource under the conditions that the D2D users' location and channel state information (CSI) are unknown. With the channel state information, the transmission mode selection for maximizing the spectrum reusing ratio is studied in [9], and a lower bound of interference distance is derived according to the transmitter density and QoS requirement of a D2D pair. By using this distance, two resource allocation schemes are proposed, namely the dual metric scheme and the tolerant interference degree (TID) scheme. These schemes achieve more uniform resource allocation to avoid the excess interference of some resources.

There are some investigations on joint applications of network coding and D2D communications. In [10], the integration of D2D and network coding (NWC) technologies in cellular network is considered. The performance of two-time-slot and three-time-slot network coding technologies are studied from the perspectives of end-to-end signal to interference plus noise ratio (SINR) and spectral efficiency. In [11], three-time-slot network coding is further investigated to assist D2D transmission. In addition, the average power consumption is also evaluated. The issues studied in [10] and [11] are further discussed in [12]. An adaptive mode selection scheme and a resource allocation algorithm are proposed in [12] to further improve the end-to-end SINR and spectral efficiency. A routing protocol for network coding is designed in [13], and each individual link quality is enhanced by using relay-based network coding. CSI is a critical factor for network design, however, each node usually only has its own CSI, and lacks the CSI of other D2D pairs. To

This work is supported in part by the National Natural Science Foundation of China (Grant Nos. 61571351 and 61231008), and ZTE Industry-Academia-Research Cooperation Funds.

enable all the nodes to get global CSI, a network-coded information exchange scheme with an emphasis on minimizing the total transmission cost for exchanging CSI between nodes is proposed in [14]. Then, a transmission scheme with the object of load balancing is proposed to achieve the minimum transmission cost. The performance of cell range extension in relay-based D2D is studied in [15]. Then, a scheme integrating mode selection, resource allocation and power control is proposed, and the performance evaluation results show that the performance of decode-and-forward with network coding is superior to both the traditional cellular and the amplify-and-forward schemes.

Existing works focus on utilization of network coding to assist D2D transmission or routing protocols. To our best knowledge, this is the first paper which uses network coding technology to solve D2D uplink interference issue, as illustrated in **Fig. 1**. D2D communication pairs reuse the resource assigned to cellular user C_1 . The quality of cellular link is degraded due to D2D interference. The helper node C_2 overhears the packets on the cellular link and decodes these packets. After being motivated, the helper node transmits the recoded packets to BS. With our proposed scheme, the total number of packets sent by the source node is less than that under the traditional scheme.

First of all, we use random linear network coding (RLNC) for the transmission of cellular nodes. In this way, the base station (BS) only concerns the total number of received packets with no need to care about what packets it received. Some helpers, which may be idle cellular users, would overhear the packets sent from the source node. The helpers would forward the received packets when they are predicted to be helpful. In this work, the selection of helpers and the transmission scheme are the two key issues. We propose the helper selection method. Moreover, we analyze the performance of the network capacity with helper nodes, which facilitates us to design an efficient

transmission scheme for the helpers.

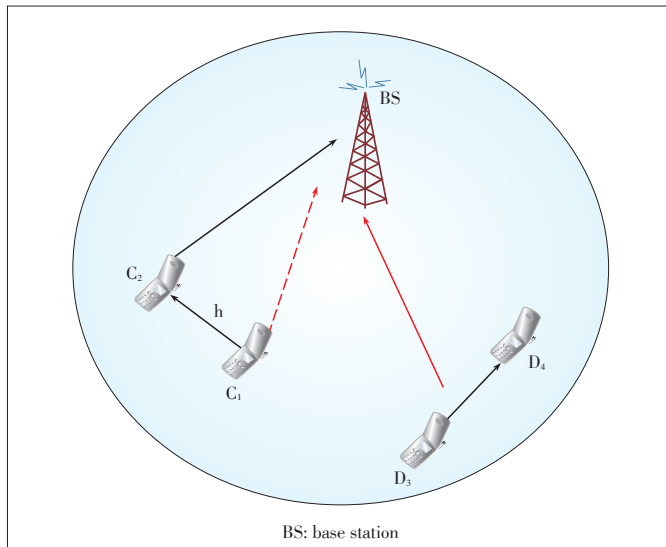
It is obvious that the performance gain provided by using helper nodes are different from that by different locations of cellular nodes. We analyze the performance gain as the function of the location of cellular users and D2D users respectively. We also conduct the simulations to demonstrate the performance gain of the proposed method.

The remainder of this paper is organized as follows. Section 2 describes the system model and presents some preliminaries. In section 3, a method for cellular users to transmit data with the assistance of helper nodes and network coding technology is stated. In section 4, we theoretically analyze the performance gain of our proposed scheme. Our simulation scenario and results are showed in section 5. Finally, we conclude the paper in section 6.

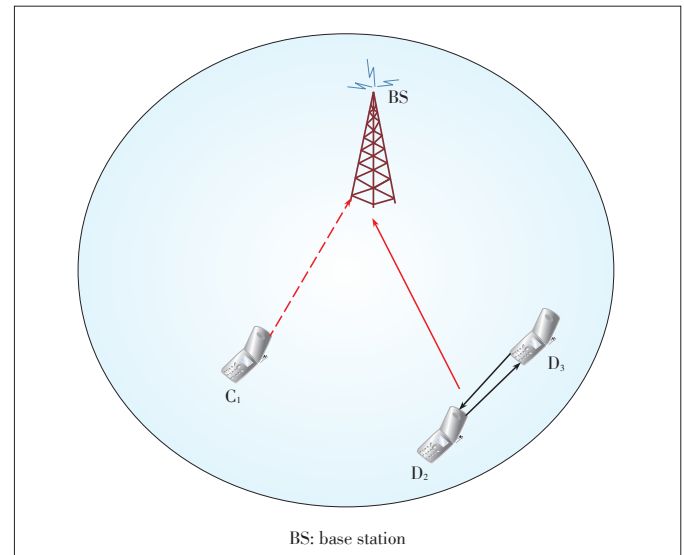
2 System Model

We consider a single-cell cellular network scenario with one cellular user communicating with BS. D2D users reuse the uplink period of network as depicted in **Fig. 2**. Since D2D users reuse the uplink resource block assigned to the cellular user, the quality of cellular user communications will be degraded. Inactive users are distributed randomly in the system, which may be selected as helpers.

We assume that the CSI of all involved links is known by the BS and that the CSI of the link between nodes can be acquired by the receiver. For each communication pair in system, we label the transmitter as i , the receiver as j and the interference node as k [15]. The large-scale fading is determined by the Euclidian distance d_{ij} between two users, and α represents for the path-loss exponent. A Rayleigh random variable f_{ij} determines the small-scale fading. The SINR at



▲ **Figure 1.** A cellular user communicates with BS via a helper node.



▲ **Figure 2.** D2D users communicate by reusing the uplink resource of cellular users in cellular network.

the receiver can be calculate as

$$SINR_j = \frac{P_{T_i} d_{i,j}^{-\alpha} h_{i,j}}{\sum_k P_{T_k} d_{k,j}^{-\alpha} h_{k,j} + \sigma^2} \quad (1)$$

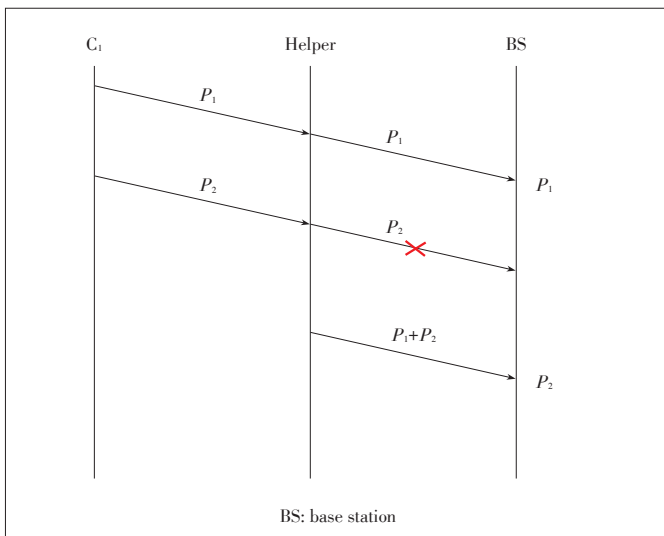
The relative bit error rate (BER) and packet error rate (PER) can be further obtained. The PER of a link is represented by e .

In this paper, RLNC technology is used for packet transmission. The helper node can simply send out a linear combination of their received packets. As shown in **Fig. 3**, the user C_1 sends out two packets P_1 and P_2 , we assume that the helper node overhears both the packets although one of the packets is lost at the BS. The helper then sends the coded packet $P_1 + P_2$ to the BS. With $P_1 + P_2$, the BS can always obtain all the packets no matter which one is lost.

The BS chooses helper nodes according to link condition information. As illustrated in Fig. 1, the BS estimates the number of packets that each helper node should store before transmitting and informs relevant helpers. Helpers overhear the packets from the cellular user and encode these packets. Traffic generated by cellular users is divided into data blocks, and each data block is called a generation. Each generation is composed of a certain number of packets. The BS needs to receive enough linearly independent packets to decode a generation. If the field size of RLNC is large enough, the probability of receiving linearly dependent coded packets is low. We consider one generation transmitting process.

3 The Proposed Interference Management Scheme

To mitigate the interference caused by D2D users, the proposed method selects helper nodes to assist transmission of cellular users, and network coding technology is also applied (Fig. 1). In the proposed scheme, the specific frequency re-



▲ Figure 3. Network coding technology.

source is assigned to one cellular user and only one D2D communication pair is assumed to reuse it.

We also assume that the BS knows link conditions of all the involved links. According to the method described in section 2, the link PER between the cellular user and BS $e_{U,B}$ can be obtained. When the value of $e_{U,B}$ is inferior to a specific level E , that is, the cellular user is severe influenced, the cellular user will be authorized for application of helper nodes. The value of E is associated with the traffic type.

3.1 Helper Selection

The geographical locations of users are different, which leads to complex link conditions between nodes. We assume that all idle nodes in coverage can be chosen as a helper node. The users can be divided into two categories according to their relative link conditions. **Fig. 4** shows the link conditions of a helper in the system, and e_i represents PER.

Quality of each individual link is also a critical factor for helper selection. We assume that e_0 is the upper bound of PER to be borne by the system. For a candidate helper node i , $e_{\max} = \max\{e_{i1}, e_{i2}\}$; if the link condition of node i satisfies the condition $e_0 \geq e_{\max}$, node i would be chosen as a candidate. The link condition of helpers are described in **Fig. 5**.

Based on Fig. 4, the type of candidate helpers can be distin-

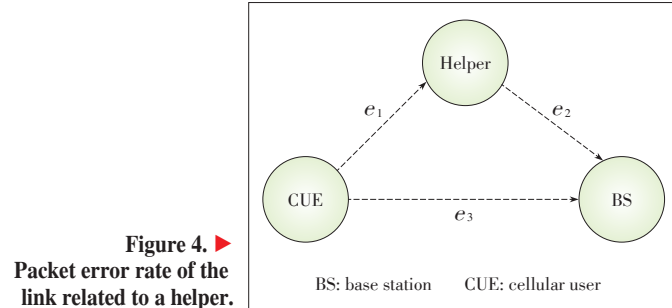


Figure 4. ▶ Packet error rate of the link related to a helper.

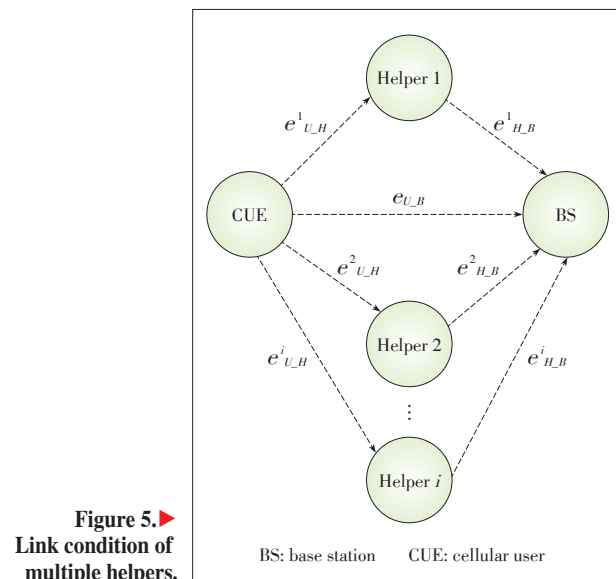


Figure 5. ▶ Link condition of multiple helpers.

guished by (2) and (3).

1) CSI of users satisfies:

$$(1 - e_1) \cdot e_3 \geq 1 - e_2, \quad (2)$$

2) CSI of users satisfies:

$$(1 - e_1) \cdot e_3 \leq 1 - e_2, \quad (3)$$

where $(1 - e_1)$ represents the probability that a helper receives a packet from the cellular user successfully. e_3 is the probability of transmission failure on cellular link to the BS. In the proposed scheme, helpers overhear the packets that the cellular user transmits to the BS. A packet received by helpers but not received by the BS is defined as an innovative packet. The product of $(1 - e_1)$ and e_3 is the probability that a helper receives an innovative packet.

Eq. (2) means that the probability of receiving an innovative packet is higher than transmitting one. From the perspective of packet number, a helper receives more innovative packets than it transmits. This kind of helper is able to transmit an innovative packet upon receiving it, since the link condition is able to support for continuous transmitting of innovative packets. Helper nodes delivering more packets means that the retransmission of the cellular user can be reduced. Therefore, this type of helpers is the prior choice.

3.2 Transmission Mode

Helper nodes overhear packets transmitted by the cellular user. Two types of helpers are motivated when enough packets are stored. Helpers transmit packets to the BS in turn after being motivated. It is obvious that this process can be divided into two parts: before and after helpers are motivated. During the period before helpers are motivated, all helpers are regarded as a whole unit of system. The packets that fail to transmit on the cellular link are innovative packets for the helper unit. If an innovative packet is received by at least one helper, we consider this packet is received by the helper unit. Helpers store packets and recode them before being motivated. Repeated transmissions can be decreased and even avoided since innovative packets are most likely coded. Each transmission from helper nodes contains new content so that the transmission efficiency is high, and that is the reason why network coding is applied. After receiving enough packets, the BS can decode all the packets and obtain the content. As stated above, the field size of linear network coding is large enough. Therefore, we consider that all the coded packets are linearly independent. The BS can decode a generation once receiving enough number of packets. The content of packets is not being concerned. Based on the theory, we study the problem from the perspective of packet number.

Assuming that BS needs to receive g packets to decode a generation. After being motivated, a helper and the cellular user transmit k packets. For the whole transmission process, the number of total packets transmitted by the cellular user is represented by X , and then we get:

resented by X , and then we get:

$$X \cdot e_{U,B} \cdot \left(1 - \prod_{i=1}^n e_{U,H}^i\right) = \sum_{j=1}^n k \{1 - e_{H,B}^j\}, \quad (4)$$

and

$$X \cdot (1 - e_{U,B}) + \sum_{j=1}^n k \left(1 - \prod_{i=1}^n (1 - e_{U,H}^i)\right) \cdot (1 - e_{H,B}^j) = g, \quad (5)$$

where n is the total number of helpers. PER of each link is constant. The value of k and X can be solved from (4) and (5). $\sum_{i=1}^n k \cdot (1 - e_{H,B}^i)$ denotes the total number of packets that the BS receives from all helpers, and $X \cdot (1 - e_{U,B})$ denotes the packet number from the cellular user. The left side of (4) represents the innovative packets the helper unit receives. Here we assume that new content is included in each packet, which means that the packets from the helper unit are innovative packets. Eq. (5) means that the packets the BS receives are from the helpers and the cellular user respectively.

4 Performance Evaluations

In this section, we give numerical simulation results to justify our analysis and to evaluate the performance of the proposed scheme. In conventional direct transmission schemes, a cellular user transmits packets to a BS continuously. When the BS receives enough packets for decoding, it sends a terminating signal to the cellular user to terminate the process.

The performance of the proposed system varies with different cellular user positions and different D2D positions. We assume that the interference sources are only D2D communications and Gaussian noise. The performance is evaluated from the perspective of packet number, and the receiving signal strength is only related with the distance. The simulation scenario and calculations of PER are described in section 2. **Table 1** shows the simulation parameters.

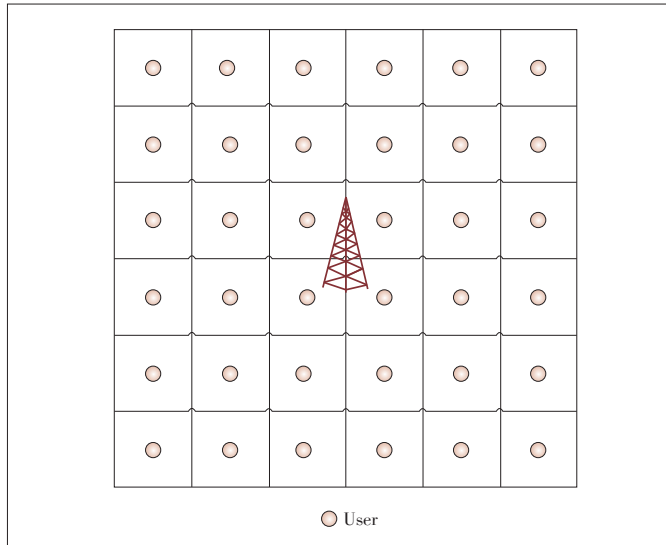
The cellular coverage is divided into many subareas (clusters) as illustrated in **Fig. 6**. The user performance at the center of the subarea is discussed to represent the users within the subarea. Cellular users and D2D users are located in different

▼Table 1. Simulation parameters

Simulation parameters	Value
Side length of cellular coverage	3000 m
Side length of subarea	200 m
BS coordinate	(1500, 1500)
Cellular transmit power	15 dBm
D2D transmit power	10 dBm
Target number of packets g	200
Pass-loss exponent α	2
Gaussian noise	-100 dBm

Network Coding-Based Interference Management Scheme in D2D Communications

WANG Shuang and HOU Ronghui



▲ Figure 6. Clusters and user locations in the cellular coverage.

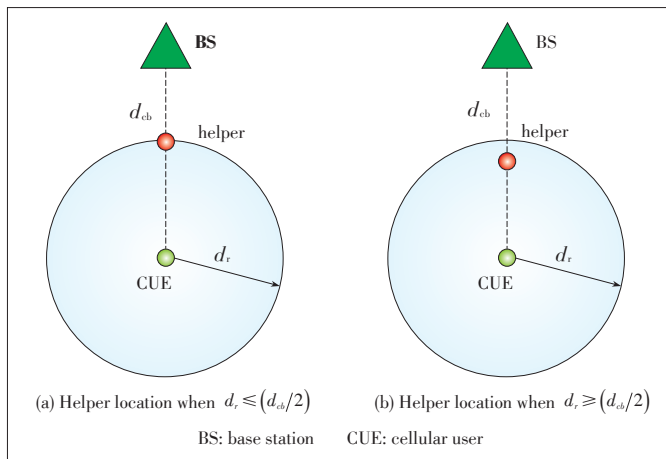
locations to evaluate the performance of the proposed scheme. These locations are represented by the red points in Fig. 6.

4.1 Location of Helper Nodes

To simplify the calculation, we assume that the location of the help node is optimal such that the network capacity is maximized. The distance between the BS and cellular user is d_{cb} , the range of helper selection is d_r . If $d_r \leq (d_{cb}/2)$, the helper node is placed at the boundary of the cluster, and located on the straight line between the BS and cellular user (Fig. 7a). If $d_r \geq (d_{cb}/2)$, the helper is placed on the straight line between the BS and cellular user (Fig. 7b). The distance between helper and cellular user is $d_{cb}/2$.

4.2 Performance Analysis of Different Cellular User Positions

We first consider that the locations of the BS and D2D users



▲ Figure 7. Helper locations.

are fixed. The coordinate of D2D is (2200, 400). Fig. 8 compares the total packet numbers for two different methods. Fig. 8a represents the performance of the method of direct transmission, while Fig. 8b represents the method with helper nodes. It is known that the farther the cellular user is from the BS, the severer the influence caused by D2D is. Our simulation results show that the proposed method decreases the packet number, especially when the cellular user is at the margin of coverage.

The performance gain of each subarea is illustrated in Fig. 9, in which the y-axis denotes the number of packets to be reduced by the proposed scheme. From the simulation result, we observe that the performance gain is related with the distance of the cellular user and BS. The farther the cellular user is from the BS, the larger the performance gain achieved by the proposed scheme is. With the increase of the distance between the cellular user and BS, the cellular link PER increases. Prob-

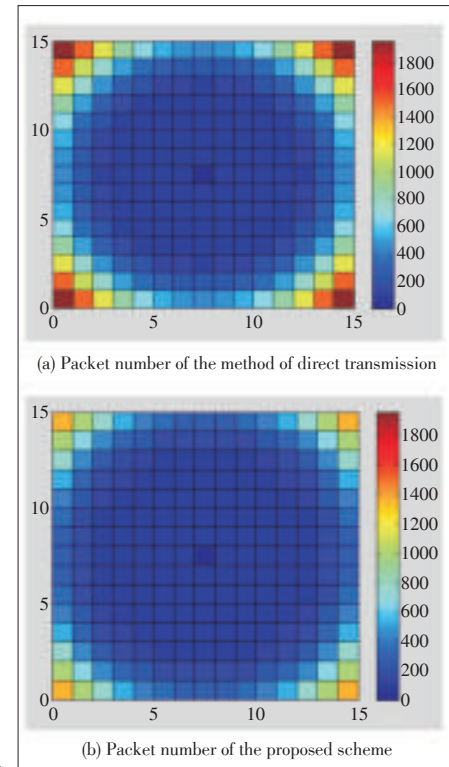


Figure 8. ▶ Comparison of the total packet numbers for direct transmission and the proposed scheme.

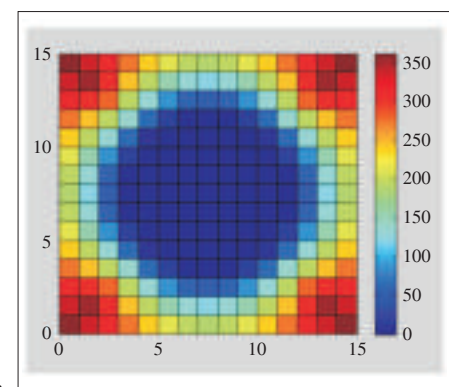


Figure 9. ▶ Promotion of packet number proposed scheme comparing with direct transmission.

ability of successful transmission for direct transmission decreases significantly, so that the cellular user has to transmit more packets to ensure enough packets received by BS. In the proposed scheme, the probability that a packet is received by at least one helper is high. With the increase of the number of helpers, the packet number would be reduced.

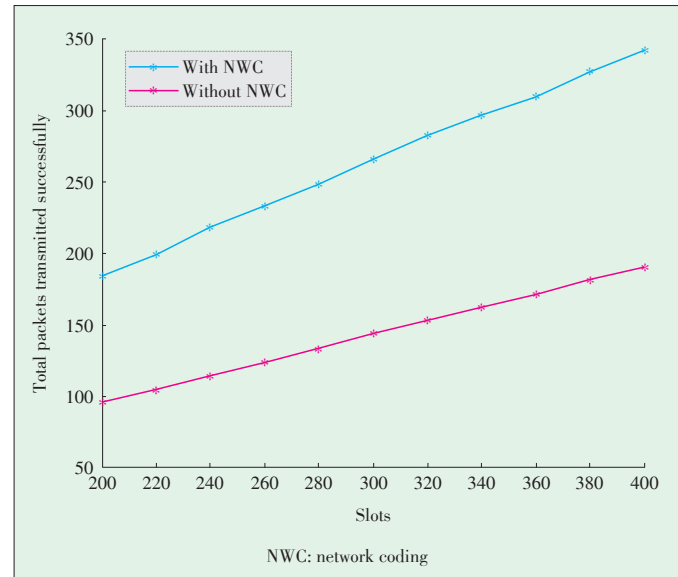
4.3 Performance Analysis of Different D2D Positions

D2D communications is not allowed if the interference on the cellular user is very severe. The interference factor W , which is defined as a constant, indicates the interference level that a transmission link can tolerate. The specific value of interference factor is determined by traffic type and the level of quality of service (QoS). The total packet number that the cellular transmits without D2D users is represented by S_1 , and the total packet number need to be delivered with the interference from D2D users is represented by S_2 . D2D users are allowed when the factors satisfy the condition: $[(S_1 - S_2)/S_1] \leq W$. Here we consider the interference factor W as 2.

Fig. 10 shows the simulation results. The subareas are colored blue when D2D communications are allowed. It is obvious that the area that D2D communications is allowed is extended when the proposed scheme is applied.

4.4 Application of Network Coding

Fig. 11 shows the comparison of whether the network coding



▲ **Figure 11.** Comparison of methods with or without NWC.

technology is applied. For the relay scheme without any helper node, the relay node overhears the packets from the source and delivers these packets to the BS directly. It can be seen that better performance can be achieved by network coding since the packets transmitted by the helper node are useful and unnecessarily repeated packets are mostly avoided.

5 Conclusions

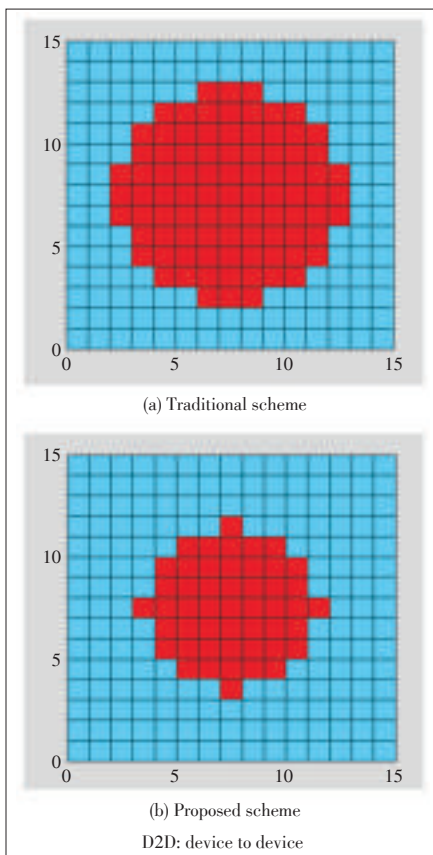
In this paper, a relay-based interference management framework with network coding technology is proposed to mitigate the interference from D2D users. We design the helper node selection scheme and the transmission policy of helper nodes. Performance evaluation verifies that the proposed scheme extends the region of D2D communication and improves the system throughput.

Acknowledgements

We would like to extend our sincere gratitude to XIE Yutang from ZTE Corporation for his instructive advice and useful suggestions on this paper.

References

- [1] G. Fodor and N. Reider, "A distributed power control scheme for cellular network assisted D2D communications," in *2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)*, Houston, USA, Dec. 2011, pp. 1–6. doi: 10.1109/GLOCOM.2011.6133537.
- [2] G. Fodor, E. Dahlman, S. Parkvall, et al., "Design aspects of cellular network assisted device-to-device communications," *IEEE Communication Magazine*, vol. 50, no.3, pp. 170–177, Mar. 2012. doi: 10.1109/MCOM.2012.6163598.
- [3] L. Wei, R. Q. Hu, Y. Qian, and G. Wu, "Enable device-to-device communications underlying cellular networks: challenges and research aspects," *IEEE Communication Magazine*, vol. 52, no. 6, pp. 90–96, Jun. 2014. doi: 10.1109/MCOM.2014.6829950.



◀ **Figure 10.** D2D communication regions in different schemes.

Network Coding-Based Interference Management Scheme in D2D Communications

WANG Shuang and HOU Ronghui

- [4] T. V. Vu, T. M. T. Nguyen, and G. Pujolle, "Distributed opportunistic and diffused coding in multihop wireless networks," in *IEEE International Communication Conference*, Ottawa, Canada, Nov. 2012, pp. 5583–5587. doi: 10.1109/ICC.2012.6364712.
- [5] X. W. Liao, J. G. Deng, and W. Wang, "Interference coordination mechanism for device-to-device communication in uplink period underlying cellular networks," in *International Conference on Wireless Communications & Signal Processing*, Huangshan, China, Oct. 2012, pp. 1–5. doi: 10.1109/WC.SP.2012.6542914.
- [6] R. Yin, T. Han, and G. D. Yu, "Interference-aware channel allocation for device-to-device communication underlying cellular networks," in *IEEE International Conference on Communications in China*, Beijing, China, Aug. 2012, pp. 422–427. doi: 10.1109/ICCChina.2012.6356919.
- [7] D. Y. WANG, X. Wang, and Z. Yuan, "An interference coordination scheme for device-to-device multicast in cellular networks," in *IEEE Vehicular Technology Conference*, Yokohama, Japan, Sept. 2012, pp. 1–5. doi: 10.1109/VTCFall.2012.6398945.
- [8] S. Shalmashi, G. W. Miao, and S. S. Ben, "Interference constrained device-to-device communications," in *IEEE International Conference on Communications*, Sydney, Australia, Jun. 2014, pp. 5245–5250. doi: 10.1109/ICC.2014.6884154.
- [9] Y. L. Xu, Y. Liu, and D. Li, "Resource management for interference mitigation in device-to-device communication," *IET Communications*, vol. 9, no. 9, pp. 1199–1207, Jun. 2015. doi: 10.1049/iet-com.2014.0976.
- [10] G. Fodor, A. Gattami, and A. Pradini, "Device-to-Device communications and network coding: friends or foes?" *IEEE COMSOC MMTC E-Letter*, vol. 9, no. 1, pp. 33–35, Jan. 2014.
- [11] A. Pradini and G. Fodor. (2014, Jun. 23–26). *Performance analysis of network coding schemes in network assisted D2D communications* [Online]. Available: https://www.metis2020.com/wp-content/uploads/publications/EuCNC_2014_Pradini_etal_D2D_NWC.pdf
- [12] G. Fodor, A. Pradini, and A. Gattami, "On applying network coding in network assisted D2D communications," in *20th European Wireless Conference*, Barcelona, Spain, 2014.
- [13] D. E. Lucani, M. V. Pedersen, and F. H. P. Fitzek, "Network coding to enhance standard routing protocols in wireless mesh networks," in *IEEE Globecom Workshops*, Georgia, USA, Dec. 2013, pp. 610–616. doi: 10.1109/GLOCOMW.2013.6825055.
- [14] X. M. Wang, C. Yuen, T. J. Li, W. T. Song, and Y. L. Xu, "Minimizing transmission cost for third-party information exchange with network coding," *IEEE Transactions on Mobile Computing*, vol. 14, no. 6, pp. 1218–1230, Aug. 2014. doi: 10.1109/TMC.2014.2350989.
- [15] A. Abrardo, G. Fodor, and B. Tola, "Network coding schemes for device-to-device communications based relaying for cellular coverage extension," in *IEEE International Workshop on Signal Processing Advances in Wireless Communications*, Stockholm, Sweden, Jun. 2015, pp. 670–674. doi: 10.1109/SPAWC.2015.7227122.

Manuscript received: 2015-12-03

Biographies

WANG Shuang (wangshuangv086@163.com) received the B.Eng. and M.Eng. degrees in communication engineering from Xidian University in 2013 and 2016, respectively. Her research interests include D2D communication and wireless networks. Since 2016, she has been with State Grid Corporation of China, where she is currently an engineer.

HOU Ronghui (rhhou@xidian.edu.cn) received the B.Eng., M.Eng., and Ph.D. degrees in communication engineering from Northwestern Polytechnical University, China in 2002, 2005, and 2007, respectively. She was a post-doctoral fellow with the Department of Electrical and Electronic Engineering, The University of Hong Kong, China, from 2007 to 2009. Since 2009, she has been with Xidian University, China, where she is currently a professor with the Department of Telecommunication Engineering. Her research interests include network quality of service issues, routing algorithm design, and wireless networks.

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min¹, ZHOU Yimin¹, LUO Minke¹, and ZUO Wen²

(1. School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China;

2. Audio and Video Technology Platform Department, ZTE Corporation, Shenzhen 518057, China)



Abstract

Rate control plays a critical role in achieving perceivable video quality under a variable bit rate, limited buffer sizes and low delay applications. Since a rate control system exhibits non-linear and unpredictable characteristics, it is difficult to establish a very accurate rate-distortion (R-D) model and acquire effective rate control performance. Considering the excellent control ability and low computing complexity of the fuzzy logic in non-linear systems, this paper proposes a bit-rate control algorithm based on a fuzzy controller, named the Fuzzy Rate Control Algorithm (FRCA), for All-Intra (AI) and low-delay (LD) video source coding. Contributions of the proposed FRCA mainly consist of four aspects. First, fuzzy logic is adopted to minimize the deviation between the actual and the target buffer size in the hypothetical reference decoder (HRD). Second, a fast lookup table is employed in fuzzy rate control, which reduces computing cost of the control process. Third, an input domain determination scheme is proposed to improve the precision of the fuzzy controller. Fourth, a novel scene change detection is introduced and integrated in the FRCA to adaptively adjust the Group-of-Pictures (GOP) length when the source content fluctuates. The FRCA can be transplanted and implemented in various industry coders. Extensive experiments show that the FRCA has accurate variable bit-rate control ability and maintains a steady buffer size during the encoding processes. Compared with the default configuration encoding under AI and LD, the proposed FRCA can achieve the target bit rates more accurately in various classical encoders.



Keywords

rate control; video coding; fuzzy control; bit per pixel; rate-distortion model

1 Introduction

With the widespread development of various video streaming and multimedia networking applications, such as mobile TV, live video broadcasting and home cinema, a desire for high quality and low delay is increasing rapidly. In the video coding technology, rate control (RC) is an important tool to strengthen the coding efficiency, which can maximize visual quality under limited bandwidth and buffer capacity. RC can be classified into two types, the constant bit rate (CBR) and the variable bit rate (VBR). The CBR allocates an uniform bit to different coding units regardless of their characteristics. Correspondingly the CBR scheme leads to frequent fluctuation and degradation of the picture quality for consecutive pictures with fast motion or scene change. Compared with the CBR, the VBR scheme can dynamically adjust the target bit rate according to characteristics of video content and obtain the consistent picture quality, but it may cause extensive buffer delay.

RC regulates the encoder output bit rate by adjusting the quantization parameters (QP) to optimize the video quality under an available channel bandwidth. Thus, many RC algorithms concentrate on an accurate rate-distortion (R-D) model and an efficient bit allocation scheme. Along with the video coding standard developing, MPEG-2, H.263, MPEG-4, H.264/AVC and HEVC integrate RC algorithms into their encoders. Meanwhile, various improved RC algorithms [1], [2], [3] have been investigated.

The R-D model is the kernel model for the great majority of RC algorithm since the QP value assignment depends on the transcendental R-D model. Based on discrete mathematics function, various R-D models were derived. T. Chiang, et al. [4] proposed a widely used quadratic R-D model by assuming the video source statistics are Laplacian distribution and expanding the rate distortion function into a Taylor series. Through observing the strong relationship between Q_{step} and the quantization parameter, Z. Li, et al. [5] improved the quadratic R-D model by using the predicted mean absolute difference (MAD) and Q_{step} instead of QP. To identify the impacts of the parameter Lambda in rate control, B. Li, et al. [6] proposed $R-\lambda$ model and adopted it into High Efficiency Video Coding (HEVC) rate control. Besides, ρ -domain rate control [7] was proposed by Z. He, et al., where ρ is the zero ratios of transformed coefficients after quantization. In our previous work, a concise exponential R-Q model [8] was proposed for H.264/AVC, which takes picture complexity, gradient and histogram information into consideration.

To provide a significant improvement in the coding efficiency, the study of RC has focused on the R-D [9] or the rate-quantization (R-Q) [10] model, and also dealt with the Intra only RC [11], the scalable video coding RC [12] and the bit allocation schemes [13]. These studies are important parts in RC and have helped develop various video compression standards,

This work is supported by ZTE Industry-Academia-Research Cooperation Funds under Grant No. CON1503180004, the Postdoctoral Science Foundation of China under Grant No. 2014M552342, and the Foundation of Science and Technology Department of Sichuan Province, China under Grant No. 2014GZ0005.

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

such as MPEG-2, H.263, MPEG-4, H.264/AVC and HEVC.

Fuzzy logic has become a successful intelligent technology for control process [14]. Compared with traditional controllers such as proportional-integral-derivative (PID), proportional-integral (PI) and proportional-derivative (PD), fuzzy control does not require any internal mathematical model and can be applied to uncertainty and imprecision system [15]. Traditional controllers have been used to control over 90% industrial processes [14]. However, for more complex systems, such as time delay, time varying parameters, models mismatch and non-linearity, those controllers cannot generate satisfactory system performance since they depend on the mathematical model [16]. With the development of intelligent control, the fuzzy controller is adopted in industrial fields such as transportation devices, intelligent machines, power engineering, and chemical processes [17]. In the medical area, fuzzy logic effects on diagnosis, treatment of illness, patient pursuit, prediction of disease risk and other medical fields [18]. Besides, fuzzy controllers are also used for food control [19], network control [20] and so on. The experiments in literature show that fuzzy controllers adopted in related areas obtain effective performance, robustness and overall stability, and overcome the shortages of traditional controllers, especially in uncertain and complex systems.

RC systems also adopted the fuzzy logic into the coding process. D. H. Tsang, et al. [21] proposed a fuzzy logic-based control scheme for real-time MPEG video to avoid long delay or excessive loss at the user-network interface (UNI) in an asynchronous transfer mode (ATM) network. S. Sheu et al. [22] presented a fuzzy adaptive rate control scheme to select the transmission rate for frame transmissions in wireless LANs. M. Rezaei, et al. [23] introduced a semi-fuzzy rate control algorithm that utilized a fuzzy rate controller and a quality controller to adjust the QP value. However, existed fuzzy controllers based RC schemes are only used for certain video coding standards, or simple experience parameters are used into the fuzzy controller. To improve the efficiency and accuracy and to reduce the computational complexity of the rate control system, this paper proposes a novel adaptive RC approach, named the Fuzzy Rate Control Algorithm (FRCA). This is an input domain determination scheme to improve the precision of the fuzzy controller and adopts a fuzzy logic look-up table to derive increment QP values to minimize the deviation between the actual and the target buffer sizes in the hypothetical reference decoder (HRD).

The proposed FRCA controls the bit rate by adjusting the QP value. It employs an improved fuzzy controller that utilizes an exponential R-D model to determine the input domain. Then, the proposed fuzzy controller is adopted to generate a fast lookup table. Through looking up the fast table, the FRCA uses the signals from the buffer to calculate the increment QP value, which is used for the RC process. The proposed FRCA also presents a novel scene change detection to adaptively adjust the Group-of-Pictures (GOP) length when the source con-

tent fluctuates. The proposed FRCA has high adaptability and can be transplanted and implemented in various industry coders. In this paper, FRCA is implemented on MPEG-2, H.263, MPEG-4, H.264/AVC and HEVC encoders. Simulations and analysis show that the proposed FRCA provides good performance on the peak signal-to-noise ratio (PSNR) gains and bit-rate savings.

The remainder of this paper is organized as follows. Section 2 introduces a fuzzy logic controller for the low delay video coding rate control. Section 3 presents a new input domain determination for the fuzzy controller. Section 4 proposes a scene change detection scheme and Section 5 gives the description of the proposed RC algorithm step by step. Simulation results and performance analysis are described in section 6. Finally, Section 7 concludes the paper.

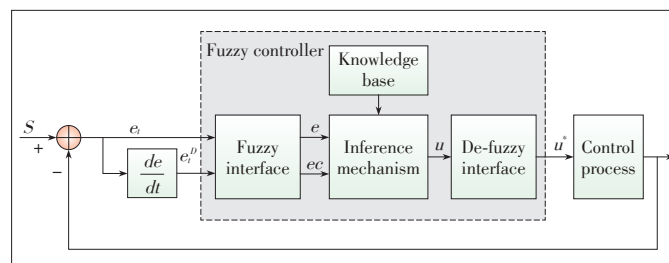
2 Fuzzy Controller for Rate Control

Different from traditional controller systems with exact mathematical models, the fuzzy controller has an intelligent control process that is set according to the experience values and can be applied into unpredictable or uncertain system. In this paper, we adopt a fuzzy controller to regulate the suitable QP value for encoding.

2.1 Structure of Fuzzy Controller

Fig. 1 shows the structure of the proposed fuzzy logic controller used for RC. It is composed of the fuzzy interface, the knowledge base (rule base), the inference mechanism and the defuzzy interface. First, a fuzzy interface transforms an exact measured value into a fuzzy value to fit the fuzzy calculation. Second, this fuzzy value is utilized to calculate the fuzzy output value by the fuzzy control knowledge base and the inference mechanism. Third, the fuzzy system converts the fuzzy output value to a precise value to control the coding process.

The fuzzy controller (Fig. 1) has two input variables and an output variable. The input variable in the video coding consists of two components, which are the buffer deviation and the buffer deviation rate of changes. Compared with the multiple-input multipleoutput fuzzy controller, the proposed controller has simple architecture that results in low computing complexity. Meanwhile, it can accurately reflect the dynamic characteristics of the output variable during the control process.



▲ Figure 1. Structure of fuzzy controller.

At the encoding time t , the variable target bit rate TBR_t can be converted to the target bit per pixel $Tbpp_t$ by (1).

$$Tbpp_t = TBR_t / (W \cdot H \cdot F_{rate}), \quad (1)$$

where F_{rate} indicates the actual frame rate and W and H are frame's width and height.

After the t -th frame is encoded, let R_t denote the actual output bits, then the current buffer size B_t can be updated by (2).

$$B_t = B_{t-1} + R_t / (W \cdot H) - Tbpp_t. \quad (2)$$

2.2 Fuzzy Interface

It is important to recognize input variables and output variables of the fuzzy control system. Generally, the fuzzy control system defines two variables as the input variables. One variable is the deviation between the actual value and system default value. Another is the change rate of this deviation. The control value is determined as the output of the system. Since QP value is the principal term that is employed to adjust the encoder output bit rate and avoid over-flow and under-flow of the buffers in video decoders, we take QP as the output of the system in this fuzzy controller.

Generally, the content and motion complexity of the successive pictures have a high correlation, so the QP at the coding time $t-1$ is very close to that of the coding time t . Without loss of generality, we define the incremental QP as output variable u^* , and then calculate the current QP based on the incremental QP.

For low delay video capture and transmission system, a relatively small buffer size is adequate. However, for random access applications such as network televisions, a large buffer size is obligatory. When feeding the bit stream into a HRD with suitable parameters, the HRD buffer should be neither overflow nor underflow. The overflow brings unexpected frame skipping in the encoder, which may result in visual quality degrades. On the other hand, the underflow can lead to under utilization of the available network bandwidth.

We define the deviation between the target buffer level B_T and current buffer size B_t as the input variable e_t , define the change rate of the deviation as another input variable e_t^D . Generally, in order to avoid overflow and underflow and to reduce the deviation between the target buffer size and the current buffer level, B_T is set to constant value 0. e_t and e_t^D can be calculated by (3) and (4).

$$e_t = B_t - B_T, \quad (3)$$

$$e_t^D = e_t - e_{t-1}. \quad (4)$$

After the fuzzy controller inputs and outputs are selected, we define the fuzzy control system. Fuzzy control is a fuzzy logic based control method. In the fuzzy logic, fuzzy input variables are used for the fuzzy inference. Thus, precise inputs

should be converted to the fuzzy subset by the fuzzy interface, which can be interpreted and compared to the rules in the rule-base. The actual input variables are e_t and e_t^D , the scaled variables are E and EC , we define the domain of the scaled variables with finite integer. The domain S can be expressed by (5).

$$S = \{-6, -5, -4, -3, -2, -1, 0, +1, +2, +3, +4, +5, +6\}. \quad (5)$$

The practical domains of input variables e_t and e_t^D are different from the domain S , so the actual input variables e_t and e_t^D should be scaled into the scaled variables in domain $[-6, 6]$. If the values of input variables e_t and e_t^D range in $[a_E, b_E]$ and $[a_{EC}, b_{EC}]$ respectively, the e_t and e_t^D can be scaled to E and EC by (6),

$$\begin{cases} E = \left\lfloor \frac{12 \cdot (e_t - (a_E + b_E)/2)}{b_E - a_E} \right\rfloor \\ EC = \left\lfloor \frac{12 \cdot (e_t^D - (a_{EC} + b_{EC})/2)}{b_{EC} - a_{EC}} \right\rfloor \end{cases} \quad (6)$$

After scaling, scaled variables E and EC are then fuzzified by input fuzzy sets. Input fuzzy sets are defined on domains S . Since the number of fuzzy segmentation determines the fuzzy control accuracy, the fuzzy segmentation is significant in the fuzzy controller. Designing more fuzzy segmentation levels results in more control rules and increases the computing complexity. On the contrary, less fuzzy segmentation level degrades the control precision and has low calculating complexity. To maintain the trade-off between precision and complexity, the most fuzzy control literature [24] adopts the fuzzy controller, which divides the linguistic variable into 7 fuzzy sets, to control systems and simulations show this fuzzy controller can obtain effective control performance. In this paper, we also use 7 fuzzy sets for input and output variables. Each input fuzzy set is assigned a linguistic name: negative big (NB), negative medium (NM), negative small (NS), zero (ZO), positive small (PS), positive medium (PM), positive big (PB). We define fuzzy sets of E , EC and U as $\{NB, NM, MS, ZO, PS, PM, PB\}$.

2.3 Knowledge Base

Fuzzification results are used in fuzzy rules to make combined membership values for fuzzy inference. Once the domain S and the fuzzy segmentation are selected, the membership function is required, which is a quantitative description of the fuzzy conception and is the basis of the fuzzy controller. It can be utilized to transform the quantized precise inputs into fuzzy sets. However, it is difficult to define the uniform membership function type. In order to decrease computational complexity and acquire low delay, we choose a simple triangle membership function to change the values of E , EC and U into a membership value that is confined to $[0, 1]$.

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

Table 1 gives the membership values of the linguistic variables E , EC and U for fuzzy sets. The first row and column of Table 1 are 13 values of the domain S and the fuzzy sets.

The fuzzy control rules are set up based on the expert knowledge or the manual operation. Control rules are critical to the fuzzy control system, since the quantity and accuracy of the rules have an effect on the performance of the control system. In this paper, we adopt the double-input single-output fuzzy controller, the double inputs are E and EC , the single output is the control variables U . So the fuzzy control rules in this paper is IF E AND EC THEN U . Since the fuzzy spatial segmentation of the proposed fuzzy controller has seven levels, we can get $7 \times 7 = 49$ fuzzy control rules which represent the relationships between inputs (E and EC) and output (U).

Table 2 gives the fuzzy control rules in this work. The first row and column of Table 2 are the fuzzy sets of E and the fuzzy sets of EC respectively. Other values are the elements of the fuzzy sets of U .

2.4 Inference Mechanism

Inference mechanism emulates the expert's decision in interpreting and applying knowledge to control the process. By inference mechanism, the membership values produced by fuzzification are transformed to the output fuzzy set.

If a rule in Table 2 is IF E is A_i AND EC is B_i THEN U is C_i , where A_i , B_i and C_i are fuzzy sets, these fuzzy sets are

▼ Table 1. Membership function

	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
PB	1	0.5	0	0	0	0	0	0	0	0	0	0	0
PM	0	0	0.5	1	0.5	0	0	0	0	0	0	0	0
PS	0	0	0	0	0.5	1	0.5	0	0	0	0	0	0
ZO	0	0	0	0	0	0.5	1	0.5	0	0	0	0	0
NS	0	0	0	0	0	0	0	0.5	1	0.5	0	0	0
NM	0	0	0	0	0	0	0	0	0	0.5	1	0.5	0
NB	0	0	0	0	0	0	0	0	0	0	0	0.5	1

NB: negative big NS: negative small PM: positive medium ZO: zero
NM: negative medium PB: positive big PS: positive small

▼ Table 2. Fuzzy control rules

	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	NS	ZO
NM	NB	NM	NM	NS	NS	ZO	PS
NS	NM	NM	NS	NS	ZO	PS	PS
ZO	NM	NS	NS	ZO	PS	PS	PM
PS	NS	NS	ZO	PS	PS	PM	PM
PM	NS	ZO	PS	PS	PM	PM	PB
PB	ZO	PS	PS	PM	PB	PB	PB

NB: negative big NS: negative small PM: positive medium ZO: zero
NM: negative medium PB: positive big PS: positive small

defined by membership function in(7), (8) and (9).

$$A_i = \int_E \mu_{A_i}(e)/E, \quad (7)$$

$$B_i = \int_{EC} \mu_{B_i}(ec)/EC, \quad (8)$$

$$C_i = \int_U \mu_{C_i}(u)/U, \quad (9)$$

where $\mu_{A_i}(e)$, $\mu_{B_i}(ec)$ and $\mu_{C_i}(u)$ are membership values. The rule deduces a fuzzy relation R_i by (10).

$$R_i = A_i \otimes B_i \otimes C_i, \quad (10)$$

where \otimes is the mini-operation rule of fuzzy implication. The ultimate fuzzy relation Re can be expressed by (11).

$$Re = \bigcup R_i, \quad (11)$$

where \bigcup represents the union operation.

When inputs E and EC are given as A^* and B^* , the control variables \hat{U} can be calculated by (12).

$$\hat{U} = A^* \otimes B^* \oplus Re, \quad (12)$$

where A^* , B^* and \hat{U} are fuzzy sets, \oplus is the sup-min compositional operator.

2.5 Defuzzy Interface

The output control value \hat{U} generated by the inference mechanism is a fuzzy value, which cannot be directly used in the monitor. The defuzzification converts the result obtained by the inference mechanism into the exact value which can be applied to control the process. In this paper, the fuzzy set \hat{U} expressed by membership values should be transformed to \hat{u} by the center of gravity method in (13).

$$\hat{u} = \frac{\sum_i \mu_{\hat{U}}(U_i) \cdot U_i}{\sum_i \mu_{\hat{U}}(U_i)}. \quad (13)$$

Fuzzy sets A^* and B^* can be acquired according to Tables 1 and 2 after two input variables are calculated. Then the control variable \hat{u} can be calculated according to (10), (12) and (13).

In order to achieve low computation work, we built a fuzzy logic querying table shown in **Table 3** to represent the relationship between the inputs e, ec and output u . Through looking up this table, we can obtain the control variable based on the input variables e_i and e_i^D .

The output control value \hat{u} in \hat{U} cannot be directly used in the RC system since the domain of the \hat{u} is different from the domain of adjustable parameter ΔQP . Thus, the scale factor K_u is employed to converts the value in Table 3 to the actual control value u^* , which can be expressed by (14).

$$u^* = [\hat{u} \cdot K_u]. \quad (14)$$

The maximum range of \hat{u} value is ± 6 , while the QP value variation allowed in rate control amplitude is ± 3 , so the scale

▼Table 3. Querying table of fuzzy control

	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-4.8	-4.8	-4.8	-4.8	-3.6	-3.6	-3.2	-3.2	-2.0	-2.0	-0.3	-0.3	0.0
-5	-4.8	-4.8	-4.8	-4.8	-3.6	-3.6	-3.2	-3.2	-2.0	-2.0	-0.3	-0.3	0.0
-4	-4.8	-4.8	-3.6	-3.6	-3.6	-3.6	-2.0	-2.0	-1.1	-1.1	0.0	0.0	0.3
-3	-4.8	-4.8	-3.6	-3.6	-3.6	-3.6	-2.0	-2.0	-1.1	-1.1	0.0	0.0	0.3
-2	-3.6	-3.6	-3.6	-3.6	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0
-1	-3.6	-3.6	-3.6	-3.6	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0
0	-3.2	-3.2	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0	2.0	3.2
1	-3.2	-3.2	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0	2.0	3.2
2	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0	2.0	3.6	3.6	3.6
3	-2.0	-2.0	-1.1	-1.1	0.0	0.0	1.1	1.1	2.0	2.0	3.6	3.6	3.6
4	-0.3	-0.3	0.0	0.0	1.1	1.1	2.4	2.4	3.6	3.6	3.6	3.6	4.8
5	-0.3	-0.3	0.0	0.0	1.1	1.1	2.4	2.4	3.6	3.6	3.6	3.6	4.8
6	0.0	0.0	0.3	0.3	2.0	2.0	3.6	3.6	3.9	3.9	4.8	4.8	4.8

factor K_u can be derived from (15).

$$K_u = \max\{\Delta QP\} / \max_{\hat{u} \in S} \{\hat{u}\} = 3/5 = 0.6. \quad (15)$$

After scaling, a fast lookup table used for encoding process is obtained (Table 4).

2.6 QP Computation in Rate Control

The incremental QP value ΔQ can be calculated based on the proposed fuzzy control scheme. This ΔQ and the previous frame QP value Q_{t-1} are adopted to calculate the QP value, which can be expressed by (16), to encode the current frame.

$$Q_t^* = Q_{t-1} + \Delta Q = Q_{t-1} + u^*. \quad (16)$$

As usual, the QP value is bounded in $[Q_{\min}, Q_{\max}]$, which in-

▼Table 4. Fuzzy rate control fast lookup table

	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-5	-5	-5	-5	-4	-4	-3	-3	-2	-2	0	0	0
-5	-5	-5	-5	-5	-4	-4	-3	-3	-2	-2	0	0	0
-4	-5	-5	-4	-4	-4	-4	-2	-2	-1	-1	0	0	0
-3	-5	-5	-4	-4	-4	-4	-2	-2	-1	-1	0	0	0
-2	-4	-4	-4	-4	-2	-2	-1	-1	0	0	1	1	2
-1	-4	-4	-4	-4	-2	-2	-1	-1	0	0	1	1	2
0	-3	-3	-2	-2	-1	-1	0	0	1	1	2	2	3
1	-3	-3	-2	-2	-1	-1	0	0	1	1	2	2	3
2	-2	-2	-1	-1	0	0	1	1	2	2	4	4	4
3	-2	-2	-1	-1	0	0	1	1	2	2	4	4	4
4	0	0	0	0	1	1	2	2	4	4	4	4	5
5	0	0	0	0	1	1	2	2	4	4	4	4	5
6	0	0	0	0	2	2	4	4	4	4	5	5	5

tends to avoid large quality fluctuation between two consecutive frames. Therefore, all the derived QP values are clipped by (17).

$$Q_t = \max\{Q_{\min}, \min\{Q_{\max}, Q_t^*\}\}. \quad (17)$$

3 Input Domain Determination for Fuzzy Controller

In a fuzzy controller, the practical domain of input variables e_t and e_t^D can be converted to the internal domain E and EC based on the practical variation range of input variables. To quantize e_t and e_t^D , we pre-calculate the practical variation range of input variables $[a_E, b_E]$ and $[a_{EC}, b_{EC}]$. This work inherits the exponential R-D model to pre-calculate the practical variation range, which is expressed in (18).

$$R_{bpp} = \alpha \cdot e^{-\beta \cdot Q}, \quad (18)$$

where R_{bpp} represents output encoded bits per pixel, Q denotes quantization parameter QP, and α and β are model parameters.

We run tests on different representative video sequences. Table 5 lists the exponent fitting results including α , β and the Pearson correlation coefficient P^2 . P^2 is the correlation between the model and actual R-Q data. The correlation is stronger with P^2 being closer to 1, and vice versa. Table 5 shows that the R-Q curve intercept α drastically changes, but the curvature β changes steadily. For example, in the 720p sequence, the min and max value of α are 6.8911 and 27.178 respectively, however β is bounded in $[0.0756, 0.0949]$. Thus, we adopt β to pre-calculate the practical variation range.

The first-order and second-order differentials of R-Q model (18) can be expressed by

$$\frac{dR_{bpp}}{dQ} = \alpha \cdot e^{-\beta \cdot Q} \cdot (-\beta) = -\beta \cdot R_{bpp}, \quad (19)$$

$$\frac{d^2 R_{bpp}}{dQ^2} = -\beta \cdot \alpha \cdot e^{-\beta \cdot Q} \cdot (-\beta) = \beta^2 \cdot R_{bpp}. \quad (20)$$

When QP varies in a small range, we suppose $\Delta R_{bpp} \approx dR_{bpp}$, $\Delta Q \approx dQ$, $\Delta R_{bpp}^2 \approx d^2 R_{bpp}$, $\Delta Q^2 \approx dQ^2$. Therefore, if ΔQ bounds in $[-3, 3]$, the practical variation range of the two input domain can be calculated by (21).

$$[a_E, b_E] = [-\Delta R_{bpp}, \Delta R_{bpp}] = [-\beta \cdot R_{bpp} \cdot 3, \beta \cdot R_{bpp} \cdot 3]. \quad (21)$$

Since β changes steadily, the mean of the β in Table 5 is adopted to calculate (21) and (22). The R_{bpp} can be approx-

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

▼ Table 5. Exponential R-D model regression results

Resolution	Seq. name	MPEG-2			MPEG-4			H.263+			H.264/AVC		
		α	β	P^2	α	β	P^2	α	β	P^2	α	β	P^2
QCIF	Akiyo	0.076	0.045	0.932	0.041	0.050	0.846	0.039	0.046	0.827	0.589	0.111	0.979
	Carphone	0.222	0.066	0.930	0.129	0.063	0.866	0.145	0.072	0.903	3.213	0.132	0.996
	Coastguard	0.538	0.083	0.958	0.314	0.088	0.937	0.376	0.090	0.944	20.495	0.168	0.998
	Deadline	0.223	0.070	0.952	0.141	0.080	0.915	0.163	0.086	0.947	3.296	0.133	0.999
CIF	Bus	3.299	0.075	0.966	2.391	0.081	0.960	2.775	0.081	0.962	60.099	0.142	1.000
	Coastguard	2.485	0.083	0.961	1.692	0.087	0.944	1.831	0.086	0.945	149.425	0.177	0.995
	Football	2.288	0.065	0.950	1.833	0.068	0.946	2.133	0.061	0.938	38.644	0.122	0.998
	Harbour	3.422	0.084	0.973	2.553	0.093	0.965	2.773	0.094	0.968	150.684	0.170	0.995
4CIF	City	5.943	0.077	0.930	3.468	0.072	0.900	3.935	0.072	0.909	411.483	0.188	0.986
	Crew	3.991	0.049	0.876	2.966	0.047	0.871	3.206	0.040	0.857	192.370	0.154	0.990
	Harbour	7.984	0.076	0.955	5.954	0.079	0.945	6.297	0.079	0.946	451.766	0.165	0.999
	Ice	1.901	0.050	0.912	1.258	0.019	0.676	1.543	0.043	0.892	27.528	0.120	0.992
WQVGA	BasketballPass	0.974	0.066	0.939	0.677	0.064	0.904	0.731	0.066	0.919	17.868	0.130	1.000
	BlowingBubbles	1.915	0.089	0.957	1.137	0.089	0.930	1.316	0.090	0.940	54.966	0.161	0.999
	BQSquare	3.133	0.083	0.984	2.444	0.096	0.969	2.756	0.099	0.981	109.042	0.175	0.995
	RaceHorses	2.094	0.081	0.941	1.362	0.079	0.919	1.501	0.079	0.918	45.569	0.141	1.000
WVGA	BasketballDrill	3.009	0.060	0.912	2.094	0.054	0.884	2.247	0.052	0.889	60.816	0.132	0.998
	BQMall	4.215	0.068	0.942	2.929	0.065	0.922	3.119	0.066	0.927	93.920	0.141	0.998
	PartyScene	12.841	0.091	0.980	9.566	0.100	0.969	10.112	0.100	0.970	385.835	0.162	0.998
	RaceHorses	8.660	0.080	0.957	6.081	0.082	0.932	6.509	0.076	0.926	375.714	0.162	1.000
720p	DucksTakeOff	27.178	0.084	0.965	20.618	0.087	0.956	21.058	0.086	0.955	1201.25	0.157	0.997
	OldTownCross	6.891	0.080	0.912	3.562	0.053	0.795	3.999	0.067	0.867	450.266	0.196	0.964
	Shields	8.914	0.076	0.904	4.908	0.063	0.872	5.247	0.065	0.888	1244.72	0.210	0.973
	Mobcal	17.292	0.095	0.954	9.766	0.089	0.921	10.886	0.092	0.934	2125.11	0.226	0.988
1080p	Beach	14.590	0.061	0.962	12.128	0.053	0.955	16.682	0.043	0.960	268.107	0.123	0.999
	BQTerrace	26.421	0.095	0.955	15.345	0.080	0.945	17.863	0.084	0.955	5276.24	0.227	0.990
	Cactus	13.595	0.077	0.936	8.201	0.056	0.917	9.026	0.060	0.937	2699.6	0.209	0.980
	ParkScene	14.508	0.081	0.944	8.636	0.062	0.915	9.245	0.066	0.927	969.87	0.179	0.991
Average			0.0746	0.9442		0.0713	0.9098		0.0728	0.9261		0.1611	0.9927

CIF: Common Intermediate Format QCIF: Quarter Common Intermediate Format WQVGA: Wide Quarter Video Graphics Array WVGA: Wide Video Graphics Array

imately arrived by the average of output bits per pixel.

$$[a_{EC}, b_{EC}] = [-\Delta R_{b_{pp}}^2, \Delta R_{b_{pp}}^2] = [-\beta^2 \cdot R_{b_{pp}} \cdot 3^2, \beta^2 \cdot R_{b_{pp}} \cdot 3^2], \quad (22)$$

$$R_{b_{pp}} = \frac{1}{n} \cdot \frac{1}{W \cdot H} \cdot \sum_{i=1}^n R_{t-i}, \quad (23)$$

where n is the window size that is empirically set to 15.

4 Scene Change Detection

When an unpredictable scene change happens, video content changes among consecutive frames. In this condition, in-

formation, such as QP, from previous frames cannot be directly employed to encode the current frame. Therefore, scene - change detection is significant to recognize changes and prevent unexpected inter prediction. There is much research work dealing with the scene change detection based on the video frame features, such as luminance component [25], YUV's mean [26], image complexity MAD [27] and color histogram information [28]. The statistical properties of video sequences indicate that the histogram information between the two adjacent frames has a greater difference when the scene change occurs. Therefore, this paper proposes scene a unified scene change detection algorithm based on the histogram statistics information of the video frame, which integrates the Pearson correla-

tion coefficient and cosine similarity to calculate the histogram correlation between the two adjacent frames.

4.1 Pearson Correlation Coefficient

The Carl Pearson correlation coefficient $P_{t-1,t}^2$ is used to represent the histogram relationship between two consecutive frames.

$$P_{t-1,t} = \frac{\sum_{i=1}^n (H_{t-1}[i] - \bar{H}_{t-1})(H_t[i] - \bar{H}_t)}{\sqrt{\sum_{i=1}^n (H_{t-1}[i] - \bar{H}_{t-1})^2} \sqrt{\sum_{i=1}^n (H_t[i] - \bar{H}_t)^2}}, \quad (24)$$

with

$$\bar{H}_t = \sum_{i=1}^n H_t[i]/n, \quad (25)$$

where H is histogram, $H_t[i]$ represents the total number of pixels at gray level i in coding the current frame, n denotes the total number of elements in H , which is usually set to 256, and $P_{t-1,t}$ ranges in $[-1, 1]$.

The correlation between two consecutive frame histograms can be measured by $P_{t-1,t}^2$. Generally, if the $P_{t-1,t}^2$ is greater than the predefined threshold 0.8, the correlation becomes obvious. Otherwise, the correlation is weak, which indicates the scene change happens.

4.2 Cosine Similarity of Histograms

Another algorithm for scene change detection between successive frames is the cosine similarity. It is expressed by (26).

$$\cos(\theta_{t-1,t}) = \frac{H_{t-1} \cdot H_t}{\|H_{t-1}\| \cdot \|H_t\|} = \frac{\sum_{i=1}^n H_{t-1}[i] \cdot H_t[i]}{\sqrt{\sum_{i=1}^n (H_{t-1}[i])^2} \sqrt{\sum_{i=1}^n (H_t[i])^2}}, \quad (26)$$

where the cosine similarity is bounded in $[0, 1]$. If the $\cos(\theta_{t-1,t})$ is close to 1, it denotes the correlation is strong. Otherwise, the correlation is weak, which shows the scene change occurs.

4.3 Unified Scene Change Detection and GOP Adjustment

To improve the accuracy of scene change detection, we propose a novel scene change detection method that combines the Pearson correlation coefficient and the cosine similarity to detect the histogram correlation between the two adjacent frames. We define Sim_t to express the similarity between two successive frames, which is calculated by (27).

$$Sim_t = P_{t-1,t} \cdot \cos(\theta_{t-1,t}). \quad (27)$$

We predefine a threshold ξ , and it means a scene change occurs if the Sim_t is lower than the ξ . Otherwise, there is no scene change. ξ is an experience value showing the sensitivity

of the scene detection, which is empirically set to 0.85.

The adaptive GOP structure is formed according to the scene structure. When a scene change occurs, the current frame type is set to I-frame. Besides, we should adjust the GOP size and allocate a new GOP.

5 Fuzzy Rate Control Algorithm

This section exhibits a new rate control algorithm (**Algorithm 1**) that integrates the proposed fuzzy controller and the unified scene change detection.

Algorithm 1 Fuzzy Rate Control Algorithm

Require:

- Initial QP value, Q_{ini} ;
- Target bit-rate(kbps), TBR ;
- Resolution of the source, $W \times H$;
- Frame rate, F_{rate} ;
- Total number of frame to be encoded, N ;
- Fuzzy rate control fast lookup Table 4, $T[13][13]$;
- Constant R-D model value, $\beta = 0.07$ for QP range $[0, 31]$ or $\beta = 0.15$ for QP range $[0, 51]$;
- Constant scene change detection empirical value, $\xi = 0.85$;
- Initial encoding time, $t = 0$;
- Initial the HRD buffer error and its deviation, $e_t = e_t^D = 0$;

Ensure:

- 1: Encode the first frame adopting Q_{ini} and $type_0 = Intra$;
- 2: $t = t + 1$;
- 3: **While** $t < N$ **do**
- 4: Collect the previous output bits R_{t-1} ; Update the target bit per pixel $Tbpp_t$ by (1); Update the HRD buffer error B_t by (2);
- 5: Calculate the HRD buffer error e_t and its deviation e_t^D by (3) and (4) respectively;
- 6: Count the average frame bits per pixel R_{bpp} by (23); Calculate the fuzzy controller input domain $[a_E, b_E]$ and $[a_{EC}, b_{EC}]$ by (21) and (22) respectively;
- 7: Calculate the fuzzy control input value e and ec by (6) and (7); Bring e and ec into table T , query the incremental control value u^* ;
- 8: Calculate the current frame QP value, $Q_t = Q_{t-1} + u^*$;
- 9: Count the previous and current frame histograms, H_{t-1} and H_t separately;
- 10: Calculate Carl Pearson coefficient of histograms $P_{t-1,t}$ by (24);
- 11: Calculate cosine similarity of histograms $\cos(\theta_{t-1,t})$ by (26);
- 12: Calculate similarity between two adjacent frames, Sim_t by (27);
- 13: Gather the current frame type, $type_t \in \{Intra, Inter\}$;
- 14: **if** $type_t = Intra$ and $Sim_t < \xi$ **then**
- 15: $type_t = Intra$;

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

```

16: else
17:    $type_i = Inter$ ;
18: end if
19: Encode one frame with  $Q_i$  and  $type_i$ ;
20:  $t = t + 1$ ;
21: end while

```

6 Simulation Results

To assess the performance of the proposed rate control scheme, the experiments were conducted on MPEG-2, H.263, MPEG-4 and H.264/AVC, and Hx265 separately. There are two prediction structures over the coding tests. One is All-Intra frames coding (I..I) and the other is low-delay frames coding; only the first frame is I-frame and the remaining frames are all P-frames (IPP..P). Comprehensive experiences have been carried out to evaluate the performance of the proposed FRCA and the default configuration encoding. Typically, the test QP points are {8,11,14,17,20} for the QP range of [0,31] and {17,22,27,32,37} for the QP range of [0,51].

Table 6 gives seven groups of video sequences generated for the coding tests. All of the encoding tests in this paper are based on these video sequences.

Four classical encoders are used to show the control accuracy of the FRCA in **Table 7**. We can find that the bit rate error (the difference between the target bit rate and actual bit rate) of the FRCA is relatively small. For AI case, the average bit rate error is less than 0.066%; for LD case, the control accuracy can achieve 0.03%.

Fig. 2 shows the simulation results of the HRD buffer size during the coding process with All-Intra and low-delay coding structure. The X-axis and Y-axis denote the picture index and the actual buffer size respectively. It is obvious that, compared with the default configuration encoding, our algorithm significantly acquires a more stable buffer size during the entire coding process. According to the upper and lower limits of the Y-

▼ **Table 6.** Video sequences used for the coding tests

	Sequence	Frequency (Hz)	Frames
QCIF	<i>Akiyo</i> (160)+ <i>Bridge</i> (180)+ <i>Claire</i> (200)+ <i>Container</i> (220)+ <i>Deadline</i> (240)	30	1000
CIF	<i>Coastguard</i> (160)+ <i>Soccer</i> (180)+ <i>City</i> (200)+ <i>Foreman</i> (220)+ <i>Silent</i> (240)	30	1000
4CIF	<i>Soccer</i> (220)+ <i>Ice</i> (240)+ <i>City</i> (260)+ <i>Harbor</i> (280)	30	1000
WQVGA	<i>BasketballPass</i> (360)+ <i>BlowingBubbles</i> (440)+ <i>BQSquare</i> (520)+ <i>RaceHorses</i> (280)	50	1600
WVGA	<i>BasketballDrill</i> (360)+ <i>BQMall</i> (520)+ <i>PartyScene</i> (440)+ <i>RaceHorses</i> (280)	50	1600
720p	<i>InToTree</i> (240)+ <i>Mobcal</i> (280)+ <i>OldTown</i> (320)+ <i>Shields_ter</i> (360)+ <i>Stockholm</i> (400)	25	1600
1080p	<i>Kimono</i> (240)+ <i>ParkScene</i> (240)+ <i>Beach</i> (250)+ <i>Cactus</i> (390)+ <i>BQTerrace</i> (480)	25	1600

CIF: Common Intermediate Format WQVGA: Wide Quarter Video Graphics Array
 QCIF: Quarter Common Intermediate Format WVGA: Wide Video Graphics Array

▼ **Table 7.** Rate control accuracy for AI and LD

	AI				LD			
	MPEG-2	MPEG-4	H.263+	H.264/AVC	MPEG-2	MPEG-4	H.263+	H.264/AVC
QCIF	0.004%	0.003%	0.004%	0.007%	0.107%	0.030%	0.107%	0.035%
CIF	0.004%	0.006%	0.005%	0.008%	0.011%	0.025%	0.032%	0.012%
4CIF	0.428%	0.005%	0.030%	0.006%	0.006%	0.002%	0.007%	0.004%
WQVGA	0.003%	0.002%	0.001%	0.007%	0.002%	0.006%	0.004%	0.009%
WVGA	0.004%	0.003%	0.003%	0.004%	0.007%	0.007%	0.017%	0.009%
720p	0.001%	0.004%	0.002%	0.005%	0.011%	0.005%	0.016%	0.010%
1080p	0.020%	0.009%	0.019%	0.028%	0.065%	0.100%	0.078%	0.111%
Average	0.066%	0.004%	0.009%	0.009%	0.030%	0.025%	0.037%	0.027%

AI: All-Intra QCIF: Quarter Common Intermediate Format
 CIF: Common Intermediate Format WQVGA: Wide Quarter Video Graphics Array
 LD: Low delay WVGA: Wide Video Graphics Array

axis given in **Fig. 2**, FRCA can control the buffer curves in a very narrow interval. Although there are large fluctuations caused by scene change, the buffer curves can more quickly response and return to steady, which illustrates that the FRCA can effectively work in video coding.

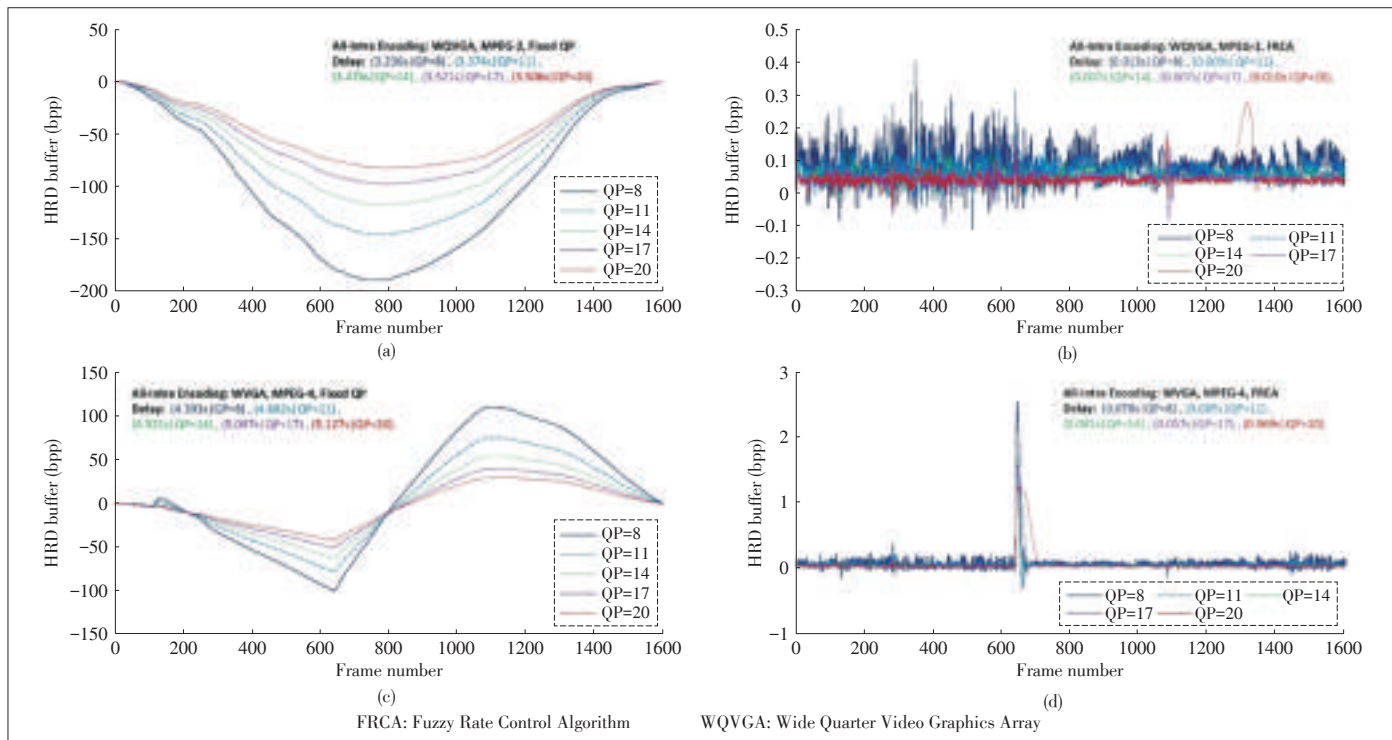
In **Figs. 2b** and **2d**, it can be seen that buffer curves are on the brink of the target buffer level (constant set to 0), and fluctuate frequently. However, our rate controller quickly and effectively controls the buffer and makes the buffer close to target buffer levels, which indicates the FRCA has a strong control ability.

Fig. 3 shows the actual output bit curves for each frame coding. The X-axis is the picture index and Y-axis denotes the actual output bit per pixel (bpp) during frame level encoding. It is evident that the curves of FRCA are consistent and very regularly fluctuates around the target bpp. The four classic encoders under default configuration value coding exhibit that frame level bpp are not controlled and present obvious volatility characteristics. In **Fig. 3**, FRCA variances σ are smaller than those of classic encoders, which indicates that FRCA can totally fulfil the purpose of rate control. The FRCA bpp variance σ in all of the sub-figures is significantly less than the σ value of default configuration encoding. Meanwhile, the σ value of FRCA with scene change ON is smaller than the σ value in the OFF condition. This means that the scene change detection with adaptive GOP length can obviously enhance FRCA performance.

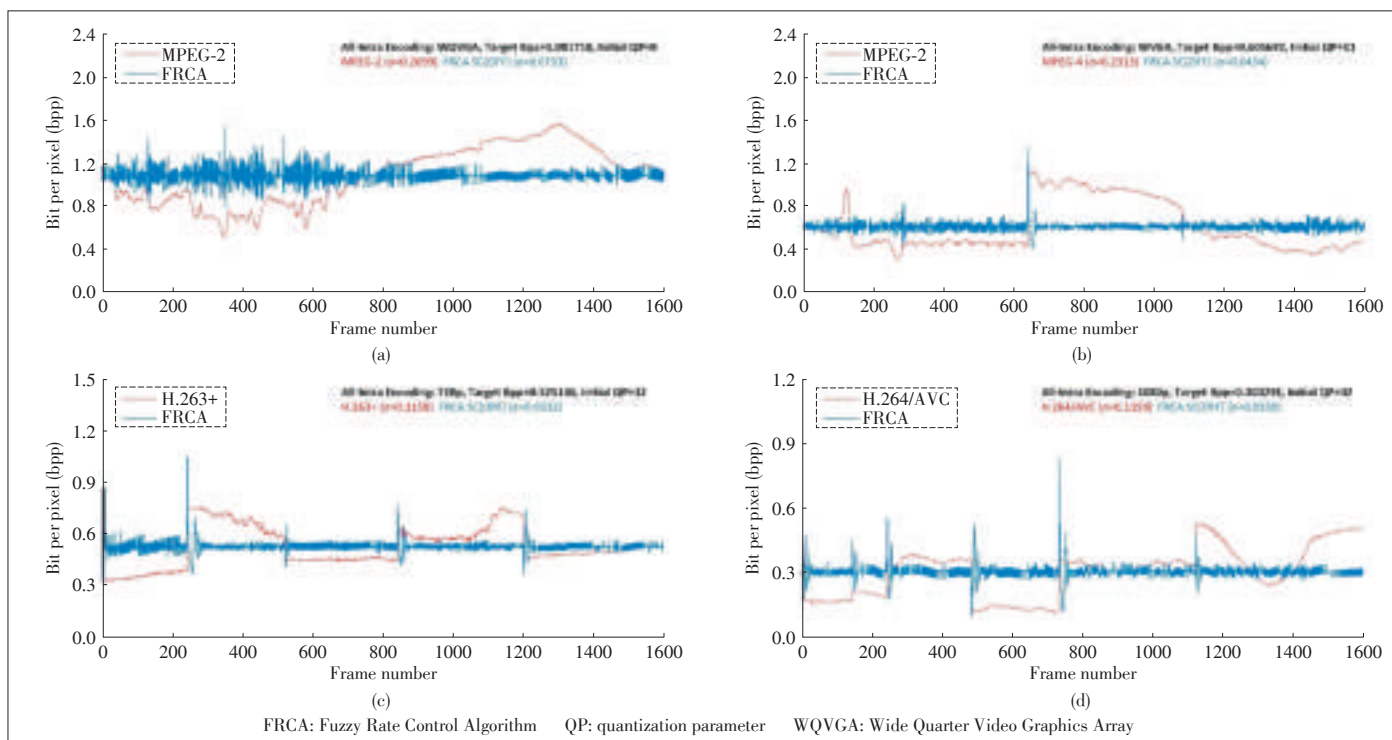
To further demonstrate the FRCA algorithm performance in the latest encoders, comprehensive experiences have been conducted on Hx265. The simulations are performed over the sequences of Class A–Class F suggested by the common test conditions (CTC) for Hx265 [29], which include 20 sequences with resolutions ranging from 4K to WQVGA. **Table 8** presents the control accuracy results for comparison with the default RC algorithm. From **Table 8** we can find that the bit rate error of the FRCA is much smaller than the default RC algorithm. It shows that the proposed algorithm works better than

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen



▲ Figure 2. HRD buffer size comparison of default configuration encoding and FRCA.



▲ Figure 3. Actual bit per pixel performance comparison of default configuration encoding and FRCA.

the default RC algorithm for both AI structure and LB structure on Hx265. To clearly represent the FRCA control accuracy on Hx265, Fig. 4 shows the simulation results of the HRD

buffer size during the coding process with All-Intra and low-delay coding structure. The HRD buffer state diagrams of the default RC algorithm are provided in (a), (b), and (c) in the figure

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

▼ **Table 8. Rate control accuracy comparison**

Sequence	Default RC		FRCA	
	AI	LD	AI	LD
HEVC-A	20.17%	7.94%	0.09%	0.08%
HEVC-B	24.12%	4.54%	0.04%	0.07%
HEVC-C	18.32%	5.51%	0.03%	0.03%
HEVC-D	7.83%	1.49%	0.01%	0.02%
HEVC-E	9.23%	5.61%	0.02%	0.07%
Overall	19.33%	5.74%	0.04%	0.06%

AI: All-Intra
FRCA: Fuzzy Rate Control Algorithm
HEVC: High Efficiency Video Coding

LD: low delay
RC: rate control

and the HRD buffer state diagrams of the FRCA are presented in (d), (e), and (f). It is obvious that, compared with the default configuration encoding, our algorithm can strictly control the HRD buffer in a very narrow interval and acquire a more stable buffer size during the entire coding process.

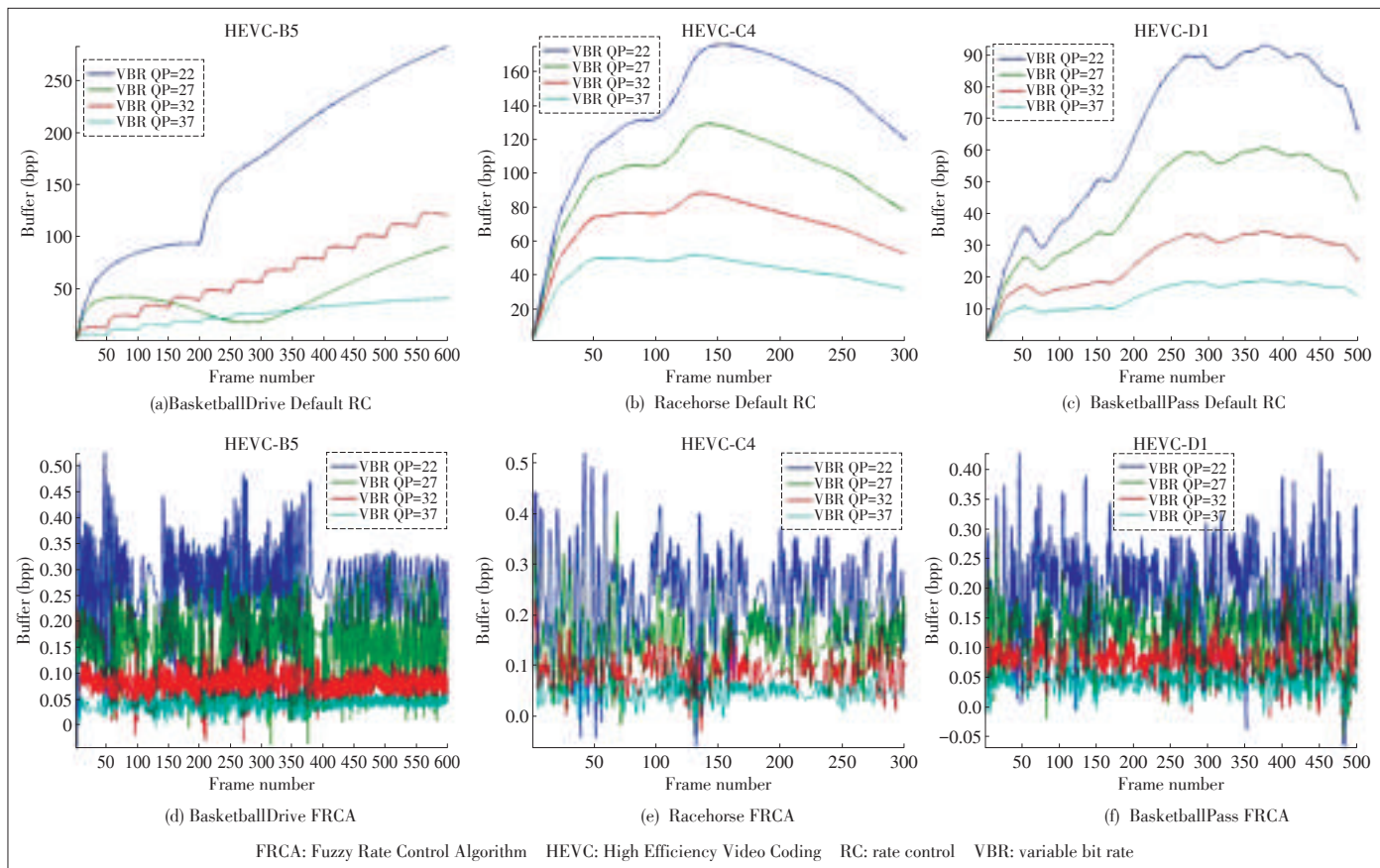
7 Conclusions

This paper develops a novel RC algorithm base on a fuzzy controller regardless of R-D model dependence, named the

Fuzzy Rate Control Algorithm (FRCA). The significant features of the FRCA are simple, efficiency and low computation complexity. Considering the efficiency and accuracy of the control system, an input domain determination scheme is adopted in the proposed fuzzy controller. The work employs a fast lookup table generated by fuzzy logic to calculate increment QP value, which reduces the computation complexity. In addition, we propose scene change detection for adaptive GOP length adjustment. FRCA achieves accurate rate control while maintains extremely low delay between the encoder and the decoder. Extensive simulations and analytical results demonstrate that the FRCA outperforms the default configuration encoding in bitrate accuracy.

References

- [1] H. Choi, J. Nam, J. Yoo, and D. Sim, "Rate control based on unified RQ model for HEVC," in *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCT-VC H0213*, San José, CA, USA, Feb. 2012.
- [2] L. Xu, S. Kwong, Y. Zhang, and D. Zhao, "Low-complexity encoder framework for window-level rate control optimization," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 5, pp. 1850–1858, May 2013. doi: 10.1109/TIE.2012.2190960.
- [3] C. W. Seo, J. H. Moon, and J. K. Han, "Rate control for consistent objective quality in high efficiency video coding," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2442–2454, Jun. 2013. doi: 10.1109/TIP.2013.2251647.



▲ **Figure 4. HRD buffer size comparison for AI structure on Hx265.**

Variable Bit Rate Fuzzy Control for Low Delay Video Coding

ZHONG Min, ZHOU Yimin, LUO Minke, and ZUO Wen

- [4] T. Chiang and Y. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 246–250, Feb. 1997. doi: 10.1109/76.554439.
- [5] Z. Li, F. Pan, K. Lim, et al., "Adaptive basic unit layer rate control for JVT," in *JVT-G012-r1, 7th Meeting*, Pattaya, Thailand, 2003.
- [6] B. Li, H. Li, L. Li, and J. Zhang, "Rate control by R-lambda model for HEVC," in *JCTVC-K0103, 11th JCTVC Meeting*, Shanghai, China, Oct. 2012.
- [7] Z. He and S. K. Mitra, "Optimum bit allocation and accurate rate control for video coding via p-domain source modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 10, pp. 840–849, Oct. 2002. doi: 10.1109/TCSVT.2002.804883.
- [8] Y. Zhou, Y. Sun, Z. Feng, and S. Sun, "New rate-distortion modeling and efficient rate control for H.264/AVC video coding," *Signal Processing: Image Communication*, vol. 24, no. 5, pp. 345–356, May 2009. doi: 10.1016/j.image.2009.02.014.
- [9] L. Xu, D. Zhao, X. Ji, et al., "Window-level rate control for smooth picture quality and smooth buffer occupancy," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 723–734, Mar. 2011. doi: 10.1109/TIP.2010.2063708.
- [10] H. Choi, J. Yoo, J. Nam, D. Sim, and I. Bajic, "Pixel-wise unified rate-quantization model for multi-level rate control," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 1112–1123, Dec. 2013. doi: 10.1109/JSTSP.2013.2272241.
- [11] B. Yan and M. Wang, "Adaptive distortion-based intra-rate estimation for H.264/AVC rate control," *IEEE Signal Processing Letters*, vol. 16, no. 3, pp. 145–148, Mar. 2009. doi: 10.1109/LSP.2008.2010813.
- [12] J. Yang, Y. Sun, Y. Zhou, and S. Sun, "Incremental rate control for H.264 AVC scalable extension," *Multimedia Tools and Applications*, vol. 64, no. 3, pp. 581–598, Jun. 2013. doi: 10.1007/s11042-011-0967-y.
- [13] Y. Zhou, Y. Sun, Z. Feng, and S. Sun, "PID-based bit allocation strategy for H.264/AVC rate control," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, no. 3, pp. 184–188, Mar. 2011. doi: 10.1109/TCSII.2011.2106350.
- [14] H. Ying, *Fuzzy Control and Modeling: Analytical Foundations and Applications*. Hoboken, USA: Wiley IEEE Press, 2000. doi:10.1109/9780470544730
- [15] Y. Zhang, "The application of the fuzzy control algorithm in the automatic track following robot," in *IEEE Second International Conference on Mechanic Automation and Control Engineering (MACE 2011)*, Hohhot, China, 2011, pp. 759–762. doi:10.1109/MACE.2011.5987038.
- [16] R. Lasri, I. Rojas, and H. Pomares, "Explaining how intelligent control has improved the way we live: a survey on the use of Fuzzy Logic Controllers in daily human life," in *IEEE International Conference on Multimedia Computing and Systems (ICMCS)*, Ouarzazate, Morocco, 2011, pp. 1–6. doi: 10.1109/ICMCS.2011.5945577.
- [17] T. Muthuramalingam, B. Mohan, and A. Rajadurai, "Monitoring and fuzzy control approach for efficient electrical discharge machining process," *Materials and Manufacturing Processes*, vol. 29, no. 3, pp. 281–286, Nov. 2013. doi: 10.1080/10426914.2013.864412.
- [18] S. Barro and R. Marín, *Fuzzy Logic in Medicine*, vol. 83. Heidelberg, Germany: Physica-Verlag Heidelberg, 2002. doi: 10.1007/978-3-7908-1804-8.
- [19] S. Birle, M. Hussein, and T. Becker, "Fuzzy logic control and soft sensing applications in food and beverage processes," *Food Control*, vol. 29, no. 1, pp. 254–269, Jan. 2013. doi: 10.1016/j.foodcont.2012.06.011.
- [20] H. Kazemian and L. Meng, "An adaptive control for video transmission over bluetooth," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 2, pp. 263–274, Apr. 2006. doi: 10.1109/TFUZZ.2005.864080.
- [21] D. Tsang, B. Bensou, and S. Lam, "Fuzzy-based rate control for real-time MPEG video," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 4, pp. 504–516, Nov. 1998. doi: 10.1109/91.728442.
- [22] S. Sheu, Y. Wang, H. Yin, and J. Chen, "Adaptive rate controller for mobile ad hoc networks," *International Journal of Mobile Communications*, vol. 1, no. 3, pp. 312–328, Sept. 2003. doi: 10.1504/IJMC.2003.003497.
- [23] M. Rezaei, M. Hannuksela, and M. Gabbouj, "Semi-fuzzy rate controller for variable bit rate video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, pp. 633–645, May 2008. doi: 10.1109/TCSVT.2008.919108.
- [24] J. Lin and R. Lian, "Intelligent control of active suspension systems," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 2, pp. 618–628, Feb. 2011. doi: 10.1109/TIE.2010.2046581.
- [25] J. Shi, L. Xu, and C. Zhang, "Scene adaptive frame-layer rate control algorithm for H.264," *Application Research of Computers*, vol. 27, no. 5, pp. 1968–1970, May 2010. doi: 10.3969/j.issn.1001-3695.2010.05.105.
- [26] Y. Wang and Y. Xue, "Rate control algorithm based on scene change," *Video Engineering*, vol. 33, no. 12, pp. 17–20, Dec. 2009.
- [27] H. Duan and T. Wang, "Improved frame-layer rate control algorithm for H.264," *Journal of Computer Applications*, vol. 29, no. 4, pp. 1008–1010, Apr. 2009. doi: 10.3724/SP.J.1087.2009.01008.
- [28] S. Kang, S. Cho, and S. Yoo, "Scene change detection using multiple histograms for motion-compensated frame rate up-conversion," *Journal of Display Technology*, vol. 8, no. 3, pp. 121–126, Mar. 2012. doi: 10.1109/JDT.2011.2167740.
- [29] F. Bossen, "Common HM test conditions and software reference configurations," in *12th Meeting of Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG*, Geneva, Switzerland, 2013, Document JCTVC-K1100.

Manuscript received: 2015-12-28

Biographies

ZHONG Min (754172961@qq.com) received the M.S. degree in computer science from the College of Computer Science, University of Electronic Science and Technology of China in 2016. She is a test development engineer with Baidu Online Network Technology (Beijing) Co., Ltd. She majored in image and video coding during the graduate study and keeps the research interest in video coding technology.

ZHOU Yimin (yiminzhou@uestc.edu.cn) received the B.S., M.S. and Ph.D. degrees in computer science from the College of Computer Science, University of Electronic Science and Technology of China (UESTC) in 2003, 2006 and 2009 respectively. He joined the College of Computer Science, UESTC in 2009 and became an associate professor in 2012. His research interests include image and video coding, streaming and processing, and visual perception and applications. He has authored or co-authored over 30 papers in journals and conferences. He has three granted patents and over 10 patent applications. He pays special attention to the video encoding standards like HEVC, IVC and AVS. His two proposals were adopted to the MPEG and over 20 proposals adopted to the AVS.

LUO Minke (544751189@qq.com) received his B.S. degree from Southwest University of Science and Technology in 2013 and M.S. degree from University of Electronic Science and Technology of China in 2016, both in computer science. During the period of postgraduate, he followed professor ZHOU Yimin to study video coding and focused on bit rate control related research. His research interests include network video transmission, quality control, etc. He has authored or co-authored two journal papers and three patent applications. His five proposals have been adopted by the AVS group.

ZUO Wen (wenz0503@qq.com) received his master's degree from Nanjing University, China in 2006. He worked with ZTE Corporation as a video system engineer. His current research interests include video encoding and application. He has authored or co-authored over 30 invention patents in his research area.

ZTE Communications Special Issue on

Wireless Data and Energy Integrated Communication Networks

In the vision of the incoming 5G era, billions of people as well as trillions of machines are expected to be connected by the next generation mobile network, as predicted by the standardisation body 5G - PPP (<http://5g-ppp.eu/>). Functions of massive communication devices have been substantially limited by insufficient power supply. As an efficient solution, dedicated radio-frequency (RF) signals are capable of carrying well-controlled energy towards the rechargeable devices in order to achieve the on-demand energy transfer. However, enabling the wireless charging capability of RF signals may significantly influence the data transfer of the communication network. Although the RF signals are capable of simultaneously carrying both the data and energy, the diverse requirements of data and energy transfers pose huge challenges in their effective integration. For example, the energy receiver and the data receiver have diverse sensitivity to the received power. The received power as low as -80 dBm is sufficient for recovering the contaminated packet, thanks to the state-of-the-art channel encoding/decoding techniques. However, only when the received power is higher than -20 dBm, the energy reception circuit can be effectively activated for converting a fraction of the energy carried by the RF signals to the direct current (DC).

As a result, the integration of the wireless data and energy transfer is worth deep exploration. For the practical implementation of RF-based energy transfers, we have to make the energy receiver adapt to a wider range of the received power, while increasing the RF-DC conversion efficiency. The advanced transceiver for the integrated data and energy transfer/reception is also required in the physical layer. The coexistence of the multiple energy and data transmitters/receivers calls for deep exploration on the interference management schemes, the medium access control (MAC) algorithms as well as the data/energy routing protocols, which systematically yield data and energy integrated communication net-

works (DEINs).

This special issue will also serve as a stimulus to educate about, promote and accelerate technical evolution towards the promising and exciting research area of DEINs. Specifically, the special issue will present tutorials, surveys and original research articles that cover the following subjects (but are not limited to):

- Design of highly-efficient RF-DC converter and the novel energy storage units.
- Information theoretic aspects of the integrated data and energy transfer.
- Highly - efficient transceiver design of integrated data and energy transfers, including the joint channel coding and modulation schemes, the waveform design and the antenna design.
- Optimal resource allocation and interference management among energy users, data users and integrated users while ensuring their quality of experiences (QoEs).
- Modelling and optimisation of the medium access control protocols in DEINs.
- Protocols and algorithms for joint energy and data routing in DEINs.
- Economic aspects and billing issues in operating DEINs.
- Practical applications, management and adaptation of DEINs.
- Prototype and practical deployment solutions as well as standardisation of DEINs.

Submission Guidance

Please directly submit your paper to zhu.li1@zte.com.cn, using the email subject "ZTE - DEIN - Paper - Submission". The draft of manuscript can be written in either PDF format or MS Word format. However, the final manuscript after the acceptance MUST BE written in MS Word format.

Guest Editors

Important Dates

First Submission Due:	September 1, 2017
Final Decision Due:	November 1, 2017
Final Manuscript Due:	December 1, 2017
Publication Date:	February 25, 2018

YANG Kun	University of Electronic Science and Technology of China, China Email: kunyang@uestc.edu.cn
HU Jie	University of Electronic Science and Technology of China, China Email: hujie@uestc.edu.cn
WONG Kai-Kit	University College London, UK Email: kai-kit.wong@ucl.ac.uk

ZTE Communications Guidelines for Authors

• Remit of Journal

ZTE Communications publishes original theoretical papers, research findings, and surveys on a broad range of communications topics, including communications and information system design, optical fiber and electro-optical engineering, microwave technology, radio wave propagation, antenna engineering, electromagnetics, signal and image processing, and power engineering. The journal is designed to be an integrated forum for university academics and industry researchers from around the world.

• Manuscript Preparation

Manuscripts must be typed in English and submitted electronically in MS Word (or compatible) format. The word length is approximately 3000 to 8000, and no more than 8 figures or tables should be included. Authors are requested to submit mathematical material and graphics in an editable format.

• Abstract and Keywords

Each manuscript must include an abstract of approximately 150 words written as a single paragraph. The abstract should not include mathematics or references and should not be repeated verbatim in the introduction. The abstract should be a self-contained overview of the aims, methods, experimental results, and significance of research outlined in the paper. Five carefully chosen keywords must be provided with the abstract.

• References

Manuscripts must be referenced at a level that conforms to international academic standards. All references must be numbered sequentially in-text and listed in corresponding order at the end of the paper. References that are not cited in-text should not be included in the reference list. References must be complete and formatted according to *ZTE Communications* Editorial Style. A minimum of 10 references should be provided. Footnotes should be avoided or kept to a minimum.

• Copyright and Declaration

Authors are responsible for obtaining permission to reproduce any material for which they do not hold copyright. Permission to reproduce any part of this publication for commercial use must be obtained in advance from the editorial office of *ZTE Communications*. Authors agree that a) the manuscript is a product of research conducted by themselves and the stated co-authors, b) the manuscript has not been published elsewhere in its submitted form, c) the manuscript is not currently being considered for publication elsewhere. If the paper is an adaptation of a speech or presentation, acknowledgement of this is required within the paper. The number of co-authors should not exceed five.

• Content and Structure

ZTE Communications seeks to publish original content that may build on existing literature in any field of communications. Authors should not dedicate a disproportionate amount of a paper to fundamental background, historical overviews, or chronologies that may be sufficiently dealt with by references. Authors are also requested to avoid the overuse of bullet points when structuring papers. The conclusion should include a commentary on the significance/future implications of the research as well as an overview of the material presented.

• Peer Review and Editing

All manuscripts will be subject to a two-stage anonymous peer review as well as copyediting, and formatting. Authors may be asked to revise parts of a manuscript prior to publication.

• Biographical Information

All authors are requested to provide a brief biography (approx. 100 words) that includes email address, educational background, career experience, research interests, awards, and publications.

• Acknowledgements and Funding

A manuscript based on funded research must clearly state the program name, funding body, and grant number. Individuals who contributed to the manuscript should be acknowledged in a brief statement.

• Address for Submission

magazine@zte.com.cn

12F Kaixuan Building, 329 Jinzhai Rd, Hefei 230061, P. R. China

ZTE COMMUNICATIONS

ZTE Communications has been indexed in the following databases:

- Abstract Journal
 - Cambridge Scientific Abstracts (CSA)
 - China Science and Technology Journal Database
 - Chinese Journal Fulltext Databases
 - Inspec
 - Ulrich's Periodicals Directory
 - Wanfang Data—Digital Periodicals
-

ZTE COMMUNICATIONS

Vol. 15 No. 2 (Issue 56)

Quarterly

First English Issue Published in 2003

Supervised by:

Anhui Science and Technology Department

Sponsored by:

Anhui Science and Technology Information Research Institute and ZTE Corporation

Staff Members:

Editor-in-Chief: CHEN Jie

Executive Associate Editor-in-Chief: HUANG Xinming

Editor-in-Charge: ZHU Li

Editors: XU Ye, LU Dan, ZHAO Lu

Producer: YU Gang

Circulation Executive: WANG Pingping

Assistant: WANG Kun

Editorial Correspondence:

Add: 12F Kaixuan Building, 329 Jinzhai Road,
Hefei 230061, P. R. China

Tel: +86-551-65533356

Fax: +86-551-65850139

Email: magazine@zte.com.cn

Published and Circulated (Home and Abroad) by:

Editorial Office of *ZTE Communications*

Printed by:

Hefei Tiancai Color Printing Company

Publication Date: April 25, 2017

Publication Licenses:

ISSN 1673-5188

CN 34-1294/ TN

Annual Subscription: RMB 80