

An International ICT R&D Journal Sponsored by ZTE Corporation

ISSN 1673-5188
CN 34-1294/ TN
CODEN ZCTOAK

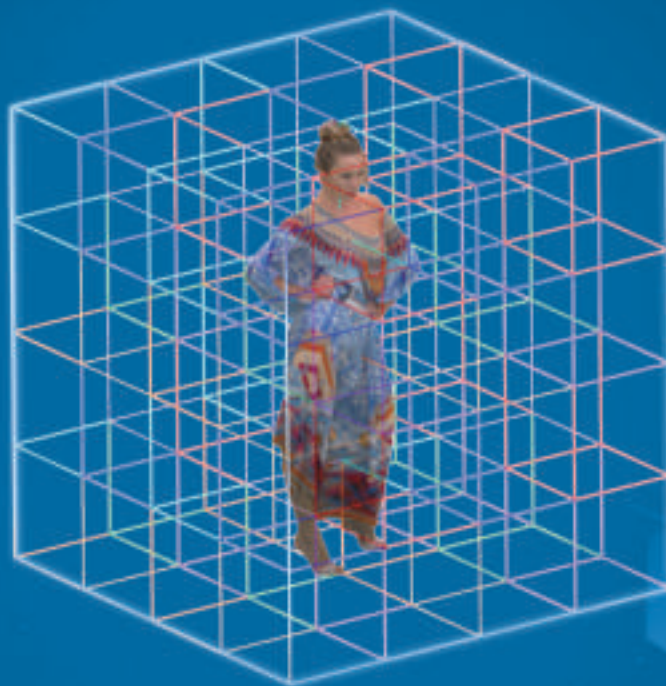
ZTE COMMUNICATIONS

中兴通讯技术(英文版)

<http://tech.zte.com.cn>

September 2018, Vol. 16 No. 3

SPECIAL TOPIC: Next Generation Mobile Video Networking



Octree decomposition



The 8th Editorial Board of ZTE Communications

Chairman

GAO Wen: Peking University (China)

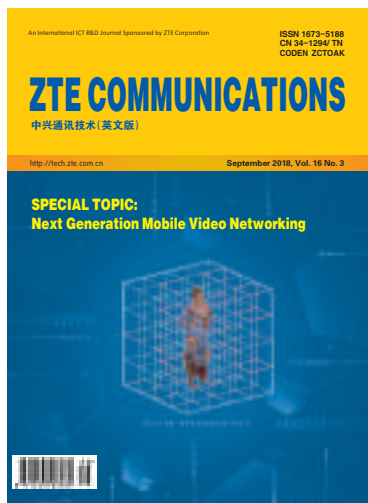
Vice Chairmen

XU Ziyang: ZTE Corporation (China) | **XU Cheng-Zhong:** Wayne State University (USA)

Members (in Alphabetical Order):

CAO Jiannong	Hong Kong Polytechnic University (China)
CHEN Chang Wen	University at Buffalo, The State University of New York (USA)
CHEN Yan	Northwestern University (USA)
CHI Nan	Fudan University (China)
CUI Shuguang	The Chinese University of Hong Kong, Shenzhen (China)
GAO Wen	Peking University (China)
HWANG Jenq-Neng	University of Washington (USA)
Victor C. M. Leung	The University of British Columbia (Canada)
LI Guifang	University of Central Florida (USA)
LIU Ming	Institute of Microelectronics of the Chinese Academy of Sciences (China)
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)
TAO Meixia	Shanghai Jiao Tong University (China)
WANG Xiang	ZTE Corporation (China)
WANG Xiaodong	Columbia University (USA)
WANG Zhengdao	Iowa State University (USA)
XU Cheng-Zhong	Wayne State University (USA)
XU Ziyang	ZTE Corporation (China)
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)
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: Next Generation Mobile Video Networking

01 Editorial

HWANG Jenq-Neng and WEN Yonggang

03 Introduction to Point Cloud Compression

Point cloud has been widely applied in services of various 3D objects and scenes. In this paper, the authors summarize the static and dynamic point cloud compression, both including irregular geometry and photometry information that represent the spatial structure information and corresponding attributes, respectively.

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie

09 Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

This paper provides an overview of several typical Forward Error Correction (FEC) codes and a novel delay-aware fountain coding (DAF) technique that maximizes the code word length under the constraint of a given delay. Moreover, the authors review video streaming technologies, focusing on Dynamic Adaptive Streaming over HTTP (DASH) and DASH over Multiple Content Distribution Servers (MCDS-DASH). They also propose a novel approach to integrating fountain codes with MCDS-DASH.

WU Kesong, CAO Xianbin, CHEN Zhifeng, and WU Dapeng

15 DASH and DASH-VR Video Multicast Systems

In this paper, the authors investigate the state-of-the-art video multicast technologies. LTE supports multicast service through evolved Multimedia Broadcast Multicast Service (eMBMS) systems, and there are different algorithms to perform the video multicast along with adaptive video quality control. The authors also propose a novel approach to improve the quality of experience for DASH-VR video multicast systems.

PARK Jounsup and HWANG Jenq-Neng

23 How to Manage Multimedia Traffic: Based on QoE or QoT?

This paper defines a new concept of Acceptable Quality of Things (AQoT) which involves IoT devices and their applications. AQoT aims at minimizing bandwidth without compromising quality in IoT devices. Experimental results based on human detection and license number plate detection use cases have demonstrated that the AQoT concept can significantly reduce bandwidth usage.

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

▶ CONTENTS

ZTE COMMUNICATIONS

Vol. 16 No. 3 (Issue 63)

Quarterly

First English Issue Published in 2003

Supervised by:

Anhui Science and Technology Department

Sponsored by:

Anhui Science and Technology Information
Research Institute;
Magazine House of ZTE Communications

Published and Circulated

(Home and Abroad) by:

Magazine House of ZTE Communications

Staff Members:

Editor-in-Chief: WANG Xiang

Executive Associate

Editor-in-Chief: HUANG Xinming

Editor-in-Charge: ZHU Li

Editors: XU Ye and LU Dan

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

Printed by:

Hefei Tiancai Color Printing Company

Publication Date:

September 25, 2018

Publication Licenses:

ISSN 1673-5188

CN 34-1294/TN

Annual Subscription:

RMB 80

Statement: This magazine is a free publication for you. If you do not want to receive it in the future, you can send the "TD unsubscribe" mail to magazine@zte.com.cn. We will not send you this magazine again after receiving your email. Thank you for your support.

30 When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

The authors present a tutorial survey on the way of using machine learning techniques to address the emerging challenges in the infrastructure and platform layer of media cloud. They review machine learning techniques and the system architecture of media cloud. They also present an outlook on the open issues.

JIN Yichao and WEN Yonggang

Research Paper

40 Mechanism of Fast Data Retransmission in CU-DU Split Architecture of 5G NR

The 5G radio access network (RAN) architecture is supposed to be split into the central unit (CU) and the distributed unit (DU). In this paper, the authors study the data fast retransmission issue introduced by this functional split in different scenarios and solutions are provided to handle this issue.

HUANG He, LIU Yang, LIU Zhuang, HAN Jiren, and GAO Yin

45 DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

This paper presents a novel DEX protection scheme, DexDefender, to withstand memory dump attack on the Android platform. Experimental results show that the proposed scheme can protect the DEX files from both reverse engineering and memory dump attacks with an acceptable performance.

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

52 A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

The authors in this paper propose a re-transmission mechanism by analyzing the security risks in Quantum key distribution (QKD) based optical networks. Numerical results indicate that the proposed re-transmission mechanism can provide strong protection degree with enhanced attack protection.

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu

Review

59 Persistent Data Layout in File Systems

In this paper, the authors introduce a recent usage of persistent layout in a file system that combines both flash memory and byte-addressable non-volatile memory. They conclude that persistent data layout in file systems may evolve dramatically in the era of emerging non-volatile memory.

LUO Shengmei, LU Youyou, YANG Hongzhang, SHU Jiwu, and ZHANG Jiacheng

Editorial

Next Generation Mobile Video Networking

▶ Guest Editors



HWANG Jenq-Neng received his Ph.D. degree from the University of Southern California, USA. In the summer of 1989, Dr. HWANG joined the Department of Electrical Engineering of the University of Washington in Seattle, USA, where he has been promoted to Full Professor since 1999. He served as the Associate Chair for Research from 2003 to 2005, and from 2011–2015. He is currently the Associate Chair for Global Affairs and International Development in the EE Department. He has written more than 330 journal papers, conference papers and book chapters in the areas of machine learning, multimedia signal processing, and multimedia system integration and networking, including an authored textbook on “Multimedia Networking: from Theory to Practice,” published by Cambridge University Press. Dr. HWANG has close working relationship with the industry on multimedia signal processing and multimedia networking.

Dr. HWANG received the 1995 IEEE Signal Processing Society’s Best Journal Paper Award. He is a founding member of the Multimedia Signal Processing Technical Committee of IEEE Signal Processing Society and was the Society’s representative to IEEE Neural Network Council from 1996 to 2000. He is currently a member of Multimedia Technical Committee (MMTC) of IEEE Communication Society and also a member of Multimedia Signal Processing Technical Committee (MMSP TC) of IEEE Signal Processing Society. He served as associate editors for *IEEE T-SP*, *T-NN* and *T-CSVT*, *T-IP* and *Signal Processing Magazine (SPM)*. He is currently on the editorial board of *ZTE Communications*, *ETRI*, *IJDMB* and *JSPS* journals. He served as the Program Co-Chair of IEEE ICME 2016 and was the Program Co-Chairs of ICASSP 1998 and ISCAS 2009. Dr. HWANG has been named a fellow of IEEE since 2001.



WEN Yonggang received the Ph.D. degree in electrical engineering and computer science (minor in western literature) from the Massachusetts Institute of Technology, USA, in 2008. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. Previously, he worked at Cisco Systems, Inc., San Jose, CA, USA, to lead product development in content delivery network. He has authored or co-authored more than 140 papers in top journals and prestigious conferences. His work in Multi-screen Cloud Social TV has been featured by global media (more than 1600 news articles from over 29 countries). His research interests include cloud computing, green data center, big data analytics, multimedia network, and mobile computing.

Prof. WEN serves on the Editorial Board of *IEEE Transactions on Circuits and Systems for Video Technology*, *IEEE Wireless Communication Magazine*, *IEEE Communications Survey & Tutorials*, *IEEE Transactions on Multimedia*, *IEEE Transactions on Signal and Information Processing over Networks*, *IEEE Access Journal*, and *Elsevier Ad Hoc Networks*. From 2014 to 2016, he was elected as the Chair for IEEE ComSoc Multimedia Communication Technical Committee. He was the recipient of the ASEAN ICT Award 2013 (Gold Medal). His work on Cloud3DView, as the only academia entry, was the recipient of the Data Centre Dynamics Awards 2015 DCD APAC. He was the recipient of the 2015 IEEE Multimedia Best Paper Award, and the Best Paper Awards at EAI/ICST Chinacom 2015, IEEE WCSP 2014, IEEE Globecom 2013, and IEEE EUC 2012.

The most recent Cisco Visual Networking Index (VNI) forecasts that more than three-fourths of the world’s mobile data traffic, which is expected to be 49 exabytes per month by 2021, will be video, i.e., a 9-fold increase between 2016 and 2021. The exponential growth in bandwidth demand of the mobile Internet is fueled by the transfer of high definition (HD)/ultra HD (UHD) video consumption to online dissemination, as well as by matured deployments of IPTV and video on demand (VOD) streaming services. It is also expected that virtual reality (VR)/augmented reality (AR) and 3D point cloud traffic will increase significantly in the near future. In the meanwhile, there are huge amount of user generated videos being uploaded to cloud servers, ranging from live streamed social media videos from smartphones, mobile surveillance videos from home/vehicles/drones, environmental monitoring videos from various Internet of Things (IoT) based cameras, etc.

To cope with this growth of video driven mobile Internet, there is an urgent need of advanced video source/channel coding techniques, effective next generation mobile networking architectures and cloud services, to disseminate and/or collect these big visual data, so as to provide best streaming services at the client side and/or perform intelligent data analytics at the cloud server side. Currently the coding technologies, such as advanced video coding (AVC, H.264) and high efficiency video coding (HEVC, H.265), mainly focus on two-dimensional video compression. A new video coding standards working group, called Future Video Coding (FVC), has also been established to fully consider the unique attributes of immersive media, such as the 360-degree panoramic VR/AR videos.

Even though 4G+/5G mobile architectures have also involved some of mobile video networking requirements, but most of them are under the framework of network slicing for diversified service, as facilitated by the incorporation of software defined networking (SDN)/network function virtualization (NFV). It is well known that wireless channel conditions vary frequently with channel environments and user behaviors. MPEG’s Dynamic Adaptive Streaming over HTTP (MPEG-DASH) is thus incorporated as an effective video streaming platform, which enables the adaptive rate selection based on the channel conditions. DASH can provide superior video experience by giving clients a chance to receive the video quality based on their channel condition and buffer status, resulting in better quality of experience (QoE). To achieve greater spectral efficiency, the use of multiple radio access technologies (Multi-RAT) or LTE-WiFi aggregation (LWA) resource allocation for video delivery is also being promoted. Moreover, most of the metrics used to measure the quality of mobile video networking are based on quality of service (QoS)/QoE, which are targeting on the human perception on streamed/distributed videos, while there are more and more uploaded videos are to be analyzed by machines without human viewing. Therefore, a new type of video quality assessment scheme which measures the quality of contents (QoC) for video

Editorial

HWANG Jenq-Neng and WEN Yonggang

analytics needs to be adopted. In this special issue, we have collected papers which address all aspects of mobile video networking mechanisms, from application, transport, network and MAC/PHY layers of future mobile networks.

More specifically, emerging immersive media services, such as 360 VR/AR videos and 3D point cloud, enable customers to feel being personally at the scene with personalized viewing perspective and enjoy real-time full interaction. A 3D point cloud video transmission demands a huge amount of data bandwidth, and causes high complexity in the scattered random distribution of the spatial distribution, which brings great challenge to the storage and transmission system. These data cannot be directly encoded with the existing H.264/H.265 and/or VR/AR compression schemes. Hence, more advanced compression coding techniques are needed to significantly reduce the amount of data, combining with the unique consumption characteristics of immersive media. The paper entitled "Introduction to Point Cloud Compression" presents a wonderful review of static and dynamic point cloud compression techniques to overcome the challenges of high bandwidth demands and dynamic users' viewing perspectives.

The paper entitled "Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey" first provides a thorough overview of several typical forward error correction (FEC) codes which can be used for combating the noisy wireless channels. A novel delay-aware fountain (DAF) coding technique is then proposed to maximize the code word length under the constraint of a given delay. Two extensions of DAF are also proposed; one is the unequal error protection DAF (UEP-DAF) for improving the video PSNR and the other is the model predictive control DAF (MPC-DAF) for reducing the computational complexity. This paper also provides an excellent review of video streaming technologies, including the dynamic adaptive streaming over HTTP (DASH) and DASH over multiple content distribution servers (MCDS - DASH) in detail. Finally, based on MCDS-DASH that adapts video bitrate at the block level to alleviate video fluctuation, a novel approach to integrating fountain codes with MCDS-DASH is proposed to achieve unprecedented high throughput.

On-demand video streaming is already the major video content platform and private broadcast is also getting more popular. In addition to fast growing of streaming videos, there are also growing demands of VR and AR data traffic, which calls huge amount of wireless resource to satisfy users' QoE. There-

fore, it is necessary to apply the wireless transmission scheme that has better spectral efficiency and the video rate adaptation to provide the best quality to the users. The paper entitled "DASH/DASH-VR Video Multicast Systems" investigates the state-of-the-art VR video multicast along with adaptive video quality control over LTE mobile systems. The QoE enhanced algorithm has the procedures of deciding the video rates, resource allocations, and user groupings.

The Internet of Things (IoT) based video networking applications, such as environmental monitoring, healthcare, surveillance, event recognition and traffic control, are amongst the most commonly deployed applications over the Internet. Since the delivery of video can be destined to a machine or human, it is important to distinguish video quality between the two, i.e., QoE for video services involves human visual system. However, what will involve a machine or process? To distinguish between the two, the paper "How to Manage Multimedia Traffic: Based on QoE or QoT?" defines a new concept of acceptable quality of things (AQoT) which involves IoT devices and their applications. The proposed AQoT metric aims at minimizing bandwidth without compromising quality in IoT devices.

Network service operators are expected to deliver significantly more network traffic with the growing video services. Media cloud inheriting the advances from cloud computing, has emerged as a promising computing paradigm to provide novel multimedia services with satisfied QoS and reduced cost. Machine learning, which has been intensively applied in various multimedia applications, can provide a natural solution to address several challenges in media cloud. In particular, machine learning represents the set of algorithms that can progressively improve the performance on a specific task without being explicitly programmed. The article entitled "When Machine Learning Meets Media Cloud: Architecture, Application and Outlook" presents a wonderful survey of how machine learning addresses the challenges in media cloud, from the infrastructure and platform perspectives.

As we conclude the introduction to this special issue and the contents of six papers, we would like to thank all authors for their valuable contributions. We also express our deep gratitude to all the reviewers for their timely and insightful comments on all submitted papers. It is our sincere expectation that the contents in this special issue are informative and useful from various aspects related to next generation mobile video networking.

Introduction to Point Cloud Compression

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie

(Cooperative MediaNet Innovation Center, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract

Characterized by geometry and photometry attributes, point cloud has been widely applied in the immersive services of various 3D objects and scenes. The development of even more precise capture devices and the increasing requirements for vivid rendering inevitably induce huge point capacity, thus making the point cloud compression a demanding issue. In this paper, we introduce several well-known compression algorithms in the research area as well as the boosting industry standardization works. Specifically, based on various applications of this 3D data, we summarize the static and dynamic point cloud compression, both including irregular geometry and photometry information that represent the spatial structure information and corresponding attributes, respectively. In the end, we conclude the point cloud compression as a promising topic and discuss trends for future works.

Keywords

immersive services; point cloud compression; geometry and photometry

1 Introduction

Emerging immersive media services are capable of providing customers with unprecedented experiences. Representing as omnidirectional videos and 3D point cloud, customers would feel being personally at the scene, personalized viewing perspective and enjoy real-time full interaction. The contents of the immersive media scene may be the shooting of a realistic scene or the synthesis of a virtual scene. Although traditional multimedia applications still play a leading role, the unique immersive presentation and consumption methods of immersive media have attracted tremendous attentions. In the near future, immersive media will form a big market in a variety of areas such as video, games, medical cares and engineering.

The technologies for immersive media have increasingly appealed to both the academic and industrial communities. Among various newly proposed content types, 3D point cloud appears to be one of the most prevalent form of media presentation thanks to the fast development of 3D scanning techniques. 3D point cloud relies on modern measurement methods to record the collection of coordinate data of the surface of the object, and obtains the effect of “stealing truth” in the form of a three-dimensional model. Furthermore, each coordinate can have multiple attributes associated to it, where the attributes may correspond to color, reflectance or other properties of the object/scene that would be associated with a single point. It forms a spatially discrete set of points by sampling point data obtained by camera arrays, laser scanners, etc. which are pre-

sented in **Fig. 1**. This media type contains complete information on the surface of the object and the image after reconstruction is the most realistic “replica” of the object. **Fig. 2** shows typical point cloud scenarios. Based on the different applications, 3D point cloud can be well-classified into three categories: static objects and scenes, dynamic objects, and dynamic acquisition.

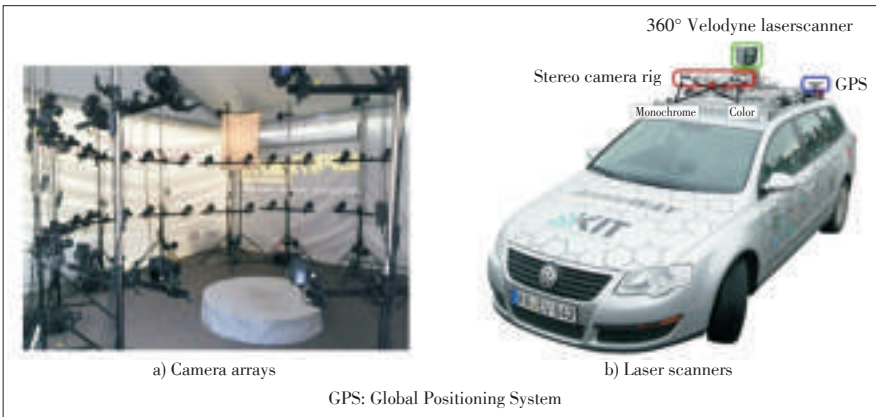
As described above, 3D Point cloud is an efficient representation as it can be seamlessly integrated and rendered in 3D virtual worlds, establishing a convergence between real and virtual realities and enabling more sophisticated applications. For instance, advances in 3D capture and reconstruction enable real-time generation of highly realistic 3D point cloud representations for 3D tele-presence. They can also improve the visual comfort related to the viewing of content with interactive parallax. Furthermore, they may enhance the performance in geographic information systems, culture heritage and autonomous navigation [1].

2 Challenges

In order to realistically represent the reconstructed scenes, a point cloud may be made up of thousands up to billions of points. This not only results in a huge amount of data, but also causes high complexity in the scattered random distribution of the spatial distribution, which brings great challenges to the storage and transmission system. Hence, more advanced compression coding techniques are needed to significantly reduce the amount of data, combining with the unique consumption

Introduction to Point Cloud Compression

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie



▲ **Figure 1. Data acquisition devices.**



▲ **Figure 2. Examples of 3D point cloud.**

characteristics of immersive media. For instance, technologies are needed for lossy compression of point clouds for use in real-time communications and Six Degrees of Freedom (6 DoF) virtual reality. In addition, technology is sought for lossless point cloud compression in the context of dynamic mapping for autonomous driving, cultural heritage applications, etc.

However, the existing coding technologies such as Advanced Video Coding (AVC) and High Efficiency Video Coding (HEVC) mainly focus on two-dimensional video compression. Although a new video coding standards working group for future video coding (FVC) has also been established, which fully considers the unique attributes of immersive media with 360-degree panoramic videos included in test sequences, FVC standards cannot be applied to three-dimensional immersive media such as point cloud yet. As a new three-dimensional spatial data model, point clouds have complex properties such as scattered distribution and time-varying irregularities compared with planar videos. Therefore, point cloud compression coding technology is urgently needed.

3 Technologies

3.1 Static Point Cloud Compression

3.1.1 Octree-Based Point Cloud Compression

In order to deal with irregularly distributed points in 3D space, various decomposition algorithms have been proposed.

In fact, the hierarchical tree data structure can effectively describe sparse 3D information. Octree based compression is the most widely used method in the literature. An octree is a tree data structure. Each node subdivides the space into eight nodes [2], [3]. For each octree branch node, one bit is used to represent each child node and called a voxel. This configuration can be effectively represented by one byte, which is considered as the occupancy node based encoding.

As shown in **Fig. 3**, each point is divided into octants when constructing octree on the point cloud. If a node contains more points than the threshold, it is recursively subdivided into eight nodes. Given a point cloud, the corners of the cube bounding box are set to the maximum and minimum values of the input point cloud-aligned bounding box. Then each point is assigned to the node it belongs to. Next, partitions and allocations are repeated until all leaf nodes contain no more than one point. Finally, an octree structure, in which each point is setted, is constructed.

By traversing the tree in different orders and outputting each occupied code encountered, the generated bit stream can be further encoded by an entropy encoding method such as an arithmetic encoder. In this way, the distribution of spatial points can be efficiently coded. In most cases, the points in each eight-leaf node are replaced by the corresponding centroid [4]. The decomposition level determines the accuracy of the data quantification and therefore may result in loss of the encoding. An octree based lossless coding algorithm has been introduced in [5]. In the sense that the quantization coordinates are preserved, it can be considered as lossless, using local surface approximations for compression.

In order to further improve the entropy encoding performance, various schemes are applied to adjust the ergodic sequence between octree voxels. By implementing a breadth-first or depth-first search on an axis-aligned grid, certain sequences



▶ **Figure 3. Octree decomposition.**

of voxels can be guaranteed and used as the basis for proper residual calculation [5]–[7]. In addition, experiments have been conducted to increase the flexible traversal order of occupancy codes based on probability reduction order or different leaf node prediction errors on the approximate surface. A prediction tree has been proposed to encode point clouds with potentially serialized point order and reduce redundancy through certain prediction rules [6]. Lossless compression is achieved by exploiting the correlation between the correction vectors, which is the difference between the predicted position and the actual position of the points.

In addition, various schemes are applied to further improve the entropy encoding performance. By implementing the breadth-first or the depth-first search on the axis-aligned grids, certain order of the voxels could be guaranteed and acts as the basis for proper residual computation [5]–[7]. Moreover, trials have been made to promote flexible traversal order for the occupancy code according to a probability descending order or the predicted error of different leaf node to an approximation surface. A prediction tree has been proposed to encode the point cloud by potentially serialized the points orders and reduce the redundancy via certain prediction rules. The lossless compression is achieved by exploiting the correlation between corrective vectors that are the difference between the predicted and real positions of a point.

3.1.2 Binary Tree Based Point Cloud Compression

We develop an efficient point cloud geometry compression scheme via binary tree partition and intra prediction [8]. As shown in **Fig. 4**, we take advantage of the binary tree structure for analyzing the geometry characteristics of the non-uniform data while decomposing the 3D space, providing the basis for the block wise efficient coding scheme. Next, we explore the optimal permutation of the points within specified leaf nodes and realize efficient intra prediction via the extended TSP in 3D space. Further proficiency is obtained for the simple residuals between sequential points. Moreover, the preserved information is largely reduced to only a single reference point for each leaf node. **Fig. 5** illustrates a novel lossless entropy coding tool PAQ which effectively combines the prediction and compression at the same time. The input context is modified to better fit point cloud and a corresponding optimal size is evaluated, achieving preferable compression performance.

3.1.3 Graph Based Point Cloud Compression

Graph is a data representation which could be used to describe signals in many applications, such as transportation and networks. Graph signal processing has obtained significant attention recently. For example, some methods using graph transform have been proposed to compress color information efficiently. As described in [9], the graph is

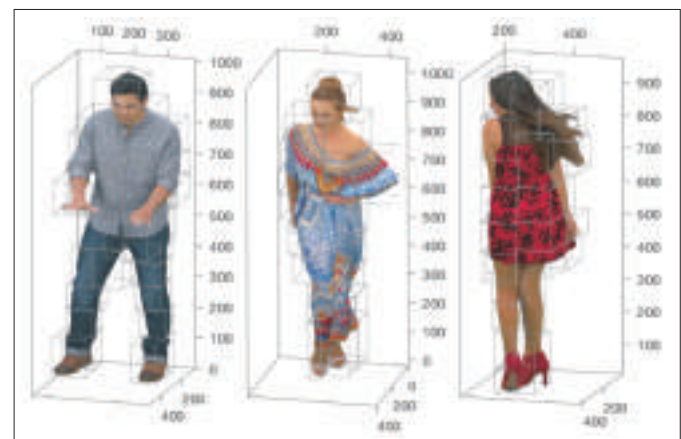
formed by connecting adjacent points in 3D space. If the distance of two points is less than the threshold, they are defined as adjacent points. The weight of the edge is inversely proportional to the distance between the two adjacent points. Then the adjacency matrix is constructed, which consists of the weights between the adjacent points. Next, the eigenvector matrix of the Laplacian matrix is calculated to implement the attribute transformation. One DC coefficient and one or more AC coefficients are obtained after the graph transformation.

Combined with block-based prediction and shape-adaptive Discrete Cosine Transform (DCT), a compact representation of the points is introduced to tackle the sparsely populated character of point clouds described in [10] and [11]. For the point cloud sequences, the approaches used in video coding like intra-frame prediction, motion estimation and compensation have been introduced into point clouds [10], [11].

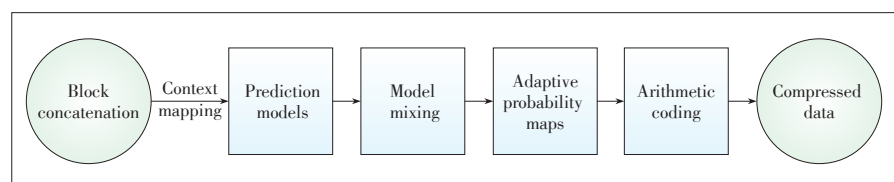
Since key points are distributed irregularly across the images, Tian et al. [12] used graph transform to represent the key point trajectories. This method makes the coding more efficient than traditional DCT based transformation and it is easier for energy compacting.

3.1.4 Clustering Based Point Cloud Compression

We propose a novel point cloud attribute compression scheme base on clustering [13]. **Fig. 6** shows our compression scheme. Global segmentation is successively implemented in photometric space and local segmentation is conducted in geometric space to split a point cloud into clusters, in which points share similar features [14]. Then, the genetic algorithm based 3D intra prediction is utilized to organize points of each



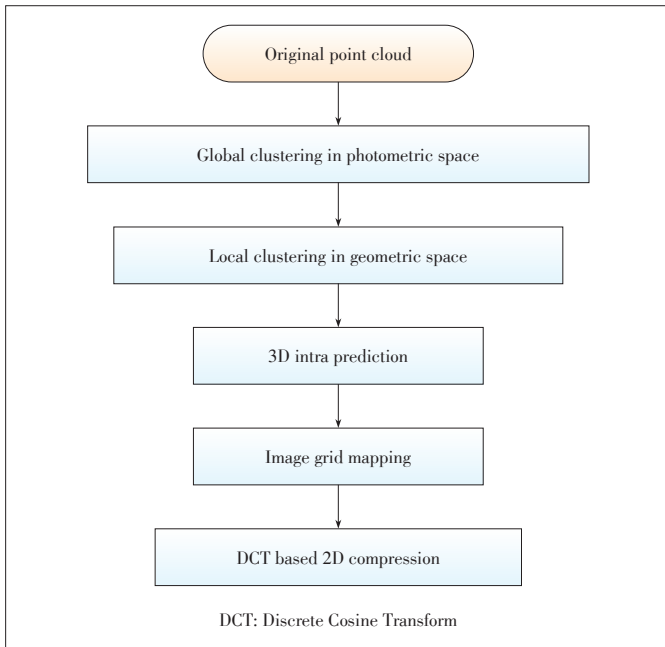
▲ **Figure 4.** Binary tree based partition and representation.



▲ **Figure 5.** Adapted PAQ8 compression procedure.

Introduction to Point Cloud Compression

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie



▲ Figure 6. Overview of the clustering based point cloud compression scheme.

cluster and then all points of each cluster are traversed in the order produced by the intra prediction algorithm. Next, the color attributes of those points are mapped to uniform grids via zigzag scan, which allows us to compress the raw point cloud data without voxelization or other preprocessing methods. A DCT based 2D image compression algorithm is also introduced to achieve impressive lossy compression performance.

3.2 Dynamic Point Cloud Compression

3.2.1 Motion-Compensated Point Cloud Compression

In Philip A Chou’s work [15]–[17], the 3D representation of choice is sparse voxel arrays, which they call voxelized point clouds. Neglecting the volumetric aspect of voxels, voxelized point clouds can be considered simply as point clouds whose points are restricted to lie on a regular 3D grid or lattice. For the kinds of data expected in 3D scene capture, voxelized point clouds are a more natural fit than dense voxels arrays, and they obviate the kinds of problems that polygonal meshes have with sampled data. Compared to color and depth maps, voxelized point clouds are a higher level representation, in which redundancies and inconsistencies between overlapping sensor maps have already been removed in a multi-camera sensor fusion step. Compared to arbitrary point clouds, voxelized point clouds have implementation advantages and are highly efficient for real-time processing of captured 3D data.

Each representation employs its own compression techniques; they believe graph-based 3D motion estimation and compensation, until recently, represented the state-of-the-art in (voxelized) point cloud color compression, with the former fo-

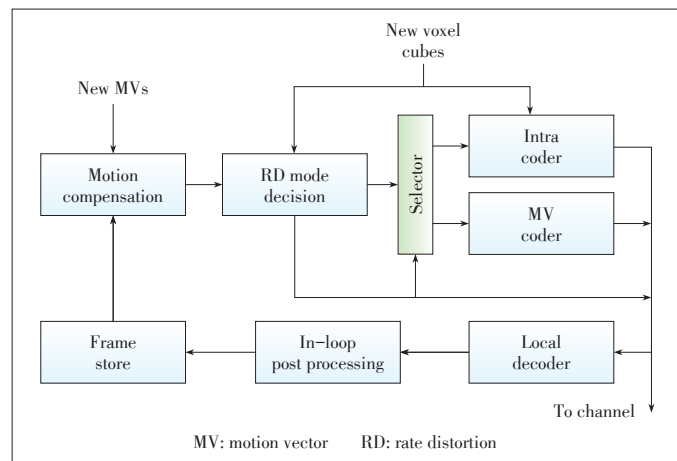
cus on intra-frame color compression and the latter extending that work to inter-frame color compression. The graph transform is a natural choice for the spatial transform of the color signal due to the irregular domain of definition of the signal. Unfortunately, the graph transform requires repeated eigen-decompositions of many and/or large graph Laplacians, rendering the approach infeasible for real-time processing. They have recently submitted a work on a coder that is able to match or outperform existing intra-frame color compression methods at a reduced cost. Such a point cloud coder is based on a region-adaptive hierarchical transform (RAHT) specially developed for point clouds and is used as a fundamental building block in the present framework for our dynamic point cloud coder, which can be considered as a 3D video coder.

Our objective is to build a coder for dynamic point clouds, which can be implemented in real time with existing technology and is expected to outperform the use of RAHT and octrees to compress color and geometry, respectively. In order to do this, they decided to explore the temporal dimension to remove temporal redundancies, i.e., to explore the fact that the geometry and color of the point cloud may not change much from one frame to another and to use $\mathcal{F}(t)$ as a predictor for $\mathcal{F}(t+1)$. At every discrete time t , the frame $\mathcal{F}(t) = \{V_{it}\}$, which is represented as a list of voxels in (1).

$$V_{it} = [x_{it}, y_{it}, z_{it}, Y_{it}, U_{it}, V_{it}], \tag{1}$$

Furthermore, they decided to explore 3D analogs of traditional video compression techniques. Motion estimation and motion compensation were used into the compression of dynamic point clouds, in order to achieve higher compression ratios at the expense of lossy coding of the geometry.

The coder (Fig. 7) is similar to a traditional video coder in essence, but they are actually quite different in details. In traditional video coders, the frame is broken into blocks of $N \times N$ pixels. However, the frame in the proposed coder is broken into blocks of $N \times N \times N$ voxels, i.e., the voxel space is



▲ Figure 7. Motion-compensated compression encoder.

partitioned into blocks and the list of occupied voxels is likewise partitioned into occupied blocks. Therefore, the occupied block at integer position (bx, by, bz) in a frame at instant $t+1$ is composed of occupied voxels $V_{i,t}+1$ within the block boundaries. Unlike traditional video coding, where the pixel position is known and the color is to be encoded, the need to encode the geometry along with the color makes it a distinct problem. So far, the geometry information has not been able to be encoded at a rate significantly lower than 2.5–3.0 bpv, which however can be achieved by using octrees without any prediction from $\mathcal{F}(t)$ to $\mathcal{F}(t+1)$. Therefore, the proposed coder does not encode geometry residuals and operates in two modes: either a block is purely motion compensated or it is entirely encoded in intra mode.

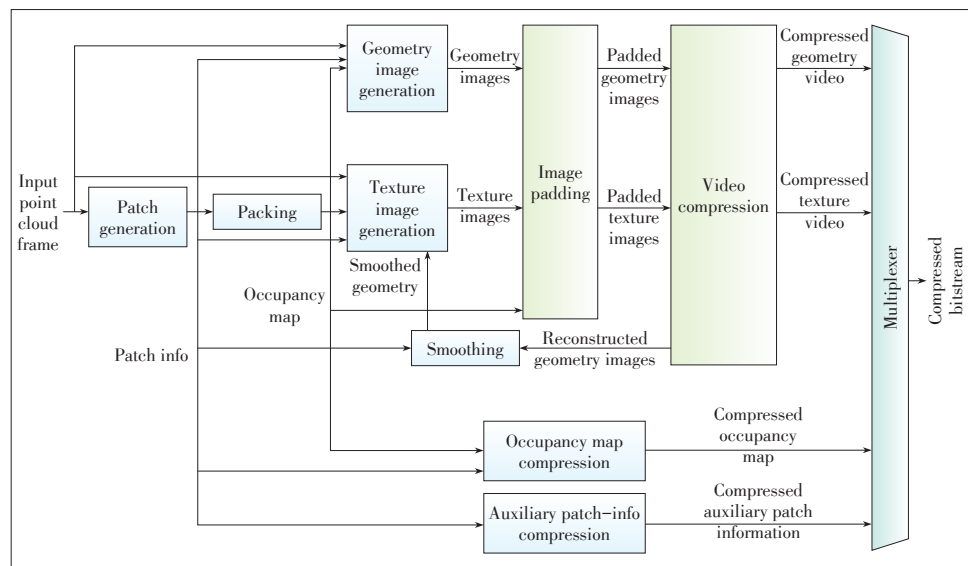
3.2.2 Video-Based Compression

Khaled proposed a video-based point cloud codec to the MPEG PCC group, aiming at test model category 2 (TMC2) [18]. Meanwhile, some studies on point cloud compression based on projection from 3D to 2D have also been proposed [19]–[21]. The main philosophy behind video-based compression is to leverage existing video codecs to compress the geometry and texture information of a dynamic point cloud, by essentially converting the point cloud data into a set of different video sequences. In particular, two video sequences, one for capturing the geometry information of the point cloud data and another for capturing the texture information, are generated and compressed using existing video codecs, e.g. using the HEVC Main profile encoder. Additional metadata that are needed to interpret the two video sequences, i.e., an occupancy map and auxiliary patch information, are also generated and compressed separately. The video generated video bitstreams and the metadata are then multiplexed together so as to generate the final point cloud TMC2 bitstream. **Figs. 8** and **9** provide overviews of the compression and decompression processes implemented in TMC2v0.

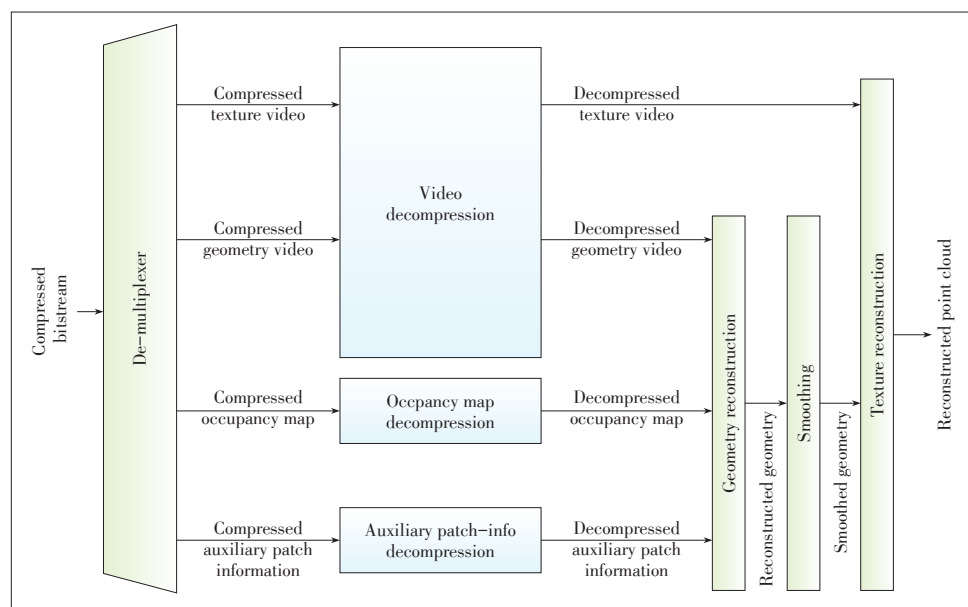
The patch generation process aims at decomposing the point cloud into a minimum number of patches with

smooth boundaries, while also minimizing the reconstruction error. First, the normal at every point is estimated from the fitting plane of the nearby points. An initial clustering of the point cloud is then obtained by associating each point with one of the following six oriented planes, defined by their normals. The packing process aims at mapping the extracted patches onto a 2D grid, while trying to minimize the unused space and guaranteeing that every TxT (e.g., 16×16) block of the grid is associated with a unique patch.

The image generation process exploits the 3D to 2D mapping computed during the packing process to store the geometry and texture of the point cloud as images. In order to better handle the case of multiple points being projected to the same



▲ Figure 8. Overview of the text model category 2 (TMC2v0) compression process.



▲ Figure 9. Overview of the text model category 2 (TMC2v0) decompression process.

Introduction to Point Cloud Compression

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie

pixel, each patch is projected onto two images, referred to as layers. The padding process aims at filling the empty space between patches in order to generate a piecewise smooth image suited for video compression. The occupancy map consists of a binary map that indicates for each cell of the grid whether it belongs to the empty space or to the point cloud. This could be encoded with a precision of a $B_0 \times B_0$ blocks and B_0 is a user-defined parameter. In order to achieve lossless encoding, B_0 should be set to 1. In practice, $B_0=2$ or $B_0=4$ will result in visually acceptable results, while significantly reducing the number of bits required to encode the occupancy map. The generated images/layers are stored as video frames and compressed using the HM16.16 video codec according to the HM configurations provided as parameters.

4 Conclusions

With the rapid development of 3D capture technologies, point clouds have been widely used in many emerging applications such as augmented reality and autonomous driving. However, a point cloud, used to represent real world objects in these applications, may contain millions of points, which results in huge data volume. Therefore, efficient point cloud compression algorithms are essential for reducing bandwidth and storage consumption.

References

- [1] C. Tulvan, R. N. Mekuria, Z. Li, S. Laserre, "Use cases for point cloud compression," ISO/IEC JTC1/SC29 WG11 MPEG Output Document, 2016.
- [2] J. Peng and C.-C. J. Kuo, "Geometry-guided progressive lossless 3D mesh coding with octree (OT) decomposition," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 609–616, Jul. 2005. doi: 10.1145/1073204.1073237.
- [3] Y. Huang, J. Peng, C.-C. J. Kuo, and M. Gopi, "A generic scheme for progressive point cloud coding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 440–453, Mar. 2008. doi: 10.1109/TVCG.2007.70441.
- [4] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, Apr. 2013. doi: 10.1007/s10514-012-9321-0.
- [5] R. Schnabel and R. Klein, "Octree-based point-cloud compression," in *3rd Eurographics/IEEE VGTC Conference on Point-Based Graphics*, Boston, USA, 2006, pp. 111–120. doi: 10.2312/SPBG/SPBG06/111-120.
- [6] S. Gumhold, Z. Kami, M. Isenbarg, and H.-P. Seidel, "Predictive point-cloud compression," in *ACM SIGGRAPH 2005 Sketches*, New York, USA, 2005. doi: 10.1145/1187112.1187277.
- [7] Y. Huang, J. Peng, C.-C. J. Kuo, and M. Gopi, "A generic scheme for progressive point cloud coding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 440–453, Jan. 2008. doi: 10.1109/TVCG.2007.70441.
- [8] W. Zhu, Y. Xu, L. Li, and Z. Li, "Lossless point cloud geometry compression via binary tree partition and intra prediction," in *IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, Luton, UK, 2017. doi: 10.1109/MMSP.2017.8122226.
- [9] C. Zhang, D. Florencio, and C. Loop, "Point cloud attribute compression with graph transform," in *IEEE International Conference on Image Processing (ICIP)*, Paris, France, 2014, pp. 2066–2070. doi: 10.1109/ICIP.2014.7025414.
- [10] R. A. Cohen, D. Tian, and A. Vetro, "Attribute compression for sparse point clouds using graph transforms," in *IEEE International Conference on Image Processing (ICIP)*, Phoenix, USA, 2016. doi: 10.1109/ICIP.2016.7532583.
- [11] R. A. Cohen, D. Tian, and A. Vetro, "Point cloud attribute compression using 3-D intra prediction and shape-adaptive transforms," in *IEEE Data Compression Conference (DCC)*, Snowbird, USA, 2016., 2016. doi: 10.1109/DCC.2016.67.

- [12] D. Tian, H. Sun, and A. Vetro, "Graph transformation for keypoint trajectory coding," *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Washington, DC, USA, 2016. doi: 10.1109/GlobalSIP.2016.7905881.
- [13] K. Zhang, W. Zhu, and Y. Xu, "point cloud attribute compression via clustering and intra prediction," to be published.
- [14] K. Zhang, W. Zhu, and Y. Xu, "Hierarchical segmentation based point cloud attribute compression," to be published.
- [15] R. L. de Queiroz and P. A. Chou, "Motion-compensated compression of dynamic voxelized point clouds," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 3886–3895, Aug. 2017. doi: 10.1109/TIP.2017.2707807.
- [16] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based motion estimation and compensation for dynamic 3D point cloud compression," in *IEEE International Conference on Image Processing (ICIP)*, Quebec City, Canada, 2015, pp. 3235–3239. doi: 10.1109/ICIP.2015.7351401.
- [17] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1765–1778, Apr. 2016. doi: 10.1109/TIP.2016.2529506.
- [18] "PCC Test Model Category 2 v0", ISO/IEC JTC1/SC29/WG11 w17248, Macau, China, 2017, 10.
- [19] L. He, W. Zhu, and Y. Xu, "Best-effort projection based attribute compression for 3D point cloud," in *Asia-Pacific Conference on Communications (APCC)*, Perth, Australia, 2017.
- [20] Nokia, "Nokia's response to cfp for point cloud compression (category 2)," ISO/IEC JTC1/SC29 WG11 Doc.m41779, Macau, China, Oct. 2017.
- [21] Technicolor, "Technicolor's response to the cfp for point cloud compression," ISO/IEC JTC1/SC29 WG11 Doc.m41822, Macau, China, Oct. 2017.

Manuscript received: 2018-03-29

Biographies

XU Yiling received her B.S., M.S., and Ph.D. from the University of Electronic Science and Technology of China in 1999, 2001, and 2004, respectively. She is a full researcher at Shanghai Jiaotong University, China. From 2004 to 2013, she was with the Multimedia Communication Research Institute of Samsung Electronics, Korea. Her research interests mainly include architecture design for next generation multimedia systems, cross-layer design, and dynamic adaptation for heterogeneous networks.

ZHANG Ke (zhang_ke@sjtu.edu.cn) received the B.S. degree from Nanjing University of Science and Technology, China in 2016. He is pursuing the master degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interest is about point cloud compression and video coding.

HE Lanyi (hly92711@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Xidian University, China in 2015. He is pursuing the master degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interest is about point cloud coding.

JIANG Zhiqian (zhiqjiang@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Xidian University, China in 2015. He is pursuing the Ph.D degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interests include reliable transmission and immersive media transmission.

ZHU Wenjie (rebecca28@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Harbin Institute of Technology, China in 2014. She is pursuing the Ph.D degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. Currently, she is studying in University of Missouri, Kansas City as an exchange scholar under the support of China Scholarship Council. Her research interests including point cloud compression and immersive media transportation. She has published around 6 conference papers and has 4 patents in processing with 3 MPEG proposals been accepted in international standard.

Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

WU Kesong¹, CAO Xianbin¹, CHEN Zhifeng², and WU Dapeng³

(1. Beihang University, Beijing 100191, China;

2. Fuzhou University, Fuzhou 350116, China;

3. University of Florida, Gainesville 32611, USA)

Abstract

Recent years have witnessed an explosive growth in mobile video-based services and efficient and reliable video delivery draws more and more attention. As a type of rateless codes, fountain codes can automatically adapt to wireless channel conditions without any knowledge of channels. This paper provides an overview of several typical Forward Error Correction (FEC) codes, such as Reed-Solomon (RS) code, Tornado code, Luby-Transform (LT) code, and Raptor code. We focus on a novel delay-aware fountain coding (DAF) technique that maximizes the code word length under the constraint of a given delay. Based on DAF, this paper also presents Unequal Error Protection DAF (UEP-DAF) which improves the Peak Signal to Noise Ratio (PSNR) without additional coordination between the encoder and the decoder, as well as Model Predictive Control DAF (MPC-DAF) which reduces the computational complexity to an affordable level for real-time video communications. Moreover, we review video streaming technologies, then introduce Dynamic Adaptive Streaming over HTTP (DASH) and DASH over Multiple Content Distribution Servers (MCDS-DASH) in detail. Based on MCDS-DASH that adapts video bitrate at the block level to alleviate video fluctuation, we propose a novel approach to integrating fountain codes with MCDS-DASH, which is capable of achieving unprecedented high throughput.

Keywords

mobile video delivery; DAF; UEP; MPC; DASH

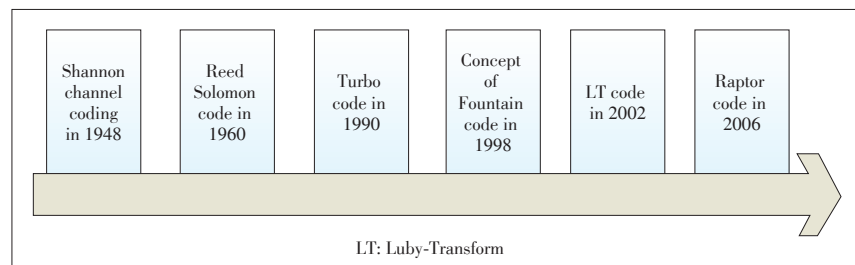
1 Introduction

During the past few years, mobile video-based services have witnessed an explosive growth, such as YouTube, FaceTime, and WeChat, with the popularity of smart mobile devices as well as the evolution of 3G, Long Term Evolution (LTE), Wi-Fi and other communication technologies. It is also expected that traffic from virtual reality (VR), augmented reality (AR), 3D games, surveillance video, vehicle networking and drone video will increase significantly in the near future. However, the problems including packet loss and insufficient bandwidth easily deteriorate in even higher demanding video applications due to the stochastic nature of wireless channels. As a result, how to deliver low latency and high quality video over wireless networks poses an unprecedented challenges for both academia and industry.

Erasure codes (Fig. 1), such as Reed-Solomon (RS) code proposed first by Reed and Solomon in 1960 and Tornado code that Luby et al. put forward in 1997, are comparatively powerful

in error correction for data transmission over network. Nevertheless, they are not suitable for wireless transmission due to high co-decoding complexity. In addition, some fixed code rate must be chosen when the encoding begins, which could also lead to bandwidth waste if the erasure rate is overestimated, otherwise poor video quality.

Luby, Byers, et al. first presented the concept of digital fountain (DF) in 1998 for large-scale data distribution services and reliable broadcast/multicast services. In this concept, the number of encoded symbols that can be generated from the source data is potentially limitless and the code rate can automatically adapt to wireless environment without assuming any knowl-



▲ Figure 1. Development of typical erasure codes.

Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

WU Kesong, CAO Xianbin, CHEN Zhifeng, and WU Dapeng

edge of channels.

In 2002, Luby further proposed the first practical DF codes, Luby-Transform (LT) code [1]. Although the LT code exhibits excellent efficiency compared to traditional erasure codes, the shortcomings are obvious: decoding cannot be successful only until all the source packets are recovered; the decoding complexity of LT code is linear logarithmic order $O(k \log k)$, k is the number of source symbols.

In 2006, Shokrollahi put forward the most effective DF code, Raptor code, which was obtained by concatenating a high rate low-density parity-check (LDPC) code with an LT code of constant average degree distribution [2]. The Raptor code exhibits linear co-decoding time while still keeping low coding overhead. More importantly, it has been employed in the Digital Video Broadcasting—Handheld (DVB-H) for IP Datacast (IP-DC) and the Third Generation Partnership Project (3GPP) standard for multimedia broadcast multicast services (MBMS) [3].

Internet video streaming services have also witnessed a tremendous growth with the evolution of Internet and the popularity of mobile intelligent devices. Traditional Real-Time Transport Protocol/Real Time Stream Protocol (RTP/RTSP) has increasingly become unable to answer the demand for the following reasons: high deployment cost, indispensable stream session management, exceedingly difficult traversing restrictive network address translators and firewalls, as well as not supported by prevalent content delivery networks (CDN).

Advanced streaming media technologies are urgently needed to solve the aforementioned problems. HTTP-based dynamic adaptive streaming media technology emerges as the times require, and has already developed rapidly.

The rest of this paper is organized as follows. Section 2 briefly describes block coding and sliding window coding, and introduces delay-aware fountain coding (DAF) in detail. Section 3 presents Unequal Error Protection DAF (UEP-DAF) and Model Predictive Control DAF (MPC-DAF) based on DAF. Section 4 reviews Dynamic Adaptive Streaming over HTTP (DASH) and proposes our novel scheme. Section 5 concludes the paper.

2 Delay-Aware Fountain Codes

Fountain codes are widely used in network transmission, such as reliable multicast, multi-source parallel downloading and distributed storage. However, if a video streaming file is delivered with conventional fountain codes, it cannot be displayed until the entire file is successfully decoded. Unfortunately, video streaming is of timeliness that means the time interval of video generation and display must not exceed a certain threshold. In addition, due to limited memory capabilities, receiver devices may also impose some restrictions on the receiving time of frames.

2.1 Block and Sliding Window Coding

The most direct solution to transmit video streaming with

fountain codes is to partition video streams into blocks (**Fig. 2**), and then encode and transmit them sequentially. In [3], Ahmad et al. present a block coding design of fountain codes for video transmission. Considering the playback delay, the smaller the block size is, the shorter it is, however, from the fountain code point of view, the larger the block size, the lower the coding overhead and the higher the coding efficiency.

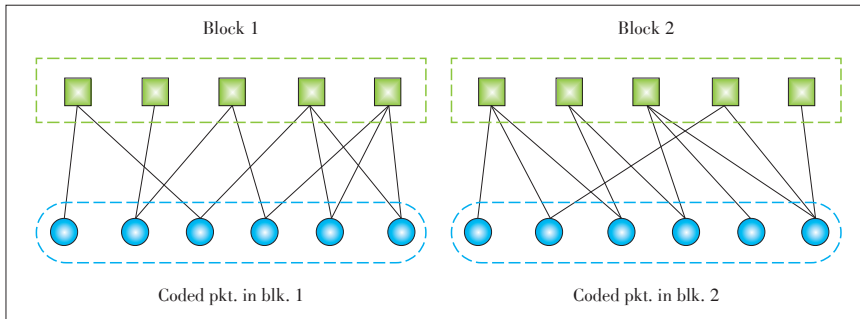
In [4], Bogino et al. proposed a sliding window approach to segment the source data, leaving some overlaps between successive steps, as shown in **Fig. 3**. Consequently, decoded packets in one window are valuable for decoding the coded packets of subsequent windows. The sliding window scheme virtually extends the size of source block with respect to the non-overlap block coding, thereby obtains a higher co-decoding efficiency with the overhead decrease. In addition, expanding window was presented in [5]. In the aforementioned three schemes, the block size is fixed in block coding, virtually extended in the sliding window, and kept growing in expanding window.

2.2 Delay-Aware Fountain Codes

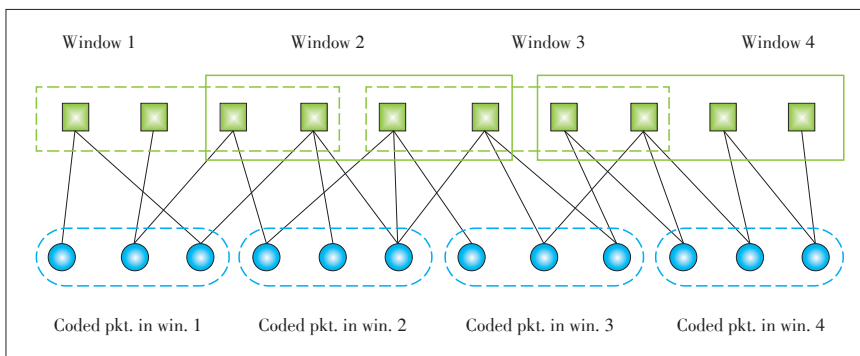
In 1977, Eliece first proposed the joint source channel coding scheme to enhance the overall performance of the communication system. Although numerous studies on joint fountain codes and video coding have also been done, such as [3], [4], [5], and [6], they have a common shortcoming that is to divide the video streaming into blocks with the fixed number of packets without considering the video content characteristics. Accordingly, the same frame is probably divided into different blocks, which has a negative impact on the decoding of compressed video stream. Moreover, due to the fluctuation of video bit rate (**Fig. 4**), each packet-based block may contain different number of frames and the resulting video jitter increases as a result of the varying latency.

As is known to all that the concepts relevant to time in video sequence can be measured with number of frames. Consequently, different from the existing sliding window schemes, Sun, Wu et al. [7] innovatively proposed to establish the sliding window with the fixed number of frames to provide the much desired delay awareness in video streaming, namely DAF. In this way, the code word length can be maximized under the condition of limited delay, so as to achieve higher coding gain. In addition, a low-complexity online version DAF-L was also proposed, adopting only one parameter—slope factor to quantify the sampling distributions.

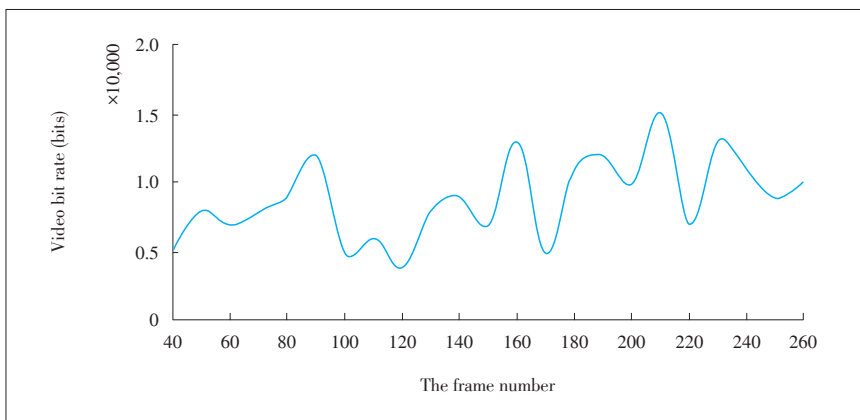
Design of degree distribution has a critical influence on the decoding performance of fountain codes, and optimization of degree distribution functions has been done in certain literatures, such as [1]. It is worth noting that in fountain codes the more uniform the sampling distribution of the source packets, the higher the coding efficiency [7]. As a result, we proposed the strategy of window-wise sampling which dynamically adjusts each window sampling mode according to the fluctuation



▲ Figure 2. Block coding.



▲ Figure 3. Sliding window coding.



▲ Figure 4. Common intermediate format (CIF) sequence foreman.

of video bit rate.

$$ASP(t) = \sum_{\omega \in \text{all windows cover } t} p_{\omega}^{pkt}(t), \quad (1)$$

where $p_{\omega}^{pkt}(t)$ denotes the average sampling probability of each packet in frame t within window ω . $ASP(t)$ denotes the total probability of every packet in frame t accumulated through all the sliding windows covering that frame. The objective function of the major optimization process in DAF is to minimize the variance of the accumulated sampling possibility $ASP(t)$ as in (1).

Fig. 5 shows the experiments, where the code rate is defined using the ratio of total number of native packets to total num-

ber of coded packets, and IDR denotes in-time decoding ratio, prove that this scheme has higher video quality than existing schemes with low delay and constant data rate.

3 UEP-DAF and MPC-DAF

All the video data has been assumed of the same importance in DAF, in order to further enhance the practical applicability, we propose a method to integrate UEP into DAF.

3.1 UEP-DAF

The aforementioned schemes are all based on the assumption that all the data has the same importance. However, the importance of different levels for video stream could be reflected in the following aspects, such as picture types, data types, position of the frame in a group of pictures (GOP), layers in scalable video coding (SVC), and picture content [8]. Fountain codes with equal error protection (EEP) are obviously inefficient, especially in the case of video delivery over wireless channel, the bandwidth of which is more often insufficient and decoding all packets with equal chance sometimes could induce suboptimal quality.

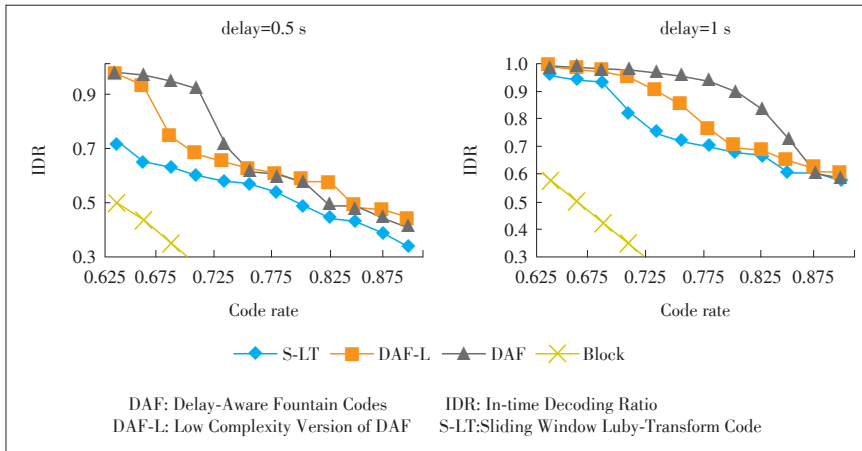
In recent years, fountain codes with unequal error protection (UEP) characteristics have attracted extensive attentions. In 2005, Rahnavard and Fekri suggested for the first time UEP-based fountain codes [9] which principle was to encode data by a method of adopting different degree distribution according to unequal importance. In [5] and [6], Sejdinovic and Vukobratovic proposed an approach of expanding window fountain codes, in which all windows had the same starting position and packets in each window must be a subset of

the subsequent window (Fig. 6), so packets in the innermost window had the highest sampling probability. In order to achieve high efficient transmission of data in multiple source relay channels, Talari et al. presented a distributed unequal error protection fountain codes [10] to meet the requirements of different terminals. Ahmad, Hamzaoui et al. divided video source data block into several segments, and then duplicated them according to the protection factors to obtain a new block [11], as shown as in Fig. 7, where Mib denotes most important bits, and Lib denotes least important bits.

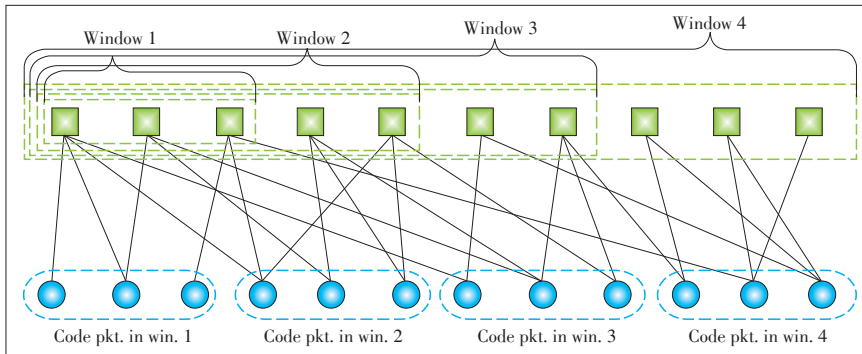
The aforementioned UEP-based schemes can improve remarkably the peak signal-to-noise ratio (PSNR) without the bit rate increasing. Nevertheless, they all have an unfavorable

Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

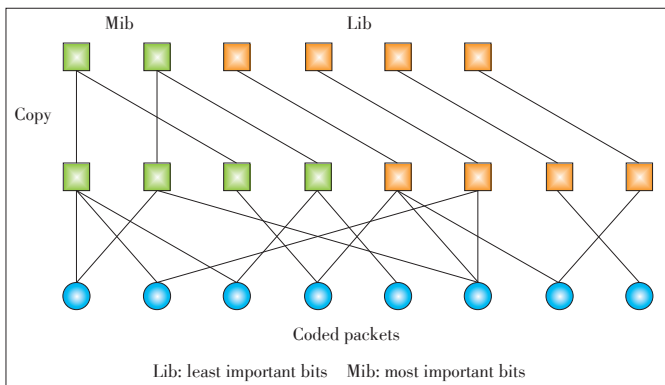
WU Kesong, CAO Xianbin, CHEN Zhifeng, and WU Dapeng



▲ Figure 5. IDR vs. the code rate of four window schemes.



▲ Figure 6. Expanding window coding.



▲ Figure 7. UEP by duplicating Mib packets.

problem which is the encoder needs to send importance description information to the decoder. Whether coordinating between encoder and decoder beforehand or explicit transmission in the packet headers, the resulting overhead will increase the possibility of packet loss.

In [8], Sun, Wu et al. proposed a novel scheme UEP-DAF to apply UEP to video communication applications based on delay-aware fountain codes. The proposed scheme does not need additional coordination between encoder and decoder; besides, the frame-level importance profile may be specified in ad-

vance. Simulation experiments show that compared to the result of DAF, UEP-DAF allocates higher sampling probability to the frames in the front of a GOP than that in the back (Fig. 8) where ASP denotes the accumulated sampling possibility. Consequently, the proposed system achieves higher decoding ratios and PSNR compared to EEP under the same network conditions.

3.2 MPC-DAF

DAF based on delay-aware sliding window and window-wise sampling, takes full advantage of channel-adaptive rateless feature, effective delay control and optimal sampling pattern, therefore it outperforms the other existing schemes. However, high computational complexity induced by the per-window optimization of sampling pattern and that the bit rate information of all frames needs to be obtained in advance, prevent its applications in live video streaming. The encoding computational complexity of DAF is $O(((T \cdot W)/\Delta t)^3)$, where T denotes video length, W denotes sliding window size, and Δt denotes step size. The detailed deduction can be found in [7]. Although the performance of low-complexity DAF-L proposed in [7] is higher than that of many other existing schemes, compared to DAF the gap is still significant, especially when packet loss is relatively serious.

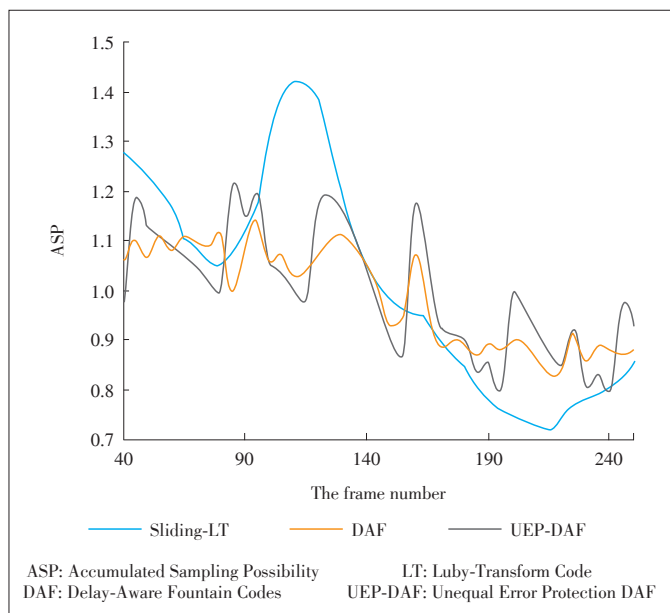
DAF-M based on the Model Predictive Control (MPC) was proposed in [12], which is an online optimization algorithm with a lower computational complexity with respect to DAF, but higher performance compared to DAF-L. More details of MPC-DAF code can be found in [12]. As a result, the computational complexity is lowered to an affordable level for real-time video communications. Moreover, the results of simulation experiments show that the decoding ratio of DAF-M is close to the global optimum in DAF codes [12].

4 MCDS-DASH Based on Fountain Codes

HTTP-based incremental download technology, overcoming the aforementioned problems, has become widespread, including Adobe's flash player, Microsoft Corp.'s Silverlight and windows media player, etc. However, it is inevitable that video playback interruption and video quality degradation occur as a result of the constantly changing network bandwidth, especially for the stochastic wireless networks.

4.1 MPEG-DASH

New generation streaming media technology, so-called the



▲ Figure 8. ASP of three sampling schemes.

adaptive streaming media scheme, becomes more and more popular, because it can provide users with real-time, smooth and high quality streaming service [13]. Meanwhile, numerous HTTP adaptive streaming schemes were also proposed, including HTTP Live Streaming (HLS) from Apple Corp., Smooth Streaming (SS) from Microsoft Corp. and HTTP Dynamic Streaming (HDS) from Adobe Corp. However, the diversity of different streaming media service solutions brings about a large number of compatibility issues, which increases the difficulty of system maintenance.

DASH, also known as MPEG-DASH, is the first international standard of HTTP-based adaptive streaming media solution. Moving Picture Experts Group (MPEG) issued a call for proposal for an HTTP streaming standard in April 2009, then started the evaluation of the submitted fifteen full proposals since July 2009. The DASH international standard draft was completed in January 2011 and became the international standard in November 2011. In April 2012, the international standard of DASH with ISO/IEC 23009 - 1 was officially promulgated. Since then DASH has been widely adopted for providing uninterrupted video streaming service to users with dynamic network conditions and heterogeneous devices [14], [15].

Media presentation description (MPD) is a manifest file of encoding information defined by DASH server, including the descriptions of time-based periods, representations based on the bit rate, frame rates and resolutions, as well as video data segments. The DASH client is capable of choosing to download and display the most appropriate video segment according to the network conditions and the receiver buffer state.

Adaptive bit stream switching algorithm, as the core technology, has always been the most critical factor affecting the efficiency of DASH. An efficient rate adaptation algorithm should

prevent frequent hopping between video bitrates, as well as frequent interruptions or non-optimal visual quality due to higher or lower than the available bandwidth.

In [16], Ojanper et al. proposed an adaptive network aided method for adaptive HTTP video stream based on cognitive network management architecture and distributed control. Müller et al. proposed an improved DASH implementation employing scalable video coding (SVC), which was suitable for mobile applications with large bandwidth variations [17]. Besides, a new model predictive control algorithm to optimize the throughput and buffer occupancy information was also proposed in [18].

4.2 DASH over Multiple Content Distribution Servers (MCDS-DASH)

Generally traditional DASH is based on single server, so no matter how excellent adaptation algorithm is adopted, all efforts seem helpless once the server is unreachable or even down. In order to improve the bandwidth and stability of the transmission, a parallel download technology based on MCDS was proposed in [19].

MCDS-DASH deploys DASH over multiple servers; thereby the same content can be available at multiple URLs concurrently and the DASH client can obtain video segments at the maximum bandwidth from the optimal server. Compared with the single server node, MCDS can obviously provide higher bandwidth, reliability and scalability.

However, rate adaptation control becomes a more challenging problem in MCDS-DASH. Because if it adapts video bitrate still at the segment level as traditional methods do, frequent video bitrate switching would occur due to diverse bandwidths over multiple heterogeneous servers. In that case, viewing experience of users will decline dramatically [20]. Moreover, disorder downloading of video segments from different servers and the compulsory requirements of playback according to the correct order would cause additional delay.

In [19], Chao Zhou et al. presented to group multiple segments into a block and adapt video bitrate at the block level rather than at the segment level to alleviate the video fluctuation. Furthermore, downloading each segment according to the playback deadline was also proposed.

4.3 MCDS-DASH Based on Fountain Codes

Bitrate smoothness and bandwidth utilization are a couple of contradictions due to the inherent bandwidth variations in MCDS-DASH. Although the scheme proposed in [19] adapting video bitrate at the block level alleviates actually the video drastic fluctuation, this is at the expense of lower bandwidth utilization.

The order of video segments download completion may not be in accordance with the one you assume based on the current network status, especially in stochastic wireless networks, which would cause additional delay. In fact, the approaches mentioned in [19] are not completely parallel, because the serv-

Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

WU Kesong, CAO Xianbin, CHEN Zhifeng, and WU Dapeng

ers having completed download will keep idle in one block unit, which actually is a waste of bandwidth resources.

Nevertheless, the block-based scheme prompts us an innovative inspiration to employ fountain codes in MCDS-DASH. In this scheme, we code video streaming files by a method of adopting fountain codes, then send the coded packets to the MCDS. In that sense DASH clients need not care about how many packets downloaded from each specific DASH server, and as long as the number of received packet reaches the threshold, the video streaming file can be decoded successfully. In that sense our novel scheme can achieve actually parallel download.

5 Conclusions

In this paper we presented an overview of several typical FEC codes, and focused on the state-of-the-art fountain codes DAF as well as its extended UEP-DAF and MPC-DAF. We also did research on respective advantages, disadvantages, suitable application scenarios etc. Furthermore, we reviewed the development of video streaming technologies, and paid a special attention to both DASH and MCDS-DASH in this work. Based on grouping multiple segments into a block and adapting video bitrate at the block level in MCDS-DASH, we put forward an innovative scheme which is able to integrate fountain codes with MCDS-DASH. Our scheme is capable of achieving unprecedented high throughput when multiple servers exert best efforts to transmit the same video file to clients.

References

- [1] M. Luby, "LT codes," in *43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Vancouver, Canada, 2002, pp. 271–280.
- [2] A. Shokrollahi, "Raptor codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006. doi: 10.1109/TIT.2006.874390.
- [3] S. Ahmad, R. Hamzaoui, and M. M. Al-Akaidi, "Unequal error protection using fountain codes with applications to video communication," *IEEE Transactions on Multimedia*, vol. 13, no. 1, pp. 92–101, 2011. doi: 10.1109/TMM.2010.2093511.
- [4] M. C. Bogino, P. Cataldi, M. Grangetto, E. Magli, and G. Olmo, "Sliding-window digital fountain codes for streaming of multimedia contents," in *IEEE International Symposium on Circuits and Systems*, New Orleans, USA, May 2007, pp. 3467–3470. doi: 10.1109/ISCAS.2007.378373.
- [5] D. Sejdinovic, D. Vukobratovic, A. Doufexi, V. Senk, and R. J. Piechocki, "Expanding window fountain codes for unequal error protection," *IEEE Transactions on Communications*, vol. 57, no. 9, pp. 2510–2516, 2009. doi: 10.1109/TCOMM.2009.09.070616.
- [6] D. Vukobratovic, V. Stankovic, D. Sejdinovic, L. Stankovic, and Z. Xiong, "Scalable video multicast using expanding window fountain codes," *IEEE Transactions on Multimedia*, vol. 11, no. 6, pp. 1094–1104, 2009. doi: 10.1109/TMM.2009.2026087.
- [7] K. Sun, H. Zhang, and D. Wu. (2016). Delay-aware fountain codes for video streaming with optimal sampling strategy [Online]. Available: <http://arxiv.org/abs/1605.03236>
- [8] K. Sun and D. Wu, "Unequal error protection for video streaming using delay-aware fountain codes," in *IEEE ICC 2017 Communications Software, Services, and Multimedia Applications Symposium*, Paris, France, May 2017. doi: 10.1109/ICC.2017.7996740.
- [9] N. Rahnavard and F. Fekri, "Finite-length unequal error protection rateless codes: design and analysis," in *Global Telecommunications Conference*, St. Louis, USA, Dec. 2005. doi:10.1109/GLOCOM.2005.1577872.
- [10] A. Talari and N. Rahnavard, "Distributed unequal error protection rateless

codes over erasure channels: a two-source scenario," *IEEE Transactions on Communications*, vol. 60, no. 8, pp. 2084–2090, 2012. doi: 10.1109/TCOMM.2012.051512.110109.

- [11] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless codes with unequal error protection property," *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1521–1532, 2007. doi: 10.1109/TIT.2007.892814.
- [12] K. Sun and D. Wu, "MPC-based delay-aware fountain codes for live video streaming," in *2016 IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6. doi: 10.1109/ICC.2016.7510691.
- [13] C. Müller and C. Timmerer, "A test-bed for the dynamic adaptive streaming over HTTP featuring session mobility," in *Second Annual ACM Conference on Multimedia Systems*, 2011, pp. 271–276. doi: 10.1145/1943552.1943588.
- [14] A. Begen, T. Akgul, and M. Baugher, "Watching video over the web: part 1: streaming protocols," *IEEE Internet Computing*, vol. 15, no. 2, pp. 54–63, Mar. 2011. doi: 10.1109/MIC.2010.155.
- [15] R. Kuschnig, I. Kofler, and H. Hellwagner, "Evaluation of HTTP-based request-response streams for Internet video streaming," in *Proc. ACM MMSys11*, Feb. 2011, pp. 245–256.
- [16] T. Ojanperä, and H. Kokkonen, "Wireless bandwidth management for multiple video clients through network-assisted DASH," in *IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks*, Coimbra, Portugal, Jun. 2016. doi:10.1109/WoWMoM.2016.7523530.
- [17] C. Müller, D. Renzi, S. Lederer, S. Battista, and C. Timmerer, "Using scalable video coding for dynamic adaptive streaming over HTTP in mobile environments," in *20th European Signal Processing Conference (EUSIPCO 2012)*, Bucharest, Romania, Aug., 2012, pp. 2208–2212.
- [18] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," in *2015 ACM Conference on Special Interest Group on Data Communication*, Larnaca, Cyprus, Jul. 2015, pp. 325–338. doi: 10.1109/ISCC.2015.7405532.
- [19] C. Zhou, C.-W. Lin, X. G. Zhang, and Z. M. Guo, "A control-theoretic approach to rate adaptation for DASH over multiple content distribution servers," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, Apr. 2014. doi: 10.1109/TCSVT.2013.2290580.
- [20] E. C. R. Mok, X. Luo, and R. Chang, "QDASH: a QoE-aware DASH system," in *Proc. ACM Multimedia Syst.*, Chapel Hill, USA, Feb. 2012, pp. 11–22. doi: 10.1145/2155555.2155558.

Manuscript received: 2018-04-03

Biographies

WU Kesong (sdtwks@163.com) received his M.S. in communication & information system from Beihang University, China in 2014. He is now a Ph.D. candidate in Department of Electronic and Information Engineering, Beihang University. His research interests include multimedia communications, digital image and video processing, channel encoding, and machine learning.

CAO Xianbin (xbcao@buaa.edu.cn) is the Dean and a professor at the School of Electronic and Information Engineering, Beihang University, China. His current research interests include intelligent transportation systems, airspace transportation management, and intelligent computation. Currently, he serves as Associate Editor of *IEEE Transactions on Network Science and Engineering*, and Associate Editor of *Neurocomputing*.

CHEN Zhifeng (10799126@qq.com) received the Ph.D. degree in electrical and computer engineering from the University of Florida, USA in 2010. He is a professor with the College of Physics and Information Engineering, Fuzhou University, China. His research interests include video coding, video transmission, computer vision, machine learning, etc.

WU Dapeng (dpwu@ufl.edu) is a professor at the Department of Electrical and Computer Engineering, University of Florida, USA. His research interests are in the areas of networking, communications, signal processing, computer vision, machine learning, smart grid, and information and network security. He serves as Editor in Chief and Associate Editor of multiple IEEE journals. He was the founding Editor-in-Chief of *Journal of Advances in Multimedia*. He has served as a member of executive committees and/or technical program committees of over 80 conferences. He is an IEEE Fellow.

DASH and DASH-VR Video Multicast Systems

PARK Jounsup and HWANG Jenq-Neng
(University of Washington, Seattle, WA 98195, USA)

Abstract

Multimedia data traffic occupies more than 70% of the Internet traffic and is still growing. On-demand video is already a major video content platform and private broadcast is getting more popular. In addition to this, virtual reality (VR) and augmented reality (AR) data traffic is increasing very fast. To provide the good quality of the multimedia service, huge amount of resource is needed because users' service experience is usually proportional to the video rates they can receive. Moreover, the variation of the bandwidth also affects to the users' experience, while more users want to use their mobile devices to see multimedia data by accessing the network through wireless links, such as Long Term Evolution (LTE) and Wi-Fi. Therefore, better spectral efficiency during wireless transmission and video rate adaptation to provide better quality to users are in great demand. Multicast system is one of the technologies that can improve the spectral efficiency drastically, and Dynamic Adaptive Streaming over HTTP (DASH) is one of the most popular video rate adaptation platforms. In this paper, we investigate the state-of-the-art video multicast technologies. LTE supports the multicast service through evolved Multimedia Broadcast Multicast Service (eMBMS) systems, and there are different algorithms to perform the video multicast along with adaptive video quality control. The algorithms include the procedure to decide the video rates, resource allocations, and user groupings. Moreover, we propose a novel approach to improve the quality of experience for DASH-VR video multicast systems.

Keywords

VR; video multicast; LTE; eMBMS

1 Introduction

The enhanced capabilities of mobile devices and the improved capacities of wireless networks have led to a massive growth in mobile video consumption. A recent report [1] shows that the video traffic occupies more than 70% of the whole Internet traffic in peak time, and moreover, half of the video consumers use mobile devices. Moreover, virtual reality (VR) and augmented reality (AR) applications are getting more popular and users can enjoy diverse experience with them. However, VR/AR applications need more data than conventional video streaming services. As multimedia data traffic is increasing over wireless networks, efficient utilization of the wireless resources is getting more important for serving more users. Moreover, wireless channel condition frequently varies with channel environments and user behaviors. MPEG's Dynamic Adaptive Streaming over HTTP (MPEG-DASH) [2] is thus proposed as an effective video streaming platform, which enables the adaptive rate selection based on the channel conditions. DASH can provide superior video experience by giving clients a chance to receive the video quality based on their channel condition and buffer status, resulting in better quality of experience (QoE). Most of Internet

video service providers, such as Netflix and Youtube, support DASH in their video streaming platforms. DASH is extended for VR video streaming (DASH-VR) and it supports tiled video rate adaptations and reconstruction of VR videos.

With the overloaded scenarios, for example, many people watch the popular live videos such as sports events, bandwidth can be easily used up and many people will suffer from delay or low video quality. To overcome the problem, video multicasting can be utilized. LTE allows using their spectrum for multicasting or broadcasting up to 60% of the spectrum and it is standardized as evolved Multimedia Broadcast Multicast Service (eMBMS) [3]. The multicast channel (MCH), that delivers eMBMS data, cannot get any advantage from either Hybrid Automatic Repeat-Request (HARQ) or retransmission since the MCH transfers the data in radio link control (RLC) unacknowledged mode (UM) [3], due to the fact that a single user's channel condition cannot represent all users' channel conditions. Besides, it is very inefficient to retransmit many lost packets to user equipment (UE) with poor channel conditions, resulting in further consumption of bandwidth. This situation makes QoE worse because it cannot transmit the appropriate video representations to the subscribed users, resulting in very high packet loss rate or possibility that users cannot get a video with

DASH and DASH-VR Video Multicast Systems

PARK Jounsup and HWANG Jenq-Neng

enough quality even when the channel condition is very good. To overcome the problem of DASH multicasting, the File Delivery over Unidirectional Transport (FLUTE) [4] protocol is thus introduced. The Internet Engineering Task Force (IETF) introduces FLUTE for unidirectional data transfer over the Internet. To avoid packet loss, FLUTE adds redundant packets to help the recovery of the lost packet, which is done by forward error correction (FEC) [5]. Moreover, if FEC is not enough to recover all the lost packets, DASH clients can request packet recovery through reliable TCP unicast transmission [6].

Combining FLUTE and eMBMS of LTE makes DASH multicasting possible with capability of adaptive video quality control, however, it introduces more complexity to the systems. Since there are multiple copies of the video with different rates, the system has to choose which video rates to be scheduled based on the channel information and user's requests. FLUTE sessions have to add redundant APP-layer FEC packets to protect the video data while not losing efficiency. Moreover, resources for each FLUTE session must be allocated in the orthogonal frequency - division multiple access (OFDMA) frames and PHY -layer modulation and coding schemes (MCS) for the chosen resource blocks also need to be selected for reliable communications. Its complexity exponentially increases as number of users and/or number of video increases to find the optimal solution. Moreover, channel condition always changes frequently, therefore, it is more difficult to optimize the whole system in real time.

DASH or scalable video coding (SVC)-based multicasting algorithms have been introduced to efficiently solve this problem and give more users better video quality [7]. Park et al. [8] show that the total utility can be improved and more users can watch better video by using DASH multicast over LTE. This algorithm allocates one video representation in one multicast video session; therefore, there is corresponding video quality when we allocate the resource to the video sessions. However, in case of tiled VR videos, the multiple tiles share the resource, and many combinations of tiles with different representations are possibly allocated in a single multicast video session. Therefore, the video quality not only depends on the allo-

cated resource but also on the tile-based rate-selection algorithm. In this paper, a new approach is proposed to allocate the DASH-VR video on LTE eMBMS systems.

The remainder of this paper is organized as follows. Section 2 summarizes the related works. Section 3 introduces the existing DASH multicasting systems and algorithms. Section 4 proposes a new approach to perform the DASH-VR multicast systems, followed by the conclusion in Section 5.

2 Related Works

2.1 LTE eMBMS

LTE supports multicasting of video streams by eMBMS [3] system (**Fig. 1**), and the broadcast multicast service center (BMSC) is responsible for managing multicast sessions. It provides membership, session and transmission, proxy and transport, service announcement, security, and content synchronization. An MBMS gateway (MBMS-GW) distributes the video data to eNBs. It performs session control signaling towards the mobile management entity (MME). Multi-cell/multicast coordination entities (MCEs) are part of eNBs and they provide admission control. They allocate the radio resource for multicast sessions and decides MCS. Multiple video multicasting sessions can thus be created and users can subscribe those sessions at the same time.

The physical layer of an LTE downlink is based on the OFDMA technology, and the basic resource unit in the LTE system is a physical resource block (RB), which has 180 KHz bandwidth with 12 subcarriers and 7 symbols [9]. Within an RB, the same MCS is applied for all subcarriers. Therefore, if we define the MCS of an RB, there is the corresponding number of bits that one RB can carry, which is

$$c(\text{MCS}) = 12(\text{subcarriers}) \times 7(\text{symbols}) \times \text{efficiency}. \quad (1)$$

Table 1 shows the MCS along with efficiency [10] for various channel quality indication (CQI) indices. In this paper, we use the CQI index as an MCS index for notational convenience. Using the information in the table, we can find what is an ex-

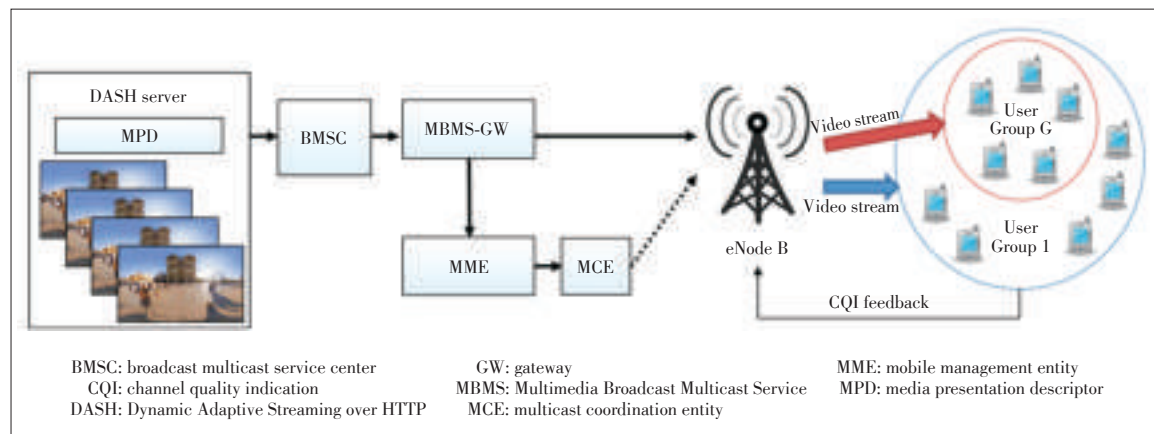


Figure 1. ▶
LTE eMBMS system for DASH multicast.

▼Table 1. CQI-MCS mapping

CQI index	Modulation	Code rate ($\times 1024$)	Efficiency
0		Out of range	
1	QPSK	78	0.1523
2	QPSK	120	0.2344
3	QPSK	193	0.3770
4	QPSK	308	0.6016
5	QPSK	449	0.8770
6	QPSK	602	1.1758
7	16QAM	378	1.4766
8	16QAM	490	1.9141
9	16QAM	616	2.4063
10	64QAM	466	2.7305
11	64QAM	567	3.3223
12	64QAM	666	3.9023
13	64QAM	772	4.5234
14	64QAM	873	5.1152
15	64QAM	948	5.5547

CQI: channel quality indication QAM: quadrature amplitude modulator
MCS: modulation and coding scheme QPSK: quadrature phase-shift keying

pected data rate when we know how many RBs are allocated to the FLUTE sessions.

2.2 File Delivery over Unidirectional Transport

The FLUTE protocol is proposed by IETF [4] to multicast a file over the networks using User Datagram Protocol (UDP)-based protocols with application-layer forward error correction (AL-FEC) being provided for protecting the file from the packet losses. Additional file repair procedures are allowed by the HTTP file repair request. A file repair response message consists of HTTP header and file repair payload. The file repair response message consists of HTTP header which informs that point-to-multipoint repair, instead of point-to-point repair, is used.

2.3 Application-Layer Forward Error Correction

The radio channel conditions vary among all the users receiving the multicast service. Therefore, the block error rate of the users that receive the video service delivered with a single MCS may have a great variance. In order to increase the robustness and reliability of multicast transmissions, FEC redundancy packets are incorporated at the APP-layer [11].

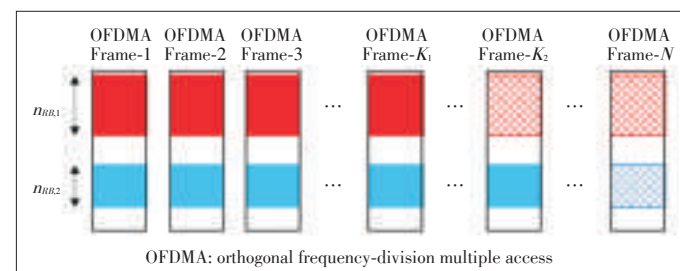
The solution proposed by 3GPP to deliver video streaming over eMBMS uses the FLUTE protocol with UDP transport to send video segments with the corresponding APP-layer FEC over multicast. An FEC block consists of N packets including K source packets and $N-K$ redundancy packets, resulting in the encoding rate K/N . The FEC decoder can ideally recover the original K source packets from any K out of N received

packets with correction capability $t=N-K$. The Reed Solomon (RS) code [12] is a well-known FEC code which operates on non-binary symbols and has the ideal correction capability. However, the RS code has a high decoding complexity because of its non-binary operations, which is not suitable for high-definition (HD) video streaming applications. The Raptor code [13] is a more attractive solution for HD video streaming services due to the flexible parameter selection and linear decoding cost. The correction capability of a Raptor code is $t=N-(1+\epsilon)K$, where ϵ is the reception overhead efficiency. The correction capability of the Raptor code is sub-optimal, however, a standardized Raptor code can closely achieve the ideal correction capability with negligible ϵ . Therefore, it is used in our scheme. In this paper, FEC block size is fixed as N and the number of source blocks K_m is determined to choose appropriate FEC code rates K_m/N for the m -th FLUTE session. Fig. 2 shows the example of an FEC block with two multicasting groups sharing the resource with different FEC code rates. $n_{RB,1}$ and $n_{RB,2}$ are the number of resource blocks allocated in an OFDMA frame for groups 1 and 2 respectively. There are total N -OFDMA frames; K_1 and K_2 of them are video data and others are redundant data.

2.4 Tiled VR Video-Streaming Systems

In a tiled video scheme, the 360-degree video is divided into smaller tiles, which can be encoded independently [14]. There can be multiple copies of the same tile with different representation qualities. These tiles are transmitted through the wireless channel. DASH extends its standard to cover tiled 360-degree videos, i.e., DASH-VR [15]. DASH-VR has included virtual reality video descriptor (VRD) and spatial relationship descriptor (SRD), in addition to media presentation descriptor (MPD), to describe the projection types and spatial relationships among tiles. VRD contains the projection format and orientation information, SRD includes the region-wise quality of rectangular videos within the projected frame, and MPD includes the size of video chunks, locations of video files, and the codec information. As the clients join the multicast system, the MPD, VRD, and SRD are provided to the client, and the client can reconstruct the VR-video by using received tiles based on the received descriptor information.

A DASH multicast system [16] is introduced to efficiently



▲ Figure 2. Application-layer forward error correction (AL-FEC) block, grouping, and resource allocation.

DASH and DASH-VR Video Multicast Systems

PARK Jounsup and HWANG Jenq-Neng

utilize the limited resource and provide better videos to the users. The DASH multicast system allocates multiple copies of the same video with different quality to satisfy more users, but it inevitably generates redundant data that decreases the spectral efficiency. Especially in case of VR videos, most of area are not visible to users. Therefore, more redundant data than conventional video are transmitted if we directly use the DASH multicast for VR-video dissemination. To be more efficient, redundant data should be removed and the tiled video [17] allows to flexibly remove or allocate lower bits to the redundant part of the video. For example, necessary parts of video are transmitted with multiple copies with different quality to satisfy users with good channel quality and the parts with lower probability of view are transmitted just once with single quality to save spectrum.

The most popular and promising technology for controlling regional quality of the video is the use of tiled videos, which has been used for the panoramic interactive video [18], since the interactive video can change its view and users cannot see the whole video at once. VR video is divided into smaller rectangular videos (tiles) and each video is encoded independently using legacy video encoders. Every tile has multiple copies with different encoding rates. Different representations of tiles are transmitted as users' viewport changes and network channel condition changes.

There are simple rate allocation algorithms for tiled-videos, which are Binary, Thumbnail, and Pyramid [19]. Binary allocates the higher representations on the visible tiles, and non-visible tiles have lowest representations to save the bandwidth. It is the most efficient way to allocate the bits, but users can easily see the lowest quality when they move their viewport since the network has latency to respond with viewport changes. Thumbnail allocates the minimum bits of lowest representations for the whole video as the background video, and the remaining bits are allocated for visible tiles with better representations. However, users still can see the lowest quality background video when they move the viewport faster than network latency. The Pyramid algorithm allocates the best representations on visible tiles and gradually lower the representations as the tiles located far from the viewport. However, these rate allocation algorithms are not network-aware and not flexible enough to provide best quality to the users with variable network channel conditions and viewport movement.

Alface et al. [20] propose a rate-selection algorithm to provide the best quality to users with a higher representation in the viewport and lower representations in the other tiles. The algorithm allocates the video rates on the tiles based on utility-over-cost ratios. The utility includes the video bitrates and a probability of view. Since it allocates the best representations for tiles in order to maximize the total utility, as long as there is available resource, the algorithm can achieve the best utility performance compared to other existing solutions.

However, none of the existing algorithms are directly appli-

cable to the multicasting scenario. A new approach to perform the VR video multicasting is proposed in this paper.

3 DASH Multicast

Heuristic algorithms have been introduced to solve the video multicasting problem. These algorithms are differentiated based on types of the video sources. SVC [7] videos are originally used for video multicasting systems because of its layer-dependent characteristics. More enhancement video layers can be combined with the base layer to create higher quality video for the users who have good channel conditions, while the users with poorer channel quality can only receive less enhancement video layers coded with more reliable but less efficient MCS. For the DASH systems, usually videos are encoded as multiple different video rates and stored at the server as small chunks, and they are transmitted to the clients who request the videos. Therefore, different video representations are independent of each other and they can be scheduled independently for multicasting. DASH can also transmit SVC type video sources, but, in this paper, for notational convenience, DASH only denotes multiple video rates without dependencies among representations and SVC denotes the layered video with dependencies among layers.

There have been studies on multicasting videos over wireless networks. Chen et al. [21] consider the fair and optimal resource allocation for LTE multicast. They also consider the unicast for some users with lower SNR without considering FEC for packet protection. Belda et al. [22] introduce a hybrid FLUTE/DASH video delivery system, which can multicast the video through FLUTE sessions and repair requests through the unicast channel to recover lost packets. They fix the FEC code rates of FLUTE sessions and provide the simulation results to show that the hybrid video delivery can improve the video quality compared to the video delivery systems using the unicast only. Nonetheless, in their approach, FEC code rates and resource allocation for multiple FLUTE sessions are not jointly optimized and it may create some repair requests through the unicast channel. Our research starts with the assumption that if we can jointly find the optimal resource allocation and FEC code rate selection, we can transmit the videos without repair requests which call for additional bandwidth. Our goal in this research is to jointly find the optimal resource allocations, the optimal MCS and FEC code rates for multiple FLUTE sessions so as to efficiently serve the DASH clients in the LTE networks without unicast channels for repairing the lost packets.

SVC-based video multicasting algorithms have been previously proposed, e.g., Conservative Multicasting Scheme (CMS) [23], Opportunistic Layered Multicasting (OLM) [24], Multicast Subgrouping for Multi-Layer (MSML) video applications [25], Median Quality Scheme (MQS) [26], Median User Scheme (MUS) [27], etc. These heuristic algorithms describe how to divide multiple users into several multicast groups

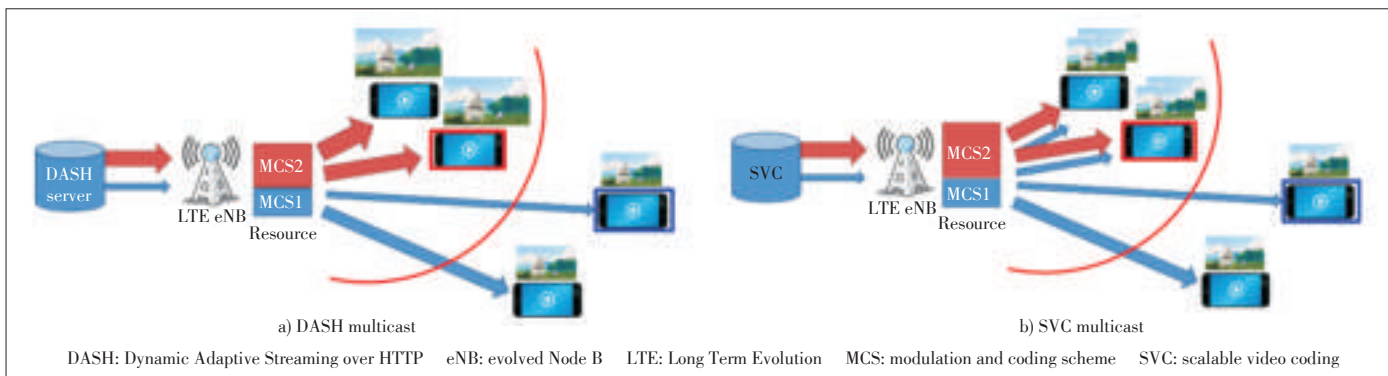
(each group corresponds to one SVC layer) and select proper resource, MCS and FEC code rates, for the different groups based on the channel quality feedbacks from the users within the same group. More specifically, the CMS [23] first allocates each sub-channel to a group of users in the multicast session based on their sub-channel gains, then a greedy algorithm is adopted for resource allocation to achieve proportional fairness among sessions. OLM [24] can choose more aggressive MCS to achieve higher spectral efficiency and protect lost packets by using FEC for each group. On the other hand, MSML [25] utilizes the frequency diversity to achieve better throughput than other schemes. For example, a user with very low average SNR can possibly have some RBs that have high channel gains, and MSML utilizes these RBs to schedule lower video layers. Since MSML can choose the best RBs for each multicast group, it can select more efficient MCS than those chosen by other conservative schemes that are constrained by the users with lowest channel quality, such as the less spectrally efficient CMS scheme. MUS and MQS choose the subgroups based on the number of users and the quality of the channel respectively. Their schemes can achieve better spectral efficiency than CMS, but less than those achieved by OLM and MSML.

SVC multicasting and DASH multicasting are differed by users' video receiving methods. Fig. 3 shows the difference between SVC and DASH multicasting systems. The users receive

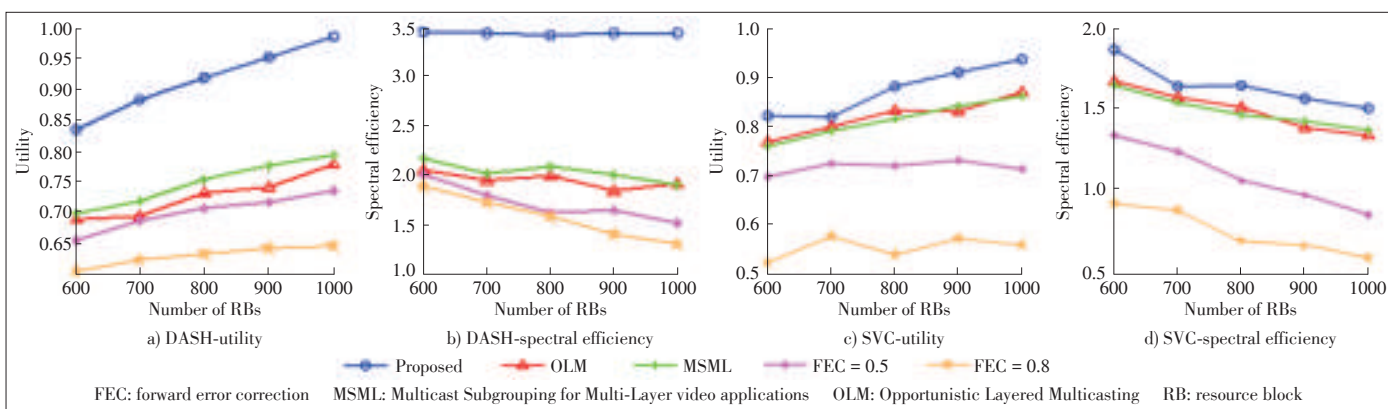
ing SVC type of videos should receive base layer as mandatory and can improve the quality of the video by receiving multiple enhancement layers. Therefore, the users need to subscribe multiple multicast sessions. However, the sources of DASH multicasting systems are independent videos. Therefore, the users only need to select one multicast session and receive single video representation to see the video. Since DASH multicasting inevitably generates some redundant data, there could be waste on resource usage. However, Park et al. [8] propose the optimal DASH multicast (ODM) algorithms and show that optimal resource allocation, video rate selection and user grouping can take advantage of multicasting. Therefore, DASH multicasting methods can achieve better utility and provide better video quality to the users. Fig. 4 shows the utility performance (a, c), and the spectral efficiency (b, d) when DASH and SVC types of video sources are used respectively. It can be found that the proposed ODM achieves the best utility and spectral efficiency performance, compared to OLM, MSML, and fixed FEC code rate methods.

4 DASH-VR Multicast

VR is getting more popular these days, and more people can enjoy more realistic experiences with VR systems [28]. Moreover, it allows people to look around the virtual world and feel



▲ Figure 3. DASH multicast and SVC multicast.



▲ Figure 4. Performance comparisons.

DASH and DASH-VR Video Multicast Systems

PARK Jounsup and HWANG Jenq-Neng

like they are actually in the environment. VR gaming can provide a more exciting experience to gamers. However, it is a more challenging task to make users satisfied with the quality of VR videos, because VR videos need much higher resolution than conventional videos. Users cannot see the whole video at the same time, they can only focus on the area that they want to see and the area is usually only 20% of whole video [29]. Therefore, 4–6 times more resolution is required for VR videos to provide the same experience as conventional videos. On the other hand, this fact allows the saving of bandwidth, because 80% of the video is unseen by the user at a given time. In an ideal case, we could save 80% of the bandwidth; but in practice, we still need to transmit redundant areas of the video because it is difficult to predict how a user's viewport will change.

The original DASH system allows the clients to do the video rate adaptation, but it is difficult to do the individual rate adaptation in multicast systems because the users grouped into the same group share the same spectrum resource and they receive the same video rate even though they have different channel quality. Rather than doing the individual rate adaptation, the server can adaptively choose the video rates of the tiles to maximize the expected total utility of the users. Feedback information of users' viewports can be used to decide which tiles should have better video rates to satisfy more users. Another way to decide which tiles are more important than others is analyzing the video at the server side. The server can analyze the video contents first and then decide which part may have higher interest from users. Saliency [30] of the video is one of the useful indicator to find the important area of the video. Therefore, we can give more bits to the area that has higher saliency to satisfy more users. Saliency detection algorithms usually find the areas that have high contrast or active movement in the video [31], because those areas usually have richer or more appealing information such as important texture or moving objects. By using the saliency information, the server can allocate more bits to the areas that have higher saliency scores to make them clearer. There are many video saliency detection algorithms to find which parts are more important and interesting to users.

There are two possible ways to do the VR video multicasting. The multicasting is featured by grouping the users to share the same resource. First, the users with the same view can be grouped into a multicast group. The number of multicast groups is the same as the number of views [32]. It can save some resource by sharing the same view with many users, but cannot take advantage of using a multicast scheme when users have different channel quality. All the multicasting groups will suffer with the user with very bad channel quality. Moreover, all the users eventually need to receive all the tiles because there is latency between the server and the client which is difficult to overcome. Second, users can be grouped with their channel quality. This grouping strategy helps to select more ef-

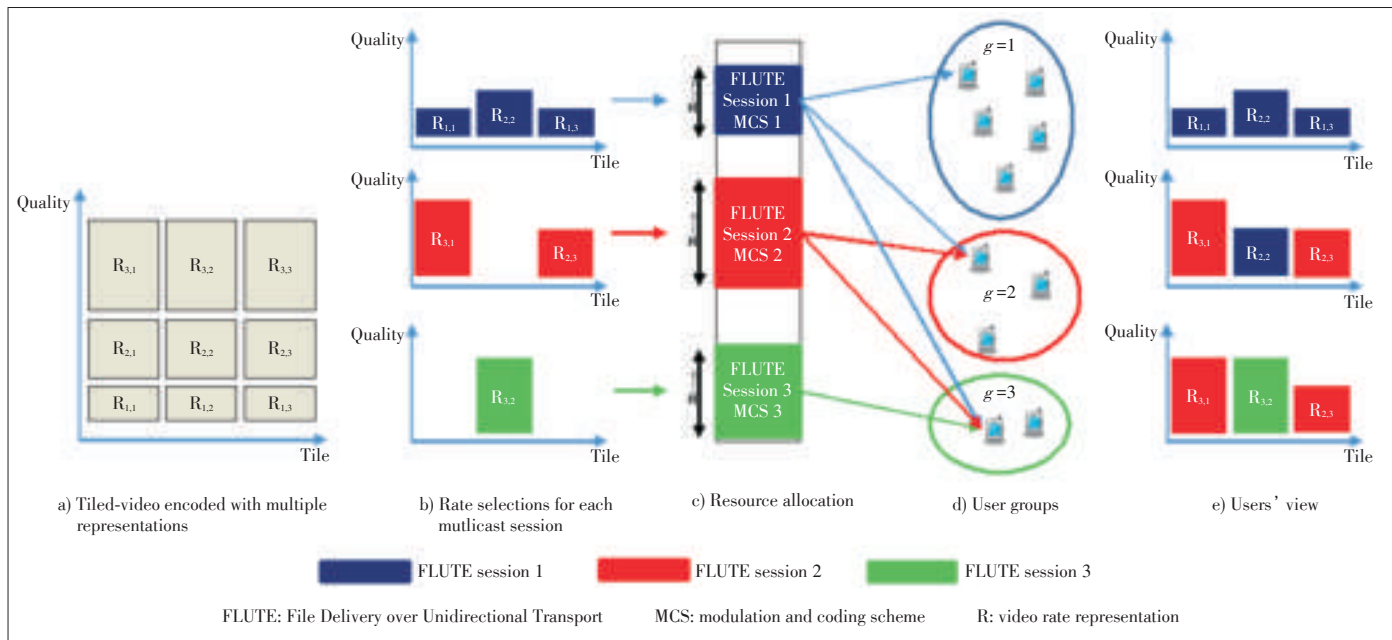
ficient MCS and AL-FEC code rate to allocate better video. As the number of users who could join the group with better video increases, total utility is also improved. Therefore, we have designed the multicast systems based on the second scheme that groups the users with their channel quality.

The clients in a DASH-VR multicast system request video chunks to the server based on MPD, SRD, and VRD information. The DASH server starts to deliver the tiled-video data. BM-SC creates the multiple video sessions that will deliver the tiled-videos with multiple video representations. BM-SC is also responsible for adding AL-FEC redundant blocks for the lost packet recovery. Multiple video multicast sessions are created to deliver multiple VR videos and multiple video representations to different user groups. A video session can contain a single tile or multiple tiles. MBMS-GW passes the video data to the eNBs and MCE allocates the resource for video sessions and assigns the proper MCS for the resource. Users participate on video sessions and the users who can participate on the multiple video sessions have chances to choose better representations. The eNB receives the CQI feedback information from the UEs to help allocating RB and choosing AL-FEC code rate and MCS for the multicasting sessions.

The difference between a multicast session and a multicast group is that a multicast session denotes a video session that uses the radio resource controlled by the MCE, while a multicasting group denotes a set of users grouped by their channel conditions and subscribing the same video. Note that users can subscribe multiple multicast sessions at the same time, therefore, the number of multicast sessions and the number of multicast groups are not necessarily the same. The multicast groups are arranged based on the channel condition and the user groups with high channel quality can take advantage of subscribing multiple multicast sessions.

We can consider two different ways to create video multicast sessions. One is the per-tile multicasting (PTM) that considers the tiles as independent videos, where each tile has its own resource and every UE subscribes all necessary sessions to regenerate the VR video. It needs to create multiple multicasting sessions as many as the number of tiles times the number of representations for a single VR-video content. All possible video representations of all tiles are available for the users based on their channel quality, and the users regenerate the VR-video with the tiles that have the best quality they can decode. For example, if there are T tiles and M representations for each tile, total $T \times M$ multicast sessions can be created. MCS, AL-FEC, and resources for all multicast sessions have to be determined to maximize the total utility. Its search space to find optimal solution is M^T . Each user selects one representation for one tile and subscribe T multicast sessions to regenerate the VR-video. It generates too much control signal and the complexity of the solution increases with the number of multicast sessions.

The other is the multi-session multicasting (MSM), which



▲ Figure 5. Multi-session multicast.

creates the same number of multicast sessions as the number of user groups. Each multicast session includes multiple tiles with different quality. Fig. 5 shows an example of MSM system with 3 groups and 3 multicast sessions. Fig. 5a shows the tiled-video encoded with multiple representations. Every tile has multiple copies with different representations (qualities) and they are generated by legacy video encoder with different quantization parameters (QP). Higher representations indicate better qualities, and they need more bandwidth to be transmitted. Fig. 5b shows the rate selection results for multiple multicast sessions. The first multicast session has all the tiles with lower representations to guarantee all the users requesting the VR video to receive at least lower quality video. The second and third multicast sessions do not need to have all the tiles. They allocate higher representations to improve the quality of the tiles for the users with better channel quality. Therefore, they are allocated on the wireless resource with more efficient MCS and AL-FEC code rates (Fig. 5c). The users can subscribe multiple multicast sessions at the same time, but their channel quality should be good enough to decode the data packets assigned with certain MCS and AL-FEC. In Fig. 5e, the user group 1 can only receive the data in the multicast session 1, while user group 2 can receive multicast sessions 1 and 2. The user group 3 can receive all three multicast sessions. Therefore, the user groups 2 and 3 have chances to choose better representations from multiple representations they can receive.

Since MSM's multicast session includes the multiple tiles, it creates less multicast sessions than PTM. Another advantage of MSM is that it can use existing rate selection algorithms introduced in Section 2.4. The rate selection algorithm [21] can work to allocate the tiles of different representations with the

bit rate constraint of each multicast session. The bit rate constraints of multicast sessions are determined by the resource allocated on the multicast sessions, MCS, and AL-FEC code rates.

5 Conclusions

In this paper we presented an overview of several wireless video multicasting systems and algorithms. SVC based video multicasting systems are introduced first, but DASH is getting more popular. DASH multicasting can take advantage of allocating multiple copies with different quality to allow users to select appropriate video quality. It improves the utility performance of the systems. The wireless multicasting systems, such as LTE eMBMS, can deliver the VR video more efficiently combined with tiled-video rate adaptation. We propose MSM system to allocate the multiple tiles on a single multicast session and generates the multiple multicast session to provide a set of tiles with different representations. The proposed tiled video multicasting scheme uses the limited wireless resource more efficiently than other VR multicasting schemes. The optimal resource allocation, MCS and AL-FEC code rate selection for multicast sessions to improve DASH-VR multicasting systems are the problems that we have to do as a future work.

References

- [1] Cisco, "Cisco visual networking index: forecast and methodology, 2014–2019," Cisco white paper, May 2015.
- [2] *Information Technology—Dynamic Adaptive Streaming over HTTP (DASH)—Part 1: Media Presentation Description and Segment Formats*, ISO/IEC 23009-1, Apr. 2012.

DASH and DASH-VR Video Multicast Systems

PARK Jounsup and HWANG Jenq-Neng

- [3] *Transparent End-To-End Packet Switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)*, 3GPP TS 26.247 Release 11, 2012.
- [4] *FLUTE—File Delivery over Unidirectional Transport*, IETF RFC 3926, Oct. 2004.
- [5] D. Gozálviz, D. Gómez-Barquero, T. Stockhammer, and M. Luby, "AL-FEC for improved mobile reception of MPEG-2 DVB-T transport streams," *International Journal of Digital Multimedia Broadcasting*, vol. 2009, Article ID 614178, 2009. doi: 10.1155/2009/614178.
- [6] B. Wang, J. Kurose, P. Shenoy, and D. Towsley, "Multimedia streaming via TCP," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 4, no. 2, pp. 1–22, 2008. doi: 10.1145/1352012.1352020.
- [7] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007. doi: 10.1109/TCSVT.2007.905532.
- [8] J. Park, J.-N. Hwang, Q. Li, Y. Xu, and W. Huang, "Optimal DASH-multicasting over LTE," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4487–500, May 2018. doi: 10.1109/TVT.2018.2789899
- [9] J. Liu, C. Wei, C. Zhigang, and K. B. Letaief, "Dynamic power and sub-carrier allocation for OFDMA-based wireless multicast systems," in *IEEE International Conference on Communications*, Beijing, China, May 2008, pp. 2607–2611. doi: 10.1109/ICC.2008.494.
- [10] M. T. Kawser, I. B. H. Nafiz, N. H. Md, M. Shah Alam, and M. Musfiqur Rahman. "Downlink SNR to CQI mapping for different multiple antenna techniques in LTE," *International Journal of Information and Electronics Engineering*, vol. 2, no. 5, p. 757, Sept. 2012. doi: 10.7763/IJIEE.2012.V2.201.
- [11] F. Alejandro, L. M. Carlos, B. Luis, et al., "Analysis of the impact of FEC techniques on a multicast video streaming service over LTE," in *European Conference on Networks and Communications (EuCNC)*, Paris, France, Aug. 2015, pp. 219–223. doi: 10.1109/EuCNC.2015.7194072.
- [12] C. Lentisco, B. Luis, F. Alejandro, et al., "A model to evaluate MBSFN and AL-FEC techniques in a multicast video streaming service," in *IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Larnaca, Cyprus, Oct. 2014, pp. 691–696. doi: 10.1109/WiMOB.2014.6962246.
- [13] *RaptorQ Forward Error Correction Scheme for Object Delivery*, IETF RFC 6330, Aug. 2011.
- [14] K. Misra, S. Andrew, H. Michael, et al., "An overview of tiles in HEVC," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 969–977, Jun. 2013. doi: 10.1109/JSTSP.2013.2271451.
- [15] L. D'Acunto, J. van den Berg, E. Thomas, and O. Niamut. "Using MPEG DASH SRD for zoomable and navigable video," in *Proc. 7th International Conference on Multimedia Systems*, Klagenfurt, Austria, May 2016. doi: 10.1145/2910017.2910634.
- [16] D. Lecompte and G. Frédéric, "Evolved multimedia broadcast/multicast service (eMBMS) in LTE-advanced: overview and Rel-11 enhancements," *IEEE Communications Magazine*, vol. 50, no. 11, pp. 68–74, Nov. 2012. doi: 10.1109/MCOM.2012.6353684.
- [17] J. Le Feuvre and C. Concolato, "Tiled-based adaptive streaming using MPEG-DASH," in *Proc. 7th International Conference on Multimedia Systems*, Klagenfurt, Austria, May 2016. doi: 10.1145/2910017.2910641.
- [18] H. Kimata, S. Shimizu, Y. Kunita, M. Isogai, and Y. Ohtani, "Panorama video coding for user-driven interactive video application," *IEEE 13th International Symposium on Consumer Electronics*, Kyoto, Japan, May 2009, pp. 112–114. doi: 10.1109/ISCE.2009.5157036.
- [19] V. R. Gaddam, M. Riegler, R. Eg, C. Griwodz, and P. Halvorsen, "Tiling in interactive panoramic video: approaches and evaluation," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1819–1831, Sept. 2016. doi: 10.1109/TMM.2016.2586304.
- [20] R. Alfacc, J. F. Macq, and N. Verzijp, "Interactive omnidirectional video delivery: a bandwidth-effective approach," *Bell Labs Technical Journal*, vol. 16, no. 4, pp. 135–147. doi: 10.1002/bltj.20538.
- [21] J. Chen, M. Chiang, J. Erman, et al., "Fair and optimal resource allocation for LTE multicast (eMBMS): group partitioning and dynamics," in *IEEE Conference on Computer Communications (INFOCOM)*, Hong Kong, China, Aug. 2015, pp. 1266–1274. doi: 10.1109/INFOCOM.2015.7218502.
- [22] R. Belda, I. de Fez, F. Fraile, P. Arce, and J. Guerri, "Hybrid FLUTE/DASH video delivery over mobile wireless networks," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 11, pp. 1070–1082, Feb. 2014. doi: 10.1002/ett.2804.
- [23] B. Kagan, M. Q. Wu, H. Liu, and S. Mathur, "Adaptive resource allocation in multicast OFDMA systems," in *Proc. IEEE WCNC*, Sydney, Australia, Apr. 2010. doi: 10.1109/WCNC.2010.5506213.
- [24] C. Huang, S. Huang, P. Wu, S. Lin, and J. Hwang, "OLM: opportunistic layered multicasting for scalable IPTV over mobile WiMAX," *IEEE Transactions on Mobile Computing*, vol. 11, no. 3, pp. 453–463, Feb. 2012. doi: 10.1109/TMC.2011.34.
- [25] M. Condoluci, G. Araniti, A. Molinaro, and A. Iera, "Multicast resource allocation enhanced by channel state feedbacks for multiple scalable video coding streams in LTE networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 2907–2921, May 2016. doi: 10.1109/TVT.2015.2449080.
- [26] A. Alexiou, C. Bouras, V. Kokkinos, A. Papazois, and G. Tschritzis, "Efficient MCS selection for MBSFN transmissions over LTE networks," in *Wireless Days (WD)*, Venice, Italy, Oct. 2010, pp. 1–5. doi: 10.1109/WD.2010.5657749.
- [27] T. Low, M. Pun, Y. Hong, and C. Kuo, "Optimized opportunistic multicast scheduling (OMS) over wireless cellular networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 2, pp. 791–801, Feb. 2010. doi: 10.1109/TWC.2010.02.090387.
- [28] S. Chen, "Quicktime VR: an image-based approach to virtual environment navigation," in *Proc. 22nd Annual Conference on Computer Graphics and Interactive Techniques*, Los Angeles, USA, 1995, pp. 29–38. doi: 10.1145/218380.218395.
- [29] X. Corbillon, S. Gwendal, A. Devlic, and J. Chakareski, "Viewport-adaptive navigable 360-degree video delivery," *IEEE International Conference on Communications*, Paris, France, Oct. 2010, pp. 1–7. doi: 10.1109/ICC.2010.7996611.
- [30] X. Hou, J. Harel, and C. Koch, "Image signature: highlighting sparse salient regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, Jul. 2011. doi: 10.1109/TPAMI.2011.146.
- [31] W. Kim and J. J. Han, "Video saliency detection using contrast of spatiotemporal directional coherence," *IEEE Signal Processing Letters*, vol. 21, no. 10, pp. 1250–1254, Jun. 2014. doi: 10.1109/LSP.2014.2332213.
- [32] J. T. Lee, D. N. Yang, Y. C. Chen, and W. Liao, "Efficient multi-view 3D video multicast with depth-image-based rendering in lte-advanced networks with carrier aggregation," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 85–98, May 2017. doi: 10.1109/TMC.2017.2707416.

Manuscript received: 2018-03-29

Biographies

PARK Jounsup (jsup517@uw.edu) received the B.S. and M.S. degrees in electrical engineering from Korea University, Seoul, Korea in 2006 and 2008, respectively, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, USA in 2018. He is currently a postdoctoral research fellow with the Coordinated Science Lab. (CSL), University of Illinois at Urbana-Champaign, USA. From 2008 to 2012, he was with Cooperate Research Institute of Samsung Electro-Mechanics, Inc., Suwon, Korea. His research interests include multimedia networking, wireless communication, and AR/VR quality of experience.

HWANG Jenq-Neng (hwang@uw.edu) received his Ph.D. from the University of Southern California, USA in 1988. In 1989, he joined the Department of Electrical Engineering of the University of Washington, USA, becoming a full professor in 1999. He served as the Associate Chair for Research from 2003–2005 and again from 2011–2015. HWANG is currently the Associate Chair for Global Affairs and International Development. He has written more than 300 journal articles, conference papers and book chapters in the areas of multimedia signal processing and multimedia system integration and networking. He is the author of the textbook "Multimedia Networking: from Theory to Practice." HWANG has a close working relationship with the industry on multimedia signal processing and multimedia networking.

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

(School of Computing, Electronics and Mathematics, University of Plymouth, Plymouth, PL4 8AA, United Kingdom)

Abstract

Internet of Things (IoT) applications such as environmental monitoring, healthcare, surveillance, event recognition and traffic control are amongst the most commonly deployed applications over the Internet. These applications involve multimedia content that has to be collected, processed and delivered appropriately over the Internet for further processing by human or machines. These applications come with their own set of requirements such as quality, computational power and bandwidth. It is, therefore, vital to minimize power consumption and bandwidth usage in IoT devices without compromising the quality of multimedia delivery. Since the delivery of the multimedia can be destined to a machine or human, it is important to distinguish multimedia quality between the two. Quality of Experience (QoE) for video services involves human visual system, but what will involve a machine or process? To distinguish between the two, this paper defines a new concept of Acceptable Quality of Things (AQoT) which involves IoT devices and their applications. AQoT aims at minimizing bandwidth without compromising quality in IoT devices. Experimental results based on human detection and license number plate detection use cases have demonstrated that the AQoT concept can significantly reduce bandwidth usage.

Keywords

QoE; QoT; IoT; bandwidth; video streaming

1 Introduction

The global Internet has been expanding at an unprecedented speed. It is now connecting over 3.7 billion people [1] and around 22 billion “smart objects” via the Internet of Things (IoT) [2]. According to the latest forecast from the Cisco Visual Networking Index [3], IP video traffic will account for about 82 percent of all consumer Internet traffic by 2021, increase threefold from 2016 to 2021. Within this five-year period, the most fast-growing IP video traffic is expected to be Internet video (such as video services provided by YouTube and Netflix) with an estimated growth of fourfold from 2016 to 2021; Internet video surveillance traffic 7-fold; live video 15-fold; gaming traffic nearly tenfold; and virtual reality and augmented reality traffic 20-fold. In addition to the above consumer Internet video traffic, machine-to-machine (M2M) communications and IoT services for multimedia applications further increase the video traffic on the Internet.

The ever-growing Internet video traffic has posed a real challenge to the healthy operation of the Internet. The Internet is feeling the strain, far beyond the imagination of its original developers in 1970s and 1980s. Any technologies or approaches

to reducing the traffic for a service to be delivered over the Internet without compromising the user’s experience for the service would be welcomed by all parties involved. For consumer-based IP video traffic, such as live video streaming, Internet TV and video gaming, keeping the customer happy and reducing the churn rate is key to the success of launching new service or maintaining an existing service for a service provider. In general, increasing video bit rate for a video streaming service will have a positive impact on end-user perceived video quality if there are no constraints on network bandwidth. However, in some cases, increasing video bitrate further does not result in a clear increase in perceived video quality or Quality of Experience (QoE). In some applications, it would be too costly to always transfer the maximum video bit rate for a multimedia service. In our previous work [4], we have demonstrated the gain in utilizing “Acceptable QoE” (i.e. Mean Opinion Score (MOS) over 3.5) in LTE downlink resource scheduling for VoIP services to improve the cell capacity. In this paper, we expand the concept to the domain of multimedia IoT applications. We define the term of ‘Quality of Things’ (QoT) [5] to refer to the quality of fulfilling an IoT task/process with multimedia IoT services and demonstrate how a similar concept, named as ‘Acceptable QoT (AQoT)’, could be applied in IoT

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

applications to reduce video traffic without compromising the quality of delivered multimedia services to a machine or a ‘thing’ over the Internet.

The remaining of the paper is organized as follows. In Section 2, related work based on the quality of multimedia in IoT is discussed. Section 3 provides QoE and QoT definitions together with QoT scheme and management. Experimental setup, results and evaluation are presented in Sections 4 and 5, respectively. Section 6 concludes the paper.

2 Related Work

Multimedia communications on the Internet of Things research has received wide attention in the literature in recent years. However, the growth and popularity of multimedia data pose new challenges to the IoT devices. Multimedia IoT (MIoT) devices consume more bandwidth and require high processing power to transfer the acquired multimedia data. Multimedia applications such as face or object detection, surveillance system and event detection are captured by the MIoT devices and then the video sequences are sent to edge nodes or cloud for further analysis depending on their tasks.

Research in edge computing framework for cooperative video analysis [6] proposed a cooperative framework for delay-sensitive multimedia IoT tasks, where high-quality video streams acquired by the camera node, are sent to the edge node to process sub-tasks e.g., feature detection and extraction and send the processed results to cloud for further video analysis if necessary. In [7], an architecture was designed to run in the hostile environment, where captured images by the camera node will be sent to the cloudlet over high-speed bandwidth connection. If the cloudlet lacks the necessary data from the database, it will send some of the tasks to the remote cloud for further processing. A two-stage procedure was implemented in [8], which included face detection, extraction and face matching. The face detection and extraction tasks are performed in a cloudlet node while the complex face matching is performed on a remote cloud node. One possible limitation of these implementations is such that the bandwidth requirement is still considerably high due to sending high-quality video and images to the edge and cloud nodes for processing.

In [9], image and video frames were divided into important premium blocks and unimportant regular blocks to save energy on IoT devices and provide high QoE to end users. A dynamic surveillance video stream was processed at fog node [10]; instead of sending a whole video frame, a sub-part of the original video frame was sent to the fog node to meet the real-time processing requirement. However, these approaches would be difficult to be used in surveillance systems since all the video frames need to be sent to the cloud for further investigations, and this requires high network bandwidth.

Authors in [11] introduced a concept of Quality of Contents (QoC) and proposed QoC based video encoding rate allocation

scheme in mobile surveillance networks. This scheme allocates different data rate constraints to each camera node based on corresponding information and delivers video tasks to the remote cloud. Although QoC could save some bandwidth, transmitting video sequences directly to the cloud would lead to congestions and delays. Edge nodes could, therefore, be used to ease congestion delays to the cloud.

In [12], an intelligent surveillance video coding technique was proposed. A background model was used to extract foreground objects and encoded in high quality while background frames were encoded with low quality. Although this approach could save some bandwidth, processing video locally at the camera node would cause computational delay due to limited network bandwidth.

A fuzzy-based approach that considers some internal and external parameters in order to define the sensing, coding and transmission configuration of visual sensors was proposed in [13]. In [14], authors defined MIoT in 3 scenarios based on the use of multimedia content such as multimedia as IoT input and output, multimedia as IoT input, and multimedia as IoT output. The paper proposed a QoE layered model for MIoT applications, presented a use case related to the remote monitoring driving practices and conducted subjective assessments to measure the QoE. However, current QoE concepts and models might not be applicable in IoT M2M concept since no humans are involved in the cycle.

In this paper, a concept of AQoT with two use cases is proposed. The goal of AQoT is to meet the acceptable quality to fulfil or complete an IoT task “successfully”. For the meaning of “successful” completion of a task, it might be a 100% detection accuracy or might be 95% accuracy (or other values) depending on applications/scenarios. By meeting the acceptable quality, the system will avoid over-provisioning of multimedia quality. Hence, it will use less bandwidth without compromising its quality for other IoT devices and applications. A similar concept was used in [4] and was termed as an acceptable QoE. It aimed at increasing the number of users in a single eNodeB of an LTE cellular network for VoIP applications.

The approach of this paper is to process MIoT tasks such as human detection, face detection and license number plate recognition at the edge nodes. If further analysis is needed, results of these tasks or some tasks will be delivered to the cloud nodes.

3 Quality, QoE and QoT

Multimedia services over the Internet can be generally categorized as human-to-human (e.g. VoIP and video conferencing services), human-to-machine (e.g. speech recognition and video recording/uploading), machine-to-machine (e.g. surveillance camera/video to a server), and machine-to-human (e.g. Internet video streaming) applications (**Fig. 1**). If a recipient involves human as depicted in black lines in the figure, the con-

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

cept of Quality of Experience (QoE) applies. Otherwise, if a recipient is a machine (including devices/things, data processes, as depicted in red lines in Fig. 1), we will utilize the Quality of Things (QoT) concept instead.

In the QoT framework for acceptable QoT, a MIoT device such as a camera is used to monitor the surveillance area and sends the acceptable quality video streams to the near edge node for further processing depending on the task. If a task is not computational intensive (light task) e.g., license plate detection or speed detection, the edge node will complete the task and send the detection results to the cloud node for post-processing and/or general management. If the task is complex (heavy task) e.g., face detection or recognition, the edge node will share or distribute the task to other neighbouring edge nodes. If all neighbouring edge nodes are busy, the task will be offloaded to the cloud node.

These processes and communications are modelled in a layered architecture consisting of things, edge and cloud layers (Fig. 2). Servers in the distributed cloud can host several IoT applications such as human detection, face recognition and license number plate recognition applications. Edge nodes can be grouped into domains associated with a set cloud nodes with particular applications or close proximity. IoT devices can also be grouped into domains with similar tasks such as temperature sensors and surveillance cameras. IoT nodes can pro-

cess tasks locally and then transmit them to edge or cloud nodes. Edge nodes can process tasks from IoT nodes or share them with other edge nodes in the same domain. Edge nodes can also forward tasks from IoT devices to cloud nodes for further processing and analysis. Cloud nodes can process tasks from IoT devices or edge nodes and send feedback to IoT devices or edge nodes. Cloud nodes can also send feedback to a human if there is a requirement.

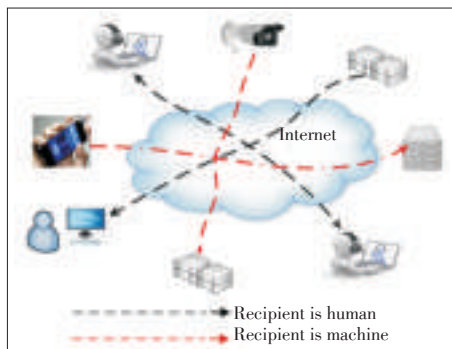
IoT applications will be residing in edge node e_j and cloud node c_k for $j=1 \dots J$ and $k=1 \dots K$. The aim of the QoT is to fulfil the minimum requirements for these applications in order to effectively execute task τ_{ij} or τ_{ik} coming from IoT device i to edge node j or cloud node k without compromising the performance of the IoT system. Assuming that overall resources are the same such as computing and network resources, the optimization of an MIoT scheme will be to maximize the number of IoT devices that can be served, subject to acceptable QoT.

Fig. 3 depicts the scenario of a surveillance IoT system consisting of a surveillance camera as an IoT node, edge node, cloud node and an IoT application which can reside either in edge or cloud nodes.

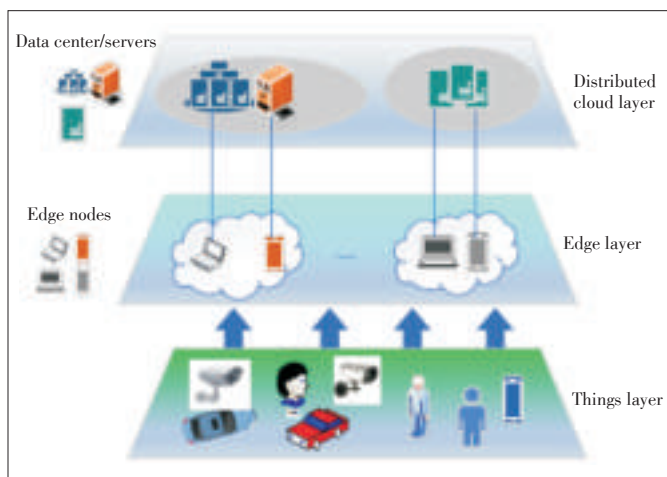
If τ_{ij}^{\min} and τ_{ij}^{\max} are minimum and maximum requirements of a task from IoT node i to edge node j , respectively, then $aQoT_{ij}$ is defined as an acceptable Quality of Things if τ_{ij}^{\min} is achieved without compromising the performance of an IoT system.

If τ_{ik}^{\min} and τ_{ik}^{\max} are minimum and maximum requirements of a task from IoT node i to cloud node k , respectively, then $aQoT_{ik}$ is defined as an acceptable Quality of Things if τ_{ik}^{\min} is achieved without compromising the performance of an IoT system.

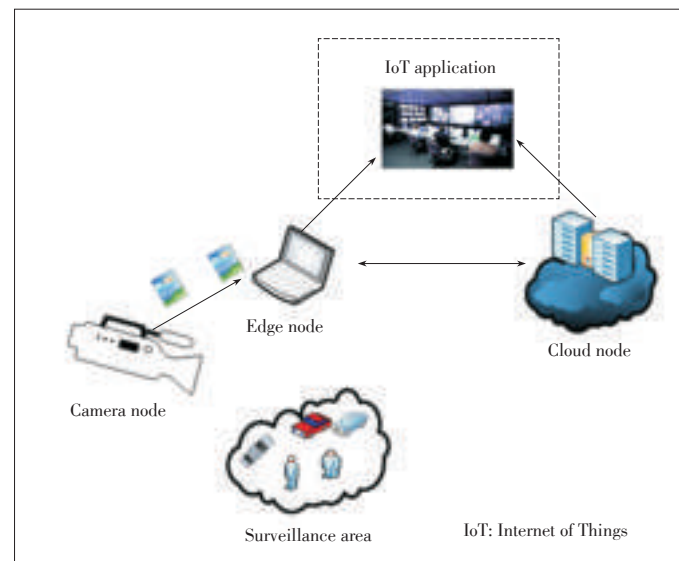
For MIoT, if a task is license number plate recognition or human detection, the requirement will be an image or video quality and the performance will be the recognition accuracy or de-



◀Figure 1. Conceptual diagram for internet multimedia services.



▲Figure 2. Things, edge and cloud layers.



▲Figure 3. An IoT scenario for surveillance.

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

tection accuracy. The minimum requirement for MIoT is taken as the minimum quality of an image or video parameter that an IoT system will still be able to effectively execute a task without compromising its performance. The minimum quality of a video task τ_{ij} is the minimum bitrate $b_{\tau_{ij}}^{\min}$ or quantization parameter $q_{\tau_{ij}}^{\min}$ that an IoT system will still be able to effectively execute a task without compromising its performance.

For the license number plate recognition task, the recognition accuracy $r_{\tau_{ij}}$ will either be 1 or 0. 1 denotes that the license number plate is accurately recognized while 0 shows that the license number plate is falsely recognized. The purpose of an acceptable QoT concept for the license number plate recognition task is to achieve $r_{\tau_{ij}}=1$ or $r_{\tau_{ik}}=1$ at $q_{\tau_{ij}}^{\min}$ or $q_{\tau_{ik}}^{\min}$.

For the human detection task, the detection accuracy $d_{\tau_{ij}}$ is defined as the ratio of a number of recognized humans to the total number of humans in a frame. If $d_{\tau_{ij}}=1$, this implies that all humans were accurately detected and if $d_{\tau_{ij}}=0$, this implies that all humans were falsely detected, otherwise, $d_{\tau_{ij}}=x$, for $0 < x < 1$. The purpose of an acceptable QoT concept for the human detection task is to achieve $0.9 \leq r_{\tau_{ij}} \leq 1$ or $0.9 \leq r_{\tau_{ik}} \leq 1$ at $b_{\tau_{ij}}^{\min}$ or $b_{\tau_{ik}}^{\min}$.

4 Experimental Setup

As a proof of concept for QoT, two use cases were considered, human detection and license number plate recognition.

The platform for development and experiment was conducted in Ubuntu 16.04 Xenial. OpenCV 3.3.0 [15] on Python was used as an IoT application for coding human detection algorithm. Histogram Oriented Gradients (HoG) [16] was applied to detect humans in video frames. HoG is a feature descriptor which uses a global feature to describe a person. This approach trained a Support Vector Machine (SVM) for classification to recognize HoG descriptors of people, which is an effective human detection method.

In the human detection use case, one video sequence was used to demonstrate the concept of the acceptable QoT at which the detection task at minimum bitrate could still be able to accurately detect humans. The video sequence information is given in **Table 1**. A human video sequence was encoded with FFmpeg version 2.8.11 as H.264/MPEG-4 AVC at a bitrate from 800 kbit/s to 5 kbit/s and the human detection algorithm was deployed at each bitrate. There is a varying number of people in each frame as humans enter and leave a scene. The maximum number of humans in some frames is 5 and the minimum is 1. The snapshots for human detection frame and li-

▼ **Table 1. Human detection video sequence**

Video sequence	Resolution (pixels)	Bitrate (kbit/s)	Frame-rate (fps)
Human detection	768×576	5–800	25

cence number plate are illustrated in **Figs. 4a** and **4b**, respectively.

For the license number plate recognition use case, Open Automatic License Plate Recognition (OpenALPR) [17] was used to recognise license number plate. In this use case, single number plate image (Car1) in Joint Photographic Experts Group (JPEG) format was used.

The image information is in **Table 2**. The Car1 image was taken close to the number plate in bright lighting conditions. The quality compression of Car1 video sequence was ranging from 90% to 1%. ImageMagick 6.8.9-9 was used to compress the JPEG images into different compression levels. The snapshot for Car1 image thumbnails is depicted in Fig. 4b.

5 Results and Discussions

For human detection, the detection accuracy is used as the performance metric to demonstrate the concept of an acceptable QoT. Detection accuracy is a ratio of accurately detected number of humans to the total number of humans in a frame. The detection ratio between 0.9 and 1 is considered accurate [18]. Three human detection video frames are selected for demonstration (**Fig. 5**). Frame 1 has two people very close to each other from the camera point of view; frame 44 has two people who are far from each other and frame 102 has three people who are far from each other.

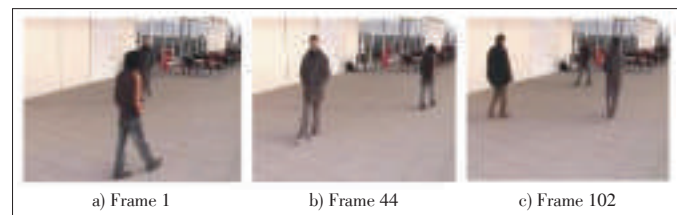
Fig. 6 depicts the human detection accuracy against the bitrate for frames 1, 44 and 102 of the 10 seconds video se-



▲ **Figure 4. Sample video sequences.**

▼ **Table 2. License number plate image**

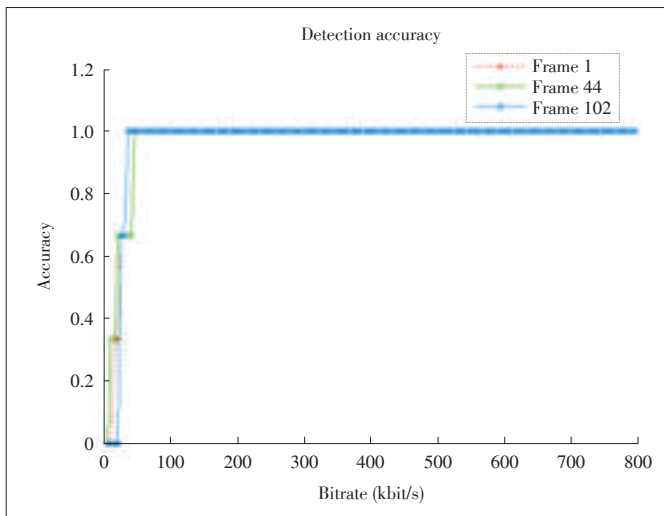
Video sequence	Resolution (pixels)	Quality range (%)
Car1	640×480	1–90



▲ **Figure 5. Human detection video frames.**

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun



▲ Figure 6. Human detection accuracy for frames 1, 44 and 102.

quence. It can be observed that for frame number 1, the human detection rate of 1 ranges from 800 kbit/s to 35 kbit/s. If the bitrate is below 35 kbit/s, the detection accuracy is significantly reduced. The detection accuracy drops to 0 at 10 kbit/s.

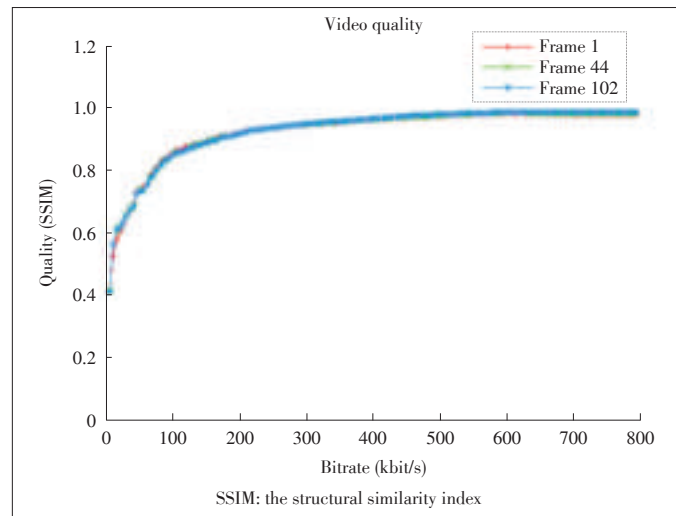
For frame number 44, it can be observed that the human detection rate of 1 ranges from 800 kbit/s to 45 kbit/s. If the bitrate is below 45 kbit/s, the detection accuracy is significantly reduced. The detection accuracy drops to 0 at 5 kbit/s.

For frame number 102, it can be observed that the human detection rate of 1 ranges from 800 kbit/s to 30 kbit/s. The bitrate below 30 kbit/s, the detection accuracy is significantly reduced. The detection accuracy drops to 0 at 20 kbit/s.

Since detection accuracy of at least 0.9 is considered as accurate, instead of transmitting the original video sequence of 800 kbit/s over the Internet to another edge node or cloud node, bitrates ranging from 50–70 kbit/s of the same video sequence will be used for transmission. This range of bitrate is considered as minimal at which the IoT system could still be able to perform human detection without negatively affecting the detection accuracy. 50–70 kbit/s is what considered as an acceptable QoT. This has resulted in a saving of more than 10 times of the original bandwidth requirement.

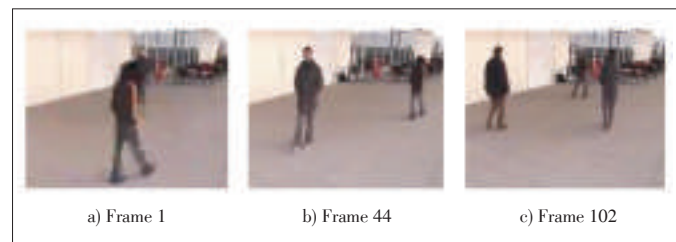
The structural similarity (SSIM) index [19] is used to measure the similarity in perceptual quality between the original images and the degraded ones. The SSIM index values are depicted in Fig. 7 for each bitrate for frames 1, 44 and 102. The SSIM index values between 0.99 and 1 are considered to be excellent in terms of QoE. The values between 0.95 and 0.99 are considered as good while those between 0.88 and 0.95 are considered as fair. The values between 0.50 and 0.88 are poor and below 0.50 are bad [20].

50–70 kbit/s is an acceptable QoT for human detection task, however, SSIM index values for this range denote poor quality in terms of QoE. This is what differentiates QoT and QoE. Fig. 8 depicts the quality of frames 1, 44 and 102 at 50 kbit/s.

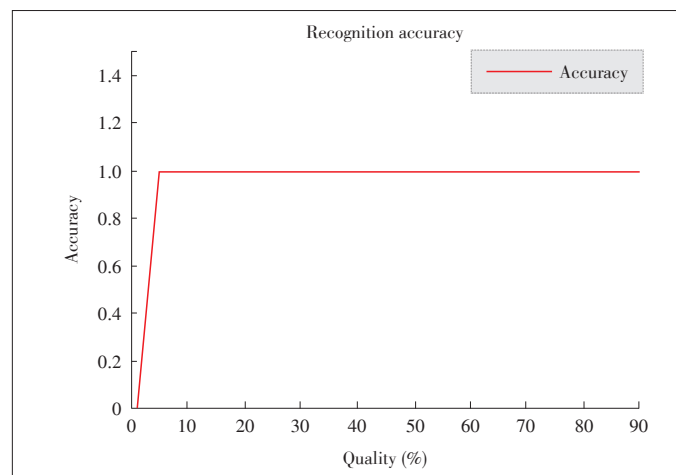


▲ Figure 7. Quality of frames 1, 44 and 102.

For the license number plate recognition task, the Car1 picture was taken close to the camera in bright weather conditions. The license plate number format was European based license plate. The recognition accuracy for the license number plate is 1 if the number plate is accurately recognized and is 0 if not. The recognition accuracy of Car1 is shown in Fig. 9 for each compression ratio. It can be observed that the recognition accuracy is still 1 at 5% compression of the original image. The original image quality was at 90%.



▲ Figure 8. Human detection video frames at 50 kbit/s.



▲ Figure 9. Car1 number plate recognition accuracy.

How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

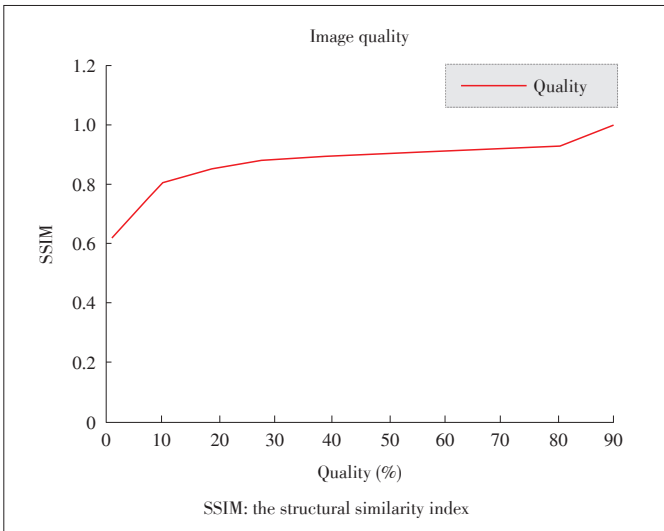
Fig. 10 illustrates the Car1 number plate sequence SSIM index values. It can be seen that although the video quality is considered poor (SSIM index value of 0.7) in terms of QoE at 5% quality level, the license number plate is still recognized. The range of 5%–10% quality level of Car1 can be considered as an acceptable QoT in this scenario of the license number plate recognition task.

Fig. 11 depicts the image size at each quality level of Car1. Since the acceptable QoT of Car1 is in the range of 5% and 10%, instead of transmitting the Car1 image in its original quality at 160 KB, the Car1 image of 40 KB at 5% quality level can be transmitted over the Internet with the same recognition accuracy of 1.

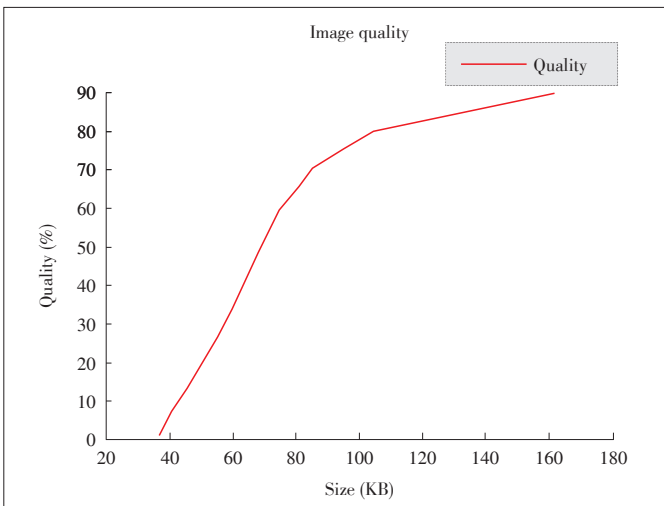
Fig. 12 depicts the Car1 number plate at 5% of the original quality. As per a human visual perception, 5% is considered poor quality, but for the QoT it can still be able to accurately recognize the license number plate.



▲ Figure 12. Car1 number plate at quality 5%.



▲ Figure 10. SSIM values of the Car1 image.



▲ Figure 11. Car1 number plate sequence frame size.

Based on the results obtained in the described uses cases, it can be observed that the QoT is different from QoE because QoE involves human and QoT involves machine/applications. If the direct mapping is considered, the acceptable QoT for human detection and license plate number recognition tasks is less than the acceptable QoE which is 3.5 and outlined by the authors in [4].

The goal of QoT for M2M communications is to meet the minimum quality that an IoT object can meet the minimum requirement of an IoT application. It focuses on the minimum quality of multimedia data captured by the camera node to be processed and delivered by edge and cloud nodes. For M2H applications (e.g., if a human being is an end user of an IoT application) the visual quality is needed for subjective viewing. Therefore, an acceptable QoE for multimedia data will be processed and delivered by the edge and cloud nodes. The ultimate goal of this study is to design such intelligent system to optimise network resources usage, so both AQoT and AQoE can be achieved depending on the use case scenario.

6 Conclusions

The IoT has been addressed as one of the biggest technological advances in the recent decades. IoT will soon be an inherent part of our daily lives ranging from smart homes, intelligent cars to aeroplanes and virtually everything we will interact with. With all the benefits that come with IoT, multimedia IoT comes with its own set of requirements such as power consumption, bandwidth usage and quality. This paper has defined a new concept, Acceptable QoT, whose experimental results have shown that it could significantly reduce bandwidth usage to fulfil IoT tasks without compromising the performance of the IoT system. Future work will be to develop intelligent IoT systems which can deliver multimedia IoT services to human or machine according to QoE and QoT automatically over edge/



How to Manage Multimedia Traffic: Based on QoE or QoT?

Amulya Karaadi, Is-Haka Mkwawa, and Lingfen Sun

cloud integrated networks.

References

- [1] Internet live stats. (2017). Internet live stats [online]. Available: <http://www.internetlivestats.com>
- [2] Statista. (2017). Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions) [online]. Available: <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide>
- [3] Cisco. (2017). Visual Networking Index: Forecast and Methodology, 2016-2021 [online]. Available: <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html>
- [4] A. Alfayly, I. Mkwawa, L. Sun, and E. Ifeachor, "QoE-driven LTE downlink scheduling for VoIP application," in *IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, USA, Jan. 2015, pp. 603–604. doi: 10.1109/CCNC.2015.7158043.
- [5] A. Karaadi, L. Sun, and I. Mkwawa, "Multimedia communications in internet of things QoT or QoE?," in *IEEE International Conference on Cyber, Physical and Social Computing (CPSCom)*, Exeter, UK, Jun. 2017, pp. 23–29. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.11.
- [6] C. Long, Y. Cao, T. Jiang, and Q. Zhang, "Edge computing framework for cooperative video processing in multimedia IoT systems," *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1126–1139, Oct. 2017. doi: 10.1109/TMM.2017.2764330.
- [7] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-vision: real-time face recognition using a mobile-cloudlet-cloud acceleration architecture," in *IEEE Symposium on Computers and Communications (ISCC)*, Cappadocia, Turkey, Jul. 2012, pp. 59–66. doi: 10.1109/ISCC.2012.6249269.
- [8] A. H. M. Amin, N. M. Ahmad, and A. M. M. Ali, "Decentralized face recognition scheme for distributed video surveillance in IoT-cloud infrastructure," in *IEEE Region 10 Symposium (TENSYP)*, Bali, Indonesia, May 2016, pp. 119–124. doi: 10.1109/TENCONSpring.2016.7519389.
- [9] W. Wang, Q. Wang, and K. Sohrawy, "Multimedia sensing as a service (MSaaS): exploring resource saving potentials of at cloud-edge IoTs and Fogs," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 487–495, Jun. 2016. doi: 10.1109/JIOT.2016.2578722.
- [10] N. Chen, Y. Chen, Y. You, et al., "Dynamic urban surveillance video stream processing using fog computing," in *IEEE Second International Conference on Multimedia Big Data*, Apr. 2016, pp. 105–112. doi: 10.1109/BigMM.2016.53.
- [11] X. Chen, J. N. Hwang, D. Meng, et al., "A quality-of-content-based joint source and channel coding for human detections in a mobile surveillance cloud," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 1, pp. 19–31, Mar. 2016. doi: 10.1109/TCSVT.2016.2539758.
- [12] L. Zhao, X. Zhang, X. Zhang, et al., "Intelligent analysis oriented surveillance video coding," in *IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, China, Jul. 2017, pp. 37–42. doi: 10.1109/ICME.2017.8019429.
- [13] D. G. Costa, M. Collotta, G. Pau, and C. D. Faundez, "A fuzzy-based approach for sensing, coding and transmission configuration of visual sensors in smart city applications," *Sensors*, vol. 17, no. 1, 2017. doi: 10.3390/s17010093.
- [14] A. Floris and L. Atzori, "Quality of experience in the multimedia internet of things: definition and practical use cases," in *IEEE International Conference on Communication Workshop (ICCW)*, London, UK, Jun. 2015, pp. 1747–1752. doi: 10.1109/ICCW.2015.7247433.
- [15] OpenCV. (2017). OpenCV [online]. Available: <https://opencv.org>
- [16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, USA, Jun. 2005, pp. 886–893. doi: 10.1109/CVPR.2005.177.
- [17] OpenALPR. (2017). OpenALPR [online]. Available: <http://www.openalpr.com>
- [18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, Sept. 2009. doi: 10.1109/TPAMI.2009.167.
- [19] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 121–132, 2004. doi: 10.1016/S0923-5965(03)00076-6.
- [20] A. N. Moldovan, I. Ghergulescu, and C. H. Muntean, "VQAMap: a novel mechanism for mapping objective video quality metrics to subjective MOS scale," *IEEE Transactions on Broadcasting*, vol. 62, no. 3, pp. 610–627, Jun. 2016. doi: 10.1109/TBC.2016.2570002.

Manuscript received: 2018-04-05

Biographies

Amulya Karaadi (Amulya.karaadi@plymouth.ac.uk) received her master's degree in computer systems networking and telecommunications from Staffordshire University, UK in 2016 and bachelor's degree in information technology from JNTU University, India in 2013. She is now pursuing her Ph.D. at the University of Plymouth, UK in the area of intelligent multimedia transmissions for the Internet of Things (IoT) applications.

Is-Haka Mkwawa (Is-Haka.Mkwawa@plymouth.ac.uk) received his Ph.D. in computing from the University of Bradford, UK in 2004. He has been working in various capacities on EU FP6, FP7 and Horizon 2020 projects since 2002 with the University of Bradford, the University College Dublin and the University of Plymouth. These projects included IASON (2002–2004), Euro-NGI (2003–2005), Euro-FGI (2005–2008), Science Foundation Ireland (2005–2006), FP6 Vital (2006–2008), FP7 ADAMANTIUM (2008–2010) and FP7 GERYON (2011–2014). He has authored several refereed publication and co-authored *Guide to Voice and Video over IP: For Fixed and Mobile Networks* (Springer, 2013). His research interests include IMS media plane security for next generation of emergency communication and services, QoE control and management, mobility management in mobile and wireless networks, software defined networking, power saving in IoT, overlay networks, performance analysis and evaluation of IMS mobility management, parallel computing, and collective communication.

Lingfen Sun (L.Sun@plymouth.ac.uk) received the B.Eng. degree in telecommunication engineering and the M.Sc. degree in communications and electronic system from the Institute of Communication Engineering, China and the Ph.D. degree in computing and communications from the University of Plymouth, UK. She is currently an associate professor (Reader) in multimedia communications and networks in the School of Computing, Electronics and Mathematics, University of Plymouth. She has been involved in several European projects including H2020 QoE-NET as PI, COST Action QUALINET as an MC member, FP7 GERYON as PI and Scientific Manager and FP7 ADAMANTIUM as Co-PI and WP leader. She has published one book and over 90 peer-refereed technical papers/book chapters since 2000. She was the Chair of QoE Interest Group of IEEE MMTC during 2010–2012, and Symposium Co-Chair for IEEE ICC' 14. She has been an AE for IEEE Transactions on Multimedia (2016–2018) and an expert reviewer for grants for EU, EPSRC (UK) and NSERC (Canada). Her main research interests include multimedia networking, multimedia quality assessment, QoS/QoE management, VoIP, DASH and SDN/NFV.

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

(School of Computer Science and Engineering, Nanyang Technological University, 639798, Singapore)

Abstract

Nowadays, media cloud and machine learning have become two hot research domains. On the one hand, the increasing user demand on multimedia services has triggered the emergence of media cloud, which uses cloud computing to better host media services. On the other hand, machine learning techniques have been successfully applied in a variety of multimedia applications as well as a list of infrastructure and platform services. In this article, we present a tutorial survey on the way of using machine learning techniques to address the emerging challenges in the infrastructure and platform layer of media cloud. Specifically, we begin with a review on the basic concepts of various machine learning techniques. Then, we examine the system architecture of media cloud, focusing on the functionalities in the infrastructure and platform layer. For each of these function and its corresponding challenge, we further illustrate the adoptable machine learning based approaches. Finally, we present an outlook on the open issues in this intersectional domain. The objective of this article is to provide a quick reference to inspire the researchers from either machine learning or media cloud area.

Keywords

machine learning; media cloud

1 Introduction

Recently, the increasing user demand on rich media experience has triggered an exponential growth of media services worldwide. According to the Cisco Visual Networking Index (VNI) report [1], the Internet video traffic would increase 3-fold from 2016 to 2021, contributing up to 82% of all Internet traffic by 2021. This trend may bring tremendous opportunities for all the stakeholders in the media service chain. Application developers can attract more customers by providing novel media experiences, such as video-on-demand, multi-screen interactions, and face/expression recognition. Platform service providers can host more applications and get more revenue. Content service providers can generate more contents and have them viewed by billions of users. Network service operators can expect to deliver significantly more network traffic. Nevertheless, such a trend also calls for novel paradigms to properly fulfil all the requirements.

Media cloud [2]–[5], inheriting the advances from cloud computing, has emerged as a promising computing paradigm to provide novel multimedia services with satisfied Quality of Service (QoS) and reduced cost. Specifically, media cloud adds media-related functions to each cloud computing layers (i.e.,

Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS)), following the cloud computing paradigm. In the infrastructure layer, media cloud schedules more virtual resources in a more dynamic style. In the platform layer, it integrates a list of media-specific functions, such as media adaptation, media streaming, and media traffic analysis, to meet various QoS requirements from different media services. In the software layer, media cloud is able to host novel media services with higher complexity than traditional web services with only text and images.

The uniqueness of media cloud posits a list of new challenges, especially in the infrastructure and platform layer. First, the process, storage, and transmission of multimedia contents need more resources, leading to more power consumption and higher failure ratio of physical and virtual resources. Second, most media services need to be delivered with low latency and high volume, thus requiring precise workload prediction and careful resource scheduling accordingly. Third, the media distribution and adaption are more resource-intensive and thus more complicated than traditional web services. Last but not least, different media functions must be orchestrated properly to better serve the media users with optimized cost.

Machine learning, which have been intensively applied in various multimedia applications, provides a nature solution to

address these challenges in media cloud. In particular, machine learning represents the set of algorithms that can progressively improve the performance of a specific task without being explicitly programmed. As a result, the adoption of machine learning makes the development of new media services and the optimization of existing media systems much easier than ever before. For example, machine learning has been already widely used in image and video processing such as face recognition, image classification, and video surveillance. However, the machine learning research in the infrastructure and platform layer of media cloud has not been as hot as the upper layer media applications.

In this article, we present a survey of how machine learning addresses the challenges in media cloud, from the infrastructure and platform perspectives. In particular, we start with the tutorial study on different machine learning strategies, as well as the concept and the challenges of media cloud. Then, we substantiate the ways of applying these machine learning techniques into media cloud via a literature review. The map between machine learning techniques and the challenges in the infrastructure and platform layer of media cloud are illustrated respectively. As a result, this allows the researchers from either machine learning or media cloud domain to quickly grasp the state-of-the-art knowledge in the overlaps of these two domains.

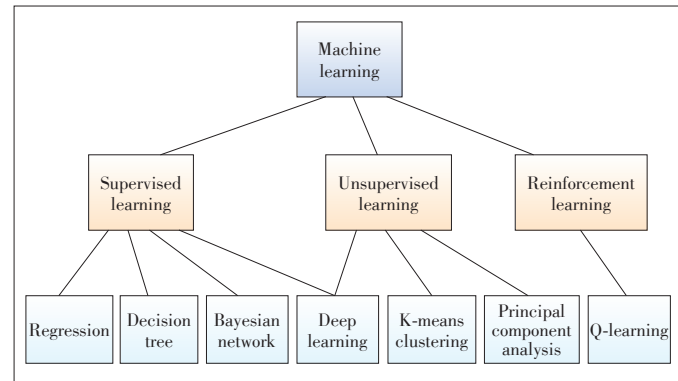
The rest of this paper is organized as follows. In Section 2, we introduce the basic ideas of machine learning as well as a layered media cloud framework and the functionalities in each layer. In Section 3, we review the machine learning efforts towards the challenges in the infrastructure layer of media cloud. In Section 4, we investigate the machine learning solutions to address the issues in the platform layer of media cloud. In Section 5, we highlight a list of open research issues in media cloud that could be addressed by machine learning techniques in the near future. Finally, in Section 6, we conclude this article.

2 Overview of Machine Learning and Media Cloud

In this section, we first introduce the basic machine learning algorithms to provide the necessary background knowledge, which will be referred to in the later sections. Then, we illustrate the media cloud framework and decompose it into a layered model. This model will serve as the blueprint to survey existing research efforts.

2.1 Machine Learning Algorithms

Existing machine learning algorithms can be generally categorized into three types [6], including supervised learning, unsupervised learning, and reinforcement learning. **Fig. 1** depicts such categorization, where each category further consists of one or more sub-categories. The brief concepts of these tech-



▲ **Figure 1.** A categorization of machine learning.

niques are presented as follows.

2.1.1 Supervised Learning

Supervised learning aims to build a model to map an input to an output based on pre-labelled input-output pairs. Typically, the input objective is a high dimension vector, the output is a low dimension or even one-dimension decision, while the objective is to minimize the difference between the labels and the output from the model. Regression, decision tree, Bayesian network, and deep neural network/deep learning are supervised learning algorithms

Regression tries to find a single function with proper parameters to represent the relationship between the input and output. There are a list of different regression models with different function types to deal with different input. For example, linear regression uses a linear function to deal with continuous input, logistic regression uses a logistic function to deal with categorical input, and non-linear regression uses non-linear functions (e.g., polynomial, logarithmic).

A decision tree uses a tree-like graph to deduct the consequences from the input. In a decision tree, each internal node refers to a control variable on an attribute, each branch refers to the consequence from the control decision, each leaf node refers to one final output, and the paths from the root to each leaf node refer to the rules.

Bayesian network is a probabilistic graphical model that represents a set of variables and their conditional dependencies via a directed acyclic graph. It calculates an estimate for the class probability from the training set based on the Bayes' theorem, and uses the estimation to map the input and output.

Deep neural network/deep learning is generally based on artificial neural networks, which consist of a collection of multiple layers of connected units (i.e., neurons). The weights between each pair of neurons are tunable to optimize the objective function. It can be used as a supervised learning approach for classification tasks.

2.1.2 Unsupervised Learning

On the contrary to supervised learning, unsupervised learn-

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

ing algorithms, such as K-means clustering, principle component analysis (PCA), and deep learning, focuses on inferring a function to describe the hidden structure from unlabeled data.

K-means clustering aims to partition n observations into k clusters, where each observation belongs to one cluster. The criteria is to ensure the overall shortest distance between the observations and the centroid of their assigned clusters accordingly.

Principle component analysis uses an orthogonal transformation to convert given observations into a set of values of linearly uncorrelated variables in lower dimension. The generated variables are often called as principal components. They serve as a projection of the original higher dimension input from its most informative perspective.

Deep learning can be also used in an unsupervised manner. Due to its multi-layer structure of fully connected neurons, deep learning can well represent complex non-linear relationships. As a result, it is able to compact the input in higher dimension into informative output with much lower dimension. Deep auto-encoder is one example in this category.

2.1.3 Reinforcement Learning

Reinforcement learning trains the model by interacting with the environment using different actions and receiving the incurred rewards iteratively. Specifically, it relies on two operations, including exploration of uncharted territory and exploitation of current knowledge to maximize the received rewards. On the one hand, exploration operation enables the algorithm to keep trying different decisions so that it can evolve without explicitly giving labelled data. On the other hand, the exploitation allows the algorithm to be aware of the explored point and move closer to the optimal decision strategy. Q-learning is a reinforcement learning algorithm.

Q-learning is most representative reinforcement learning technique. Specifically, it uses Q-value to represent the quality of a state-action combination, and iteratively update this Q-value for the improvement. Q-learning can compare the expected utility of the available actions without requiring a model of the environment. Moreover, it has been proven that Q-learning is able to eventually get the optimal action-selection policy, for any finite Markov decision process.

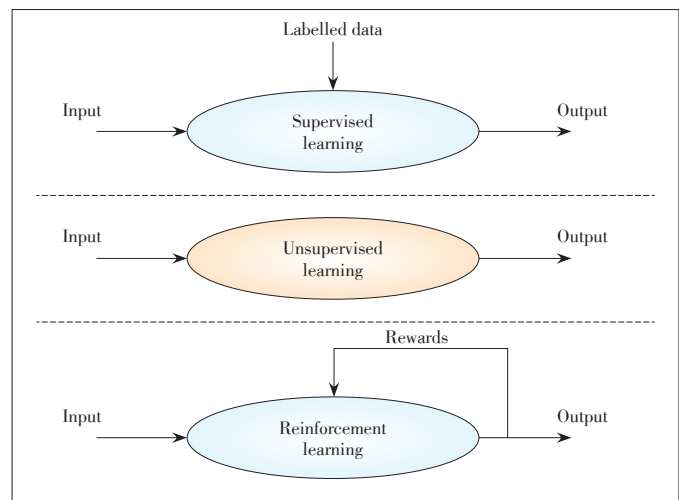
Fig. 2 illustrates a comparison of supervised learning, unsupervised learning, and reinforcement learning. In particular, supervised learning relies on the pre-defined labelled input and output pairs as the target. On the other hand, unsupervised learning does not need labelled data, and it uses the internal features of the dataset instead of any labelled data as the objective. Whereas reinforcement learning does not have the labelled data in advance. It has to sense the results by performing different actions, and use the previous outputs as the objective.

We will illustrate how the machine learning techniques from different categories can be applied in media cloud in Sections

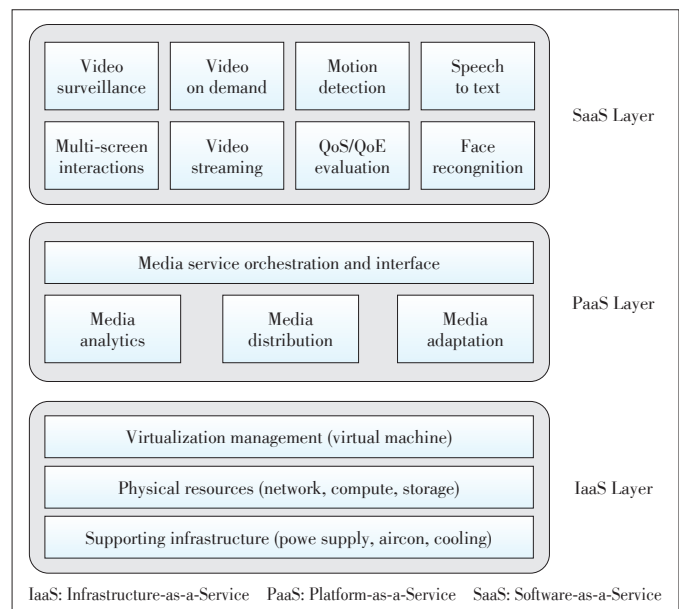
3 and 4.

2.2 Media Cloud Framework

Media cloud aims to leverage cloud computing paradigm together with a list of media-related functions to enhance the media experience. From a cloud-centric view [2], [5], it still can be defined as a cloud-centric layered model as shown in Fig. 3. Each layer consists of traditional cloud services (e.g., virtualization and resource management) and corresponding media services (e.g., media adaptation and media analytic). This conceptual hierarchy provides a clear clue for us to characterize the technical challenges and existing works in different layer. Note that, this paper mainly focuses on the machine learning works for infrastructure and platform layer, whereas the efforts



▲ Figure 2. Comparison of three different machine learning categories.



▲ Figure 3. A layered architecture of media cloud, consisting of three layers, including IaaS, PaaS, and SaaS from the bottom up.

towards multimedia software applications have been intensively reviewed by many other literatures [7]–[11].

The infrastructure layer aggregates all the physical ICT resources together via virtualization technology, with the objective to allocate them in a fine-granular, on-demand, and fault-free manner. According to the different functionalities, we can further classify this layer into the following three sub-layers.

- **Supporting Infrastructure:** This layer refers to the power supply, air-conditioner, and other cooling systems, which support the smooth operations of datacenters [12] as well as the cloud services on top of them. It focuses on those bottom level schemes, for instance the datacenter layout schedule, power consumption optimization, and cooling system management [13].
- **Physical Resources:** This layer consists of servers (including CPUs, hard disks, memory, network interface, etc.) and switches/routers, which provide the networking, computation, and storage capacity. These resources need to be monitored in real time and well maintained once there is any fault [14]. As a result, the hosted cloud services will not be affected.
- **Virtualization Management:** This layer virtualizes the underlying physical resources into a virtual resource pool in terms of virtual machines [15], [16]. These resources are then exposed to the cloud platform services to meet the specific Service Level Agreement (SLA) with the lowest possible cost. To achieve this target, the resource provision needs be allocated elastically and dynamically via virtual machine configuration and migration.

Following this hierarchy, machine learning algorithms can be developed and applied to each of the sub-layer to address the corresponding challenges. This will be the main focus of Section 3.

The platform layer encapsulates various fundamental media services into a layer of middle-ware, by utilizing the virtual resources provided by the infrastructure layer. This middle-ware is then exposed to the software layer via a set of APIs. According to the functionalities, we cast these media services into the following four types.

- **Media Analytics:** This service refers to the data mining schemes that focus on the nature of media contents as well as the user request patterns. Typical examples include media content popularity prediction [17] and content recommendation [18].
- **Media Distribution:** This service is in charge of acquiring media contents from the origin servers, and delivering them to end users throughout the media cloud. The objective is to improve the delivery efficiency. Content caching [19] and pre-fetching [20] are two representative examples under this category.
- **Media Adaptation:** This service modifies the original media contents into the target ones with different domains (e.g., format, rate, resolution, and annotation). Typical examples of

such services are content encoding/transcoding [21], content quality estimation/assessment [22], and media mashup [23].

Similarly, intelligent mechanisms powered by machine learning algorithms can be adopted by these services to improve the quality. The in-depth survey of adopting machine learning in the cloud-based media platform services will be covered in Section 4.

3 Machine Learning in Infrastructure Layer

In this section, we present three typical scenarios that can be benefited from machine learning techniques in the infrastructure layer of media cloud (Fig. 3). They are datacenter power consumption prediction and control, cloud resources failure prediction and operation, and virtual machine configuration and operation.

3.1 Power Consumption Prediction and Control

Datacenters nowadays have become a large energy consumption center, resulting in the fact that even modest improvements are able to yield significant cost cut and avert millions of carbon emissions globally. In particular, power consumption from datacenter comprises around 1.4% of global energy usage and 2% of global carbon emissions [24]. Among all datacenters in the world, the majority of them has a power usage effectiveness (PUE) at 1.6–2.0 while the ideal efficiency should be around 1.1–1.2 [24]. As a result, there are sufficient improvement spaces, and any small one improvement can bring great impact.

Regression is one of the classic ways to predict the power consumption. Choi et al. [25] proposed a three-dimension regression model to predict the power usage using work intensity and CPU utilization. This model achieved 9% error margin on average comparing with real observed usage. Similarly, Lewis et al. [26] developed a linear regression model to correlate processor power, bus activity, and system ambient temperatures with real-time server power consumption by considering. Their model gave an error of 4% as verified using a set of benchmarks. In addition, some further studies [27] indicated that the linear regression model based on CPU usage and workload is only able to provide reasonable prediction accuracy for CPU-intensive jobs, while a Gaussian mixture regression model can perform consistently well with different workload (e.g., IO or memory intensive jobs).

Neural network/deep learning is another powerful tool to predict the data-center power consumption with the ability to take into much more input parameters. Gao [28] from Google, presented a simply three layers neural network model by considering 19 different factors as the input, including the IT load, weather conditions, number of chillers and cooling towers running, equipment set points and so on. This generated a promising prediction performance already, with a mean absolute error

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

of 0.4% and standard deviation of 0.005. Li et al. [29] further deepened the neural network with more layers. Specifically, it used a linear recursive auto-encoder to process the input, and added an additional layer before the final output to correct the prediction results of auto-encoder. This model was fed by 11-dimension input, including CPU/memory/disk usage, network traffic, and file system workload. The results pushed the performance a bit further by reducing around 40% prediction error comparing with a widely-used regression based time-series prediction model.

3.2 Cloud Failure Prediction and Operation

Given the scale and complexity of cloud infrastructure, the failure prediction and operation desires significant high levels of automation. In particular, such failure consists of physical hardware failure such as disk/CPU/memory/network error and virtual jobs failure due to software or configuration issues. It is challengeable but important to properly identify these failures and take actions accordingly on time if not in advance, so that the high standard Service Level Agreement can be well maintained.

Decision trees have become a popular method for failure prediction and detection. Pelleg et al. [30] collected system metrics including execution count, CPU usage, waiting time, blocked time, and IO count, on top of Xen virtual machine, and fed them into a decision tree classifier. By using this classifier, they were able to detect the potential system problems with 0.94 receiver operating characteristic (ROC) curve as the accuracy. Fu [31] proposed a framework to combine the decision tree model together with the principle component analysis algorithm. Specifically, the principle component analysis is first used to reduce feature dimensions from the set of collected cloud infrastructure parameters, then only the principle components are input into a decision tree classifier to identify anomalies in the cloud. A following-up work [32] further integrated a Bayesian model with the decision tree. It first reduced 50 plus system metrics including CPU statistics, memory swap statistics, IO requests, and network traffic into principle components. Then this solution fed these generated components into both the Bayesian predictor and decision tree, and did an ensemble between the output. This generated a promising result with 0.99 ROC curve.

At the same time, there is an increasing popularity of applying neural network or deep learning into the cloud failure prediction task recently. Prevost et al. [33] presented a neural network model to predict the cloud datacenter workload. Specifically, the model takes historical data points as input to predict the future trend, with the objective to minimize the Rooted Mean Squared Error (RMSE) of sample data. Chen et al. [34] developed a recurrent neural network (RNN) based model to learn the temporal characteristics of resource usage metrics including CPU and memory usage, which are in turn used to calculate the failure possibility of a running job in the cloud.

They then verified the model by using real Google cluster workload traces. The results indicated a reasonable accuracy with a false positive rate at around 6%, and the following-up operations based on the prediction were able to save 6% to 10% cost saving by early killing and restarting jobs with high failure possibility. Zhu et al. [35] also explored the performance of back propagation based neural network combined with a boosting approach, in driver failure prediction for large scale storage system. Moreover, it compared the results with a traditional Supported Vector Machine (SVM) model on a real world database. The evaluation showed the proposed neural network model achieves over 95% detection accuracy which is much better than 68% achieved by SVM.

3.3 Virtual Resource Configuration and Consolidation

Cloud infrastructure virtualizes the physical resources into a virtual machine pool and operates them in a fine-grained model, thus providing significant flexibilities to host different services. In particular, virtual machines can be dynamically turned on/off, migrated from one physical machine to another. As a result, there is a chance to significantly increase the cost efficiency by properly orchestrating the virtual machines to consolidate the workload in an on-demand manner.

Bayesian networks have become a popular tool to consolidate virtual resources for cloud environment. Sohrabi et al. [36] proposed a virtual machine migration heuristic based on Bayesian networks. In particular, this solution evaluates the probability of a physical server host to be overloaded, then migrates the virtual machines away from those servers. As a result, not only energy consumption can be saved by consolidating virtual machines, but also the performance is improved by balancing the workload into multiple hosts. Li et al. [37] discussed a very similar Bayesian based approach to estimate the resource utilization in physical machines and then used it to predict the migration probability of virtual machines. Shyam et al. [38] presented a Bayesian model to determine both short and long term virtual resource requirements for CPU or memory intensive applications running in cloud environment. They built the Bayesian model based on a list of parameters, including day of week, time-interval of application access, workload, benchmarks, and availability of virtual machines. All of these works were able to generate better performance in terms of either lower energy consumption of cloud infrastructure or higher accuracy in predicting virtual resource utilization, by comparing with a few other non-machine-learning methods.

Reinforcement learning has also been applied into this task. Masoumzadeh et al. [39] presented a Q-learning based model, which takes multiple virtual machine metrics (including CPU performance, disk storage, memory usage and network bandwidth) as the input, the migration action as the output, and the energy consumption combined with SLA score as the reward function. The trained model outperforms virtual machine selection policies using fixed criteria for decision making. Jin et al.

[40], [41] built the virtual machine migration model specifically for the cloud media scenario by using the same technique. In particular, this work used the user interactive behaviors in multi-screen applications as the input, the backend virtual machine migration decision as the output, and the total monetary cost of operating cloud resources as the rewards. The result revealed a significant cost saving compared with some heuristics. The model also showed a very closed performance to an offline optimal solution. Liu et al. [42] introduced deep reinforcement learning into the virtual machine allocation and consolidation problem. Specifically, deep reinforcement learning integrates deep neural network with reinforcement learning, enabling the algorithm to deal with larger state space while keeping the fast coverage speed. Thus, this work is able to take the real-time metrics for each job and virtual machine pair as the input, the job and virtual machine matching decision as the output, and the combined job latency and energy consumption as the rewards. Similarly, this approach also achieves cost saving while at the same time the latency improvement.

3.4 Summary

We demonstrate a list of works that use different machine learning techniques to tackle three major infrastructure challenges in this section. **Table 1** matches the specific machine learning approaches with the topic domains for each work, so that the interested readers can quickly obtain the ways how machine learning can be applied in the infrastructure layer in media cloud.

4 Machine Learning in Platform Layer

In this section, we discuss the machine learning applications in three major media platform services (Fig. 3). Specifically, this section covers content popularity prediction and recommendation in media analysis domain, content caching and pre-fetching in media distribution domain, and content transcoding in media adaptation domain.

4.1 Content Popularity Prediction and Recommendation

The tremendous growth of multimedia content generation has changed not only the user content consumption behaviors, but also the way of operating the media services. Millions of hours of video are generated and uploaded to YouTube every day [43]. As opposed to the traditional TV programs where all the audiences watched the same content at the same time, mul-

timedia content users have much more options to spend their video watching time. As a result, given such a large amount of available user generated content, their popularity are much more difficult to be predicted. Moreover, the personalized video recommendation becomes increasingly important for better user experience.

Regression is the simplest yet feasible machine learning tool for dealing with the content popularity prediction task. Szabo et al. [44] found the long-term content popularity on YouTube had a strong correlation with their early popularity. Such correlation can be represented by a linear regression model to predict the long-term content popularity. Borgho et al. [45] confirmed the efficiency of using the linear model to predict the popularity, and further derived a multi-linear regression model by taking more factors such as video quality, number of keywords, uploader view count, uploader followers, and uploader video count. Chu et al. [46] adopted a similar approach by using a bilinear regression framework to achieve a personalized content recommendation system. They used this regression model to associate the attributes in user profiles with the potential content that might be interested to the user. Unsupervised learning tools provide another angle to examine the content popularity task. Szabo et al. [44] used k-means algorithm to separate video contents into two clusters, where the content popularity in one group grew faster than the average, and the other grew slower. Borgho et al. [45] applied PCA to characterize the relationships between different content/user profiles and the content popularity. In this way, they were able to identify the groups of variables which were responsible for the variation of popularity prediction. Ahmed et al. [47] introduced another clustering algorithm known as affinity propagation to the content popularity prediction task. This method does not require a predefined number of clusters, which differs from the k-means algorithm. By properly cluster the similarity score for the content popularity, this approach is able to outperform the traditional k-means and the linear regression models.

There are also a few works making use of deep learning for content recommendation. Ma et al. [48] developed an auto-encoder model backed by unsupervised deep learning technique, to cluster the similarity among different videos. They could recommend different videos to different users according to their categories. Covington et al. [49] designed a YouTube recommendation system based on a fully-connected deep neural network. It first embeds the video profile, video watch history, search tokens, previous impressions, and user profile into high-

▼ **Table 1. Mapping between machine learning methods and cloud infrastructure services for each literature work**

	Regression	Decision tree	Bayesian network	PCA	Q-learning	Deep learning
Power predict and control	[25], [26], [27]					[28], [29]
Failure predict and operate		[30], [31], [32]		[31], [32]		[33], [34], [35]
VM configure and consolidate			[36], [37], [38]		[39], [40], [41], [42]	

PCA: principle component analysis

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

dimension vectors, and uses the concatenation of them as the input to the neural network. And the output can be directly used as ranked recommendations for each individual user.

4.2 Media Content Caching and Pre-fetching

It is a common practice nowadays to cache multimedia content data in the intermediate nodes between users and the host servers, to improve the user experience as well as the media service operational cost. In particular, because the sizes of multimedia contents are much larger than the traditional text or images, it takes more time to transmit them from the host to the end-users. To relieve the pain, content delivery networks has been proposed to cache contents in some middle places. However, it is not efficient to cache all or just blindly choose a few at all the time. Therefore, the key factor of this task is how to choose the right content to be cached at the right time. Bayesian network is a promising tool for content personalization prefetching task by identifying the right content to be cached in the content delivery network [50]. Venketesh et al. [51] introduced the naive Bayesian classifier to calculate the probability of viewing a potential content based on the browsing pattern exhibited by the end users in sessions. This approach helps to increase the cache hit rate and minimize access latency, especially when user has long browsing sessions. Ali et al. [52] used naive Bayesian classifier in the same task but in a different way. Specifically, they incorporated the Bayesian classifier with the classical caching strategy (e.g., Least-Recently-Used and Greedy-Based), by learning the dependency probability between the content access log and the content re-visit event. As a result, when doing cache eviction, the content with higher probability of re-visit will be kept. Clustering is another promising way for the content personalization prefetching task [50]. Yan et al. [53] uses K-means to cluster users based on their geo-location and temporal access patterns. In this way, the contents for different mobile applications can be prefetched into the mobile device, thus reducing the app launching delay and improving the user experience. Hu et al. [54] applied the affinity propagation clustering algorithm to group users in different communities, based on their social relationships, geo-locations, and video watching interests. As a result, the content caching decision can be made specifically for different communities, thus improving the caching efficiency.

It is also possible to use reinforcement learning to optimize this content prefetching process. Hu et al. [55] formulated the content prefetching problem as a Markov Decision Process (MDP), with the objective to strike a balance between the increased content caching cost from incorrect prediction and the reduced content download delay from correct prediction. A Q-learning based approach was then proposed to address this problem.

4.3 Media Content Adaptation

There is an increasing trend to consume online video via mo-

bile phones rather than via fixed terminals like TV and PCs. This means the video contents must adapt to the terminals by providing different resolution, bitrates, and quality versions for different screen sizes under different network environments. Such video adaptation tasks can be computation intensive, but at the same time, they also pose an opportunity to improve the user experience with different devices.

Deep learning is the most widely used machine learning tool for such tasks. Covell et al. [56] explored a neural network based framework to predict the parameters of a model that relates the bitrate to various video properties. Specifically, in video transcoding, the perceptual video quality for a given bandwidth constraint can be achieved by controlling the quantization levels. In this context, they used deep neural network to correlate this quantization level with the bitrate, and achieved a much higher accuracy than the traditional alternative. Dash et al. [57] proposed to use deep neural network to assess the quality of images after encoding/decoding or transcoding, and the model is able to achieve as high as 98% image-level accuracy for the assessment. Zhang et al. [58] further extended the quality of experience assessment from images into video by using an even deeper neural networks with more hidden layers and unique structures.

4.4 Summary

In this section, we demonstrate the way of using machine learning techniques for three important media platform services. **Table 2** maps each work according to the adopted machine learning technique as well as the detailed platform services that it focused on. As a result, the interested readers can quickly obtain the ways how machine learning can be used in the platform layer services in media cloud.

5 Open Research Issues

The research on applying machine learning to media cloud is at the infancy stage, while there are still many open challenges. In this section, we present a brief outlook on these open issues, aiming to provide insights for researchers from either machine learning or media cloud area.

5.1 Media Traffic Classification and Flow Control

Real-time media traffic classification and flow information are important for network management and optimizing the service operational cost. The traditional way of classifying Internet traffic is based on the network protocols (e.g., TCP or UDP). However, such static methods are not enough for media contents as they roughly use the same protocol for network transmission.

Machine learning is able to either learn from the historical media traffic data with fine-grained categories as per different metric set in a supervised manner, or cluster the real-time media traffic based on their internal features into different groups

▼Table 2. Mapping between machine learning methods and cloud platform services for each literature work

	Regression	Bayesian network	K-means	PCA	Affinity propagation	Q-learning	Deep learning
Content recommendation	[44], [45], [46]		[44]	[45]	[47]		[48], [49]
Content prefetching		[50], [51], [52]	[53]		[54]	[55]	
Media data adaptation							[56], [57], [58]

PCA: principle component analysis

in an unsupervised manner. For example, for the former one, it is possible to make use of distributed SVM [59], and deep learning [60]. While for the later one, K-means [61] and deep auto-encoder [62] can be the right tools.

5.2 Media Service Chain Orchestration

Recently, network function virtualization (NFV) emerges to transform the way of operating communication networks. Specifically, NFV implements network functions in software, orchestrating various services dynamically instead of follow the pre-defined workflows from hardware. As a result, it provides an opportunity to dramatically increase the infrastructure flexibility, simplify the resource management process, and reduce both hardware and operational cost.

The emergence of NFV-enabled media cloud framework [63] offers the opportunity to further improve the performance of media services running on top of the media cloud, and machine learning can be one of the best candidates to achieve this target. In particular, media services are not standalone. Most media services require a list of functions to be orchestrated in a chain. For example, the consumption of online video streaming via a mobile phone involves content caching, prefetching, adaptation and personalization. Machine learning can be used to learn the most efficient pattern on how to orchestrate these services in the large scale.

5.3 Media Security

Nowadays, it is much easier to access, download, and upload multimedia contents via the Internet, making the Digital Rights Management (DRM) much more difficult and complicated than before. Previously, audio and video DRM was usually achieved by physical subscription and rental, but this method does not work well today, because any subscriber can simply upload the copyrighted contents as user-generated contents to popular video distribution platforms like YouTube. It is hard to restrict such behavior giving the huge amount of uploaded contents every day.

Machine learning can be applied in this field too. In particular, it can be used to classify or identify if the uploaded audio or video has a copyright issue, by learning from a set of labelled contents from their commercial owners. It is also possible to use machine learning to improve the performance of DRM techniques such as digital watermarking by learning from the failed cases. In fact, such operation has been introduced to the music and audio DRM system [64], while it is on

the way to extend to video contents.

6 Conclusions

In this article, we presented a tutorial survey on applying machine learning techniques to address challenges in the infrastructure and platform layers of media cloud. In particular, we first reviewed the basic concept of different machine learning techniques. Then, we examined the system architecture of media cloud framework, focusing on the functionalities in the infrastructure and platform layers. For each functionality and its corresponding challenge, we further illustrated the adopted machine learning techniques. Finally, we present an outlook on a few open issues in this domain, aiming to inspire the researchers from either machine learning or media cloud area.

References

- [1] Cisco. (2017). Cisco visual networking index: Forecast and methodology, 2016-2021 [Online]. Available: <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf>
- [2] W. Zhu, C. Luo, J. Wang, and S. Li, "Multimedia cloud computing," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 59–69, May 2011. doi: 10.1109/MSP.2011.940269.
- [3] M. Tan and X. Su, "Media cloud: when media revolution meets rise of cloud computing," in *IEEE 6th International Symposium on Service Oriented System Engineering (SOSE)*, Irvine, USA, 2011, pp. 251–261. doi: 10.1109/SOSE.2011.6139114.
- [4] Y. Xu and S. Mao, "A survey of mobile cloud computing for rich media applications," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 46–53, Jun. 2013. doi: 10.1109/MWC.2013.6549282.
- [5] Y. Wen, X. Zhu, J. Rodrigues, and C. Chen, "Cloud mobile media: Reflections and outlook," *IEEE Transactions on Multimedia*, vol. 16, no. 4, pp. 885–902, Jun. 2014. doi: 10.1109/TMM.2014.2315596.
- [6] S. Marsland, *Machine Learning: An Algorithmic Perspective*. Boca Raton, USA: CRC Press, 2015.
- [7] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys*, vol. 35, no. 4, pp. 399–458, 2003.
- [8] R. Poppe, "A survey on vision-based human action recognition," *Image and vision computing*, vol. 28, no. 6, pp. 976–990, 2010.
- [9] M. Wang and X.-S. Hua, "Active learning in multimedia annotation and retrieval: a survey," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 2, article no. 10, Feb. 2011. doi: 10.1145/1899412.1899414.
- [10] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 41, no. 6, pp. 797–819, Nov. 2011. doi: 10.1109/TSMCC.2011.2109710.
- [11] L. Deng and X. Li, "Machine learning paradigms for speech recognition: an overview," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1060–1089, May 2013. doi: 10.1109/TASL.2013.2244083.
- [12] L. A. Barroso, J. Clidaras, and U. Ho'lzle, "The datacenter as a computer: an introduction to the design of warehouse-scale machines," *Synthesis Lectures on Computer Architecture*, vol. 8, no. 3, pp. 1–154, 2013.
- [13] W. Zhang, Y. Wen, Y. Wong, K. Toh, and C.-H. Chen, "Towards joint optimization over ict and cooling systems in data centre: a survey," *IEEE Communica-*

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

- tions *Surveys and Tutorials*, vol. 18, no. 3, pp. 1596–1616, 2016. doi: 10.1109/COMST.2016.2545109.
- [14] M. Isard, “Autopilot: automatic data center management,” *ACM SIGOPS Operating Systems Review*, vol. 41, no. 2, pp. 60–67, Apr. 2007. doi: 10.1145/1243418.1243426.
- [15] P. Barham, B. Dragovic, K. Fraser, et al., “Xen and the art of virtualization,” in *ACM Symposium on Operating Systems Principles*, New York, USA, 2003.
- [16] S. Soltész, H. Pözl, M. E. Fuczynski, A. Bavier, and L. Peterson, “Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors,” *ACM SIGOPS Operating Systems Review*, vol. 41, no. 3, pp. 275–287, 2007. doi: 10.1145/1272996.1273025.
- [17] A. Tatar, M. D. de Amorim, S. Fdida, and P. Antoniadis, “A survey on predicting the popularity of web content,” *Journal of Internet Services and Applications*, vol. 5, no. 1, pp. 1–20, 2014. doi: 10.1186/s13174-014-0008-y.
- [18] F. Xia, N. Y. Asabere, A. M. Ahmed, J. Li, and X. Kong, “Mobile multimedia recommendation in smart communities: a survey,” *IEEE Access*, vol. 1, pp. 606–624, Sept. 2013. doi: 10.1109/ACCESS.2013.2281156.
- [19] S. Podlipnig and L. Böszörményi, “A survey of web cache replacement strategies,” *ACM Computing Surveys*, vol. 35, no. 4, pp. 374–398, 2003.
- [20] J. Famaey, F. Ierbeke, T. Wauters, and F. De Turck, “Towards a predictive cache replacement strategy for multimedia content,” *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 219–227, 2013. doi: 10.1016/j.jnca.2012.08.014.
- [21] 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.
- [22] Y. Chen, K. Wu, and Q. Zhang, “From QoS to QoE: a tutorial on video quality assessment,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 1126–1165, 2015. doi: 10.1109/COMST.2014.2363139.
- [23] M. K. Saini, R. Gadde, S. Yan, and W. T. Ooi, “Movimash: online mobile video mashup,” in *Proc. 20th ACM International Conference on Multimedia*, Nara, Japan, 2012, pp. 139–148.
- [24] M. Avgerinou, P. Bertoldi, and L. Castellazzi, “Trends in data centre energy consumption under the european code of conduct for data centre energy efficiency,” *Energies*, vol. 10, no. 10, p. 1470, 2017.
- [25] J. Choi, S. Govindan, B. Urgaonkar, and A. Sivasubramaniam, “Pro-filing, prediction, and capping of power consumption in consolidated environments,” in *IEEE International Symposium on Modeling, Analysis and Simulation of Computers and Telecommunication Systems*, Baltimore, USA, 2008, pp. 1–10. doi: 10.1109/MASCOT.2008.4770558.
- [26] A. W. Lewis, S. Ghosh, and N.-F. Tzeng, “Run-time energy consumption estimation based on workload in server systems,” *HotPower*, vol. 8, pp. 17–21, 2008.
- [27] G. Dhiman, K. Mihic, and T. Rosing, “A system for online power prediction in virtualized environments using gaussian mixture models,” in *ACM/IEEE 47th Design Automation Conference (DAC)*, Anaheim, USA, 2010, pp. 807–812. doi: 10.1145/1837274.1837478.
- [28] J. Gao, “Machine learning applications for data center optimization,” Google White Paper, 2014.
- [29] Y. Li, H. Hu, Y. Wen, and J. Zhang, “Learning-based power prediction for data centre operations via deep neural networks,” in *Proc. 5th International Workshop on Energy Efficient Data Centres*, Waterloo, Canada, 2016, pp. 1–10. doi: 10.1145/2940679.2940685.
- [30] D. Pelleg, M. Ben-Yehuda, R. Harper, L. Spainhower, and T. Adeshiyani, “Vigilant: out-of-band detection of failures in virtual machines,” *ACM SIGOPS Operating Systems Review*, vol. 42, no. 1, pp. 26–31, Jan. 2008.
- [31] S. Fu, “Performance metric selection for autonomic anomaly detection on cloud computing systems,” in *IEEE Global Telecommunications Conference (GLOBECOM 2011)*, Kathmandu, Nepal, 2011, pp. 1–5. doi: 10.1109/GLOCOM.2011.6134532.
- [32] Q. Guan, Z. Zhang, and S. Fu, “Ensemble of bayesian predictors and decision trees for proactive failure management in cloud computing systems,” *Journal of Communications*, vol. 7, no. 1, pp. 52–61, 2012. doi: 10.4304/jcm.7.1.52-61.
- [33] J. J. Prevost, K. Nagothu, B. Kelley, and M. Jamshidi, “Prediction of cloud data center networks loads using stochastic and neural models,” in *IEEE International Conference on System of Systems Engineering (SoSE)*, Albuquerque, USA, 2011, pp. 276–281. doi: 10.1109/SYSOSE.2011.5966610.
- [34] X. Chen, C.-D. Lu, and K. Pattabiraman, “Failure prediction of jobs in compute clouds: A google cluster case study,” in *IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, Naples, Italy, 2014, pp. 341–346. doi: 10.1109/ISSREW.2014.105.
- [35] B. Zhu, G. Wang, X. Liu, et al., “Proactive drive failure prediction for large scale storage systems,” in *IEEE 29th Symposium on Mass Storage Systems and Technologies (MSST)*, Long Beach, USA, 2013, pp. 1–5. doi: 10.1109/MSST.2013.6558427.
- [36] S. Sohrabi, A. Tang, I. Moser, and A. Aleti, “Adaptive virtual machine migration mechanism for energy efficiency,” in *Proc. 5th International Workshop on Green and Sustainable Software*, Austin, USA, 2016, pp. 8–14. doi: 10.1145/2896967.2896969.
- [37] Z. Li, C. Yan, X. Yu, and N. Yu, “Bayesian network-based virtual machines consolidation method,” *Future Generation Computer Systems*, vol. 69, pp. 75–87, 2017. doi: 10.1016/j.future.2016.03.002.
- [38] G. K. Shyam and S. S. Manvi, “Virtual resource prediction in cloud environment: a bayesian approach,” *Journal of Network and Computer Applications*, vol. 65, pp. 144–154, Apr. 2016.
- [39] S. S. Masoumzadeh and H. Hlavacs, “Integrating VM selection criteria in distributed dynamic VM consolidation using fuzzy q-learning,” in *IEEE International Conference on Network and Service Management (CNSM)*, Zurich, Switzerland, 2013, pp. 332–338. doi: 10.1109/CNSM.2013.6727854.
- [40] Y. Jin, Y. Wen, and H. Hu, “Minimizing monetary cost via cloud clone migration in multi-screen cloud social tv system,” in *IEEE Global Communications Conference (GLOBECOM)*, Atlanta, USA, 2013, pp. 1747–1752. doi: 10.1109/GLOCOM.2013.6831326.
- [41] Y. Jin, Y. Wen, H. Hu, and M. Montpetit, “Reducing operational costs in cloud social TV: an opportunity for cloud cloning,” *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1739–1751, Oct. 2014. doi: 10.1109/TMM.2014.2329370.
- [42] N. Liu, Z. Li, J. Xu, et al. (2017, Mar. 13). A hierarchical framework of cloud resource allocation and power management using deep reinforcement learning [Online]. Available: <https://arxiv.org/abs/1703.04221>
- [43] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, “I tube, you tube, everybody tubes: analyzing the world’s largest user generated content video system,” in *Proc. 7th ACM SIGCOMM Conference on Internet Measurement*, San Diego, USA, 2007, pp. 1–14. doi: 10.1145/1298306.1298309.
- [44] G. Szabo and B. A. Huberman, “Predicting the popularity of online content,” *Communications of the ACM*, vol. 53, no. 8, pp. 80–88, 2010.
- [45] Y. Borghol, S. Ardon, N. Carlsson, D. Eager, and A. Mahanti, “The untold story of the clones: content-agnostic factors that impact youtube video popularity,” in *Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Beijing, China, 2012, pp. 1186–1194. doi: 10.1145/2339530.2339717.
- [46] W. Chu and S.-T. Park, “Personalized recommendation on dynamic content using predictive bilinear models,” in *Proc. 18th International Conference on World Wide Web*, Madrid, Spain, 2009, pp. 691–700. doi: 10.1145/1526709.1526802.
- [47] M. Ahmed, S. Spagna, F. Huici, and S. Niccolini, “A peek into the future: Predicting the evolution of popularity in user generated content,” in *Proc. Sixth ACM International Conference on Web search and Data Mining*, Rome, Italy, 2013, pp. 607–616. doi: 10.1145/2433396.2433473.
- [48] X. Ma, H. Wang, H. Li, J. Liu, and H. Jiang, “Exploring sharing patterns for video recommendation on youtube-like social media,” *Multimedia Systems*, vol. 20, no. 6, pp. 675–691, 2014. doi: 10.1007/s00530-013-0309-1.
- [49] P. Covington, J. Adams, and E. Sargin, “Deep neural networks for youtube recommendations,” in *Proc. 10th ACM Conference on Recommender Systems*, Boston, USA, 2016, pp. 191–198. doi: 10.1145/2959100.2959190.
- [50] G. Pallis and A. Vakali, “Insight and perspectives for content delivery networks,” *Communications of the ACM*, vol. 49, no. 1, pp. 101–106, 2006.
- [51] P. Venketesh, R. Venkatesan, and L. Arunprakash, “Semantic web prefetching scheme using naïve bayes classifier,” *International Journal of Computer Science and Applications*, vol. 7, no. 1, pp. 66–78, 2010.
- [52] W. Ali, S. M. Shamsuddin, and A. S. Ismail, “Intelligent naïve bayes-based approaches for web proxy caching,” *Knowledge-Based Systems*, vol. 31, pp. 162–175, 2012.
- [53] T. Yan, D. Chu, D. Ganesan, A. Kansal, and J. Liu, “Fast app launching for mobile devices using predictive user context,” in *Proc. 10th International Conference on Mobile Systems, Applications, and Services*, Low Wood Bay, UK, 2012, pp. 113–126. doi: 10.1145/2307636.2307648.
- [54] H. Hu, Y. Wen, T.-S. Chua, et al., “Joint content replication and request routing for social video distribution over cloud CDN: a community clustering method,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 7, pp. 1320–1333, Jul. 2016. doi: 10.1109/TCSVT.2015.2455712.
- [55] W. Hu, Y. Jin, Y. Wen, Z. Wang, and L. Sun, “Towards wi-fi AP-assisted content prefetching for on-demand TV series: a learning-based approach,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 7, pp.

When Machine Learning Meets Media Cloud: Architecture, Application and Outlook

JIN Yichao and WEN Yonggang

- 1665–1676, Jul. 2017. doi: 10.1109/TCSVT.2017.2684302.
- [56] M. Covell, M. Arjovsky, Y.-C. Lin, and A. Kokaram, "Optimizing transcoder quality targets using a neural network with an embedded bitrate model," *Electronic Imaging*, vol. 2016, no. 2, pp. 1–7, 2016. doi: 10.2352/ISSN.2470-1173.2016.2.VIPC-237.
- [57] P. P. Dash, A. Mishra, and A. Wong. (2016, Sept. 22). Deep quality: a deep no-reference quality assessment system [Online]. Available: <https://arxiv.org/abs/1609.07170v1>
- [58] H. Zhang, H. Hu, G. Gao, Y. Wen, and K. Guan. (2018, Apr. 10). Deepqoe: a unified framework for learning to predict QoE [Online]. Available: <https://arxiv.org/abs/1804.03481>
- [59] V. D' Alessandro, B. Park, L. Romano, *et al.*, "Scalable network traffic classification using distributed support vector machines," in *IEEE 8th International Conference on Cloud Computing (CLOUD)*, New York, USA, 2015, pp. 1008–1012. doi: 10.1109/CLOUD.2015.138.
- [60] L. Vu, C. T. Bui, and Q. U. Nguyen, "A deep learning based method for handling imbalanced problem in network traffic classification," in *Proc. Eighth International Symposium on Information and Communication Technology*, Nha Trang, Vietnam, 2017, pp. 333–339.
- [61] J. Ran, X. Kong, G. Lin, D. Yuan, and H. Hu, "A self-adaptive network traffic classification system with unknown flow detection," in *3rd IEEE International Conference on Computer and Communications (ICCC)*, Chengdu, China, 2017, pp. 1215–1220.
- [62] J. Hochst, L. Baumgartner, M. Hollick, and B. Freisleben, "Unsupervised traffic flow classification using a neural autoencoder," in *IEEE 42nd Conference on Local Computer Networks (LCN)*, Singapore, Singapore, 2017, pp. 523–526. doi: 10.1109/LCN.2017.57.
- [63] Y. Jin and Y. Wen, "When cloud media meets network function virtualization: challenges and applications," *IEEE MultiMedia*, vol. 24, no. 3, pp. 72–82, 2017. doi: 10.1109/MMUL.2017.3051519.
- [64] H. Jagadish, J. Gehrke, A. Labrinidis, *et al.*, "Big data and its technical challenges," *Communications of the ACM*, vol. 57, no. 7, pp. 86–94, 2014. doi: 10.1145/2611567.

Manuscript received: 2018-04-05

Biographies

JIN Yichao (yjin3@ntu.edu.sg) received the B.S and M.S degree from Nanjing University of Posts and Telecommunications (NUPT), China, in 2008 and 2011 respectively, and Ph.D degree from School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore, in 2016. His research interests are cloud computing and multimedia network.

WEN Yonggang (ygwen@ntu.edu.sg) is an associate professor with School of Computer Science and Engineering at Nanyang Technological University, Singapore. He received his PhD degree in Electrical Engineering and Computer Science (minor in Western Literature) from Massachusetts Institute of Technology (MIT), Cambridge, USA. Previously he has worked in Cisco to lead product development in content delivery network, which had a revenue impact of 3 Billion US dollars globally. Dr. Wen has published over 150 papers in top journals and prestigious conferences. His research interests include cloud computing, green data center, big data analytics, multimedia network and mobile computing.

Mechanism of Fast Data Retransmission in CU-DU Split Architecture of 5G NR

HUANG He, LIU Yang, LIU Zhuang, HAN Jiren, and GAO Yin

(ZTE R&D Center, Shanghai 201203, China)

Abstract

The 5G radio access network (RAN) architecture is supposed to be split into the central unit (CU) and the distributed unit (DU) in order to support more flexible transport networks and provide enhanced user experience. However, such functional split may also introduce some new technical issues. In this paper, we study the data fast retransmission issue introduced by this functional split in different scenarios and solutions are provided to handle this issue. With the fast data retransmission mechanism proposed in this paper, the retransmitted data packets could be identified and handled with high priority. In this way, the data delivery between the CU and DU in 5G RAN is assured.

Keywords

5G RAN; central unit; distributed unit; fast retransmission

1 Introduction

The 5G RAN architecture is supposed to be split into the central unit (CU) and the distributed unit (DU) in order to support various types of transport networks and multi-vendor requirement. The latency-tolerant network function resides in the CU entity and the latency-sensitive network function resides in the DU entity [1].

For the higher layer split (HLS) solution, there are eight possible CU-DU split options as shown in [2]. According to the agreements from the 3GPP RAN3#95bis meeting [3], the 3rd Generation Partnership Project (3GPP) has decided to select Option 2 (based on Packet Data Convergence Protocol (PDCP) and decentralized Radio Link Control (RLC)) as HLS solution for normative work in Release 15. As for the lower layer split (LLS) solution, the possible solutions can be found in [4] and the corresponding discussions are still going on. Additionally,

the gNB-CU can be further split into the control plane and user plane [5] in order to support more flexible data services. In this paper, we focus on HLS only.

The interface between CU and DU is called F1 interface [6]. One CU can connect to multiple DUs and one DU can support one or more cells. The general principles can be found in [7] and the application protocol for the F1 interface can be found in [8]. Considering the CU-DU split scenario, one newly raised issue is that the data transmission may suffer outage due to different reasons, such as handover and temporary Radio Link Failure (RLF). For example, when user equipment (UE) performs handover from the source DU to the target DU, the packets which are still buffered in the source DU will be lost. Additionally, the air interface between UE and DU may become unstable and the packets which are already in the air may not be confirmed by the UE. Therefore, the DU triggered fast data retransmission needs to be investigated.

The lost PDCP protocol data units (PDUs) need to be identified and retransmitted with high priority in order to facilitate the fast transmission for CU-DU split scenario. The network slicing based handover procedures and mobility management mechanisms are discussed in [9], but the lost data packets during the handover procedure is not mentioned. The Fog Radio Access Network architecture is introduced in [10] and it is claimed that the associated mobility and resource management mechanisms can reduce the signaling cost. However, the lost data packets scenario is not covered. In the following, we will first describe the typical scenarios for this issue and the corresponding solutions for fast retransmission of the lost PDCP PDUs are also provided. In the following, we denote CU/DU as gNB-CU/gNB-DU to make them aligned with the 3GPP specifications.

2 Scenario Description

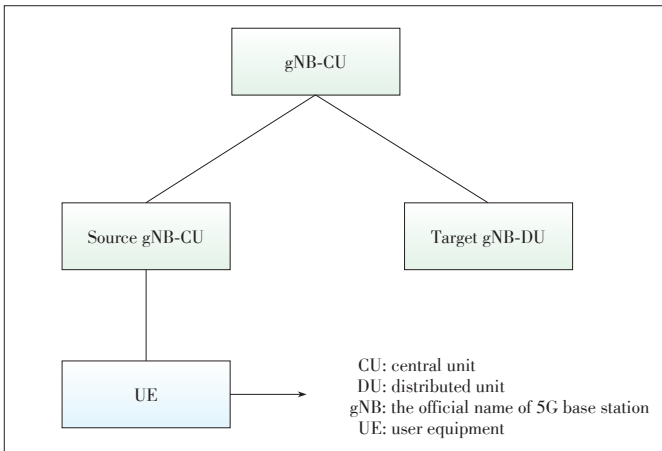
In this section, we will illustrate the PDCP PDU retransmission issue in three typical scenarios: single connectivity, multi-connectivity, and E-UTRAN-NR dual connectivity (EN-DC).

2.1 Single Connectivity Scenario

Single connectivity means that UE is connected to only one gNB-DU at a certain time instance. **Fig. 1** shows the downlink transmission of the intra-gNB-CU inter-gNB-DU handover scenario. When UE moves from the source gNB-DU to the target gNB-DU, one common situation is that there are still remaining data packets buffered in the source gNB-DU waiting for transmission. Since gNB-CU is unaware of the delivery status for these data packets, the current mechanism in LTE cannot assure the fast retransmission of these remaining packets in the source gNB-DU and these data packets will be lost.

2.2 Multi-Connectivity Scenario

The multi-connectivity indicates that UE is served by at



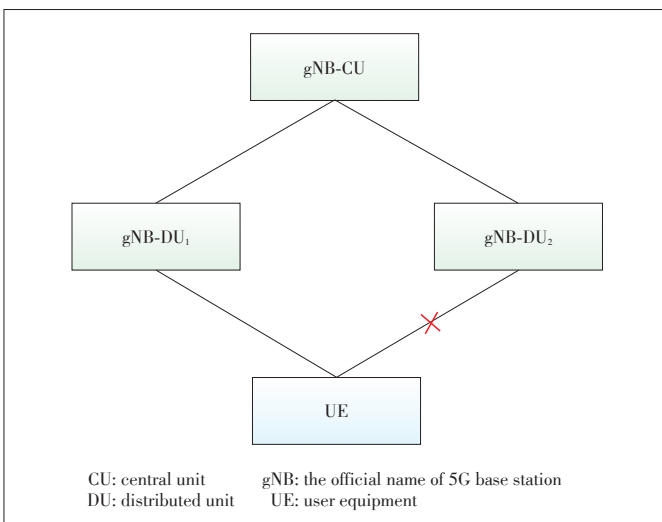
▲ Figure 1. Single connectivity scenario.

least two gNB-DUs simultaneously. This scenario focuses on the fast retransmission of data packets, during which UE is served by one of the gNB-DUs with a radio link that is subject to RLF. Such a scenario is typically encountered with radio links at high frequencies.

Fig. 2 shows the scenario where UE connects to the gNB-CU via two gNB-DUs simultaneously. The data packets are delivered from gNB-CU to the UE via gNB-DU 1 and gNB-DU 2. At a certain time point, the radio link of gNB-DU 2 encounters RLF and becomes unavailable, thus all the data packets that has not been successfully delivered to the UE via gNB-DU 2 should be retransmitted to the UE via gNB-DU 1. Once the radio link of gNB-DU 2 becomes available again, it is expected that data traffic can be centrally forwarded to the gNB-DU 2 hosting the previously broken link. The traffic transmission in gNB-DU 2 will resume as it was before the RLF.

2.3 EN-DC Scenario

Fast data retransmission in the EN-DC scenario can be con-



▲ Figure 2. Multi-connectivity scenario.

sidered as another typical case, which indicates that UE may be configured to utilize radio resources provided by two distinct schedulers in two different nodes connected via non-ideal backhaul, one providing E-UTRAN access (eNB) and the other providing NR access (gNB).

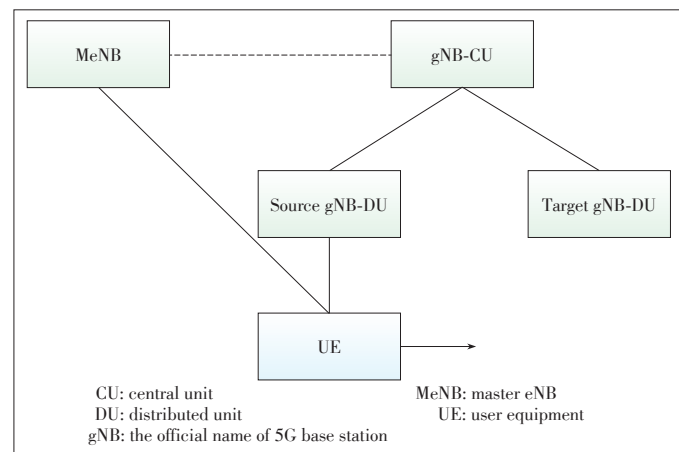
As shown in Fig. 3, we take the inter-gNB-DU mobility using Master Cell Group (MCG) Signalling Radio Bearer (SRB) as an example, i.e., UE moves from one gNB-DU to another gNB-DU within the same gNB-CU when MCG SRB is available during EN-DC operation.

3 Solutions

In this section, we will demonstrate that the gNB with CU-DU architecture can be enhanced to enable fast retransmission of PDCP PDUs in a centralized way. The enhanced fast retransmission solutions for above scenarios will be illustrated independently.

3.1 Solution for Single Connectivity Scenario

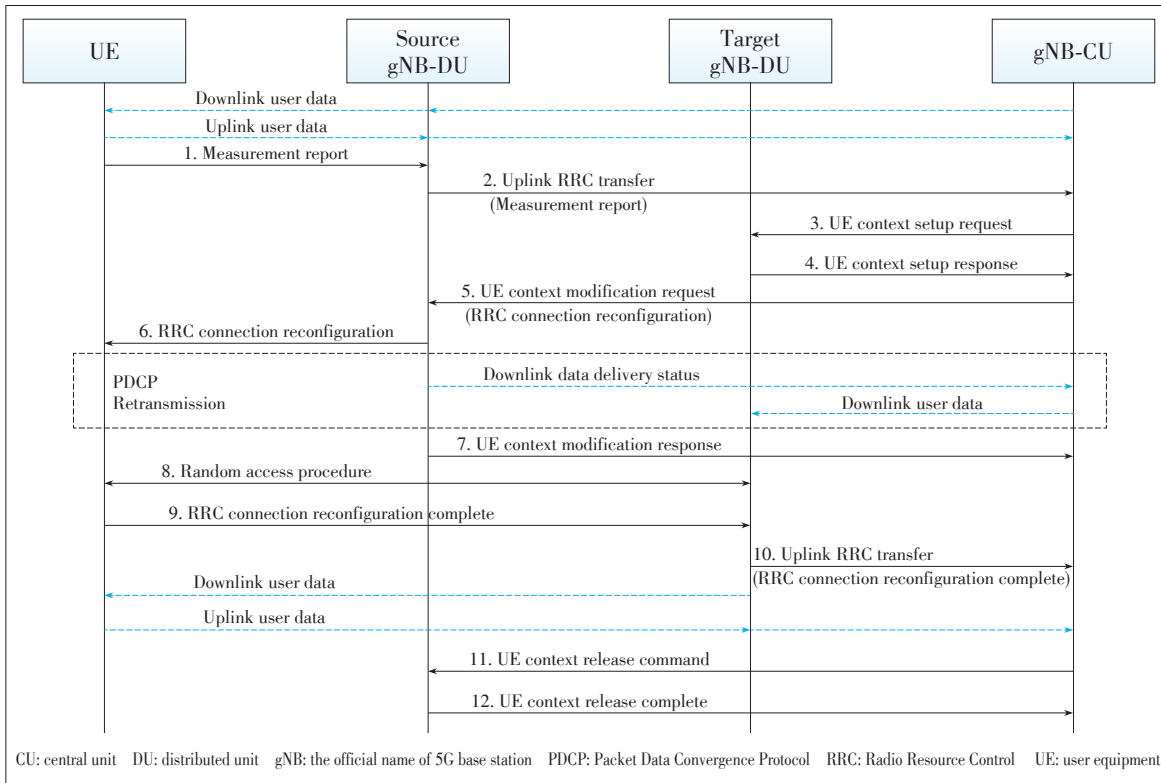
As shown in Fig. 4, the Intra-gNB-DU mobility procedure [5] can be used to demonstrate the retransmission mechanism for single connectivity scenario. It can be seen that at Steps 5 and 6, the gNB-CU sends a UE Context Modification Request message to the source gNB-DU, which includes a generated Radio Resource Control (RRC) Connection Reconfiguration message and indicates to stop the data transmission for UE. The source gNB-DU could respond with a Downlink Data Delivery Status (DDDS) frame via F1-U to inform the gNB-CU about the highest PDCP PDU SN successfully delivered in sequence to UE and the highest NR PDCP PDU sequence number transmitted to the lower layers, which is the key step for the fast retransmission of lost PDCP PDUs (shown in the dashed rectangle in Fig. 4). Based on this information, the gNB-CU is able to identify the unsuccessfully transmitted PDCP PDUs in the source gNB-DU side. Those unsuccessfully delivered PDCP PDUs will be sent from the gNB-CU to the target



▲ Figure 3. E-UTRAN new radio-dual connectivity (EN-DC) scenario.

Mechanism of Fast Data Retransmission in CU-DU Split Architecture of 5G NR

HUANG He, LIU Yang, LIU Zhuang, HAN Jiren, and GAO Yin



◀Figure 4. Procedure of fast data retransmission in single connectivity scenario (Based on Figure 8.2.1.1-1 shown in TS38.401 [6]).

gNB-DU for retransmission.

3.2 Solution for Multi-Connectivity Scenario

For the multi-connectivity scenario, the centralized retransmission procedure in intra-gNB-CU [5] is shown in Fig. 5. This mechanism allows to perform the retransmission of PDCP PDUs that are not delivered by a gNB-DU (gNB-DU 1) because the corresponding radio link toward the UE is subject to outage.

As shown in Fig. 5, UE is receiving data from gNB-DU 1 and gNB-DU 2 simultaneously. At a certain time, gNB-DU 1 realizes that the radio link towards UE is experiencing outage and sends the “Radio Link Outage” notification to the gNB-CU over the F1-U, which is a part of the DDDS frame of the concerned data radio bearer. The message includes the highest PDCP PDU Sequence Number (SN) successfully delivered in sequence to UE and the highest NR PDCP PDU sequence number transmitted to the lower layers in gNB-DU 1. Based on the received information, gNB-CU will retransmit the potentially undelivered PDCP PDUs and the new PDUs via gNB-DU 2. If gNB-DU 1 realizes that the radio link is back in normal operation, it may send a “Radio Link Resume” notification to inform the gNB-CU that the radio link can be used again. Then the gNB-CU may start sending traffic via gNB-DU 1 again.

3.3 EN-DC Scenario

This procedure is used for the case where the UE moves from one gNB-DU to another gNB-DU within the same gNB-

CU when MCG SRB is available during the EN-DC operation. Fig. 6 shows the inter-gNB-DU mobility procedure using MCG SRB in EN-DC.

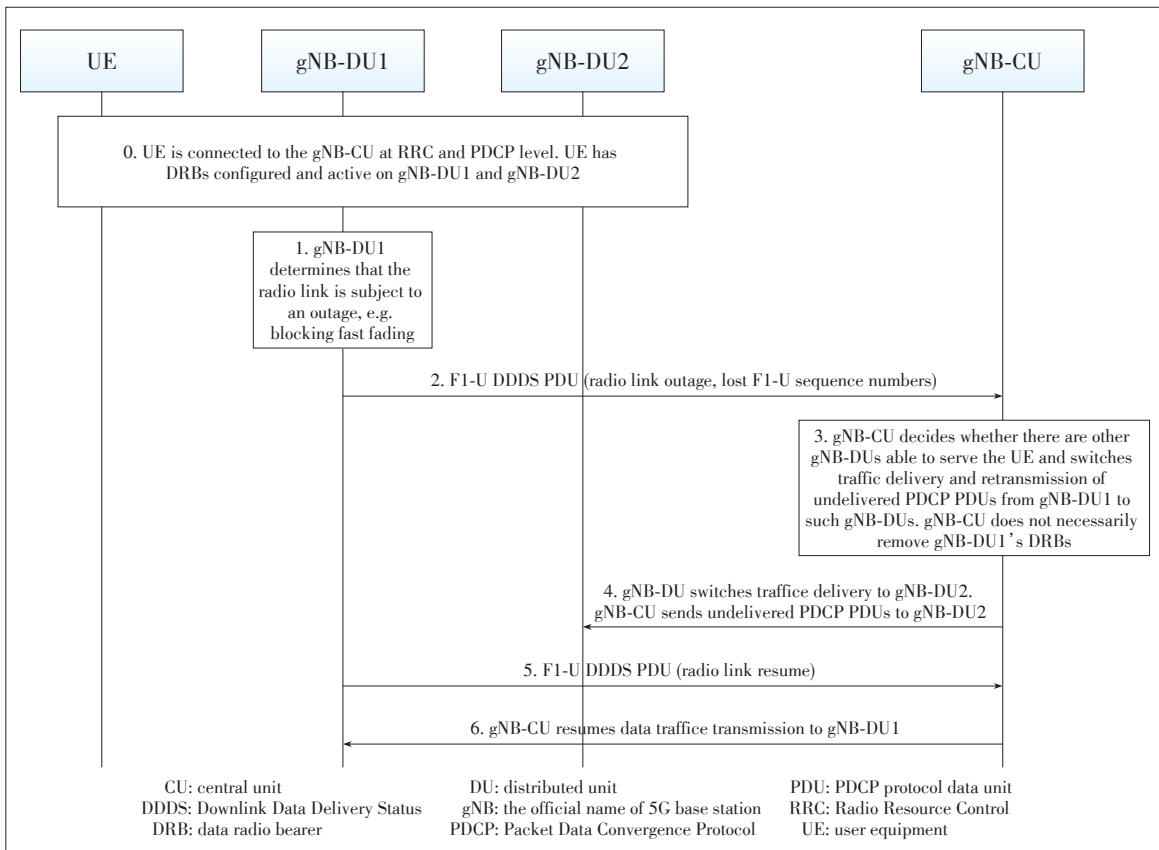
As shown in Fig. 6, UE sends a Measurement Report message to master eNB (MeNB). Then MeNB performs the SgNB Modification procedure and the UE context in the target gNB-DU is created. In the following, the gNB-CU sends a UE Context Modification Request message to the source gNB-DU indicating to stop the data transmission to UE. Correspondingly, the source gNB-DU replies a Downlink Data Delivery Status frame to inform the gNB-CU about the unsuccessfully transmitted downlink data to UE or lower layers. After UE performs RRC Reconfiguration procedure with MeNB, the MeNB sends an SgNB Reconfiguration Complete message to the gNB-CU. Thus the unsuccessfully transmitted PDCP PDUs in the source gNB-DU are sent from the gNB-CU to the target gNB-DU. After the random access procedure is performed between UE and the target gNB-DU, the downlink packets will be sent to the UE.

3.4 Handling Retransmitted Packets with High Priority

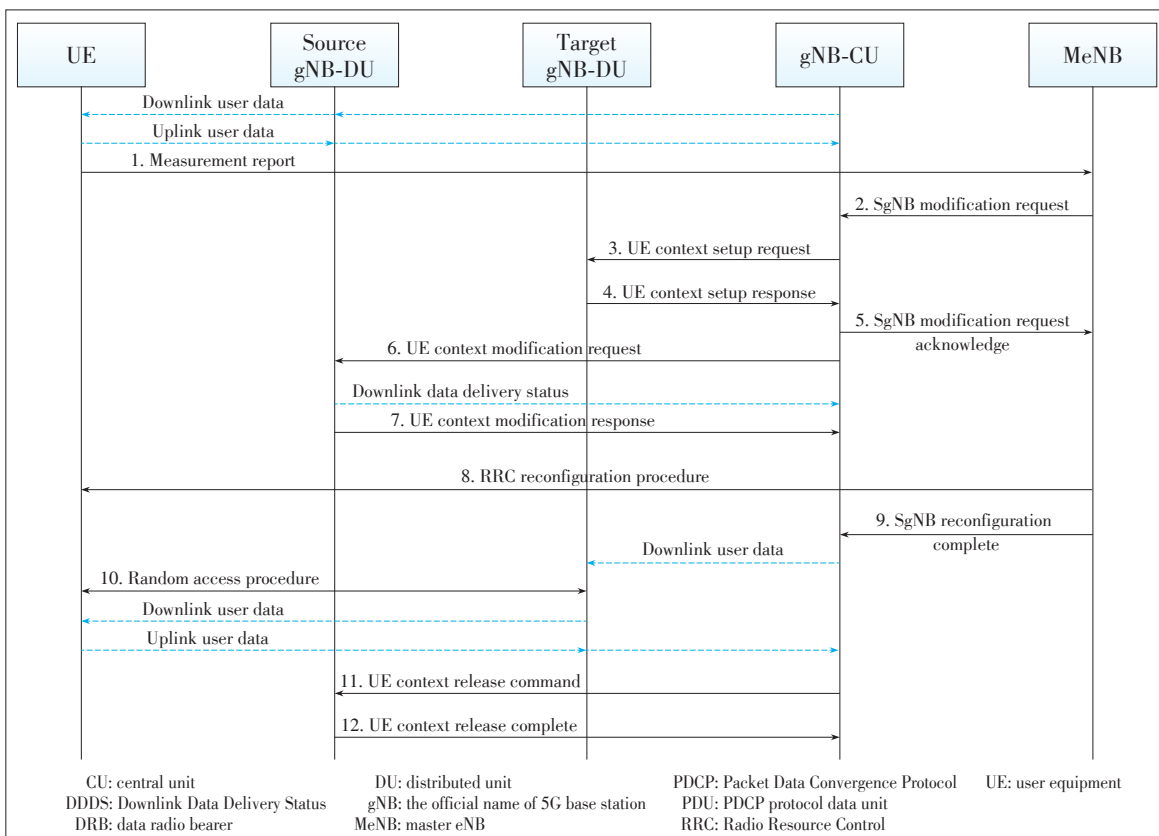
The retransmission mechanism still has a problem. The only way for the gNB-CU to acquire the packet delivery status to UE is DDDS [11]. However, DDDS only reports “the highest delivered/transmitted PDCP SN” to gNB-CU, thus the gNB-CU cannot know whether the retransmitted packets are delivered or not. In case the PDCP SN of retransmitted packets is lower than the one already buffered in the gNB-DU, the gNB-CU

Mechanism of Fast Data Retransmission in CU-DU Split Architecture of 5G NR

HUANG He, LIU Yang, LIU Zhuang, HAN Jiren, and GAO Yin



◀Figure 5. Procedure for fast data retransmission in the multi-connectivity scenario (Based on Figure 8.3.1-1 in TS38.401 [6]).



◀Figure 6. Procedure of fast data retransmission in the EN-DC scenario (Based on Figure 8.2.2.1-1 in TS38.401 [6]).

Mechanism of Fast Data Retransmission in CU-DU Split Architecture of 5G NR

HUANG He, LIU Yang, LIU Zhuang, HAN Jiren, and GAO Yin

may clean the buffer and the retransmitted packets could not be delivered.

In order to facilitate the data transmission procedure, retransmitted data packets need to be identified and handled with high priority at the gNB-DU side. One straightforward way is to introduce an indication in the retransmitted packet. To be specific, a “retransmission flag” is introduced in the spare bit of DL_USER_DATA. The detailed illustration for the DL_USER_DATA frame structure can be found in [12]. When the gNB-CU performs retransmission, it indicates to the gNB-DU whether the packets are retransmitted or not by the value of this flag, which can help the gNB-DU to identify retransmitted packets easily. At the gNB-DU side, once the retransmitted data packets are identified, the gNB-DU will handle the corresponding data frame separately. For example, besides the normal transmission queue, gNB-DU will additionally provide a retransmission queue for the retransmitted packets in order to guarantee the scheduling of retransmitted data with high priority.

4 Conclusions

In this paper, we study the fast data retransmission issue introduced by the functional split of the 5G RAN architecture. Three typical scenarios (the single connectivity scenario, multi-connectivity scenario, and EN-DC scenario) are respectively described in order to illustrate the PDCP PDU data retransmission issue. We also provide the solutions targeting the above scenarios, which have already been agreed and captured by the 3GPP specifications [6]. With the data retransmission mechanism described in this paper, the retransmitted data packets could be identified and handled with high priority. Thus the data delivery between the gNB-CU and the gNB-DU in 5G RAN is assured.

References

- [1] NTT DOCOMO, INC., “Revised WID on new radio access technology,” 3GPP RP-172109, 2017.
- [2] *Study on New Radio Access Technology: Radio Access Architecture and Interfaces*, 3GPP TR38.801, 2016.
- [3] I. Toufik, “Report of 3GPP RAN WG3 #95bis,” R3-171412, Spokane, USA, Apr. 2017.

- [4] *Study of CU-DU Low Layer Split for NR (Release 15)*, 3GPP TS 38.816, 2018.
- [5] *Study of Separation of NR Control Plane (CP) and User Plane (UP) for Split Option 2 (Release 15)*, 3GPP TS 38.806, 2018.
- [6] *NG-RAN, Architecture Description (Release 15)*, 3GPP TS 38.401, 2018.
- [7] *NG-RAN, F1 General Aspects and Principles (Release 15)*, 3GPP TS 38.470, 2018.
- [8] *NG-RAN, F1 Application Protocol (Release 15)*, 3GPP TS 38.473, 2018.
- [9] H. J. Zhang, N. Liu, X. L. Chu, et al., “Network slicing based 5G and future mobile networks: mobility, resource management, and challenges,” *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, Aug. 2017. doi: 10.1109/MCOM.2017.1600940.
- [10] H. J. Zhang, Y. Qiu, X. L. Chu, K. P. Long, and V. C. M. Leung, “Fog radio access networks: mobility management, interference mitigation and resource optimization,” *IEEE Wireless Communications*, vol. 24, no. 6, pp. 120–127 Dec. 2017. doi: 10.1109/MWC.2017.1700007.
- [11] *NG-RAN, NR User Plane Protocol (Release 15)*, 3GPP TS38.425, 2018.
- [12] ZTE Corporation, “R3-180131 further discussion on data retransmission indication,” 3GPP R3-180610, 2018.

Manuscript received: 2018-02-05

Biographies

HUANG He (huang.he4@zte.com.cn) received the bachelor’s degree in computer science and technology from Shanghai Jiao Tong University, China in 2004. He is currently the chief engineer of wireless innovation laboratory of ZTE Corporation and leads the research and standardization work on 5G RAN. He has filed more than 60 patents. He was the rapporteurs of multiple CCSA/3GPP SIs/WIs and the editors of the related protocols.

LIU Yang (liu.yang31@zte.com.cn) received the Ph.D. degree in communication and information systems from Beijing University of Posts and Telecommunications (BUPT), China in 2016. He was a visiting scholar at Department of Electrical and Computer Engineering of North Carolina State University, USA from 2013 to 2015. He is currently a 5G research engineer at ZTE R&D center, Shanghai. His research interests include 5G wireless communications and signal processing.

LIU Zhuang (liu.zhuang2@zte.com.cn) received the master’s degree in computer science from Xidian University, China in 2003. He is currently a senior 5G research engineer at ZTE R&D center, Shanghai. His research interests include 5G wireless communications and signal processing.

HAN Jiren (han.jiren@zte.com.cn) received the master’s degree in wireless communication systems from University of Sheffield, UK in 2016. He is an advanced research engineer at the Algorithm Department, ZTE Corporation. His research interests include 5G wireless communications and signal processing.

GAO Yin (gao.yin1@zte.com.cn) received the master’s degree in circuit and system from Xidian University, China in 2005. Since 2005 she has been with the research center of ZTE Corporation and been engaged in the study of 3G/4G/5G technology. She has authored or co-authored about hundreds of proposals for 3GPP meetings and journal papers in wireless communications and has filed more than 100 patents. She was the rapporteurs of multiple 3GPP WIs. From August 2017, she has been elected as the vice chairman of 3GPP RAN3.

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu¹, LIU Yiyi¹, LI Hui¹, and WANG Wei²

(1. Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. Government & Enterprise Communications Institute, ZTE Corporation, Nanjing 210012, China)

Abstract

Since Dalvik Executable (DEX) files are prone to be reversed to the Java source code using some decompiling tools, how to protect the DEX files from attackers becomes an important research issue. The traditional way to protect the DEX files from reverse engineering is to encrypt the entire DEX file, but after the complete plain code has been loaded into the memory while the application is running, the attackers can retrieve the code by using memory dump attack. This paper presents a novel DEX protection scheme to withstand memory dump attack on the Android platform with the name of DexDefender, which adopts the dynamic class-restoration method to ensure that the complete plain DEX data not appear in the memory while the application is being loaded into the memory. Experimental results show that the proposed scheme can protect the DEX files from both reverse engineering and memory dump attacks with an acceptable performance.

Keywords

Android; DEX; memory dump; reverse engineering

1 Introduction

Although the Android platform employs multi-level security mechanisms, the adoption of Java language in most of Android applications makes the applications on the platform prone to be decompiled and vulnerable to reverse engineering. An attacker can

obtain the Java source code by decompiling an application's Android Package (APK) file, and then repackage them as another APK, which may cause a serious problem with the copyright protection of the software. For example, the game developer of "Dead Trigger", Madfinger, was forced to provide software for free because of software piracy [1], which has brought huge loss. More seriously, attackers can also insert malicious codes into the application that has been cracked [2], and then it will be disguised as a legitimate application to steal user's sensitive information. This not only violates the developer's copyright, but also harms the interests and personal privacy of users.

In order to prevent the applications from being decompiled and reassembled, various methods have been proposed. In 2012, Moon et al. designed a software protection system based on symmetric and asymmetric cryptography [3], in which the users buy applications from a specific application market. The purchased applications use users' public key to encrypt, the users can decrypt applications with the private key, so that only the legitimate users can run the applications. However, attackers can also obtain the applications by copying its codes from the path of mobiles: /data/App.

In the same year, Jeong et al. proposed a mechanism for anti-piracy based on component separation and dynamic loading [4], in which the applications are divided into main programs and plugins. Users install the main program, and then the main program downloads the plugins from the web before the system reminds users to pay. These plugins are protected by encryption, only paid authorized users can decrypt correctly and the decrypted plugins are stored into the phone's security area. However, malicious users can also get root privilege to copy the code of plugins.

All of the above mentioned methods provide ideas for software protection on the Android platform. However, they have their own shortcomings. One possible way to protect the applications is to always keep the key parts of the applications confidential, only decrypt the key parts in memory when it is running, and clear the memory after use, so that the decryption process and calling process will be difficult to track. In this paper, we define the Dalvik Executable (DEX) file as the key part of applications because it contains main information of the applications' source codes. The DEX file is a kind of Dalvik binary byte code file generated by the java source code and can directly run on the Dalvik virtual machine.

The concept of code obfuscation proposed by Collberg et al [5] can be used to protect DEX files by making data promiscuous or obfuscating the control flow so that the code and program become obscure and complex, which can protect the application from being reversed and can prevent the software from direct static analysis. However, the obfuscated executable code can still be deobfuscated by the general approach proposed in [6].

Another way to protect the applications is to encrypt the

This work was supported by ZTE Industry-Academia-Research Cooperation Funds.

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

DEX files and then hide them in the applications, which has been applied by Bangbang [7], 360 [8], and Dong et al [9]. They encrypt the DEX files of the source APK with encryption algorithms and replace them with a fake DEX file prepared previously. When the program is being executed, the fake DEX file will be run first, and then the fake DEX file can lead the original DEX file to run. Since all these methods protect the DEX files completely, the plaintext of the DEX data will appear in the memory in the run time, which makes it possible for attackers to dump the DEX file from the memory by using Interactive Disassembler (IDA), ZjDroid, Drizzle Dumper or other tools.

To solve this problem, Fan et al. proposed a method to prevent Android App repackaging based on code splitting [10], in which the DEX files are divided into multiple fragments in accordance with the DEX file’s format, making the application’s executable code be fragmented in its entire life cycle in the memory. Since each DEX file fragment in this approach has a certain feature for attackers to identify, they can get the complete DEX file by dumping and combining from the memory.

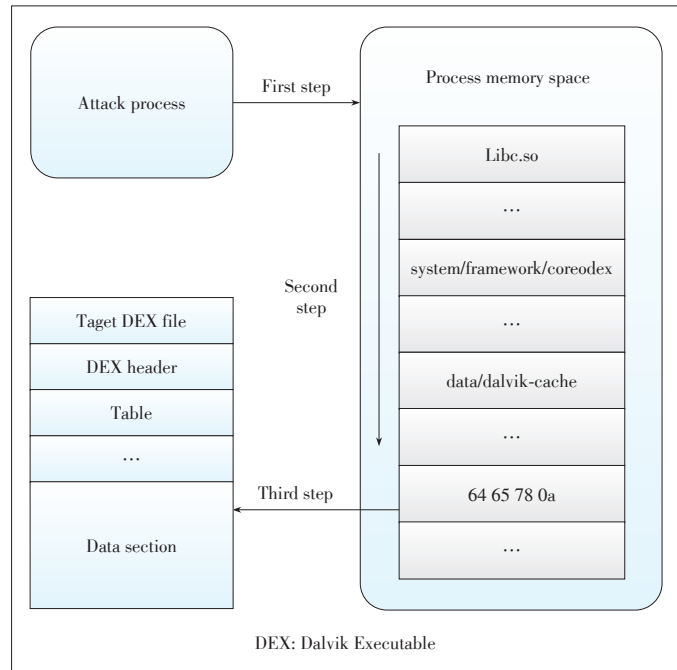
In order to prevent the direct copy of DEX files from memory, this paper presents a scheme named DexDefender to withstand memory dump attacks. It extracts the code fields of classes in the DEX files and then restores each class dynamically into the memory when the program is running. The snippets of the extracted code have no features to be identified by attackers so that it can effectively prevent attackers from cracking the applications by dumping DEX data from the memory.

The rest of this paper is organized as follows. Section 2 presents a memory dump attack approach to obtain DEX data on the Android platform. The third section describes the proposed protection scheme which uses the dynamic class - restoration method to avoid the complete plain DEX data from appearing in the memory. In Section 4 the proposed scheme is analyzed and evaluated. Finally, we conclude our work in Section 5.

2 Memory Dump Attack

The traditional way to enhance the security of DEX files is to protect the files completely. No matter how to hide the DEX file, even if the DEX file is encrypted, the whole plain DEX data must exist in the memory while an application is running. Attackers can dump the DEX data from memory through such tools as IDA, ZjDroid, and Drizzle Dumper. This kind of attack is called DEX memory dump attack.

Such an attack includes three steps as shown in Fig. 1. In the first step, when an application reinforced by an existing approach is running, the attacker attaches its process to the application’s process. In the second step, the attacker locates the DEX in the memory. The DEX file usually has a consistent and specific format. The DEX file header records some basic information of the DEX file and has a constant length of 0x70 bytes. The first 8 bytes of the file header are named magic field



▲ Figure 1. DEX memory dump attack steps.

that is used to identify a valid DEX file of a specific value 64 65 78 0a 30 33 35 00. An attacker can search for those 8 bytes in memory, and if the magic field is found in a virtual memory area, the attacker can locate the DEX successfully. Finally the attacker can dump the complete plain DEX data from the memory.

After the attacker obtains the application’s DEX file, the application can be cracked so that the attacker can steal the program logic, insert malicious program or repackage the APK.

3 The Proposed Solution

3.1 Overview

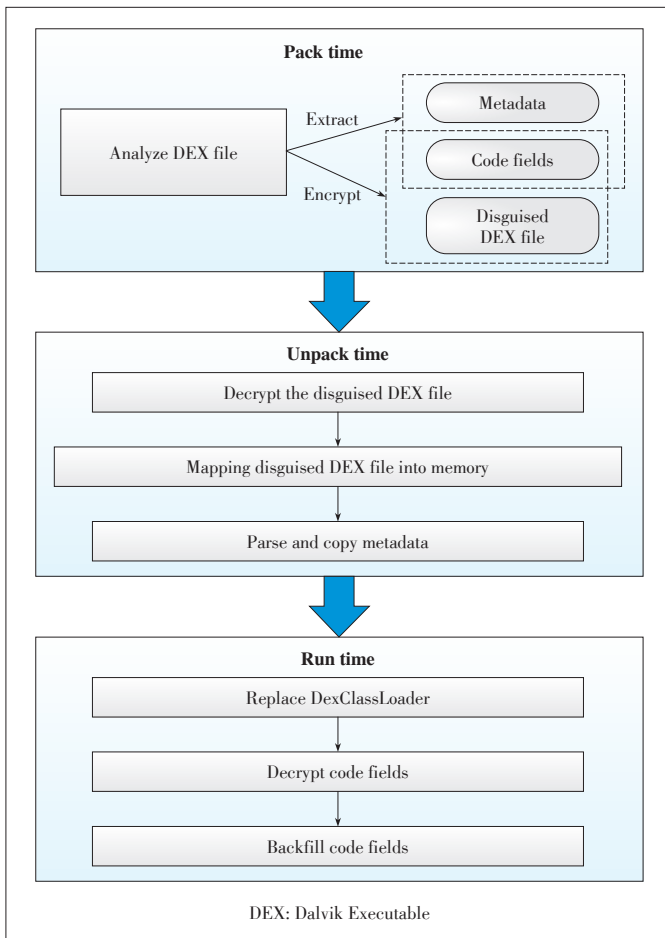
Because the traditional reinforcement technologies cannot resist the memory dump threat as described in Section 2, this paper presents a DEX protection scheme to withstand memory dump attacks. The purpose of this scheme is to ensure that the complete plain DEX data not appear in the memory when an application is being loaded into the memory. This can better protect the DEX file from being completely dumped from the memory and reduce the possibility of crack applications.

Fig. 2 shows the overall framework of the proposed scheme, which is divided into three phases: pack time, unpack time and run time.

In the pack time, the DEX parser will first analyze the DEX file of APK, extract the code fields of DEX file and encrypt it. Then the code fields’ metadata (the offset and length of code fields) is saved, the code fields of the original DEX file (disguised DEX file) is cleared and replaced with the fake DEX

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei



▲ Figure 2. Framework of DexDefender.

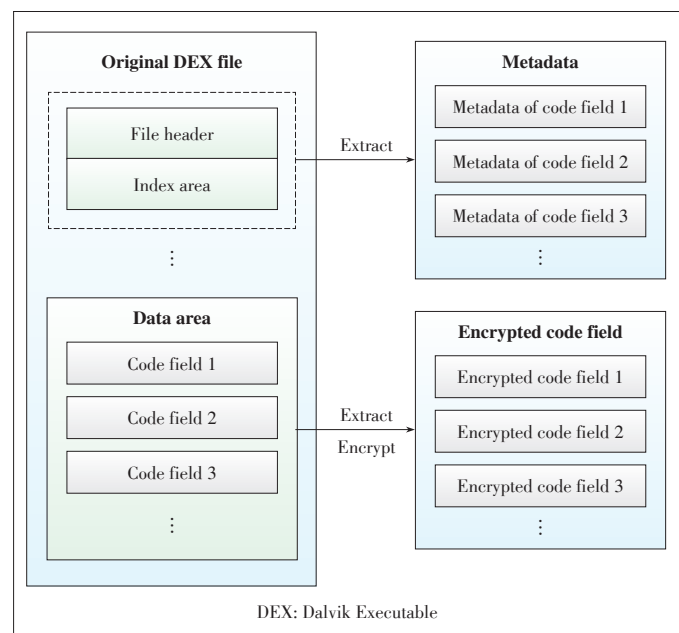
file. Thirdly, the disguised DEX file, metadata, and the encrypted code fields are used as input during the unpack time. The unpack time process is mainly responsible for loading the disguised DEX file from the fake DEX file. Because the disguised DEX file's code fields have been cleared, the complete plain DEX will not appear both in the file system and the memory. In order to keep the original APK running normally, a code field will be decrypted according to the class name that belongs to and backfilled to restore the disguised DEX during the running time. The process of analyzing the original DEX file and restoring the class dynamically in memory is described in Sections 3.2 and 3.3 of this paper.

3.2 Analysis of Original DEX File

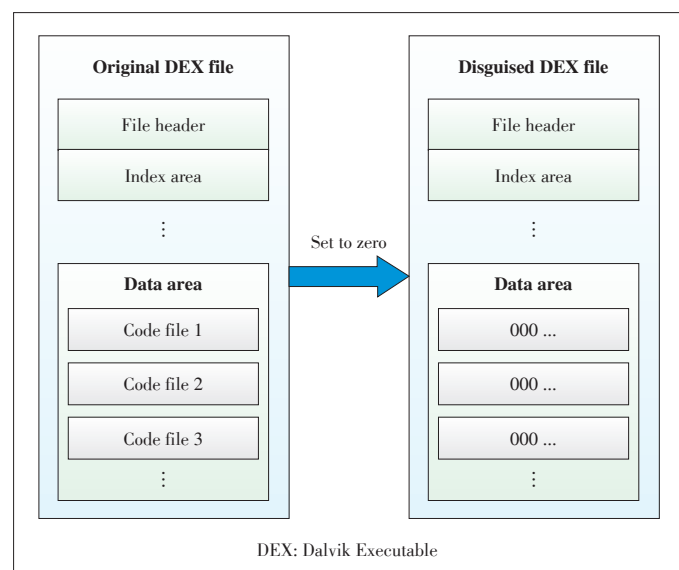
DEX file structure mainly contains three parts: the DEX file header, index area, and data area. Code fields that contain primary information of the applications are in the data area. The offset and length of the code fields in the DEX file are called code fields' metadata.

The process of extracting code fields and metadata is shown in Fig. 3. Because the code fields are not stored in the data area continuously, it is necessary to extract the metadata of each

code field, i.e. extract the offset and length of the code field in the original DEX file according to the file header and index area. The metadata will be used to first restore code fields in order to ensure that APK run successfully, and then extract and encrypt each code field. The code fields will be decrypted in the memory to restore the DEX while the application is being loaded into the memory. Finally, the values of code fields of the original DEX need to be changed to zero as shown in Fig. 4. With the metadata and encrypted code fields stored separately, it is hard for an attacker to restore the whole DEX file if it only obtained one of them. In addition, since the code fields do not have fixed identifiable features, it is difficult for attack-



▲ Figure 3. Process of extracting code fields and metadata.



▲ Figure 4. Process of setting code fields as zero.

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

ers to locate all real code fields in the memory.

Because the values of code fields of disguised DEX file are zero, even if the attacker can find and dump the disguised plain DEX data from the memory, he cannot get any information about the application.

3.3 Dynamic Class Restoration

When the application is being loaded into the memory, Android system will create a default class DexClassLoader for the application to load class, in which the values of code fields in the disguised DEX are zero so that the DexClassLoader cannot find the real class. Therefore, we need to use our customized DexClassLoader to replace the default DexClassLoader. First, the system’s default DexClassLoader is inherited. Then, the findClass method is rewritten. In the findClass method, the dynamic class restoration process is implemented. When the program needs to load a class of the original DEX, the customized DexClassLoader will first index the name of the class, find the corresponding encrypted code fields based on the metadata extracted before, and then decrypt and backfill it to the correct position of the DEX in the memory. The process of class restoration is shown in Fig. 5.

By this way, the original APK can run correctly and the memory will be cleared after running the APK.

4 Analysis of DexDefender

DexDefender has been implemented on Android 4.4.4, Android 5.1.1 and Android 6.0. Android 4 uses the Dalvik mode, in which DEX is optimized to Optimized Dalvik Executable (ODEX). Android 5 or Android 6 uses Android Runtime (ART) mode, in which DEX is optimized to Optimized Android Runtime Machine Code (OAT). The specific implementation of

DexDefender in Dalvik and ART modes is slightly different, but the structures of DEX are the same in ODEX and OAT. Therefore, the customized DexClassLoader can be used to load the DEX in both Dalvik and ART modes, which makes the number of codes required to be modified in different modes minimal.

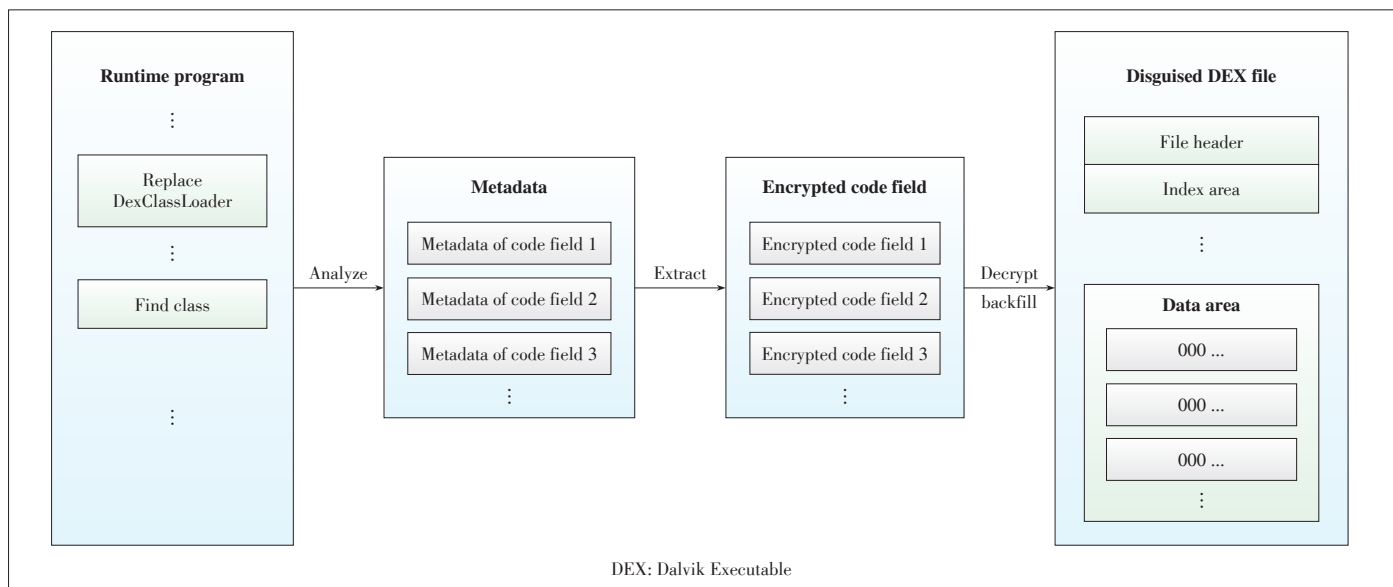
DexDefender adopts the symmetric encryption algorithm of Cipher Block Chaining (CBC) mode. This section will analyze and evaluate the effectiveness and performance of the proposed scheme through the experiment.

4.1 Analysis of Effectiveness

The purpose of designing the approach to withstand the memory dump attack is to avoid loading the whole DEX file into the memory at once. In the proposed scheme, the code fields which contain the most important information are not stored in the DEX file. When the program needs to run and load a class, corresponding code fields will be located through the previously saved metadata and be decrypted to restore the DEX. By this way, only disguised DEX and DEX fragments (code fields) are in the memory and this make it difficult to obtain DEX files at once.

Even if the attacker can locate disguised DEX and dump it from the memory according to the characteristics of the DEX file, the values of DEX file’s code fields are zero and attackers cannot get any information about the class of the application. As described in Section 1 of this paper, the attacker could not crack the application even if all the attack steps are completed.

If an attacker wants to retrieve the complete DEX file, it must analyze the characteristics of each field code, and then find and dump all the code fields from the memory. In addition, the attacker would also require a lot of time to restore the DEX file and this process is prone to making mistakes, which



▲ Figure 5. Process of dynamic class restoration.

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

greatly increases the cost of the attack.

To prove that the proposed scheme can prevent the complete plain DEX data from being dumped from the memory, we implemented the attack scenario as described in section 2, using IDA pro for dynamic debug attacks. We installed and ran the reinforced APK, attached to the program's process with IDA pro, found the position of DEX in the memory is 0x74f99028, as shown in Fig. 7.

The location of the code field corresponded to the class appstore. Appstore_codec.CharEncoding was 0x7500CD3C. The values of code fields in the corresponding location were changed to zero, as shown in Fig. 7. The length of this code fields was 8 bytes.

It can be seen from the corresponding location in the original DEX file as shown in Fig. 8 that the values of code fields can be successfully changed to zero. The code fields will be restored at the corresponding location in the memory when the program needs to get the class.

In summary, the proposed scheme can ensure that from loading to running time of the application, the complete plain DEX data are not appear in the memory, which makes the cracking more difficult and provides defense against memory dump attacks.

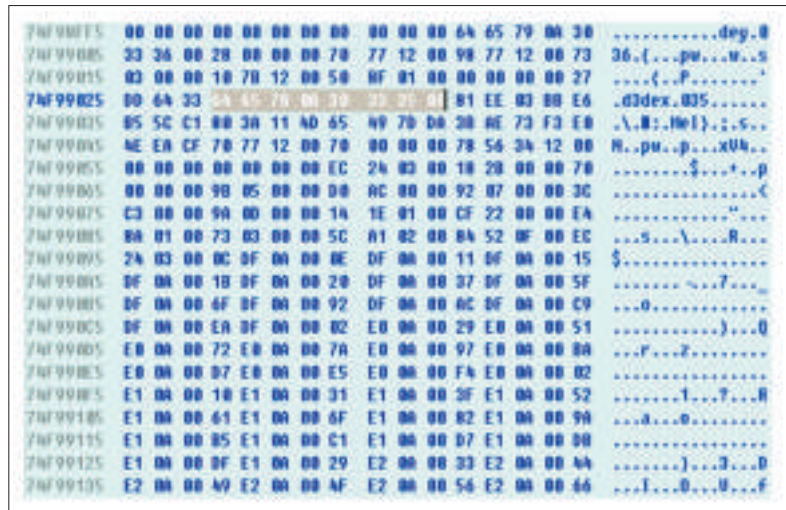
4.2 Analysis of Performance

20 popular applications were selected and tested on an Intel core i5 computer. Both space consumption and time consumption were measured using LG Nexus5. Experimental results show that the increase of the size of applications is less than 1 M. In the Dalvik mode, the increase of the initial startup time of applications is no more than 5 s as shown in Table 1. In the ART mode, the increase of the initial startup time of applications does not exceed 5 s, and the restart time does not exceed 2 s, as shown in Table 2.

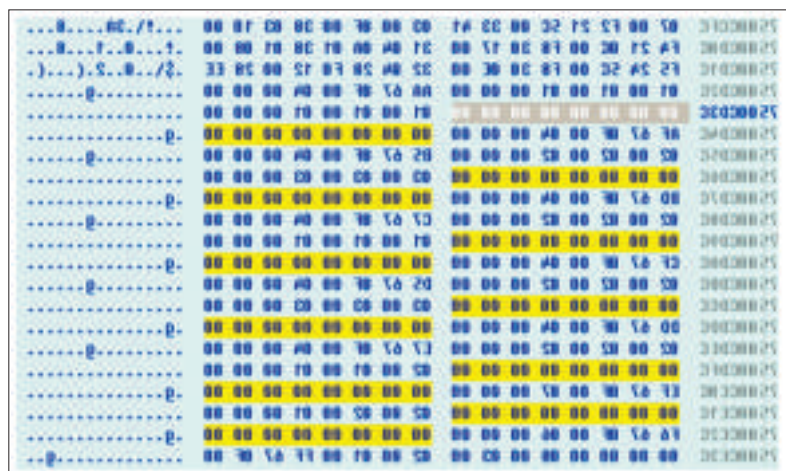
From the experimental results, the space overhead and time overhead of the scheme are within the acceptable range.

5 Conclusions

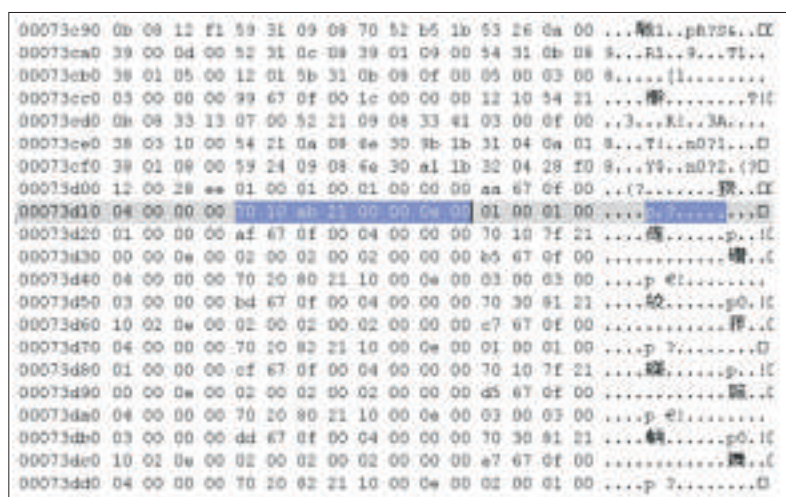
The traditional methods to protect the DEX files cannot withstand the memory dump attack because the whole plain DEX data can be copied after the application is loaded into the memory. In order to protect the DEX files from memory dump attack, DexDefender, a novel DEX protection scheme is proposed. It extracts the code fields in the DEX file and dynamically restores the code fields of each class while the application is loaded. In this way, no complete plain-text of DEX files exist in the memory during the



▲ Figure 6. The DEX in memory by using IDA pro.



▲ Figure 7. The code fields corresponding to class appstore.appstore_codec.CharEncoding in the memory by using IDA pro.



▲ Figure 8. The code fields corresponding to original apk's class appstore.appstore_codec.CharEncoding.

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

▼Table 1. Time consumption in the Dalvik mode

APK	APK version	Mean of the initial startup time before reinforcement (ms)	Mean of the initial startup time after reinforcement (ms)	Mean of the restart time before reinforcement (ms)	Mean of the restart time after reinforcement (ms)	Initial startup time increment (ms)	Restart time increment (ms)
DieProvider	1	399	717	378	276	318	-102
file_rc4	1	377	815	162	352	438	190
calculator	1	299	680	180	320	381	140
appstore	1	568	942	480	496	374	16
autorun	null	223	1213	219	246	990	27
iietransfer	2.1.0901.2146	785	2037	835	813	1252	-22
baifashop	1.0.0	2920	4184	1827	2875	1264	1048
MicroMessage	1	346	761	340	524	415	184
KuaiGeng	2.1.1	1119	4853	550	2534	3934	1984
Ofo	1.8.9	2358	4784	2524	3170	2426	646
Flipboard	3.5.3.0	2563	5819	2487	4239	3256	1752
Course plaid	9.0.4	1140	4694	1425	2254	3554	829
Gaokao Bang	4.1.1	1198	3969	603	2251	2771	1648
Dubbing hall	1.6.02.01	620	3378	595	1216	2758	621
Translator	5.8.1	2208	6216	2435	3056	4008	621
Tuhua	7.9.A.2.0	948	4113	847	1935	3165	1088
Lily	6.9.0	2368	5990	2385	3899	3622	1514
Yaolan	2.2.2	2580	6172	2297	4290	3592	1993
Xiao D Location	1.0.1	844	3693	645	1610	2849	965
Chuangbie Bookstore	4.1.1	756	3018	1567	2684	2262	1117

APK: Android Package

▼Table 2. Time consumption in the ART mode

APK	APK version	Mean of the initial startup time before reinforcement (ms)	Mean of the initial startup time after reinforcement (ms)	Mean of the restart time before reinforcement (ms)	Mean of the restart time after reinforcement (ms)	Initial startup time increment (ms)	Restart time increment (ms)
DieProvider	1	388	820	398	704	432	306
file_rc4	1	364	740	390	731	376	341
calculator	1	273	701	277	712	428	435
appstore	1	437	862	838	813	425	-25
autorun	null	338	744	218	711	406	493
iietransfer	2.1.0901.2146	972	1533	1055	1491	561	436
baifashop	1.0.0	1647	3756	1679	3356	2109	1677
MicroMessage	1	376	702	343	722	326	379
KuaiGeng	2.1.1	1882	2525	706	1885	643	1179
Ofo	1.8.9	4369	5923	2850	3734	1554	884
Flipboard	3.5.3.0	1926	4604	1595	3586	2678	1991
Course plaid	9.0.4	1925	2733	1099	2184	808	1085
Gaokao Bang	4.1.1	871	2662	368	1349	1791	981
Dubbing hall	1.6.02.01	527	2526	559	2268	1999	1709
Translator	5.8.1	1786	4236	1108	2703	2450	1595
Tuhua	7.9.A.2.0	1051	2728	842	2221	1677	1379
Lily	6.9.0	2340	5909	1306	2529	3569	1223
Yaolan	2.2.2	1429	4073	1284	3169	2644	1885
Xiao D Location	1.0.1	708	2138	663	2590	1430	1927
Chuangbie Bookstore	4.1.1	493	2510	1070	2947	2017	1877

APK: Android Package

whole lifecycle of the application, which increases the difficulty of dumping DEX directly from the memory and cracking the

applications. The experimental results show that the proposed scheme can resist memory dump attack with an acceptable per-

DexDefender: A DEX Protection Scheme to Withstand Memory Dump Attack Based on Android Platform

RONG Yu, LIU Yiyi, LI Hui, and WANG Wei

formance in both the Dalvik and ART modes on the Android platform.

References

- [1] E. Ravenscraft. (2012, Jul. 31). Just how bad is app piracy on android anyway? Hint: we're asking the wrong question. [Online]. Available: <http://www.android-police.com/2012/07/31/editorial-just-how-bad-is-app-piracy-on-android-anyways-hint-were-asking-the-wrong-question>
- [2] M. T Yuan, "China Mobile Payment Security Report," *Business Culture*, pp. 54–56, May 2014.
- [3] Y. C. Moon, J. H. Noh, A. R. Kim, et al., "Design of copy protection system for android platform," in *International Conference on Information Technology, System and Management*, Chongqing, China, 2012.
- [4] Y. S. Jeong, J. C. Moon, D. Kim, et al., "An anti-piracy mechanism based on class separation and dynamic loading for android application," in *ACM Research in Applied Computation Symposium*, San Antonio, USA, 2012, pp. 328–332. doi: 10.1145/2401603.2401674.
- [5] C. Collberg, C. Thomborso, and D. Low. (1997). A taxonomy of obfuscating transformations [Online]. Available: <http://www.cs.auckland.ac.nz/staff-cgi-bin/mjd/csTRcgi.pl?serial>
- [6] B. Yadegari, B. Johannesmeyer, B. Whitely, and S. Debray, "A generic approach to automatic deobfuscation of executable code," in *IEEE Symposium on Security and Privacy*, San Jose, USA, pp. 674–691, 2015. doi: 10.1109/SP.2015.47.
- [7] W. Zhou, Y. Zhou, M. Grace, X. Jiang, and S. Zou, "Fast, scalable detection of 'piggybacked' mobile application," in *ACM Conference on Data and Application Security and Privacy*, San Antonio, USA, pp. 185–196, 2013. doi: 10.1145/2435349.2435377.
- [8] W. Zhou, X. Zhang, and X. Jiang, "AppInk: watermarking android apps for repackaging deterrence," in *Proc. 8th ACM SIGSAC Symposium on Information, Computer and Communications Security (ASIA CCS'13)*, Hangzhou, China, pp. 1–12, 2013. doi: 10.1145/2484313.2484315.
- [9] Z. J. Dong, W. Wang, H. Li, et al., "SeSoa: security enhancement system with online authentication for android APK," *ZTE Communications*, vol. 14, no. S0, pp. 44–50, Jun. 2016. doi: 10.3969/j.issn.1673-5188.2016.S0.005.
- [10] R. X. Fan, D. Y. Fang, Z. Y. Tang, et al., "A method of preventing android app repackaging based on code splitting," *Journal of Chinese Mini-Micro Computer Systems*, vol. 37, no. 9, pp. 1969–1974, Sept. 2016.

Manuscript received: 2017-12-14

Biographies

RONG Yu (463397867@qq.com) graduated from Xidian University, China in 2015 and now she is studying for her master's degree at the Beijing University of Posts and Telecommunications (BUPT), China. Her research interests are software security and information security.

LIU Yiyi (793645428@qq.com) graduated from University of Electronic Science and Technology of China (UESTC) in 2016 and now she is studying for her master's degree at the Beijing University of Posts and Telecommunications (BUPT). Her research interests are software security and information security.

LI Hui (lihuill@bupt.edu.cn) got her Ph.D. in cryptography from BUPT, China in 2005. From July 2005, she has been working at BUPT as lecturer and associate professor. Her research interests are cryptography and its applications, information security, and wireless communication security.

WANG Wei (wang.wei8@zte.com.cn) received her B.S. degree from Nanjing University of Aeronautics and Astronautics, China. She is an engineer and project manager in the field of mobile Internet at Government & Enterprise Communications Institute of ZTE Corporation. Her research interests include new mobile Internet services and applications, PaaS, terminal application development, and other technologies. She has authored five academic papers.

A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua¹, ZHAO Yongli¹, WANG Dajiang²,
WANG Jiayu², and WANG Zhenyu²

(1. State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, 100876, China;

2. Skill Transfer Management Department, ZTE Corporation, Beijing 100000, China)

Abstract

Due to the vulnerability of fibers in optical networks, physical-layer attacks targeting photon splitting, such as eavesdropping, can potentially lead to large information and revenue loss. To enhance the existing security approaches of optical networks, a new promising technology, quantum key distribution (QKD), can securely encrypt services in optical networks, which has been a hotspot of research in recent years for its characteristic that can let clients know whether information transmission has been eavesdropped or not. In this paper, we apply QKD to provide secret keys for optical networks and then introduce the architecture of QKD based optical network. As for the secret keys generated by QKD in optical networks, we propose a re-transmission mechanism by analyzing the security risks in QKD-based optical networks. Numerical results indicate that the proposed re-transmission mechanism can provide strong protection degree with enhanced attack protection. Finally, we illustrated some future challenges in QKD-based optical networks.

Keywords

optical networks; security; QKD; re-transmission

1 Introduction

The explosive growth of services has led to a growing demand for bandwidth and transmission quality, which poses a serious challenge to network operators. At the same time, operators need to manage both IP layer and optical layer in the optical networks, which

This work has been supported in part by NSFC project (Grant No. 61571058 and 61601052), Science and Technology Project of State Grid Corporation of China: The Key Technology Research of Elastic Optical Network (Grant No. 526800160006), China Postdoctoral Science Foundation Project (2016M600970), and ZTE Industry-Academia-Research Cooperation Funds.

results in a waste of time and energy overhead and rapidly increase in operating costs by repeating resource construction. The developed technology of IP over optical layer can solve this problem.

However, because optical network is a communication infrastructure to support people's daily life, it is widely recognized that the optical layer in IP over optical networks is crucial in supporting the rapidly growing traffic. Therefore, issues related to optical layer security become very important, which suffers more and more security incidents which are mainly by the method of eavesdropping information in optical networks to carry out harmful behavior. For instance, the world's largest bit-maker trading platform was attacked in 2014 and the loss was estimated about \$467 million, which was caused by eavesdropping information in the fiber. Hence, it is crucial to solve the security problem of the optical layer, which also means the security problem in optical networks.

Caused by the weak defense of physical layer and the simplicity of logic layer in optical networks, services in transmission are vulnerable to security threats; the solution to this security issue depends on the encryption of services. Standard optical network encryption approaches typically utilize complex mathematical questions and decrypting them is not difficult but needs time only. This may be effective in the presence of failures under normal circumstances, but may fail to provide adequate protection for the services under deliberate eavesdropping.

To deal with this problem, an "absolutely safe" solution for the above problems in optical networks is quantum communication which could let the clients notice whether the quantum key has been eavesdropped based on the quantum mechanics inside itself [1]. The "absolutely safe" is guaranteed by quantum key distribution (QKD) over "one time padding" system [2]. Due to the above advantages, the topic about quantum communication in optical networks has been hot around the world. Quantum communication has been listed as the one of the top ten key technologies to promote the development of "13th five-year" plan in China. The National Institute of Standards Department of Defense and Technology of USA has regarded quantum as one of the key research directions. The Europe has invested billions of dollars in its quantum projects. Japan has proposed a long-term research strategy for quantum communication. The introduction of quantum communication into optical networks as a security support can effectively avoid the risk of unsafe communication and ensure the "absolute security" of optical networks, which has a very important innovative value and practical significance.

2 QKD Fundamentals

2.1 QKD Protocols and Networks

QKD is a process which enables both sides in communica-

A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu

tion to share a secure key by encrypting and decrypting services, which needs corresponding protocols and networks to formulate the rules and realize the wide-spread confidential communication.

A QKD protocol is used to arrange the behavior of both sides in communication to achieve the proposal of security. The BB84 protocol is the first international quantum key distribution protocol, which has been proposed since 1984 to increase the safety of communication distance, improve the security rate and improve the real system security. **Fig. 1** shows the point-to-point quantum key communication procedure, where the sender (commonly known as Alice) and receiver (commonly known as Bob) use quantum channels to transmit quantum states, taking into account the possibility that both channels are eavesdropped by a third party (commonly known as Eve). Other related protocols include the B92 protocol, six-state protocol, and E91 protocol [3].

QKD networks refer to the operability among multiple nodes in secure communication. The quantum network of Defense Advanced Research Projects Agency (DARPA), an agency of the United States Department of Defense, uses multi-optical switches and trusted relays in the backbone to connect multiple subnets [4]. The Secure Communication Based on Quantum Cryptography (SECOQC) network in the Europe and the QKD network in Tokyo, Japan use the trusted relay to build quantum networks [5], [6]. Moreover, University of Science and Technology of China has designed a full-time all-quantum router based on the wave division multiplexer, and used it as the core technology to build the “four-node star” QKD network in Beijing, China and “multi-level” quantum government network in Wuhu, Anhui Province of China [7]–[9], which is in the forefront of the world. Shandong Institute of Quantum Science and Technology in China took the application demonstration of quantum communication integrated in optical networks in 2015, which passed the testing and achieved the QKD network under a multi-user environment. To promote the QKD as the core technology of quantum network construction, China launched the world’s first quantum science experimental satellite “Micius” in 2017. Following it, a long-distance quantum communication backbone optical network in China is being completed between Beijing and Shanghai to achieve the backbone network QKD and promote wide area quantum communi-

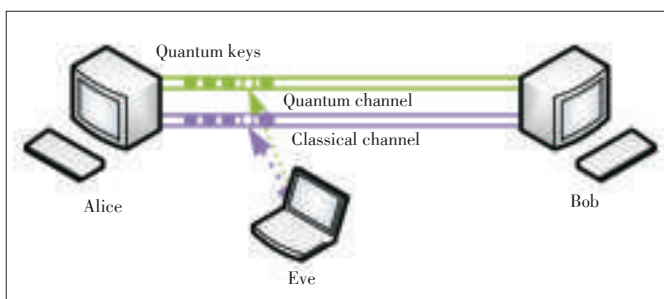
cation [10].

2.2 Key Technologies of QKD in Optical Networks

Nowadays, the main studies of quantum communication in optical networks are focused on the mixed transmission of quantum and classical light, deployment of quantum relay or trusted relay, quantum coding and quantum storage in optical networks, and other research directions. However, we mainly discuss the compatibility of quantum communication integrated with optical networks and the related transmission technology of mixing quantum signals and classical light.

The compatibility of quantum communication in classical optical networks is one of the crucial factors that directly affect the performance of quantum optical network and cost of network construction. The energy in optical pulse of a single photon (quantum key) transmitted in a QKD channel is about 1.28×10^{-19} J at 1550 nm. In previous experiments, QKD systems used a single mode fiber to realize the longest transmitted distance of QKD, which up to 250 km with ultra-low loss [11]. In the case of point-to-point QKD connection in fiber, quantum can reach Mbit/s level rate [12]. Because of the high cost of laying and leasing fiber, the way that both quantum and classical light are multiplexed and transmitted in a fiber can effectively save cost and improve fiber utilization, which is significant for the development of quantum communication. For the same reason, the research in transmission of mixing QKD channels and classic channels in a common single fiber with wavelength division multiplexing (WDM) technology is gradually increasing. The transmission of combined QKD and services using WDM technology was first demonstrated in 1997 [13]. Subsequently, the quantum channel is accurate to O-band (1260 nm–1360 nm) to achieve confidential communication [14], [15].

In order to transmit weak quantum and dense classical light with WDM technology, we need solve two key problems: 1) Due to the large number of services, effective isolation is needed to prevent the quantum from being flooded by the classical light; 2) nonlinear noise is caused by the Raman scattering and the four-wave mixing effect, which would cause the quantum deteriorate seriously. Different solutions to the above problems have been proposed. A classical and quantum mixed transmit mechanism was proposed, which could effectively inhibit the four wavelengths and noisy filtering effect by non-uniform wavelength interval over C-band [16]. A multi-stage band-stop filter technique was developed then, which utilizes multi-stage filter to realize the effective isolation of quantum channel, synchronization channel and classical channel [17]. The wavelengths of quantum and synchronization signals are 1550.12 nm and 1556.55 nm, the quantum error rate is as low as 0.9% to 2%, which could achieve the optical transmission distance up to 45 km [18]. Classic channels and quantum channels cannot near the position of long wavelength was found, which could avoid the Raman noise, and working away from the optical fiber zero dispersion wavelength can effectively reduce the



▲ Figure 1. Point-to-point quantum key communication procedure.

A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu

generation of four-wave mixing effect.

3 Security Analysis of QKD in Practical Optical Networks

Today’s optical networks provide suitable infrastructure for kinds of services ranging from government networks, financial networks, military networks, social networks to communicating or trade online networks, which are supposed to be protected by at least one quantum key according to the security requirements of users; one key can only be used once. Therefore, a large number of quantum keys are transmitted in the optical network for real-time protecting services. While the “unconditional security” of QKD was proven, several practical security concerns in QKD integrated in optical networks are still need to be solved for compatibility. We analyze this complex security issue in a systematic way with respect to quantum key transmission failure, eavesdropping, and authentication failure.

3.1 Quantum Key Transmission Failure

With the development of computer technology, security requirements of data service users are also increasing. Therefore, it is necessary to transmit a large number of quantum keys in a limited amount of resources in the optical network. If there is no resource in the network that can be provided for the quantum key, or if the quantum key is distributed at the receiver, we believe that the quantum key transmission fails once the quantum bit error rate is higher than a certain threshold.

3.2 The Security of Keys in Other Ways

The behavior of eavesdropping is inevitable, which is inherent to the attacks in optical networks that need to be protected using quantum. Because practical QKD devices are immature and fibers are vulnerable, the keys generated by QKD are still vulnerable to some attacks since keys still have the risk of leakage [19]. In order to prevent service leakage, they still need to be encrypted with the permitted conditions.

3.3 Quantum Key Authentication Failure

The quantum key is used for secure encryption of data information in multi-side quantum communication. The related protocols ensure a secure key reaches the receiver, while the identity of both sides in communication cannot be guaranteed and building a fake receiver could make the information eavesdropped. Thus, the communication sides need to be authenticated before the data transmission.

4 Quantum Key Re-Transmission Mechanism

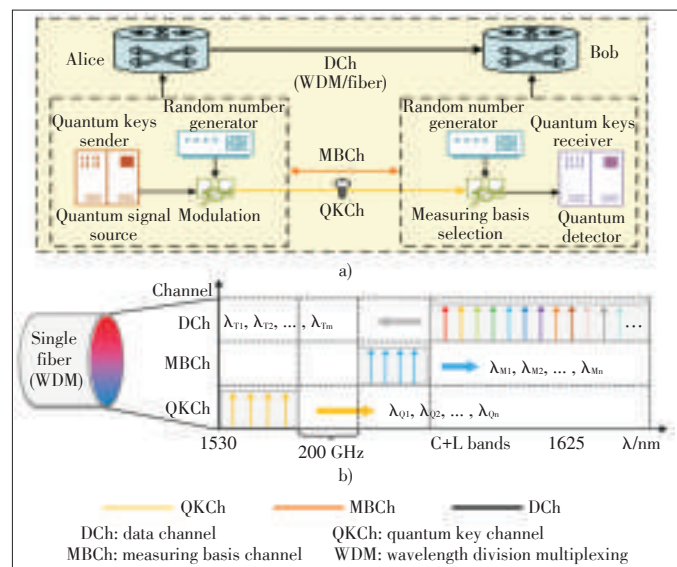
4.1 Architecture of QKD-Based Optical Networks

Optical networks are important infrastructure of communica-

tion systems. With the continuous improvement in flexibility and intelligence of optical networks, the concept of using quantum communication to enhance its security has been put forward [4]. Quantum keys are generated by QKD technology to encrypt the services, following which the network administrator selects paths and allocate resources for the keys.

The point-to-point communication in QKD-based optical networks is shown in Fig. 2. The architecture has the application plane, management plane, QKD plane and data plane from top to bottom. To realize point-to-point protection for services, QKD communication is realized by sharing a quantum key between quantum transmitter and receiver through quantum key channel (QKCh) and measurable basis channel (MBCh) (Fig. 2a). QKCh and MBCh can share the same fiber with data channel (DCh) over C-band (Fig. 2b) by WDM technology to save fiber resources and reduce costs [20]. Optical cross-connect devices (OXCs) are deployed at the data plane and QKD plane using trusted-nodes.

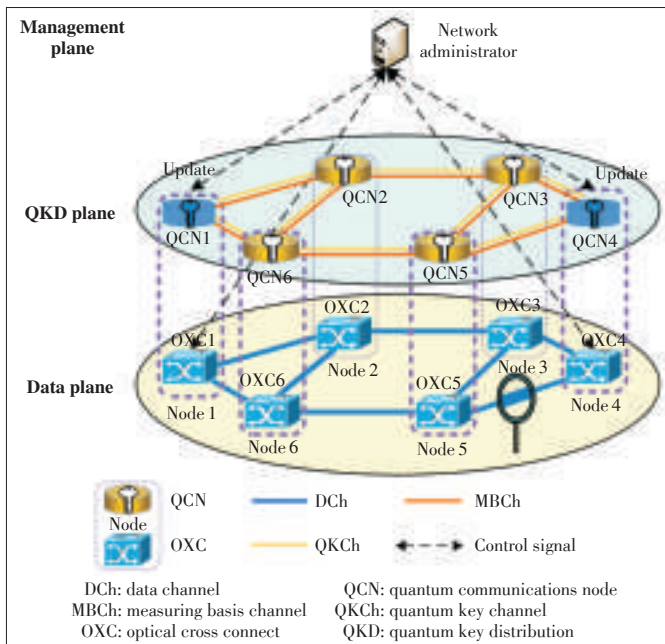
Fig. 3 shows the architecture of QKD-based optical networks. To realize end-to-end protection for services, secure communication requests are first generated from clients. Then this would be received by the management plane which is responsible for route forwarding and resource allocation at the QKD and data planes. The QKD plane is logically separated from the data plane but in the same physical entity. The QKD plane provides quantum keys to protect the services at the data plane, which includes the management of quantum keys and the service encryption process, such as update of quantum keys and the process of quantum key distribution. The management of quantum keys becomes flexible and intelligent for the network administrator, and the administrator is able to adaptively change the keys to effectively guarantee the whole net-



▲ Figure 2. Point-to-point communication in quantum key distribution (QKD) based optical networks: a) point-to-point communication system; b) wavelength allocation in fiber [21].

A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu



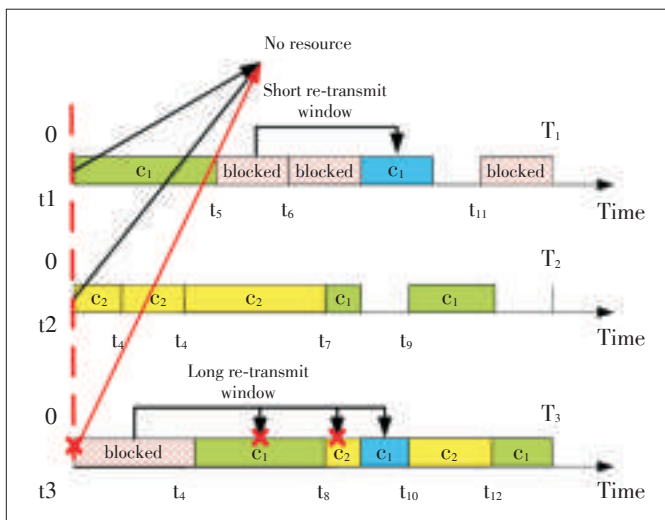
▲ Figure 3. Architecture of QKD-based optical networks.

work security.

4.2 Quantum Key Re-Transmission Mechanism

In response to the above analysis, we propose a quantum key re-transmission mechanism, analogous to the Advanced Encryption Standard (AES) in classical optical networks [22].

As shown in Fig. 4 and Algorithm 1, the mechanism is a re-transmission process of failed quantum keys. As there are lots of services transmitted in a QKD-based optical network, the start of re-transmission of the failed quantum key is always caused by the limited optical network resource. When one of these cases occurs, the failed quantum key needs to formulate a re-transmission time window, which could try many times



▲ Figure 4. Re-transmission of quantum keys.

within a range. The re-transmission time window depends on the security degree required by users. A high secure degree service needs a large re-transmission time window to try many times for safely reaching the receiver, just like the third axis. A low secure degree service re-transmits within a short time window. For example, there are six wavelengths in one fiber used for services, quantum keys and measurable basis information, respectively. When all the quantum key channels are occupied, the quantum keys need wait a certain time to re-transmit.

Algorithm 1: quantum key re-transmission

1. For each quantum key {
2. While (failed quantum key been detected) {
3. Select random distribution;
4. Set the range of Δt ;
5. Do {
6. Generate t_{ri} in the range of time window
7. utilizing the distribution;
8. } While ($t_{ri} < T_{ei}$) t_{si}
9. }
10. While (clock comes to t_{ri}) {
11. Compute one path d utilizing Dijkstra algorithm;
12. If $d \neq \emptyset$, Then First Fit algorithm for time-slot assignment;
13. Else the quantum key failed to transmit;
14. }
15. }
16. }
17. }

We give specific quantum key re-transmission algorithms for users in need of different secure requirements. A quantum key in the algorithm is denoted as $q_r(s, d, t_s, t_h, t_r, \Delta t)$, where u is the number of quantum keys, s and d represent sources and destination nodes, t_s and t_h are its start time and hold time respectively, and Δt is the time window width. The arrival time of each update key is denoted as t_{si} , which should be generated before the leaves of data service T_{ei} . The hold time of each re-transmission quantum key is a fixed value of 1s. Firstly, once a quantum key transmission failure is detected, the secure degree of the service is judged and a re-transmission time window is set. If the secure requirement is in a high degree, we set a long range for the time window and vice versa. The range values are designed according to the simulation results. Then, when the clock goes to t_{si} , the network administrator computes one shortest path among several available paths by the Dijkstra algorithm with the same source node and destination node with the services. If there is no available path or time slot, this quantum key is failed to be transmitted and then would be thrown away.

4.3 Simulation Results

Simulations were conducted to evaluate the proposed mecha-

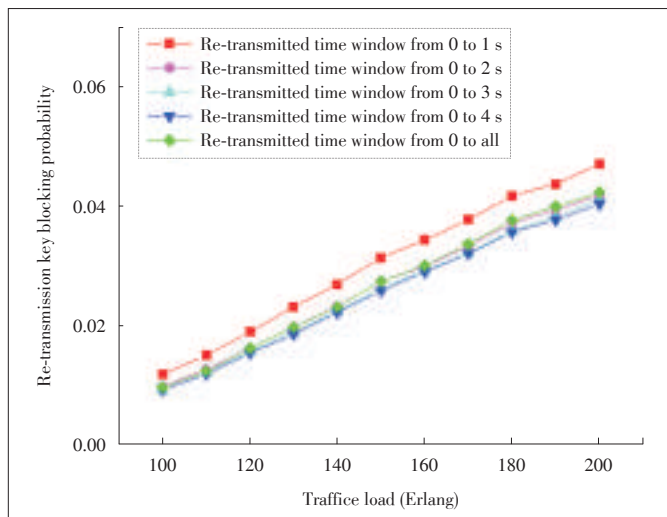
A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu

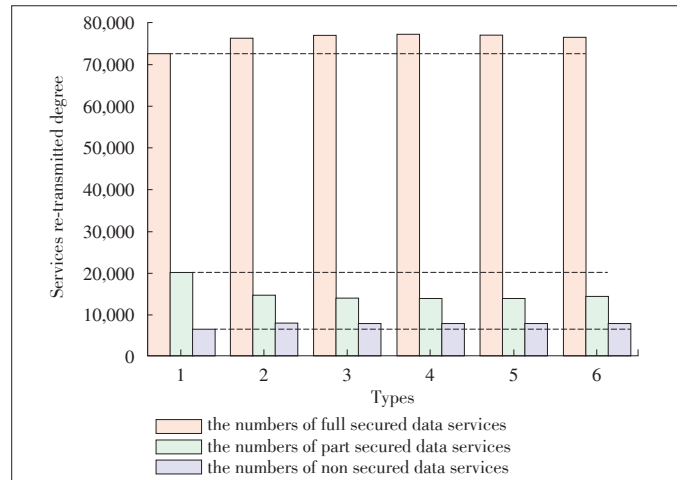
nism and ensure the feasibility of the re-transmission mechanism. In the simulation, the topology is a national science fund network (NSFNET) with 14 nodes and 21 links. The number of services is 100,000. The wavelength numbers of DCh, QKCh and MBCh are set as 28, 4, and 4, respectively. The simulations were carried out in the software virtual studio that is based on C++ language. We studied the performance of QKD-based optical networks in terms of blocking probability, resource utilization probability, re-transmission protection degree and re-transmission successful probability.

We simulated re-transmission of quantum keys random generated from different size time windows (Fig. 5). The quantum key was re-tried from the current failure time, and the time window is increased by 1 s each time until the data service transmission time ends. It can be seen that the blocking rate of a re-transmission quantum key becomes stable gradually as the traffic load increases. We found that larger key re-transmission time windows could result in lower blocking probability by comparing the time windows with different sizes. This is because more time is given to make the failed quantum key have more chances to try re-transmission. This could increase the security of data services.

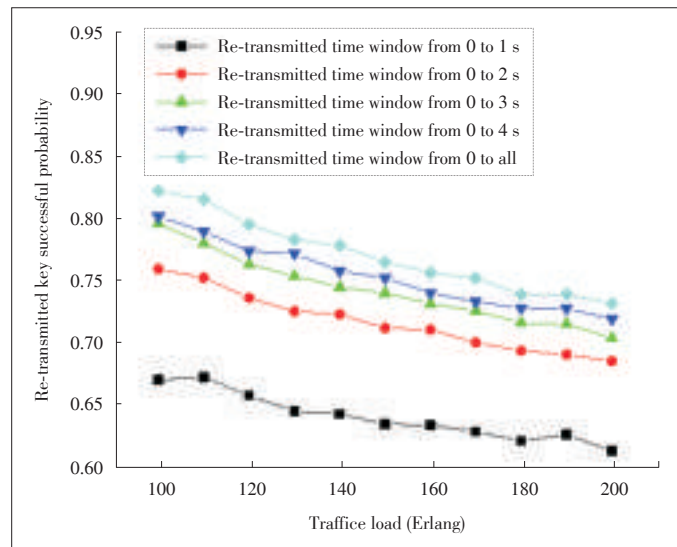
Fig. 6 shows the protection degrees of data services after re-transmission compared with no re-transmission. The abscissa indicates the types of re-transmission time window in the order which are no re-transmission (type 1), [0, 1] (type 2), [0, 2] (type 3), [0, 3] (type 4), [0, 4] (type 5), [0, all] (type 6). The re-transmission has a certain increase in full-protect data services which are suitable for the high security level services compared with no re-transmission. The number of part-protect data services after re-transmission is reduced while the number of none-protect data services is slightly higher. The overall security level of the full-protect and part-protect data services is increased compared to the services with no re-transmission.



▲ Figure 5. Re-transmission key blocking probability in time windows with different sizes.



▲ Figure 6. Services re-transmitted protection degree.



▲ Figure 7. Re-transmit quantum key successful probability.

The successful probability of key re-transmission in different sizes of time window is shown in Fig. 7. With the increase in traffic load (the density of data services), the overall trend of successful re-transmission is gradually small. The higher successful re-transmission probability is always with bigger time windows, which could reduce the blocking probability in a big degree (Fig. 4) to enhance the network security. Therefore, bigger re-transmission time windows can result in lower blocking probability, higher resource utilization and bigger numbers of successful re-transmission.

5 Main Research Challenges

With optical networks becoming more virtualized and intelligent, they are facing with various security risks. For these security problems, quantum communication can provide a reliable and secure scheme for optical networks, helping guarantee the

backbone security of telecommunication networks and reduce the complexity of management. QKD-based optical networks are developing from point-to-point application to multi-node application. However, further research is needed, especially on the important issues shown in Fig. 8.

5.1 Quantum Key Management

In recent years, quantum communication in optical networks has made great progress and entered the trial stage, in which the quantum nodes achieve receive and forwarding function both for quantum and classical light signals. It has become a consensus that quantum can be used for the medium that carries critical information, so the management of quantum keys has attracted much research attention because it is the basis for secure optical networks. Storing quantum keys at a node, updating quantum keys to ensure key security, and allocating resources for a large number of quantum keys are hot topics in the research of quantum key management.

5.2 Quantum Key Survivability

Survivability is an issue every network has to take into consideration, and QKD-based optical networks are no exception. It also means the disaster resistance of quantum keys in the network. In order to achieve the protection of services in optical networks, we should study protection and recovery measures of QKD-based optical networks, as well as the collaborative protection of quantum keys and services.

5.3 Network Construction Cost Reduction

Cost reduction plays a decisive role in the development and practical application of QKD-based optical networks. The high cost of network construction is always caused by the high cost of hardware equipment. The transmission of quantum combined with classic signal can not only ensure "absolute security" of services in optical networks, but also help to reduce the

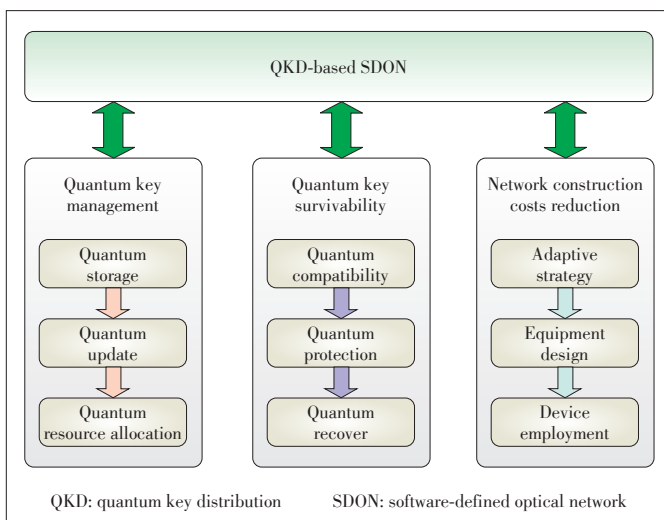
laying cost of fiber and that of its management and maintenance. The following issues are crucial for the cost reduction: how to select wavelength for quantum keys to reduce crosstalk with classical channels; how to make a long-distance safe transmission for reducing the use of hardware devices; how to deploy hardware devices at a minimum cost.

6 Conclusions

With the development of quantum networks in metro areas, quantum communication is becoming a key technology to support optical security in the future. In this paper, we describe quantum communication as part of a secure communications solution, and specifically introduce the architecture of QKD-based optical networks for flexibly and dynamically protecting services. A more secure quantum key re-transmission mechanism is proposed to solve the security risk issue in QKD-based optical networks. The numerical simulation results show the good performance of the mechanism. Our future work will focus on quantum management, quantum survivability, and the cost reduction of network construction in QKD-based optical networks.

References

- [1] D. J. Griffiths, *Introduction to Quantum Mechanics*. Cambridge, England: Cambridge University Press, 2016.
- [2] S. S. Kute and G. C. Desai, "Quantum cryptography: a review," *Indian Journal of Science and Technology*, vol. 10, no. 3, 2017. doi: 10.17485/ijst/2017/v10i3/110635.
- [3] W. K. Hong, M. O. Foong, and J. T. Low, "Challenges in quantum key distribution: a review," in *Proc. ACM 4th International Conference on Information and Network Security*, Kuala Lumpur, Malaysia, 2016, pp. 29–33.
- [4] C. Elliott, A. Colvin, D. Pearson, et al., "Current status of the DARPA quantum network," in *Proc. SPIE 5815, Quantum Information and Computation III*, Orlando, USA, 2005, pp. 138–149. doi: 10.1117/12.606489.
- [5] M. Peev, C. Pacher, R. Alléaume, et al., "The SECOQC quantum key distribution network in Vienna," *New Journal of Physics*, vol. 11, article no. 075001, Jul. 2009. doi: 10.1088/1367-2630/11/7/075001.
- [6] M. Sasaki, M. Fujiwara, H. Ishizuka, et al., "Field test of quantum key distribution in the Tokyo QKD network," *Optics Express*, vol. 19, no. 11, pp. 10387–10409, 2011. doi: 10.1364/OE.19.1010387.
- [7] S. Aleksic, D. Winkler, G. Franzl, et al., "Quantum key distribution over optical access networks," in *NOC/OC&I*, Graz, Austria, 2013, pp. 11–18. doi: 10.1109/NOC-OCI.2013.6582861.
- [8] F. X. Xu, W. Chen, S. Wang, et al., "Field experiment on a robust hierarchical metropolitan quantum cryptography network," *Chinese Science Bulletin*, vol. 54, no. 17, pp. 2991–2997, Sept. 2009. doi: 10.1007/s11434-009-0526-3.
- [9] S. Wang, W. Chen, Z.-Q. Yin, et al., "Field test of wavelength-saving quantum key distribution network," *Optics Letters*, vol. 35, no. 14, pp. 2454–2456, Jul. 2010. doi: 10.1364/OL.35.002454.
- [10] S. Wang, W. Chen, Z.-Q. Yin, et al., "Field and long-term demonstration of a wide area quantum key distribution network," *Optics Express*, vol. 22, no. 18, pp. 21739–21756, Sept. 2014. doi: 10.1364/OE.22.021739.
- [11] S. Wang, W. Chen, J.-F. Guo, et al., "2 GHz clock quantum key distribution over 260 km of standard telecom fiber," *Optics Letters*, vol. 37, no. 6, pp. 1008–1010, Mar. 2012. doi: 10.1364/OL.37.001008.
- [12] A. R. Dixon, Z. L. Yuan, J. F. Dynes, A. W. Sharpe, and A. J. Shields, "Continuous operation of high bit rate quantum key distribution," *Applied Physics Letters*, vol. 96, no. 16, Mar. 2010. doi: 10.1063/1.3385293.
- [13] D. P. Townsend, "Simultaneous quantum cryptographic key distribution and conventional data transmission over installed fiber using wavelength-division multiplexing," *Electronics Letters*, vol. 33, no. 3, pp. 188–190, Jan. 1997.



▲ Figure 8. Future development of QKD-based optical networks.

A Quantum Key Re-Transmission Mechanism for QKD-Based Optical Networks

WANG Hua, ZHAO Yongli, WANG Dajiang, WANG Jiayu, and WANG Zhenyu

- [14] J. R. Runser, T. E. Chapuran, P. Toliver, et al., "Demonstration of 1.3 μm quantum key distribution (QKD) compatibility with 1.5 μm metropolitan wavelength division multiplexed (WDM) systems," in *OFC/NFOEC*, Anaheim, USA, 2005.
- [15] N. I. Nweke, R. J. Runser, S. R. McNown, et al., "EDFA bypass and filtering architecture enabling QKD+WDM coexistence on mid-span amplified links," in *Conference on Lasers and Electro-Optics/Quantum Electronics and Laser Science Conference*, Long Beach, USA, 2006.
- [16] L. He, J. Niu, Y. Sun, and Y. Ji, "The four wave mixing effects in quantum key distribution based on conventional WDM network," in *12th International Conference on Optical Internet*, Jeju, South Korea, 2014.
- [17] L. Wang, L.-K. Chen, L. Ju, et al., "Experimental multiplexing of quantum key distribution with classical optical communication," *Applied Physics Letters*, vol. 106, no. 8, Feb. 2015. doi: 10.1063/1.4913483.
- [18] T. F. da Silva, G. B. Xavier, G. P. Temporal, et al., "Impact of raman scattered noise from multiple telecom channels on fiber-optic quantum key distribution systems," *Journal of Lightwave Technology*, vol. 32, no. 13, pp. 2332–2339, Jul. 2014. doi: 10.1109/JLT.2014.2322108.
- [19] H. Wang, Y. Zhao, Y. Li, et al., "A flexible key update method for software-defined optical networks (SDON) secured by quantum key distribution," *Optical Fiber Technology*, vol. 45, pp. 195–200, Nov. 2018. doi: 10.1016/j.yofte.2018.07.005.
- [20] I. Choi, R. J. Young, and P. D. Townsend, "Quantum key distribution on a 10Gb/s WDM-PON," *Optics Express*, vol. 18, no. 9, pp. 9600–9612, 2010. doi: 10.1364/OE.18.009600.
- [21] Y. Cao, Y. Zhao, X. Yu, et al., "Resource allocation in software-defined optical networks secured by quantum key distribution," in 2017 Opto-Electronics and Communications Conference (OECC) and Photonics Global Conference (PGC), Singapore, Singapore, 2017. doi: 10.1109/OECC.2017.8114769.
- [22] F. P. Miller, A. F. Vandome, and J. McBrewster, *Advanced Encryption Standard*. South San Francisco, USA: Alpha Press, 2009.

Manuscript received: 2017-11-03

Biographies

WANG Hua (Whua@bupt.edu.cn) is currently working toward her Ph.D. degree in information and communications engineering at Beijing University of Posts and Telecommunications (BUPT), China. Her research interests include software defined optical networking and quantum communication.

ZHAO Yongli (yonglizhao@bupt.edu.cn) is currently an associate professor of the Institute of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications (BUPT), China. He received the B.S. degree in communication engineering and Ph.D. degree in electromagnetic field and microwave technology from BUPT. During Jan. 2016 to Jan. 2017, he was a visiting associate professor at UC Davis, USA. He has published more than 300 international journal and conference papers. Since 2015, he has become a senior member of IEEE. His research focuses on software defined optical networking, elastic optical networks, datacenter networking, and optical network security.

WANG Dajiang (wang.dajiang@zte.com.cn) is an experienced senior engineer and product planning manager of intelligent optical networks with ZTE Corporation. With 12 years of R&D experience in the intelligent optical network field, he has many optical-oriented SDN patents and was the core researcher of two "863" national scientific research projects of China.

WANG Jiayu (Wang.jiayu@zte.com.cn) is an experienced senior engineer and product planning manager of intelligent optical networks with ZTE Corporation. His research interest is intelligent optical networking.

WANG Zhenyu (Wang.zhenyu@zte.com.cn) is an experienced senior engineer and product planning manager of intelligent optical networks with ZTE Corporation. His research interest is intelligent optical networking.

Persistent Data Layout in File Systems

LUO Shengmei¹, LU Youyou², YANG Hongzhang¹,
SHU Jiwu², and ZHANG Jiacheng²

(1. ZTE Corporation, Shenzhen 518057, China;

2. Tsinghua University, Beijing 100084, China)

1 Introduction

The organization of data stored outside of a core is data layout in file systems. The data layout influences the performance of a file system greatly. Data can be stored in various kinds of storage mediums.

In traditional file systems, disks are employed for storing data out of a core. The performance of a disk is greatly influenced by the time of tracking and locating a sector [1], [2]. To reduce the time of tracking, Fast File System (FFS), ext2, ext3 and other traditional file systems organize the storage space into groups of cylinders, which increases the locality of accesses [1], [3], [4]. To reduce the time of locating a sector, traditional file systems based on disks utilize caching, prefetching, pre-allocating and other technologies to access the data sequentially [4]. Thus, the data layout gathers the data in the successive physical sections as far as possible. Then it can increase the succession and locality of the data accessing, improving the performance of the file system.

In the state-of-the-art file system, solid state drives (SSDs) are used as the external storages [5]–[7]. Because SSDs are electronic memories, they do not have units of mechanical tracking and sector locating. Therefore, the influence of the optimization by gathering data and accessing data sequentially is futile. Nevertheless, the imbalance of read and write latency in SSDs and the lifetime problem are new challenges to SSDs. The random write performances worse than the random read in SSDs. Thus, the data layout in SSDs has the best able data to be written sequentially, and the lifetime problem of SSDs requires slighter write amplification. Then, fine-grained writes to SSDs are employed in file systems to update the data, and this prevents write amplification from page alignment. In addition, the garbage collection of file systems in SSDs should move the valid pages out before wiping the block. In conclusion, the da-

Abstract

Data layout in a file system is the organization of data stored in external storages. The data layout has a huge impact on performance of storage systems. We survey three main kinds of data layout in traditional file systems: in-place update file system, log-structured file system, and copy-on-write file system. Each file system has its own strengths and weaknesses under different circumstances. We also include a recent usage of persistent layout in a file system that combines both flash memory and byte-addressable non-volatile memory. With this survey, we conclude that persistent data layout in file systems may evolve dramatically in the era of emerging non-volatile memory.

Keywords

data layout; file system; persistent storage; solid state drive (SSD)

ta layout in SSDs not only influences the performance of the file system but also influences the lifetime of SSDs.

Besides the storage performance, the integrity and consistency of data should be taken into consideration. These two properties play an important role in the file system. For example, if a user is updating data in the file system through the cache and a power failure occurs, the data in the cache will not be written back to the external storage yet. Then it is unable for the user to know if the storage stores the latest data. It is crucial that the data will not lose and be inconsistent when there are some failures.

Besides security, the performance of file systems always attracts a lot of attention from researchers. Based on SSDs, we propose a multi-level file system named stageFS. It utilizes the cache to update data not in a granularity of a page but in a granularity of a record. It can decrease the number of writing to a large extent and then improve the performance of the file system.

The rest of this paper is organized as follows. Section 2 shows three kinds of traditional data layout in file systems. Section 3 describes the persistent storage in file systems. Section 4 presents the built-in multi-level persistent file system and the conclusion is given in Section 5.

2 Data Layout in File System

This section will introduce three kinds of traditional data layouts. They are in-place update file system, long-structured file system (LSF), and copy-on-write file system. All the exist-

Persistent Data Layout in File Systems

LUO Shengmei, LU Youyou, YANG Hongzhang, SHU Jiwu and ZHANG Jiacheng

ing data layouts belong to these three kinds.

2.1 In-Place Update File System

In-place update file systems, such as ext2, ext3 and ext4, support overwrite operation. They enable data with continuous logical addresses to be stored continuously in disks, which improves the locality of data when being read. Nevertheless, in-place update file systems may lead data to be written dispersedly, which has impact on the performance of writing data. We will take ext4 as an example to illustrate the in-place update file system.

Ext4 is a representative in-place update file system. The data layout on the disk is shown in Fig. 1. An ext4 file system is divided into a lot of block groups. The block allocator always tries to allocate the blocks belonging to the same file into the same block group, which may reduce the time of tracking. In a block group, data are distributed as shown in Fig. 1. The first 1024 bytes in the block group 0 are used to install the boot block, and other block groups have no section like this. The super block describes and maintains the state of the file system, such as the gross of index nodes (inodes) and the used blocks. A portion of block group stores the redundant copies of super blocks and group descriptors. Not all the block groups have the copies. If one has no copy, it will start with the data block bitmap. Reserved Group Description Table (GDT) blocks are used for file system extensions.

In order to ensure that the logical continuous data are stored on the disk continuously, ext4 adopts five kinds of allocating strategy:

1) Multi-block allocator

When a new file is created, the block allocator assumes that it will grow with a high speed, thus allocating 8 KB of continuous disk space to it. When the file is closed, if this space does not be used, the unused part will be recycled; if used, the data are in a physically continuous space.

2) Delayed allocation

This strategy works with the cache. When a file needs more blocks to write due to updating data, the controller will not allocate blocks for it immediately, but until the data in cache must be written back to the disk (such as sync operation occurs and the memory is full). In this way, as many data as possible are stored in the cache, which is beneficial for allocating.

3) Allocating inodes and data blocks in the same block group

When the file system reads inodes of a

file system and obtains the locations of data blocks, if the two are in the same block group, the time for tracking will reduce.

4) Inode and its directory in the same block group

When the file system reads the directory and obtains the ID of the inode, if the two are in the same group, the time for tracking will reduce.

5) Dividing the disk into several block groups

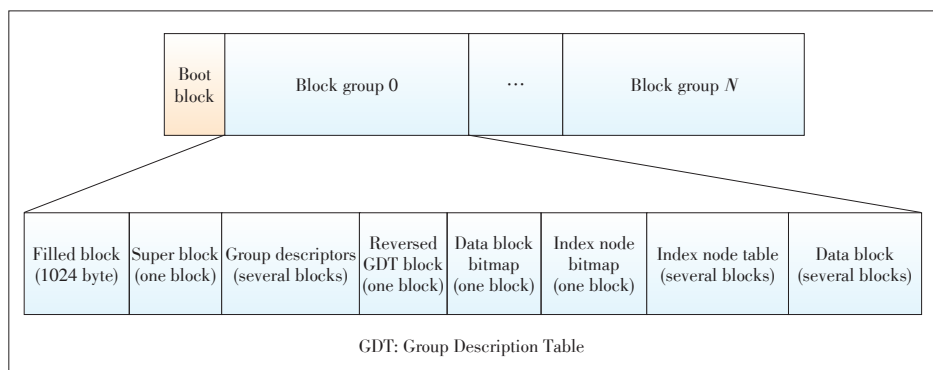
Trying best to allocate the blocks belonging to the same file in the same block group [1], [3], [8] will mitigate the problem of file fragmentation [4], [9], [10].

In in-place update file systems, there are allocation modes that allocating several blocks continuously with the extended segment mode, except the block-level allocation mode.

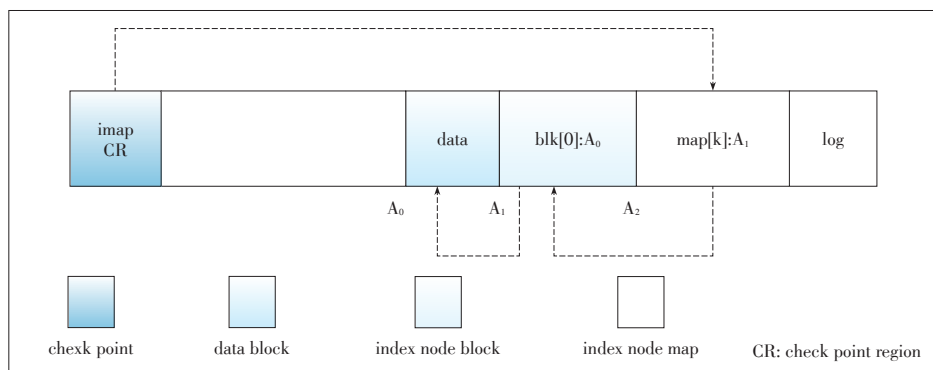
2.2 Log-Structured File System

LSF write data using the mode of append write to the external storage sequentially [11]. This write mode performs well when writing data, while it has a bad performance due to the random read. We can find that the in-place file system is good for reading while LSF is good for writing.

Fig. 2 shows the principles of LSF. It caches the file data in the memory and then writes data using append write to the external memory when the cache has no space. This ensures the sequential write to the storage. LSF also updates inodes sequentially. In order to solve the problem of file location, LSF introduces Inode Map (map of inodes) and check point region



▲ Figure 1. Data layout of the ext4 file system.



▲ Figure 2. Data layout of a typical log-structured file system.

(CR). Inode Map records the location of each inode. Through the Inode Map, the controller can locate the corresponding inode quickly and then finds the corresponding file data blocks. The Inode Map is also updated sequentially and the CR can locate the latest version of the Inode Map. The process of file updating is shown as the follow:

- 1) The file system data is written to the cache.
- 2) The data is written to the external memory when the capacity of the cache reaches a threshold.
- 3) During the sequential write process, the file data is updated first, followed by updating the inode; then the Inode Map is updated.
- 4) The region of CR is updated periodically.

In LSF, the following process of searching the file data according to the inode is different from that in in-place update file system:

- 1) Locate the latest version of Inode Map according to CR
- 2) Search the address of the inode in the Inode Map according to the ID of the inode
- 3) Locate the file data according to the inode.

Take Flash Friendly File System (F2FS) [12] as a typical example to introduce the LSF. F2FS is developed by Samsung based on SSDs. It divides a disk into a number of segments. Each segment has the fixed size: 2 MB. Each section is composed of adjacent segments and several adjacent sections compose a zone. Through the command `mkfs`, one can easily change the sizes of section and zone.

The layout of F2FS is shown in Fig. 3. F2FS divides the disk into two regions. One is the metadata region and the other is the data region. Each region is composed of several segments except the super block. The super block is located at the start of the zone, including some information of partition and default parameters. There are two backups of the super block in a system. The check point (CP) includes the states of a file system, the bitmap of Node Address Table (NAT)/Segment Information Table (SIT), the linked list of orphan nodes, the number of current active segments and other information. `Segment Information Table` includes the information of each segment, such as the number of valid blocks and the valid bit-

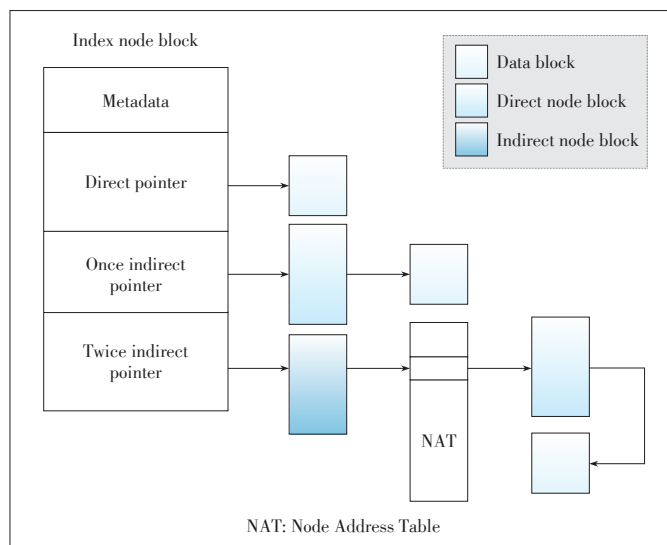
maps of blocks in main memory. `Node Address Table` is used for searching the physical address according to the node ID. `Segment Summary Area` (SSA) stores the owners' information of all blocks in main memory, such as the father node ID of one node and the offset of the node/data. The information in this section is mainly used for garbage collection. The main area includes the data of files and directories and their indexes. It also contains six logs for hot/warm/cold data and metadata.

F2FS mainly solves two following problems based on log-structured file system:

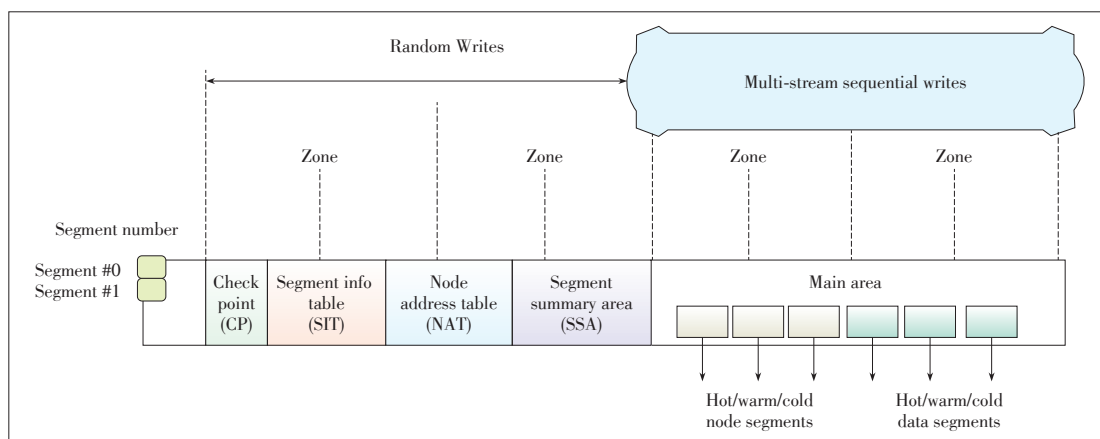
- 1) Wandering tree problem

The wandering tree problem is that when updating the block, the controller needs to update the pointer pointing to this block, update the pointer of this pointer, and then recur until the pointer pointing to the inode is updated.

F2FS solves this problem by introducing NAT, shown in Fig. 4. In traditional LFS, the ID of an inode is transferred into a physical address by Inode Map. F2FS extends this strategy. In F2FS, there are three kinds of node blocks: inode block, di-



▲ Figure 4. Node blocks in the flash friendly file system.



◀ Figure 3. Data layout of the flash friendly file system.

Persistent Data Layout in File Systems

LUO Shengmei, LU Youyou, YANG Hongzhang, SHU Jiwu and ZHANG Jiacheng

rect node block and indirect node block. Inode block includes the metadata of a file, such as the file name, the inode ID, the file size, the update time, the access time, the pointer directly pointing to the block, the pointer pointing to the direct pointer, the pointer pointing to the indirect pointer and other kinds of pointers. Each node has its own unique ID, called node ID. Through this ID, NAT can obtain the physical address of this node. For a file whose size is larger than 4 GB, LFS needs to update three pointer blocks, while F2FS only needs to update the direct node block and NAT. Actually, NAT in F2FS is in-place updated, only data in the main area are updated in the way of log-structured strategy.

2) Garbage collection

F2FS divides the data region into three levels: hot, warm and cold. Then it divides the data region into six levels by combining the division of node blocks and the division of data blocks. Compared with traditional LFS, only the log region is different in F2FS. F2FS maintains six active log regions for data to be written.

2.3 Copy-on-Write File System

The copy-on-write [13]–[15] refers to a new version of the file unit data created in a different location. Typically, an in-place file system updates the data to its original location, while the copy-on-write technology updates the data to a new location and update the file pointer. We take btrfs as an example here.

Btrfs is a Linux file system based on copy-on-write, developed by several companies. It supports a variety of advanced features and is expected to become the next generation of Linux standard file system. Btrfs supports copy-on-write, B-tree metadata management, and dynamic inode allocation.

1) Copy-on-Write

Fig. 5 shows the updating process of traditional file system data. When the file is updated, the data is written to the original location. If the system crashes, it will cause the data block to be in the semi-updated state, and destroy the consistency of the file data.

By using the copy-on-write technology (**Fig. 6**), the file will remain consistent before updating, if the system does not crash. If the crash does not occur and the file pointer is updated after the data block update is completed, the file can keep a consistent state. Therefore, copy-on-write is very effective to

maintain file consistency.

2) B-tree metadata management

For ext2, ext3 and other file systems, their directory organization hinders their scalability. In ext2/3, their directories are linearly organized; when there are too many files in one directory, the number of corresponding directory entries increases, which results in increasing lookup times. Btrfs uses B-tree to manage metadata, which solves the problem of time-consuming searching of directory entries, so it has strong expansibility.

3) Dynamic inode allocation

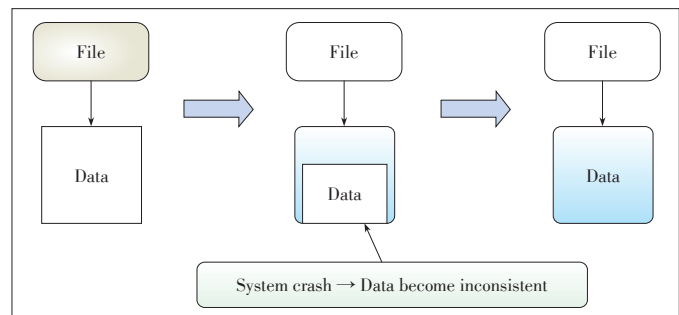
In each block group of ext2, the inode area is allocated fixedly in advance, which means that it can accommodate up to a limited number of inodes. Therefore, each partition creates a limited number of files, which may seriously affect its scalability. In btrfs, the physical storage location of an inode is no longer fixed, so users can create unlimited files anywhere. Therefore, btrfs has better scalability.

3 Persistent Storage in File System

The ultimate goal of a file system is to store a large number of data to a persistent storage in an organized way. These storage devices are different from the memory when an emergency power-off occurs: the persistent storages do not lose data while the memory will. How to realize the persistent storing is a critical issue, which can ensure the integrity and the persistency of data. The following subsections show two main kinds of persistent storages in file systems.

3.1 File System with Journaling

A journaling file system [16]–[21] uses a data structure



▲ **Figure 5. In-place update of traditional file system data.**

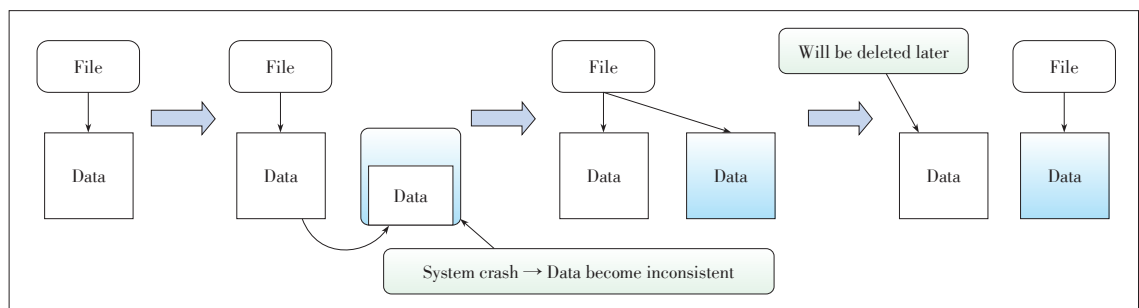


Figure 6.▶ Copy-on-write update.

named journal to record the changes of data which have not been committed to the main part of the file system.

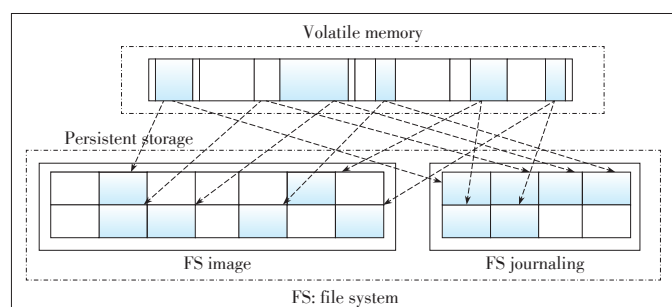
The basic structure of journal is shown in **Fig. 7**. A journaling file system can be recovered more quickly from a system crash or a power failure [18]. It may only keep the track of metadata in the actual implementation. This will improve the performance. A journaling file system may track both the metadata and the corresponding data and some implementations allow users to select the behaviors to use. No matter what the condition is, it needs several separate write operations to reflect changes of data to files when updating file systems. We take deleting a file from a file system as an example to explain why journaling is essential. Deleting a file goes three steps:

- 1) Remove the directory entry of the file
- 2) Release the inode and add the released inode to the free inode pool
- 3) Return all of disk blocks used by this file to the free disk block pool.

If there is a crash between step 1 and step 2, an orphan node occurs and a storage link happens. It is same bad when the crash happens between step 2 and step 3, because the file which has not been deleted yet will be marked deleted and something will be written on the block to cover the undeleted block.

To prevent these problems, a journaling file system provides a journal structure which records changes of data before the change operation occurs [22]. The journal in some systems can change its size dynamically like a regular file, while in other systems it has a fix size and must be allocated in a certain contiguous area. In the second situation, the journal cannot be moved and the file system is mounted. There are also some file systems that allow the journal to be allocated on external separate device, such as SSDs and other non-volatile memories (NVMs). The journal may be distributed on several storages in order to avoid device crash.

When the journal itself is being written to, the journal must guard against crashes. Many journal implementations (such as the JBD2 layer in ext4) gather each change logged with a checksum. If a crash leaves a partially written change with a missing (or mismatched) checksum, the system can simply ignore it when replaying the journal after the recovery from the crash.



▲ Figure 7. Basic structure of journaling file system.

There are two kinds of journals, one is physical journal and the other is logical journal. A physical journal is used to log copies of blocks which will be written to the file system latter. If a crash occurs when the blocks are being written to the file system, the system just needs to replay the write in the journal to complete the operation when the file system recovers from the crash. If a crash occurs when the write is being logged to the journal, the partial write will miss or mismatched checksum and can be ignored when the file system recovers from the crash. A physical journal takes a performance penalty because each block changed must be committed twice. However, this may be acceptable when absolute fault protection is required.

A logical journal is used to store changes to metadata in the journal. A file system with a logical journal can recover quickly after a crash, but may allow the inconformity of unlogged file data and logged metadata. For example, appending to a file may involve three separate writes:

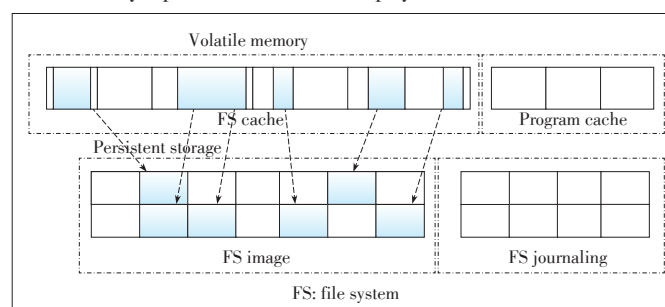
- 1) Writes to the inode of the file and note in the metadata of the file that its size has increased
- 2) Writes to the free space map and mark out an allocation of space for the data that will be appended
- 3) Writes to the newly allocated space and write the appended data actually.

In a metadata-only journal, step 3 is not logged. If step 3 is not done, but steps 1 and 2 are replayed during recovery, the file will be appended with garbage.

3.2 File System with Virtual Memory

Virtual memory [23]–[26] is a technique to manage memory. It employs both software and hardware to map virtual addresses used by a program to physical addresses. The translation hardware in CPU translates virtual addresses to physical addresses automatically. The software in a file system with virtual memory can extend capabilities by providing a larger virtual address space when more physical storages are added in the system. File systems with virtual memory divide a virtual address space into pages. Pages are blocks whose virtual memory addresses are contiguous. The size of a page is usually at least 4 kilobytes.

The basic structure of file systems with virtual memory is shown in **Fig. 8**. It is expected that applications have a continuous memory space, however, the physical blocks are actually



▲ Figure 8. Basic structure of file system with virtual memory.

Persistent Data Layout in File Systems

LUO Shengmei, LU Youyou, YANG Hongzhang, SHU Jiwu and ZHANG Jiacheng

stored dispersedly. Some blocks may even be stored in external storages; they are swapped into the memory when they will be used.

Virtual memory benefits applications by freeing them from managing a shared memory space, which will improve the security because the memory is isolated well [26]. It can also use more memory conceptually by the paging technique. When the memory is full, it will employ the persistent storage to work as an extended part of memory. In a traditional file system, a part of disk is used as the extended memory. With the rise of SSDs, there are some systems employing an SSD (due to its low cost, power efficiency and so on) as the extended memory [27]–[29], such as NVMalloc [30] and FlashVM [31].

NVMalloc was proposed to employ NVM as a secondary memory partition for applications to allocate explicitly and use memory regions in it. NVMalloc provides an NVMalloc library with a series of services, enabling applications to access NVM storage. With NVMalloc, files can be accessed in a byte-addressable fashion by using the memory mapped I/O interface. The approach in NVMalloc is able to re-energize computations outside of the core on large scale machines. This increases the capacity of the memory. NVMalloc shows that it can compute larger size of problem than the physical memory whose manner is cost-effective manner. In addition, it has better performance and efficiency when computing time or data access locality increases.

FlashVM focuses on high performance, reduced flash wear-out for improved reliability, and efficient garbage collection. It modifies the code paths for allocating/reading/writing pages in order to optimize the performance of flash. FlashVM further uses zero-page sharing and page sampling to reduce the number of page writes. It also makes full use of the discard command and provides fast online garbage collection of free VM pages.

4 Built-in Multi-Levels Persistent File System

The file layout of the flash file system not only affects the performance of the flash storage system, but also has impact on the control of life loss of flash [12], [32]–[34]. However, the data layout of the flash file system has different requirements for different operations, and the file system step-by-step operation limits the optimization of the data layout, which is mainly reflected in two aspects:

- 1) Fine-grained writes conflicts with page granularity reads. Flash memory write operations expect fine-grained writes to extend flash memory life. Flash read operations expect a page granularity read to improve read performance.
- 2) The conflict between synchronization and data layout is optimized. Because of consistency or persistence requirements, the file system provides an application that is explicitly called synchronous connections (such as fsync) or uses operating system background processes (such as pflush) to syn-

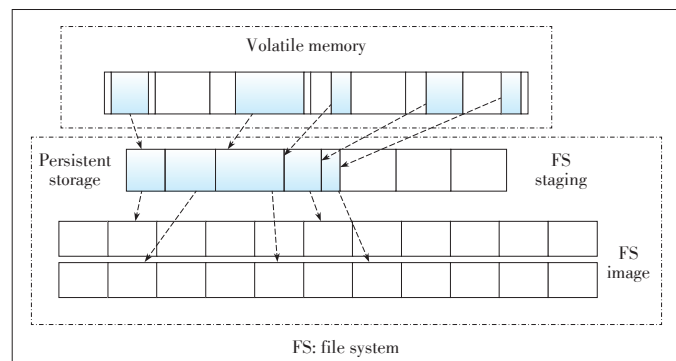
chronize data frequently to external memory. Synchronization reduces the duration between data persistence and data buffering. This reduces the probability of data merging on the same page. The update of valid data in a single shell is small. On the other hand, synchronization reduces the amount of data buffer, resulting in lacking data for effective classification, reducing the accuracy of data packets and affecting the optimization of the data layout effect.

Thus, StrageFS is proposed as a solution to these issues.

The basic structure of stageFS is shown in **Fig. 9**. The SSD managed by stageFS is divided into two spaces: FS Staging and FS Image. FS Staging provides the persistent storage for the recent write to the file system, while FS Image stores the other data in the file system. In the staging phase, only the dirty parts need to be written back into FS Staging in the way of recording. A record that has been marked with a unique ID is a dirty part in a page. In FS Staging, files are deleted in the grain of a page and the delete operation happens when the space of FS Staging is going to be used up. At this time, stageFS merges the pages in FS Staging with the pages in FS Image and patch them to FS Image. After the patching operation, stageFS erases the whole space in FS Staging.

StageFS includes two phases, the first one is staging phase and the second one is patching phase.

The staging phase is designed to provide efficient persistence to file system, for example, to write file system updates to persistent storage efficiently. The goal includes both high write performance and low writes amplification. The staging phase provides data durability to the content updates, including updates to file pages, directory entry pages, and index nodes. StageFS tracks the dirty bytes of each page instead of marking a page dirty. It records the write location of each write request, including the offset and length in each page where the request updates. When I/O synchronization is required, StageFS iterates all dirty files in the file system. For each dirty file, its dirty pages are performed using either full-page steal or record-level logging, according to their dirty granularity, hotness, etc. Full-page steal write is to steal pages from the hidden area in FS image and write dirty pages in full pages. Record-level logging write is to update data in the granularity of record and



▲ Figure 9. Basic structure of StageFS.

compact these dirty parts to the staging area. Each record has a logical ID for identification, an offset for marking the start address and a length. In the implementation of StageFS, the dirty parts of a file are tracked in the logical ID tuples in the page cache. These dirty parts are indexed in a hash table, which keeps an ordered linked list to store the tuples for each file.

The patching phase is designed to accumulate data in the input datasets and improve the effect of data layout optimization. In the patching phase, space allocation in the FS image for file system updates is performed lazily. In the staging phase, each update is appended to the staging log with only the logical ID. With the ID, its offset and length in the file are known. In the patching phase, space allocation is performed to reorganize the data into a better layout:

- Page-level indexing that is transformed from the non-indexed record-level logging
- More sequential accesses by merging and reordering random writes.

The updates in the file system are written to FS Image in the granularity of page, and are written back by using the memory copies. As memory pages of the staging data are pinned in the main memory, it does not need to scan and merge the variable length records in the staging area. Therefore, file system updates are written twice: one is to FS Staging in record level for data durability, and the other is to FS Image in page level for data indexing.

StageFS needs to ensure consistency in both the staging phase and the patching phase. In the staging phase, file system updates are written in a log-structured way. StageFS treats each synchronized write as a transaction and uses the padding record as the commit record, which indicates the end of a transaction. For a synchronized write, a new page is allocated to be the padding record. Therefore, every synchronized write has a padding record to indicate its completeness. Though the padding record is used as the commit record, there is no ordering between data/inode record writes and the padding record write. An unwritten page has all '0's, and the partially written page is detected by checking the Error Correction Code (ECC). If any page in one transaction is not written, the transaction is not committed. During recovery after failures, content updates in the staging area need to be merged with corresponding pages in the file system image. StageFS reads the updates of files or directories in the staging area, and marks their inode pages in icache as obsolete by setting their obsolete bits. StageFS delays the merge operation to the succeed I/O accesses. Therefore, I/O operations during recovery need to check the obsolete bit in icache before performing read or write operations. If the obsolete bit is set, data pages in the file system image are read to the page cache followed by the updates from the staging area. In the patching phase, StageFS pre-allocates all the space that is needed in the current patching phase when starting the patching operation. Then it writes the bitmap changes to the tail of the staging logging. Only after the bitmap changes are

persistently written, the patching writes are performed. If system fails during patching, bitmap changes are read to check the write statuses of the staging data. If the patching fails, StageFS marks all corresponding pages of the bitmap changes as invalid, and then restarts the patching phase by allocating space and writing the staging data to the FS image.

5 Conclusions

This paper introduces the data layout in file systems. First, we give the introduction of disks and SSDs. Their difference requires us to design a suitable data layout for SSDs instead of directly using the data layout in file systems based on disks. Second, we introduce three kinds of traditional data layouts in file systems and analyze their advantages and disadvantages in different circumstances. Third, we take persistent storage into consideration. We introduce journaling file system first, and then we introduce virtual memory. Besides, we give a brief introduction on SSDs used as the extended memory in virtual memory file systems. Finally, we propose a new file system based on SSDs, named as stageFS. It employs FS Staging which likes a cache in the system and each updating only writes the dirty parts of the page into FS Staging. At the moment that FS Staging is almost full, these data are being written back to FS Imaging in the grain of a page. StageFS employs the technologies performing well in SSDs and also has a new multi-level structure, archiving better performance in the file system based on SSDs.

References

- [1] M. K. McKusick, W. N. Joy, S. J. Leffler, and R. S. Fabry, "A fast file system for unix," *ACM Transactions on Computer Systems (TOCS)*, vol. 2, no. 3, pp. 181–197, 1984.
- [2] D. Hitz, J. Lau, and M. A. Malcolm, "File system design for an NFS file server appliance," in *Proc. USENIX Winter*, San Francisco, 1994, vol. 94, pp. 19–19.
- [3] S. Tweedie, "Ext3, journaling filesystem," in *Ottawa Linux Symposium*, Ottawa, Canada, 2000, pp. 24–29.
- [4] M. Cao, S. Bhattacharya, and T. Ts'o, "Ext4: the next generation of ext2/3 filesystem," in *Linux Storage & Filesystem Workshop (LSF)*, San Jose, USA, 2007.
- [5] F. Chen, D. A. Koufaty, and X. Zhang, "Understanding intrinsic characteristics and system implications of flash memory based solid state drives," in *ACM SIGMETRICS/Performance*, Seattle, USA, 2009.
- [6] G. Soundararajan, V. Prabhakaran, M. Balakrishnan, and T. Wobber, "Extending SSD lifetimes with disk-based write caches," in *USENIX Conference on File and Storage Technologies*, San Jose, USA, 2010, pp. 101–114.
- [7] N. Agrawal, V. Prabhakaran, T. Wobber, et al., "Design tradeoffs for SSD performance," in *USENIX Annual Technical Conference*, Boston, USA, 2008, pp. 57–70.
- [8] R. Card, T. Ts'o, and S. Tweedie, "Design and implementation of the second extended filesystem," in *Proc. First Dutch International Symposium on Linux*, Groningen, Netherlands, 1994.
- [9] O. Rodeh, J. Bacik, and C. Mason, "Btrfs: the linux b-tree filesystem," *ACM Transactions on Storage (TOS)*, vol. 9, no. 3, article no. 9, 2013. doi: 10.1145/2501620.2501623.
- [10] R. Y. Wang and T. E. Anderson, "XFS: a wide area mass storage file system," in *IEEE Fourth Workshop on Workstation Operating Systems*, Napa, USA, 1993, pp. 71–78. doi: 10.1109/WWOS.1993.348169.

Persistent Data Layout in File Systems

LUO Shengmei, LU Youyou, YANG Hongzhang, SHU Jiwu and ZHANG Jiacheng

- [11] M. Rosenblum and J. K. Ousterhout, "The design and implementation of a log-structured file system," *ACM Transactions on Computer Systems (TOCS)*, vol. 10, no. 1, pp. 26–52, Feb. 1992. doi: 10.1145/146941.146943.
- [12] C. Lee, D. Sim, J. Hwang, and S. Cho, "F2FS: A new file system for flash storage," in *Proc. 13th USENIX Conference on File and Storage Technologies (FAST)*, Santa Clara, USA, 2015, pp. 273–286.
- [13] Z. N. J. Peterson, "Data placement for copy-on-write using virtual contiguity," Ph.D. dissertation, University of California Santa Cruz, USA, 2002.
- [14] D. Hitz, M. Malcolm, J. Lau, and B. Rakitzis, "Copy on write file system consistency and block usage," U.S. Patent 6 892 211, May 10, 2005.
- [15] W. A. Sawdon and F. B. Schmuck, "Deferred copy-on-write of a snapshot," U.S. Patent 6 748 504, Jun. 8, 2004.
- [16] B. J. Fuller, "Single transaction technique for a journaling file system of a computer operating system," U.S. Patent 6 021 414, Feb. 1, 2000.
- [17] J. Piernas, T. Cortes, and J. M. Garcia, "Dualfs: a new journaling file system without meta-data duplication," in *ACM 16th International Conference on Supercomputing*, New York, USA, 2002, pp. 137–146. doi: 10.1145/514191.514213.
- [18] V. Prabhakaran, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau, "Analysis and evolution of journaling file systems," in *USENIX Annual Technical Conference*, Anaheim, USA, 2005, pp. 105–120.
- [19] Z. Zhang and K. Ghose, "yFS: a journaling file system design for handling large data sets with reduced seeking," in *2nd USENIX Conference on File and Storage Technologies (FAST)*, San Francisco, USA, 2003, pp. 59–72.
- [20] M. T. Jones, "Anatomy of linux journaling file systems," IBM DeveloperWorks, USA, 2008.
- [21] M. I. Seltzer, G. R. Ganger, M. K. McKusick, et al., "Journaling versus soft updates: asynchronous meta-data protection in file systems," in *USENIX Annual Technical Conference*, San Diego, USA, 2000, pp. 71–84.
- [22] V. Chidambaram, T. Sharma, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau, "Consistency without ordering," in *10th USENIX Conference on File and Storage Technologies (FAST)*, San Jose, USA, 2012, pp. 9–9.
- [23] K. Li and P. Hudak, "Memory coherence in shared virtual memory systems," *ACM Transactions on Computer Systems (TOCS)*, vol. 7, no. 4, pp. 321–359, 1989.
- [24] K. Li, "Shared virtual memory on loosely coupled multiprocessors," Yale University, New Haven, USA, Tech. Rep., 1986.
- [25] P. J. Denning, "Virtual memory," *ACM Computing Surveys (CSUR)*, vol. 2, no. 3, pp. 153–189, 1970.
- [26] A. W. Appel and K. Li, "Virtual memory primitives for user programs," in *4th International Conference on Architectural Support for Programming Languages and Operating Systems*, Santa Clara, USA, 1991, vol. 26, no. 4.
- [27] A. Badam and V. S. Pai, "SSDAllo: hybrid SSD/RAM memory management made easy," in *Proc. 8th USENIX Conference on Networked Systems Design and Implementation*, Boston, USA, 2011, pp. 211–224.
- [28] S. Kannan, A. Gavrilovska, K. Schwan, and D. Milojicic, "Optimizing checkpoints using NVM as virtual memory," in *IEEE 27th International Symposium on Parallel & Distributed Processing (IPDPS)*, Boston, USA, 2013, pp. 29–40. doi: 10.1109/IPDPS.2013.69.
- [29] M. Hadwiger, J. Beyer, W.-K. Jeong, and H. Pfister, "Interactive volume exploration of petascale microscopy data streams using a visualization-driven virtual memory approach," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2285–2294, Dec. 2012. doi: 10.1109/TVCG.2012.240.
- [30] C. Wang, S. S. Vazhkudai, X. Ma, et al., "Nvmmalloc: Exposing an aggregate ssd store as a memory partition in extreme-scale machines," in *IEEE 26th International Parallel & Distributed Processing Symposium (IPDPS)*, Shanghai, China, 2012, pp. 957–968. doi: 10.1109/IPDPS.2012.90.
- [31] M. Saxena and M. M. Swift, "Flashvm: Virtual memory management on flash," in *USENIX Annual Technical Conference*, Boston, USA, 2010.
- [32] Y. Lu, J. Shu, and W. Zheng, "Extending the lifetime of flash-based storage through reducing write amplification from file systems," in *Proc. 11th USENIX Conference on File and Storage Technologies (FAST)*, San Jose, USA, 2013.
- [33] Y. Lu, J. Shu, and W. Wang, "ReconFS: a reconstructable file system on flash storage," in *Proc. 12th USENIX Conference on File and Storage Technologies (FAST)*, Santa Clara, USA, 2014, pp. 75–88.
- [34] J. Zhang, J. Shu, and Y. Lu, "ParaFS: a log-structured file system to exploit the internal parallelism of flash devices," in *USENIX Annual Technical Conference*, Denver, USA, 2016.

Manuscript received: 2017-10-26

Biographies

LUO Shengmei (luo.shengmei@zte.com.cn) received his master's degree from Harbin Institute of Technology, China. He has been working with ZTE Corporation for over 20 years. His research interests include cloud computing and big data. He is a member of CIE and CCF.

LU Youyou (luyouyou@tsinghua.edu.cn) received the B.S. degree from Nanjing University, China in 2009 and the Ph.D. degree from Tsinghua University, China in 2015, both in computer science. He is currently an assistant researcher in the Department of Computer Science and Technology, Tsinghua University. His current research interests include nonvolatile memories and file systems. He received the best paper award at IEEE NVMSA '14 and the best paper runner-up at MSST' 15. He is a member of the IEEE, ACM and CCF.

YANG Hongzhang (yang.hongzhang@zte.com.cn) received his master's degree in computer science and technology from University of Chinese Academy of Sciences, China in 2015. He has been working with ZTE Corporation for 3 years. His research interests include distributed file system and cloud computing. His paper on pNFS was received by HPCA '15. He is a member of the IEEE, ACM and CCF.

SHU Jiwu (shujw@tsinghua.edu.cn) received the Ph.D. degree from the Department of Computer Science and Technology, Nanjing University, China. He is currently a professor in the Department of Computer Science and Technology, Tsinghua University, China. His current research interests include nonvolatile memories and file systems. He is IEEE Fellow and CCF Fellow.

ZHANG Jiacheng (zhang - jc13@mails.tsinghua.edu.cn) received the B.S. degree from Harbin Institute of Technology, China, in software engineering in 2013 and is now a Ph.D. candidate student in Tsinghua University, China, majoring in computer science. His current research interests include nonvolatile memories and storage system. His paper on flash-based file system was received by USENIX ATC '16.



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

<http://mc03.manuscriptcentral.com/ztecom>
12F Kaixuan Building, 329 Jinzhai Rd, Hefei 230061, P. R. China

ZTE COMMUNICATIONS

中兴通讯技术(英文版)

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

ZTE COMMUNICATIONS

Vol. 16 No. 3 (Issue 63)

Quarterly

First English Issue Published in 2003

Supervised by:

Anhui Science and Technology Department

Sponsored by:

Anhui Science and Technology Information Research Institute;
Magazine House of ZTE Communications

Published and Circulated (Home and Abroad) by:

Magazine House of ZTE Communications

Staff Members:

Editor-in-Chief: WANG Xiang

Executive Associate Editor-in-Chief: HUANG Xinming

Editor-in-Charge: ZHU Li

Editors: XU Ye and LU Dan

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

Annual Subscription: RMB 80

Printed by:

Hefei Tiancai Color Printing Company

Publication Date: September 25, 2018

Publication Licenses:

ISSN 1673-5188

CN 34-1294/TN