



An International ICT R&D Journal Sponsored by ZTE Corporation

ISSN 1673-5188

CN 34-1294/ TN

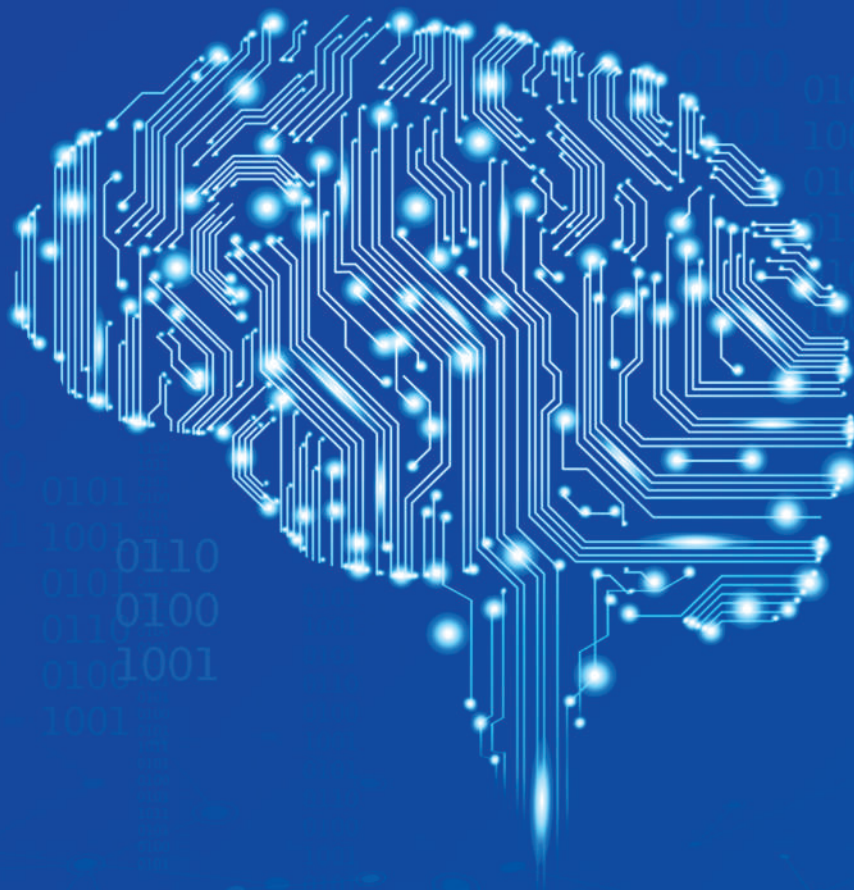
ZTE COMMUNICATIONS

中兴通讯技术(英文版)

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

June 2020, Vol. 18 No. 2

Machine Learning at Network Edges



The 8th Editorial Board of ZTE Communications

Chairman GAO Wen, Peking University (China)

Vice Chairmen XU Ziyang, ZTE Corporation (China) | XU Chengzhong, University of Macau (China)

Members (Surname in Alphabetical Order)

AI Bo	Beijing Jiaotong University (China)
CAO Jiannong	Hong Kong Polytechnic University (China)
CHEN Chang Wen	The State University of New York at Buffalo (USA)
CHEN Yan	Northwestern University (USA)
CHI Nan	Fudan University (China)
CUI Shuguang	UC Davis (USA) and The Chinese University of Hong Kong, Shenzhen (China)
GAO Wen	Peking University (China)
GAO Yang	Nanjing University (China)
GE Xiaohu	Huazhong University of Science and Technology (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)
LI Xiangyang	University of Science and Technology of China (China)
LI Zixue	ZTE Corporation (China)
LIN Xiaodong	ZTE Corporation (China)
LIU Chi	Beijing Institute of Technology (China)
LIU Jian	ZTE Corporation (China)
LIU Ming	Institute of Microelectronics of the Chinese Academy of Sciences (China)
MA Jianhua	Hosei University (Japan)
MA Zheng	Southwest Jiaotong University (China)
NIU Zhisheng	Tsinghua University (China)
PAN Yi	Georgia State University (USA)
REN Fuji	Tokushima University (Japan)
REN Kui	Zhejiang University (China)
SHENG Min	Xidian University (China)
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 Haiming	Southeast University (China)
WANG Xiang	ZTE Corporation (China)
WANG Xiaodong	Columbia University (USA)
WANG Xiyu	ZTE Corporation (China)
WANG Yongjin	Nanjing University of Posts and Telecommunications (China)
WANG Zhengdao	Iowa State University (USA)
XU Chengzhong	University of Macau (China)
XU Ziyang	ZTE Corporation (China)
YANG Kun	University of Essex (UK)
YUAN Jinhong	University of New South Wales (Australia)
ZENG Wenjun	Microsoft Research Asia (China)
ZHANG Chengqi	University of Technology Sydney (Australia)
ZHANG Honggang	Zhejiang University (China)
ZHANG Jianhua	Beijing University of Posts and Telecommunications (China)
ZHANG Yueping	Nanyang Technological University (Singapore)
ZHOU Wanlei	University of Technology Sydney (Australia)
ZHUANG Weihua	University of Waterloo (Canada)

CONTENTS

ZTE COMMUNICATIONS June 2020 Vol. 18 No. 2 (Issue 70)

Special Topic

Machine Learning at Network Edges

Editorial 01

TAO Meixia, HUANG Kaibin

Enabling Intelligence at Network Edge: 02 An Overview of Federated Learning

A comprehensive introduction of federated learning is delivered. Specifically, the authors first survey the basis of federated learning, including its learning structure and the distinct features from conventional machine learning models. They then enumerate several critical issues associated with the deployment of federated learning in a wireless network, and show why and how technologies should be jointly integrated to facilitate the full implementation from different perspectives, ranging from algorithmic design, on-device training, to communication resource management. Finally, the paper is concluded by shedding light on some potential applications and future trends.

Howard H. YANG, ZHAO Zhongyuan, Tony Q. S. QUEK

Scheduling Policies for Federated Learning in 11 Wireless Networks: An Overview

A new distributed training framework called federated learning (FL) has emerged and attracted much attention from both academia and industry. In FL, participating devices iteratively update the local models based on their own data and contribute to the global training by uploading the model updates until the training converges. Therefore, the computation capabilities of mobile devices can be utilized and the data privacy can be preserved. The authors first introduce the backgrounds and fundamentals of FL. Then, the key challenges in deploying FL in wireless networks are discussed, and several existing solutions are reviewed. Finally, the authors highlight the open issues and future research directions in FL scheduling.

SHI Wenqi, SUN Yuxuan, HUANG Xiufeng, ZHOU Sheng, NIU Zhisheng

20 Joint User Selection and Resource Allocation for Fast Federated Edge Learning

By periodically aggregating local learning updates from edge users, federated edge learning (FEEL) is envisioned as a promising means to reap the benefit of local rich data and protect users' privacy. However, the scarce wireless communication resource greatly limits the number of participated users and is regarded as the main bottleneck which hinders the development of FEEL. To tackle this issue, the authors propose a user selection policy based on data importance for FEEL system. In order to quantify the data importance of each user, they first analyze the relationship between the loss decay and the squared norm of gradient and then formulate a combinatorial optimization problem to maximize the learning efficiency by jointly considering user selection and communication resource allocation. By problem transformation and relaxation, the optimal user selection policy and resource allocation are derived, and a polynomial-time optimal algorithm is developed. Finally, the authors deploy two commonly-used deep neural network (DNN) models for simulation.

JIANG Zhihui, HE Yinghui, YU Guanding

31 Communication-Efficient Edge AI Inference over Wireless Networks

The principles of efficient deployment of model inference at network edge to provide low-latency and energy-efficient AI services are presented. This includes the wireless distributed computing framework for low-latency device distributed model inference as well as the wireless cooperative transmission strategy for energy-efficient edge cooperative model inference. The communication efficiency of edge inference systems is further improved by building up a smart radio propagation environment via intelligent reflecting surface.

YANG Kai, ZHOU Yong, YANG Zhanpeng, SHI Yuanming

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.

CONTENTS

ZTE COMMUNICATIONS June 2020 Vol. 18 No. 2 (Issue 70)

Knowledge Distillation for Mobile Edge Computation Offloading 40

Edge computation offloading allows mobile end devices to execute compute-intensive tasks on the edge servers. End devices can decide whether the tasks are offloaded to edge servers, cloud servers or executed locally according to current network condition and devices' profile in an on-line manner. The authors propose an edge computation offloading framework based on deep imitation learning (DIL) and knowledge distillation (KD), which assists end devices to quickly make fine-grained decisions to optimize the delay of computation tasks online. The authors formalize computation offloading problem into a multi-label classification problem. Training samples for our DIL model are generated in an offline manner.

After the model is trained, the authors leverage KD to obtain a lightweight DIL model, by which they further reduce the model's inference delay. Numerical experiment shows that the offloading decisions made by the proposed model not only outperform those made by other related policies in latency metric, but also have the shortest inference delay among all policies.

CHEN Haowei, ZENG Liekang, YU Shuai, CHEN Xu

Joint Placement and Resource Allocation for UAV-Assisted Mobile Edge Computing Networks with URLLC 49

An unmanned aerial vehicle (UAV) assisted mobile edge computing (MEC) network with ultra-reliable and low-latency communications (URLLC) is investigated, in which a UAV acts as an aerial edge server to collect information from a set of sensors and send the processed data to the corresponding actuators. In particular, the authors focus on the round-trip URLLC from the sensors to the UAV and to the actuators in the network. By considering the finite block-length codes, the authors' objective is to minimize the maximum end-to-end packet error rate (PER) of these sensor-actuator pairs, by jointly optimizing the UAV's placement location and transmitting power allocation, as well as the users' block-length allocation, subject to the UAV's sum transmitting power constraint and the total block-length constraint. Although the maximum-PER minimization problem is non-convex and difficult to be optimally solved, the authors obtain a high-quality solution to this problem by using the technique of alternating optimization.

ZHANG Pengyu, XIE Lifeng, XU Jie

Review

57 Adaptive and Intelligent Digital Signal Processing for Improved Optical Interconnection

To pursue the improved interconnection performance of capacity, energy efficiency and simplicity, effective approaches are demonstrated including particularly advanced digital signal processing (DSP) methods. The authors present a review about the enabling adaptive DSP methods for optical interconnection applications, and a detailed summary of the recent and ongoing works in this field. In brief, the works focus on dealing with the specific issues for short-reach interconnection scenarios with adaptive operation, including signal-to-noise-ratio (SNR) limitation, level nonlinearity distortion, energy efficiency consideration and the decision precision.

SUN Lin, DU Jiangbing, HUA Feng, TANG Ningfeng, HE Zuyuan

Research Paper

74 Crowd Counting for Real Monitoring Scene

Crowd counting is a challenging task in computer vision as realistic scenes are always filled with unfavourable factors such as severe occlusions, perspective distortions and diverse distributions. Recent state-of-the-art methods based on convolutional neural network (CNN) weaken these factors via multi-scale feature fusion or optimal feature selection through a front switch-net. L2 regression is used to regress the density map of the crowd, which is known to lead to an average and blurry result, and affects the accuracy of crowd count and position distribution. To tackle these problems, the authors take full advantage of the application of generative adversarial networks (GANs) in image generation and propose a novel crowd counting model based on conditional GANs to predict high-quality density maps from crowd images. Furthermore, they innovatively put forward a new regularizer so as to help boost the accuracy of processing extremely crowded scenes. Extensive experiments on four major crowd counting datasets are conducted to demonstrate the better performance of the proposed approach compared with recent state-of-the-art methods.

LI Yiming, LI Weihua, SHEN Zan, NI Bingbing

Serial parameters: CN 34-1294/TN*2003*Q16*82*en*P*¥ 20.00*5000*09*2020-06

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.



Editorial: Special Topic on Machine Learning at Network Edges

**Guest Editor**

TAO Meixia is currently a professor with the Department of Electronic Engineering, Shanghai Jiao Tong University, China. She received the B. S. degree in electronic engineering from Fudan University, China in 1999, and the Ph. D. degree in electrical and electronic engineering from Hong Kong University of Science and Technology, China in 2003. Her current research interests include wireless caching, edge computing, physical layer multicasting, and resource allocation. She has published over 200 peer-reviewed IEEE journal and conference papers. Dr. TAO is the recipient of the 2019 IEEE Marconi Prize Paper Award and the 2013 IEEE Heinrich Hertz Paper Award. She also receives the IEEE/CIC ICC 2015 Best Paper Award and the WCSP 2012 Best Paper Award. She served as a member of the Executive Editorial Committee of *IEEE Transactions on Wireless Communications* during 2015 – 2019. She was also on the Editorial Board of several other journals as Editor or Guest Editor, including *IEEE Transactions on Communications* and *IEEE Journal on Selected Areas in Communications*. She served as Symposium Oversight Chair of IEEE ICC 2019, Symposium Co-Chair of IEEE GLOBECOM 2018, the TPC Chair of IEEE/CIC ICC 2014 and Symposium Co-Chair of IEEE ICC 2015. She is a Fellow of IEEE.

**Guest Editor**

HUANG Kaibin received the B. Eng. (first-class honors) and the M. Eng. from the National University of Singapore, respectively, and the Ph. D. degree from The University of Texas at Austin (UT Austin), USA, all in electrical engineering. Presently, he is an associate professor in the Department of Electrical and Electronic Engineering at The University of Hong Kong, China. He has served on the editorial boards of numerous IEEE journals including *IEEE Transactions on Green Communications and Networking*, *IEEE Transactions on Wireless Communications*, *IEEE Journal of Selected Areas in Communication*, and *IEEE Wireless Communications Letters*. Dr. HUANG received several awards from IEEE Communication Society including the Best Tutorial Paper Award in 2019, two Asia Pacific Outstanding Paper Awards in 2015 and 2019, and Best Paper Awards from IEEE GLOBECOM 2006 and IEEE/CIC ICC 2018. Other recognitions include a Second Class Award in Research Achievements from China Ministry of Education in 2018, an Outstanding Teaching Award from Yonsei University, and a University Continuing Fellowship from UT Austin. He is named a Highly Cited Scientist by Clarivate Analytics in 2019.

With the proliferation of end devices, such as smartphones, wearable sensors and drones, an enormous amount of data is generated at the network edge. This motivates the deployment of machine learning algorithms at the edge that exploit the data to train artificial intelligence (AI) models for making intelligent decisions. Traditional machine learning procedures, including both training and inference, are carried out in a centralized data center, thus requiring devices to upload their raw data to the center. This can cause severe network congestion and also expose users' private data to hackers' attacks. Thanks to the recent development of mobile edge computing (MEC), the above issues can be addressed by pushing machine learning towards the network edge, resulting in the new paradigm of edge learning. The notion of edge learning is to allow end devices to participate in the learning process by keeping their data local, and perform training and inference in a distributed manner with coordination by an edge server. Edge learning can enable many emerging intelligent edge services, such as autonomous driving, unmanned aerial vehicles (UAVs), and extended reality (XR). For this reason, it is attracting growing

interests from both the academia and industry.

The research and practice on edge learning are still in its infancy. In contrast to cloud-based learning, edge learning faces several fundamental challenges, including limited on-device computation capacities, energy constraints, and scarcity of radio resources. This special issue aims at providing a timely forum to introduce this exciting new area and latest advancements towards tackling the mentioned challenges in edge learning.

To begin with, the first paper "Enabling Intelligence at Network Edge: An Overview of Federated Learning" by YANG et al. serves as a comprehensive overview of federated learning (FL), a popular edge learning framework, with a particular focus on the implementation of FL on the wireless infrastructure to realize the vision of network intelligence.

Due to the salient features of edge learning (notably, FL), such as the non independent and identically distributed (i. i. d) dataset and a dynamic communication environment, device scheduling and resource allocation should be accounted for in designing distributed model training algorithms. To this end, the second paper "Scheduling Policies for Federated Learning in Wireless Networks: An Overview" by SHI et al. provides a comprehensive survey of existing scheduling policies of FL in wireless networks and also points out a few promising relevant

➔ **To Page 30**

Enabling Intelligence at Network Edge: An Overview of Federated Learning



Howard H. YANG¹, ZHAO Zhongyuan², Tony Q. S. QUEK¹

(1. Singapore University of Technology and Design, Singapore 487372, Singapore;

2. Beijing University of Post and Telecommunication, Beijing 100876, China)

Abstract: The burgeoning advances in machine learning and wireless technologies are forging a new paradigm for future networks, which are expected to possess higher degrees of intelligence via the inference from vast dataset and being able to respond to local events in a timely manner. Due to the sheer volume of data generated by end-user devices, as well as the increasing concerns about sharing private information, a new branch of machine learning models, namely federated learning, has emerged from the intersection of artificial intelligence and edge computing. In contrast to conventional machine learning methods, federated learning brings the models directly to the device for training, where only the resultant parameters shall be sent to the edge servers. The local copies of the model on the devices bring along great advantages of eliminating network latency and preserving data privacy. Nevertheless, to make federated learning possible, one needs to tackle new challenges that require a fundamental departure from standard methods designed for distributed optimizations. In this paper, we aim to deliver a comprehensive introduction of federated learning. Specifically, we first survey the basis of federated learning, including its learning structure and the distinct features from conventional machine learning models. We then enumerate several critical issues associated with the deployment of federated learning in a wireless network, and show why and how technologies should be jointly integrated to facilitate the full implementation from different perspectives, ranging from algorithmic design, on-device training, to communication resource management. Finally, we conclude by shedding light on some potential applications and future trends.

Keywords: federated learning; edge intelligence; learning algorithm; communication efficiency; privacy and security

DOI: 10.12142/ZTECOM.202002002

<http://kns.cnki.net/kcms/detail/34.1294.TN.20200610.1007.002.html>, published online June 10, 2020

Manuscript received: 2020-02-10

Citation (IEEE Format): H. H. Yang, Z. Y. Zhao, and T. Q. S. Quek, "Enabling intelligence at network edge: an overview of federated learning," *ZTE Communications*, vol. 18, no. 2, pp. 02 – 10, Jun. 2020. doi: 10.12142/ZTECOM.202002002.

1 Introduction

The networking system is experiencing a paradigm shift from a conventional cloud computing architecture that aggregates the computational resources at a data center, to mobile edge systems which largely deploy com-

putational power to the network edges to meet the demands from mobile applications—which are most thriving today—and support resource-constrained nodes reachable only over unreliable network connections^[1]. Moreover, along with the burgeoning progress of machine learning research, it is expect-

ed that by integrating machine learning algorithms to the edge nodes, future networks will be able to utilize local data to conduct intelligent inference and control on many activities, e.g., learning activities of mobile phone users, predicting health events from wearable devices, or detecting burglaries within smart homes^[2].

However, as the data is usually generated at the end-user devices, the sheer volume of the dataset as well as the rising concerns about sharing private information often makes the users reluctant to send their raw data to the edge server for the training of any model—even that can eventually benefit them in return. In response to this dilemma, a new machine learning model has emerged, namely federated learning, that allows decoupling of data acquisition and computation at the central unit^[3–5]. Specifically, rather than collecting all the data to a central unit for training, federated learning brings the models directly to the end-user devices for training, where only the resultant parameters shall be sent to the edge servers that reside in an edge node. This salient feature of on-device training brings along great advantages of eliminating the large communication overheads as well as preserving data privacy, and hence making federated learning particularly relevant for mobile applications. These properties also identify the federated learning as one of the most promising factors to an intelligent mobile edge network^[6–9].

Nevertheless, in order to deliver a successful deployment of federated learning, one also needs to tackle new challenges that require a fundamental departure from the standard methods designed for distributed optimization^{[3],[10]}. Particularly, unlike many traditional machine learning models, where an algorithm runs on a large dataset partitioned homogeneously across multiple servers in the cloud, the federated learning often operates in a mobile edge system, in which a server orchestrates the training with a union of end-user devices, which have non independent and identically distributed (i.i.d.) and unbalanced dataset, and communicate over a resource-limited spectrum^[11–12]. In that regard, the staleness becomes more paramount to the training process^[13] and security issues also arise that make the learning architecture vulnerable^[14]. Addressing these issues requires joint studies from many aspects, including the learning algorithm, system design, and communication and information theory^[15–16]. In response, Ref. [10] discussed the possible directions to improve the training efficiency when encountering with heterogeneous datasets. Moreover, Ref. [6] investigated the end-to-end latency, reliability, and scalability of a federated learning empowered edge network. In the particular context of deep learning, Ref. [8] explored the challenges and approaches to integrate the learning algorithm into network edge via a federated approach; Ref. [9] discussed a number of guidelines for the implementation of federated learning with the wireless channels. With these efforts, the results are fruitful: As will be detailed in Section 4, there are numerous applications that can benefit a lot by adopting federated learn-

ing. To that end, the central thrust of this paper is to deliver a comprehensive introduction to the federated learning system as well as to appeal for more research devoted into this emerging field. It is also noteworthy that while a few surveys on the topic of federated learning have been now available, our work puts a particular focus on the integration of the wireless infrastructures (such as the mobile edge network) as a supporting platform and the federated learning as an operation system, which ultimately achieves the network intelligence by jointly running them together.

The remainder of this paper is organized as follows. In Section 2, we introduce the basic structure and the defining characteristics of a federated learning model. The techniques to the core of a practical implementation of the federated learning system are elaborated in Section 3. Section 4 discusses the potential applications and future trends of federated learning, followed by the conclusion remarks in Section 5.

2 Federated Learning: Basis and Properties

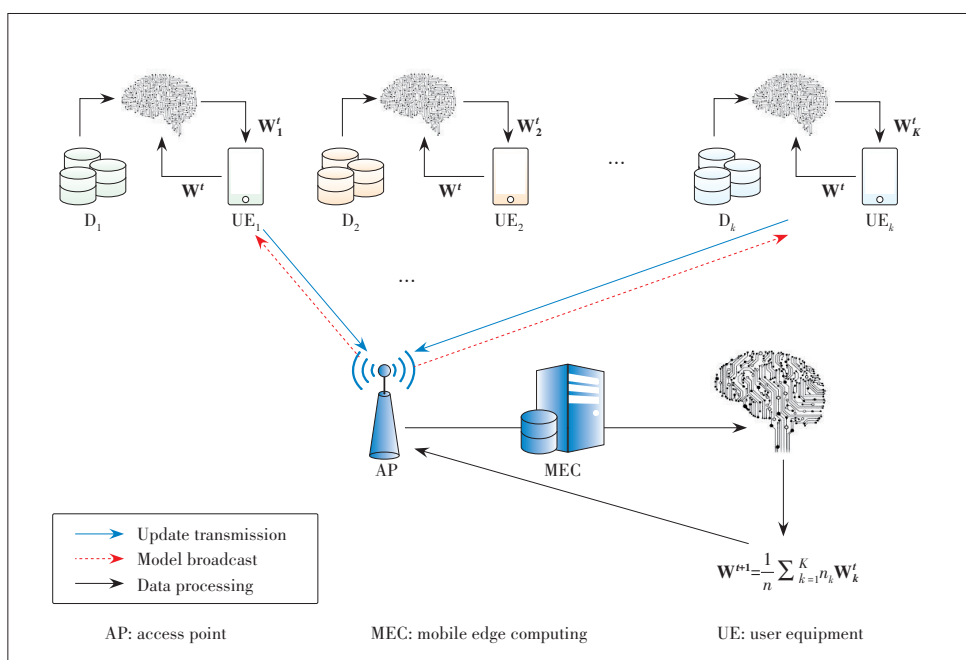
In this section, we detail the basic architecture of a federated learning model running on the mobile edge system. A number of key features associated with such a setting will also be presented.

2.1 Basic Architecture

As illustrated in **Fig. 1**^[17], the network elements involved in the federated learning include a central unit, e.g., the edge server that resides at a base station or access point and a number of end-user devices, in which they collaboratively learn a statistical model. The model is typically devised by a model engineer for a particular application, with which the server then orchestrates the training process with the end-user devices by repeating the following steps^[3–4].

- 1) Client selection: The server selects from a subset of its clients, namely the end-user devices, which meet the eligibility requirements, e.g., mobile phones or tablets that currently have a wireless connection, for one round of training.
- 2) Broadcast: The selected clients download the current model, including the weights and a training program, from the server for local computing.
- 3) End-user computation: Each selected device performs a local computation, usually in the form of stochastic gradient descend (SGD), for a given period, and uploads the resultant parameters to the server.
- 4) Update aggregation: The server collects the updates from the end-user devices—in the form of either trained parameters or gradients—and aggregates, in general by a weighted average, the collected results.
- 5) Model update: The server locally updates the shared model based on the aggregated update computed from the clients that participated in the current round.

After a sufficient number of training and update exchanges



▲ Figure 1. Illustration of the network architecture, in which a mobile edge system is integrated with federated learning.

(usually termed as communication rounds) between the server and the clients, the global statistical model is able to converge to its optimal and the end users can benefit from a collaboratively learned model.

1) The advantage: By training via federated learning, end users are able to directly download the model, perform computing on the devices, and send back the resultant trained parameters; in this way, the end users decouple the necessity of sharing local data and hence reserves privacy. Additionally, the local training also abbreviates the upload of raw data, which can be very large in size and consume a lot of energy for the upload. To that end, the federated learning is particularly relevant to wireless applications.

2) The challenge: The potential drawback of federated learning is also obvious. As the training is at a large scale amongst heterogeneous entities, e.g., different end terminals can have various processing power and communication conditions, the learning efficiency can be much lower than that in a data center. On top of this, the communication is often unreliable in the federated learning environment and security issue is more paramount under such a setting.

In the sequel, we will point out the possible directions to overcome the crux and finally realize the potential of federated learning. Before that, let us pause a while and clarify the most distinguishing features of such a learning model.

2.2 Distinguishing Features

At the first sight, it might seem that the federated learning is simply another format of distributed learning. These two machine learning models share several properties in common; for

instance, the computing is carried out by a number of end terminals and the terminals iteratively collaborate via a central entity. However, there are many more features that distinguish the federated learning from those more conventional models. We highlight the key features of federated learning as follows.

- **Non-i.i.d. dataset:** The most distinct feature of federated learning is that the dataset of each end-user device is highly personalized and hence the dataset is usually non-i.i.d. across users. The sources of the dependence and non-identicalness are due to the fact that data collected at each device corresponds to a particular user, a particular geographic location, and/or a particular time period.

As such, unlike situations in the conventional setup where the dataset is completely shuffled and i.i.d., in federated learning, the non-i.i.d. structure may lead to the local minimum of each device diverting from the global minimum, and requires a rethinking of learning model to take into account such differences in the process.

- **Unbalanced data size:** Aside from being non-i.i.d. distributed, the dataset of each end-user device also differs in size. Therefore, the training procedure at each end terminal can be highly unbalanced, because some terminals that have small datasets can complete the training in a short period of time, while those with large dataset sizes may take a longer time to complete the local training. Moreover, due to the unbalanced nature, some devices, e.g., those with a large dataset, may contribute more to the overall model than others, and hence how to account for such difference in the learning algorithm is also important.

- **Limited communication resources:** As the communications between end-user devices and a central entity often take place at the network edge, where spectrum is the medium to conduct communications, the transmissions are by nature unreliable. Moreover, as the wireless resources are usually limited, it is necessary to select the appropriate number of users each round for the communication. All these can impose more significant impact of staleness on the overall training efficiency.

- **Privacy/security issues:** Whilst learning under the federated setting abbreviates the sharing of local data, it does not promise a perfect protection of privacy. In fact, one can still extract leaky information from upload parameters and retrieve the original information to an approximation extend^[11]. Moreover, under the federated setting, the end-devices are more

vulnerable to malicious attacks in this case and it is easily for some adversary users to inject malicious information into the system.

Note that a marked property of many of the features/problems discussed above is that they are inherently interdisciplinary and solving them likely requires not just machine learning, but also techniques from distributed optimization, security, differential privacy, fairness, compressed sensing, systems, information theory, statistics and more. In fact, many of the hardest problems are at the intersections of these areas and hence a cross-area study/collaboration is essential to ongoing progress.

3 Towards Practical Implementation

As mentioned in the previous section, despite the potentials to endow the mobile edge network with a higher degree of intelligence via federated learning, it requires a full cooperation between computing and communication to realize the full potential of such a scheme. In this section, we elaborate several key aspects that we believe to be lying at the core of achieving the final goal.

3.1 Efficient Learning Algorithms

The primary factor to the implementation of federated learning is an efficient algorithm. Due to the non-i.i.d. nature of the dataset, a model training process of the federated learning can be very different from the conventional counterparts. In particular, unlike scenarios under the distributed computing, where each end terminal possesses a statistically identical model (namely the empirical loss function), in the federated learning, each end-user device can have very different empirical loss due to the personalized dataset. As such, the local minimum may differ from the global minimum and the learning algorithm shall be reengineered to account for this fact^[10]. Besides, as the communication resource is limited, the edge server can only choose a subset of users for the update in each round of communication. Therefore, how to select users appropriately also plays a critical role in the overall learning efficiency^[12].

3.1.1 Optimization and Model Aggregation

Because of the non-i.i.d. nature of user dataset, treating all samples equivalently at the global model may not make a solid sense. Therefore, how to craft a more appropriate objective function is an important aspect to research. Besides, the current state-of-the-art training is mostly SGD-base, which is well-known for slow converging. Therefore, how to develop more effective algorithm will also determine the efficiency of federated learning. Moreover, owing to the vast number, each device is likely to participate only a few rounds in the training of a global model, so stateless algorithms are necessary to investigate.

In the aggregation stage, the common approach is the Feder-

ated Averaging algorithm, an adaption of parallel SGD that takes a weighted average of the collected parameters according to their dataset size. While the effectiveness of such an approach has been demonstrated in different models, it is still unknown whether this is the optimal way of aggregating parameters and further investigation is necessary.

3.1.2 Sampling and Client Selection

Due to the unbalanced structure of datasets as well as the limited bandwidth, the sampling, of not just the data points for computing but also the clients to conduct local trainings in each communication round, plays a critical role that determines the overall learning efficiency. In particular, as each end-user device may correspond to a specific local minimum of empirical loss, spending a lot of time on the local training may bear the risk of leading the parameters to diverge from the global minimum. On the other hand, as the global communication can take up a much longer period than the local computing, it is also desirable to reduce the communication rounds. As such, how to strike a balance between local computing and global communication is important to the efficiency of federated learning. In response, it is suggested that the sampling data size of each local training shall be adaptively adjusted across the global learning period.

On top of the sampling of dataset for local training, in the global aggregation stage, the edge server can only select a portion of users out of the total due to the limited bandwidth available. Therefore, for the client, i.e., end-user device, selection is also critical for the performance of federated learning. In the context of mobile edge system, it has been shown that by taking the channel quality into consideration and selecting the end-user devices with the best channel qualities, the learning efficiency can be effectively boosted up^[12], as demonstrated in **Fig. 2**. Besides, it is also important to take into account the staleness and the significance of updates in the client selection stage^[17].

3.2 Model Compression

Although the processing power of mobile devices has surged over the last decade by the hardware revolution, these terminals are still subject to power and storage constraints, making it problematic to deploy the federated learning toward a deep and large scale. The difficulty mainly attributes to two reasons. One is that a deep neural network often consists of an abundant amount of activation units and interconnecting links, and hence training such a model will inevitably incur excessive energy consumption and, if not worse, memory occupation. The other is that, even the task of model training can be accomplished at the user side, sending the resultant parameters, which are generally high dimension vectors, to the server requires not just high transmit power but also wide mobile spectrum, which imposes very high communication cost. Nonetheless, this does not mean one has no hope to adopt the most

fruitful achievement of machine learning, namely the deep neural network, in the federated setup. Two powerful approaches shed the light for overcoming the setbacks:

1) **Architecture compression:** This approach aims to save the cost from the computing perspective of neural network via pruning the connecting links and shrinking the size of the network^[7]. The idea of link pruning stems from the fact that the majority of links connecting different layers of neurons are usually associated with very small weights. In other words, the most effective component of a neural network is architecturally sparse. Therefore, it is feasible to mute a number of links that have small weights—so as to skimp on the caching memory—without affecting the overall accuracy. Moreover, despite the unprecedented success brought by deep learning, there are many applications in which using a small neural network is able to achieve as good the performance as a large one. As such, directly reducing the size of neural network at the user side is also an appropriate choice to attain marked savings in both energy and memory consumption.

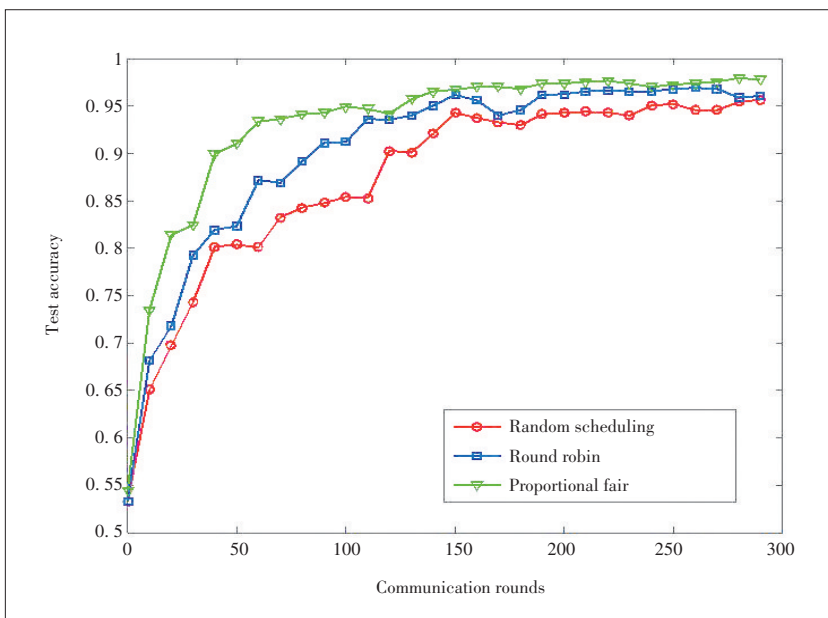
2) **Gradient compression:** This approach tackles the issue from the perspective of communication, by trading the estimation accuracy for better communication efficiency. In particular, by noticing that practical applications of machine learning often do not require very high accuracy, one can compress the high-dimension trained gradients (which can include millions of coefficients) into low dimension surrogates via different levels of quantization^[18–19]. As a result, the packet size to encapsulate the trained results can be significantly reduced, which not only saves the radiated power at each device, but also facilitates the decoding process at the server. It is noteworthy that to balance the tradeoff between communication cost and training accuracy, the level of quantization shall be adapted to

the particular location of a user. For instance, for users located in proximity to the edge node, they can conduct less quantization and maintain the high accuracy of the results, while for those located far away, they shall compress the trained results more aggressively in order to succeed the communication and engage in the training process.

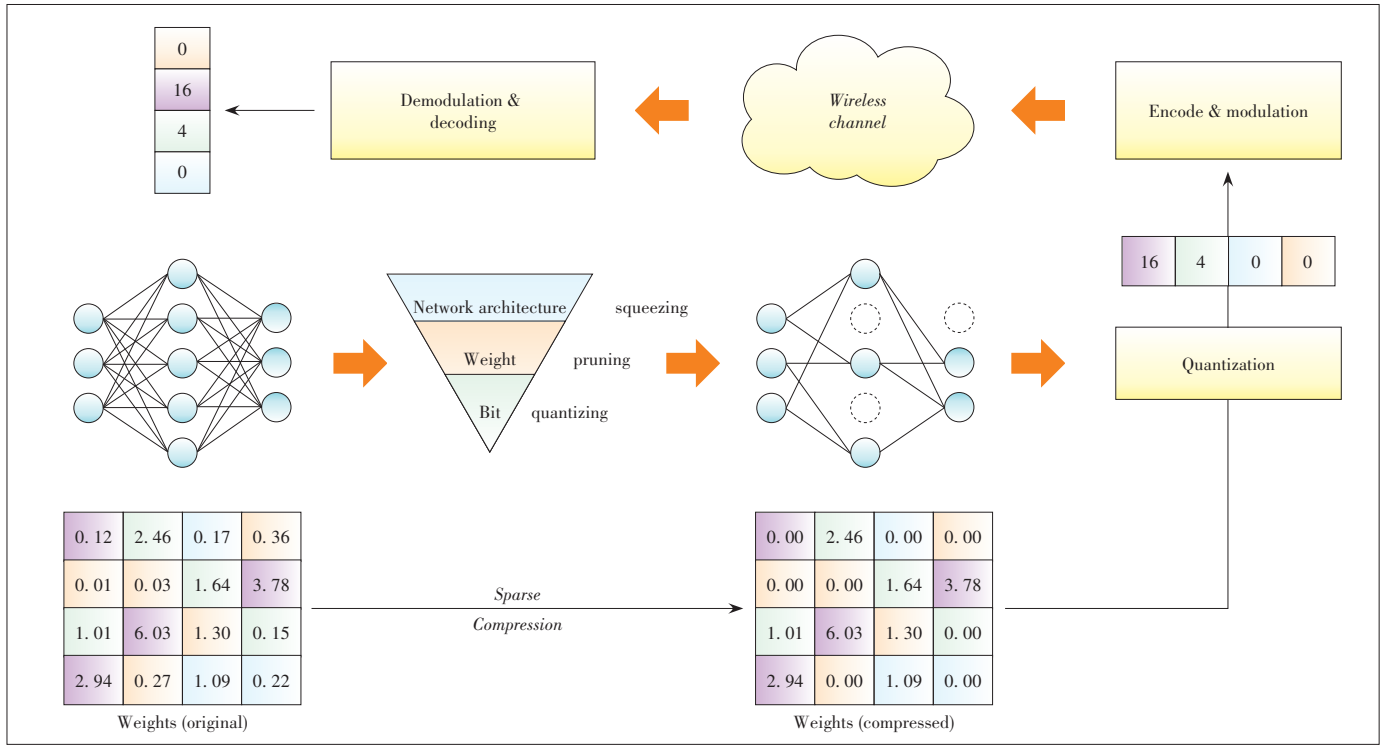
A complete process of model compression is illustrated in **Fig. 3**; we can see that it is feasible to remove a number of links with small weights in the neural network. Moreover, some neurons with only a few connections can also be muted. The architecture compression can thus transform the learning model into a sparse version, which can achieve almost the same performance as the original neural network. Another part is associated with the gradient compression, as the generated forms of parameters are often continuous with long digits, which are not suitable for the transmissions via wireless channels. By using appropriate quantization methods, the data volume of the update results can be significantly reduced, which not only saves the power consumption of end-user devices, but also facilitates the decoding procedure at the server side. To mitigate the impact of quantization noise, sophisticated parameter strategies are also necessary to minimize the model accuracy loss. It is worthwhile to mention that due to potential failure and retransmissions, the weights before and after the encoding/decoding process may appear in different orders. Nonetheless, the server can still leverage the sequential number to rearrange the weights before the global aggregation.

3.3 Advanced Communication and Networking Techniques

It has become a consensus that the communication efficiency is also one of the first-order concerns of federated learning, particularly due to the fact that the training involves a vast number of end-user device communications through a limited wireless bandwidth. In that respect, the technologies that enhance the spectral efficiency can be a critical solution to this dilemma. Specifically, the development of new technologies, e.g., the massive multiple input multiple output (MIMO), full duplex or non-orthogonal multiple access (NOMA), that are able to support more channel accesses over the same bandwidth will facilitate the deployment of federated learning. For instance, by deploying an excessive number of antennas at the base station, multiple devices can be simultaneously selected for parameter update in each round of communication, which, as demonstrated by a number of literatures, can help accelerate the convergence of federated learning algorithm. In a similar spirit, one can also leverage the techniques from full duplex or NOMA to increase the number of updates collectible in each global aggregation and hence speedup the



▲ Figure 2. Test accuracy of federated learning under different scheduling policies.



▲ Figure 3. Basic flow of model compression in the federated learning system.

training process. Besides, the ultra-reliable low latency communication (URLLC) that reduces the latency in the transmission is also a good candidate for more real-time learning tasks. A joint design that takes in the processing power and communication capability from both sides will also enhance the operation efficiency^{[15–16], [20–22]}.

Aside from communication efficiency, advanced networking technology is also important for the federated learning. In general, the federated learning involves a central server that orchestrates the training process and receives the contributions of end-user devices. Being as a central player, the server also represents a potential point of great failure^[10]. As such, even though large companies or organizations can take this role in certain applications, a reliable and powerful central server may not always be available in more collaborative learning scenarios. Moreover, the server may even become a bottleneck when the number of clients is very large. To that end, it is suggested to replace communication with the server by a more distributed manner, namely peer-to-peer communication between individual devices. For that reason, advanced device-to-device (D2D) communication and interference management schemes can be a dominant factor to the overall performance. The self-organized networking techniques may have significant influence on the performance.

3.4 Privacy Preserving Technologies

Despite the raw data is not explicitly shared in the context of federated learning, it is still possible for adversaries to retrieve the original information to an approximation extent, es-

pecially when the learning architecture and parameters are not completely protected. In fact, due to the share nature of wireless medium, the intermediate results such as parameter update from an optimization algorithm are exposed during the transmission, which may leak out private information. Moreover, the existence of malevolent users may incur further security issues. Therefore, the design of federated learning into a mobile edge system needs further protection of parameters as well as investigations on the tradeoffs between the privacy security-level and the system performance^[23].

In the federated learning process, there exists several fatal points that have privacy and security issues. We enumerate them into the following categories^[14].

3.4.1 Privacy Protection at User Side

In a federated learning algorithm, end users need to iteratively upload their learning results to the edge server for global aggregation, but these users may not trust the server since a curious entity might take a look at the uploaded parameter to infer the underlined information. To address this concern, the end users can employ some privacy-preservation technologies as follows.

1) **Perturbation:** The idea of perturbation is adding noise to the uploaded parameters by clients. This line of work often uses differential privacy^[24] to obscure certain sensitive attributes until the third party is not able to distinguish the individual, thereby making the data impossible to be restored so as to protect user privacy.

2) **Dummy:** The concept of dummy method stems from the

location privacy protection. By sending dummy model parameters along with the true one to the server, the end users can thus hide their contribution during training. Because of the aggregation processed at the server, the system performance can still be guaranteed.

3.4.2 Privacy Protection at Server Side

After collecting the updated parameters from the end-user devices, the server generally performs a weighted average to produce a new model. However, when the server broadcasts the aggregated parameters back to the users, the information may leak out as there may exist eavesdroppers. Thus, protections at the server side are also of significance.

1) Privacy-enabled aggregation: While the general purpose of aggregation at the server side is to produce an improved learning model, it is possible to scramble parameters before aggregating or enlarging the set of collected clients, which can prevent the adversaries or untrusted server from inspecting client information according to the aggregated parameters.

2) Secure multi-party computation (SMC): The central idea of SMC is to use encryption to increase protection of user updates, instead of only revealing the sum after a sufficient number of updates. Specifically, SMC is a four-round interactive protocol optionally enabled during the reporting phase of a given communication round. In each protocol round, the server gathers messages from all devices, and then uses the set of device messages to compute an independent response and return to each device. The third round constitutes a commit phase, during which devices upload cryptographically masked model updates to the server. Finally, there is a finalization phase during which devices reveal sufficient cryptographic secrets to allow the server to unmask the aggregated model update.

3.4.3 Security Protection for Learning Framework

This aspect mainly considers the model stealing attacks. In particular, any participant in the training process may introduce hidden backdoor functionality into the global model, e.g., to ensure that an image classifier assigns an attacker-chosen label to images with certain features, or that a word predictor completes certain sentences with an attacker. Consequently, there are also some protecting measures on the security design for this.

1) Homomorphic encryption: Homomorphic encryption aims to protect the parameter exchange process via encryption mechanism, by means of encoding the parameters before upload, and to transmit along with the public-private decoding keys for the intended entity to decipher.

2) Back-door defender: This is a crucial issue with the federated learning, as a malicious user may act as an innocent user but injecting certain parameters to pollute the global parameter. In con-

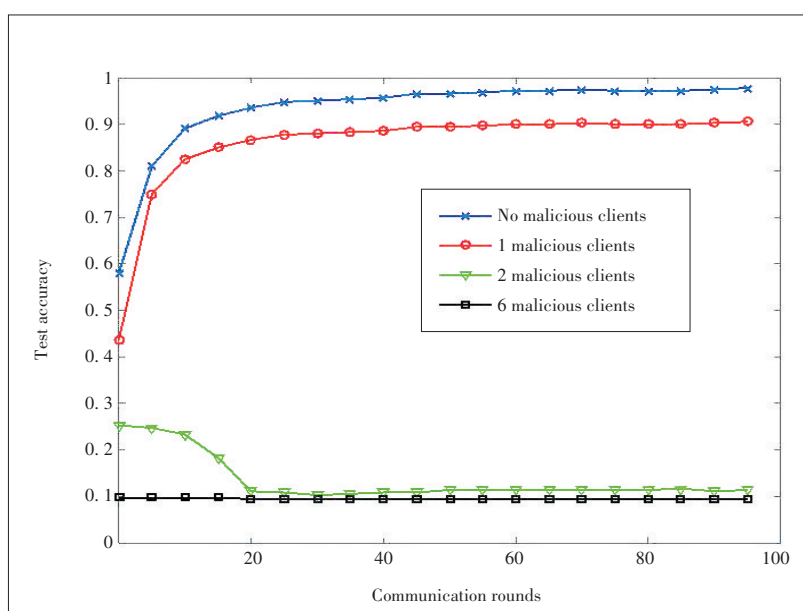
sequence, other end-user devices may encounter severe malfunctioning and breakdown. Therefore, effective approaches shall be developed to protect the users from these attacks.

In order to illustrate the impact of malicious attacks on the performance of federated learning, we carry out an experiment (Fig. 4^[14]). Particularly, a convolutional neural network (CNN) is set up with 30 end-user devices participated in, whereas the malicious clients will upload fake values of parameters in each communication round. It can be seen that the system performance can significantly curtail by malicious attacks, and even enter a breakdown when there are too many malicious clients participating in. As such, security is of significant to the performance of federated learning.

While we have listed out several concerns on the implementation of federated learning and the approaches to address these issues, another important practical consideration for federated learning is the composability of these methods. The schemes of tackling each of these aspects shall not be devised in isolation but need to be combined with each other. For instance, the efficient learning algorithm will need to be designed in consideration of learning efficiency as well as privacy preserving. Also, the model compression shall also be contended with privacy preserving.

4 Potential Applications and Future Trends

The future trends of mobile edge networks are to integrate the supply and demand of services, being able to identify a particular application to the network and respond promptly. By employing the federated learning as an operational system to the network architecture, a more intelligent network system can be foreseen in the future^[2].



▲ Figure 4. Performance of federated learning with different malicious users.

From the perspective of network architecture, the federated learning can be integrated with content caching and edge computing at the edge of a mobile network to reduce backhaul traffic loads. The general idea of caching at the network edge is, when availed with a priori information of each individual preference distributions, to optimally place the desired content resource in the edge server so as to respond to user request more swiftly. It thus simultaneously enhances the energy efficiency, reduces the service latency, and relieves the backhaul load. Despite such benefits, the gain from caching alone is only pronounced when the users' preference distributions are a priori and highly homogeneous, i.e., the users tend to request the same contents. These two constraints, however, are less likely to be satisfied in next-generation wireless applications that possess a higher degree of heterogeneity. On one hand, the users' preference distributions vary drastically across time and space, thus making them extremely difficult to be estimated and tracked, especially when the number of mobile devices becomes large. On the other hand, in practice, the users' preference distributions are highly diverse due to the personality differences. Therefore, conventional model-base designs may not be suitable for such a task because it is not capable of considering multitude of factors that influence content popularity. Moreover, directly accessing the privacy-sensitive user data for content differentiation may not be possible in practice. Federated learning with the premise of utilizing the locally trained models rather than directly accessing the user data seems to be a match made in heaven for content popularity prediction in proactive caching in wireless networks. For instance, in augmented reality (AR), federated learning can be used to learn certain popular elements of the augmentations from the other users without obtaining their privacy-sensitive data directly. This popular information is then pre-fetched and stored locally to reduce the latency.

From the perspective of resource management^[6], the federated learning paradigm can be used to improve the spectrum sensing efficiency, and thus flexible and adaptive sharing and reuse strategies can be implemented to the communication system. Apart from the radio access, the next-generation network needs to deal with more volatile traffic conditions. Along with the warp speed of progress of mobile applications, different types of traffic, which may be bursty, long-lasting, or with short packet size, coexist in the network. Consequently, centralized strategies, where information about traffic pattern is gathered in the database of a server to infer the circumstance, may not always be appropriate. Therefore, the future of network traffic management will be dependent on the decentralized training approaches such as the federated learning. In this context, the on-device training can provide more real-time reaction to schedule the traffic of the most appropriate users. A specific instance of application is the coexistence of dedicated short-range communication and

cellular-connected vehicle-to-everything in the same intelligent transport systems.

Finally, from the perspective of end user applications, federated learning is expected to find many landing grounds. For instance, by equipping sensors with federated learning algorithms, one can construct a local Internet-of-Things (IoT) network with intelligent monitoring system that can quickly identify certain events and quickly respond to them. Hospitals, if endowed with a federated learning system for disease monitoring, might increase the doctors' intention to share information and prevent certain catastrophe in the early stage. In the area of retailing, the federated learning system can leverage data from a wide range of entities to increase the accuracy of prediction on demands, and thus help providers/owners prepare supplies in a proper manner. In self-driving cars, information related to traffic can be learned through vehicles on the road using federated learning and stored in the roadside units, which facilitates the efficiency of an autonomous driving operation system.

Notably, a number of future studies immediately follow from the above discussions. For instance, one can investigate how to adopt the federated learning to inference the distribution of local demand so as to provide appropriate guidance on the allocation of caching contents on the network edge that can reduce communication burden. In the context of mobile resource management, how to leverage the federated learning to extract the individual traffic distributions to further benefit the allocation of global spectral resources is also a concrete direction. To sum up, the integration of federated learning and mobile edge network can provide a unified platform to support a variety of applications, and we also advocate for subsequent studies to build up the federated intelligence ecosystem.

5 Conclusions

In this paper, we provided an overview to the federated learning system. Specifically, we elaborated the basic architecture of the federated learning model and the salient features, in particular the non-i.i.d. and unbalanced dataset, unreliable and limited communication resource, as well as privacy and security issues, that distinguish it from the conventional ones. Furthermore, we presented a number of practical approaches that enable the implementation of federated learning into a mobile edge system. Among them, we emphasized the importance from aspects of algorithm design, model compression and communication efficiency. Lastly, we presented several applications that are most foreseeable to benefit from applying federated learning. In summary, we believe that federated learning is one of the building blocks in achieving an intelligent network and we expect that more interesting research issues will appear in this area.

References

- [1] MAO Y Y, YOU C S, ZHANG J, et al. A survey on mobile edge computing: the communication perspective [J]. *IEEE communications surveys & tutorials*, 2017, 19(4): 2322 – 2358. DOI: 10.1109/comst.2017.2745201
- [2] LETAIEF K B, CHEN W, SHI Y M, et al. The roadmap to 6G: AI empowered wireless networks [J]. *IEEE communications magazine*, 2019, 57(8): 84 – 90
- [3] KONEČNÝ J, MCMAHAN H B, YU F, et al. Federated learning: strategies for improving communication efficiency [EB/OL]. (2016-10-18) [2019-09-17]. <https://arxiv.org/abs/1610.05492>
- [4] MCMAHAN H B, MOORE E, RAMAGE D, et al. Communication - efficient learning of deep networks from decentralized data [EB/OL]. (2016-02-17) [2019-09-17]. <https://arxiv.org/abs/1602.05629>
- [5] SMITH V, FORTE S, MA C X, et al. CoCoA: a general framework for communication - efficient distributed optimization [J]. *Journal of machine learning research*, 2018, 18(230): 1 – 49
- [6] PARK J, SAMARAKOON S, BENNIS M, et al. Wireless network intelligence at the edge [J]. *Proceedings of the IEEE*, 2019, 107(11): 2204 – 2239. DOI: 10.1109/jproc.2019.2941458
- [7] ZHAO Z Y, FENG C Y, YANG H H, et al. Federated-learning-enabled intelligent fog radio access networks: fundamental theory, key techniques, and future trends [J]. *IEEE wireless communications*, 2020, 27(2): 22 – 28. DOI: 10.1109/mwc.001.1900370
- [8] ZHOU Z, CHEN X, LI E, et al. Edge intelligence: paving the last mile of artificial intelligence with edge computing [J]. *Proceedings of the IEEE*, 2019, 107(8): 1738 – 1762. DOI: 10.1109/jproc.2019.2918951
- [9] ZHU G X, LIU D Z, DU Y Q, et al. Toward an intelligent edge: wireless communication meets machine learning [J]. *IEEE communications magazine*, 2020, 58(1): 19 – 25. DOI: 10.1109/mcom.001.1900103
- [10] KAIROUZ P, MCMAHAN H B, AVETET B, et al. Advances and open problems in federated learning [EB/OL]. (2019-12-10) [2019-09-17]. <https://arxiv.org/abs/1912.04977>
- [11] WANG S Q, TUOR T, SALONIDIS T, et al. Adaptive federated learning in resource constrained edge computing systems [J]. *IEEE journal on selected areas in communications*, 2019, 37(6): 1205 – 1221
- [12] YANG H H, LIU Z, QUEK T Q S, et al. Scheduling policies for federated learning in wireless networks [J]. *IEEE transactions on communications*, 2020, 68(1): 317 – 333
- [13] DAI W, ZHOU Y, DONG N Q et al. Toward understanding the impact of staleness in distributed machine learning [C]//International Conference for Learning Representations (ICLR). New Orleans, Louisiana, 2019: 1 – 6
- [14] MA C, LI J, DING M, et al. On safeguarding privacy and security in the framework of federated learning [J]. *IEEE network*, 2020: 1 – 7. DOI: 10.1109/mnet.001.1900506
- [15] TRAN N H, BAO W, ZOMAYA A, et al. Federated learning over wireless networks: optimization model design and analysis [C]//IEEE Conference on Computer Communications (INFOCOM). Paris, France, 2019. DOI: 10.1109/infocom.2019.8737464
- [16] CHEN M Z, YANG Z H, SAAD W, et al. A joint learning and communications framework for federated learning over wireless networks [EB/OL]. [2019-09-17]. <https://arxiv.org/pdf/1909.07972>
- [17] YANG H H, ARAFA A, QUEK T Q S, et al. Age-based scheduling policy for federated learning in mobile edge networks [C]//IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Barcelona, Spain, 2020. DOI: 10.1109/icassp40776.2020.9053740
- [18] DU Y Q, YANG S, HUANG K B. High-dimensional stochastic gradient quantization for communication-efficient edge learning [J]. *IEEE transactions on signal processing*, 2020, 68: 2128 – 2142.
- [19] ZHU G X, DU Y Q, GÜNDÜZ D, et al. One-bit over-the-air aggregation for communication-efficient federated edge learning: design and convergence analysis [EB/OL]. [2020-01-16]. <https://arxiv.org/pdf/2001.05713>
- [20] ZHU G X, WANG Y, HUANG K B. Broadband analog aggregation for low-latency federated edge learning [J]. *IEEE transactions on wireless communications*, 2020, 19(1): 491 – 506. DOI: 10.1109/twc.2019.2946245
- [21] YANG K, JIANG T, SHI Y M, et al. Federated learning via over-the-air computation [J]. *IEEE transactions on wireless communications*, 2020, 19(3): 2022 – 2035. DOI: 10.1109/twc.2019.2961673
- [22] AMIRI M M, GUNDUZ D. Machine learning at the wireless edge: distributed stochastic gradient descent over-the-air [J]. *IEEE transactions on signal processing*, 2020, 68: 2155 – 2169
- [23] PHONG L T, AONO Y, HAYASHI T, et al. Privacy-preserving deep learning via additively homomorphic encryption [J]. *IEEE transactions on information forensics and security*, 2018, 13(5): 1333 – 1345. DOI: 10.1109/tifs.2017.2787987
- [24] DWORK C, MCSHERRY F, NISSIM K, et al. Calibrating noise to sensitivity in private data analysis [M]//Theory of cryptography. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 2006: 265 – 284. DOI: 10.1007/11681878_14

Biographies

Howard H. YANG received the B.Sc. degree in communication engineering from Harbin Institute of Technology (HIT), China, in 2012, the M.Sc. degree in electronic engineering from Hong Kong University of Science and Technology (HKUST), China, in 2013, and the Ph.D. degree in electronic engineering from Singapore University of Technology and Design (SUTD), Singapore, in 2017. His background also features appointments at the University of Texas at Austin, USA and Princeton University, USA. His research interests cover various aspects of wireless communications, networking and signal processing, currently focusing on the modeling of modern wireless networks, high dimensional statistics, graph signal processing and machine learning. He received the IEEE WCSP 10-Year Anniversary Excellent Paper Award in 2019 and the IEEE WCSP Best Paper Award in 2014.

ZHAO Zhongyuan (zyzhao@bupt.edu.cn) received the B.S. and Ph.D. degrees from Beijing University of Posts and Telecommunications (BUPT), China, in 2009 and 2014, respectively. He is currently an associate professor with BUPT. His research interests include mobile cloud and fog computing and network edge intelligence. Dr. ZHAO serves as an editor of *IEEE Communications Letters* (since 2016). He was the recipient of the Best Paper Awards at the IEEE CIT 2014 and WASA 2015. He was also the recipient of Exemplary Reviewers-2017 of *IEEE Transactions on Communications*, and Exemplary Editor Award 2017 and 2018 of *IEEE Communication Letters*.

Tony Q. S. QUEK received the B.E. and M.E. degrees in electrical and electronics engineering from Tokyo Institute of Technology, Japan. At MIT, USA, he earned the Ph.D. in electrical engineering and computer science. Currently, he is the Cheng Tsang Man Chair Professor with Singapore University of Technology and Design (SUTD). He also serves as the acting head of Information System Technology and Design (ISTD) Pillar, sector lead for SUTD AI Program, and the deputy director of SUTD-ZJU IDEA. He is currently serving as an editor for the *IEEE Transactions on Wireless Communications*, the chair of IEEE VTS Technical Committee on Deep Learning for Wireless Communications as well as an elected member of the IEEE Signal Processing Society SPCOM Technical Committee. He received the 2008 Philip Yeo Prize for Outstanding Achievement in Research, the 2012 IEEE William R. Bennett Prize, the 2016 IEEE Signal Processing Society Young Author Best Paper Award, 2017 CTTC Early Achievement Award, 2017 IEEE ComSoc AP Outstanding Paper Award, and 2016-2019 Clarivate Analytics Highly Cited Researcher. He is a Distinguished Lecturer of the IEEE Communications Society and a Fellow of IEEE.



Scheduling Policies for Federated Learning in Wireless Networks: An Overview

SHI Wenqi, SUN Yuxuan, HUANG Xiufeng, ZHOU Sheng, NIU Zhisheng

(Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

Abstract: Due to the increasing need for massive data analysis and machine learning model training at the network edge, as well as the rising concerns about data privacy, a new distributed training framework called federated learning (FL) has emerged and attracted much attention from both academia and industry. In FL, participating devices iteratively update the local models based on their own data and contribute to the global training by uploading model updates until the training converges. Therefore, the computation capabilities of mobile devices can be utilized and the data privacy can be preserved. However, deploying FL in resource-constrained wireless networks encounters several challenges, including the limited energy of mobile devices, weak onboard computing capability, and scarce wireless bandwidth. To address these challenges, recent solutions have been proposed to maximize the convergence rate or minimize the energy consumption under heterogeneous constraints. In this overview, we first introduce the backgrounds and fundamentals of FL. Then, the key challenges in deploying FL in wireless networks are discussed, and several existing solutions are reviewed. Finally, we highlight the open issues and future research directions in FL scheduling.

Keywords: federated learning; wireless network; edge computing; scheduling

DOI: 10.12142/ZTECOM.202002003

<https://kns.cnki.net/kcms/detail/34.1294.TN.20200610.1010.004.html>, published online June 10, 2020

Manuscript received: 2020-02-10

Citation (IEEE Format): W. Q. Shi, Y. X. Sun, X. F. Huang, et al., "Scheduling policies for federated learning in wireless networks: an overview," *ZTE Communications*, vol. 18, no. 2, pp. 11 – 19, Jun. 2020. doi: 10.12142/ZTECOM.202002003.

1 Introduction

With the deployment of deep learning algorithms on Internet-of-Things (IoT) devices at the network edge^[1] and the explosive growth of mobile data^[2], technologies like edge learning^[3] emerge and focus on running deep learning algorithms at the wireless access net-

work. To ensure the performance of deep learning in practical scenarios, such as auto-driving and user preference prediction, efficient training of the learning model with the data generated at the network edge is necessary. However, transmission of massive training data from edge devices to servers is challenging due to limited wireless communication resources, as well as the privacy requirement, which makes it difficult to exploit centralized training for updating the learning model. To solve this problem, federated learning (FL)^[4] is proposed, which exchanges learning models rather than raw data between edge devices and edge servers by deploying the training

This work is supported in part by the National Key R&D Program of China under Grant No. 2018YFB1800800 and the Nature Science Foundation of China under Grant Nos. 61871254, 91638204 and 61861136003.

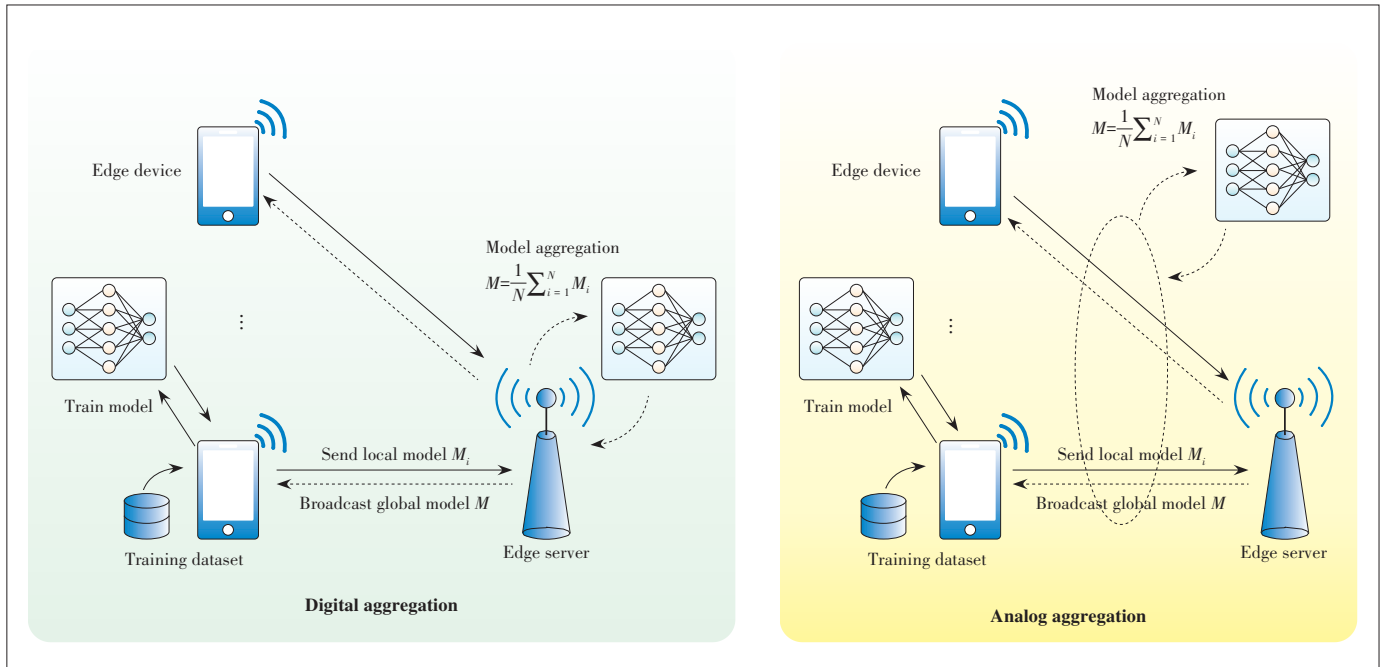
algorithms on edge devices. Since mobile devices will consume their limited computation and communication resources when participating in FL, mobile devices may not be willing to contribute. Therefore, some incentives have been introduced, such as the access to the high-quality models trained by FL, as well as some payment after participating in the FL training.

In a typical FL system, there is an edge server and several edge devices, which collaboratively train a learning model. The architecture of FL system is shown in **Fig. 1**. In each iteration (also known as communication round), the edge server aggregates the local models from edge devices in order to update the global model. Then the edge server broadcasts the newest model to edge devices for model training in the next round. After receiving the newest model, each edge device improves this model based on its own data to obtain a new local model. This process goes on until the global model converges. When aggregating the local model, each device can send the gradient of the local model back to the edge server as well as the whole local model. Compared with sending the whole model, sending gradients can reduce the information loss under the constraint of signal-to-noise ratio (SNR) and thus perform better than sending the whole model in analog aggregation (refer to Section 2.2 for details), because the norm of the gradient is smaller than the model generally. Except for analog aggregation, aggregating gradients and models are equivalent from the scheduling point of view, thus we consider that edge devices upload their updated local models rather than model gradients in the following parts of this paper unless otherwise specified. Fig. 1 shows two different model aggregation schemes, analog aggregation

and digital aggregation. In the analog aggregation scheme, edge devices send local models to the sever simultaneously and the aggregation is performed in the wireless channel according to the waveform-superposition property. In this way, the system can reduce the transmission latency since the transmission latency will not scale linearly with the number of devices. However, stringent synchronization between devices is needed during the model uploading, and the aggregation is vulnerable under the attack of third-party devices. In the digital aggregation, the model can be encoded for compression, encryption, and other purposes, which prevents the model from being aggregated in the wireless channel and is not suitable for analog aggregation. Although the digital aggregation is more convenient than the analog aggregation, long transmission latency will be introduced when the number of devices is large.

By distributing model training to the edge devices, FL mitigates the problem of privacy leaks caused by sending the raw training data from devices to the server. With the advantage of protecting data privacy, FL has been applied in some data sensitive scenarios, such as health artificial intelligence (AI)^[5]. However, some studies show that the learning model can still result in privacy leaks^[6]. To solve this problem, differential privacy-based methods^[7-8], collaborative training-based methods^[9-10] and encryption-based methods^[11-12] are proposed, which can protect the privacy of parameters of learning model.

Another advantage of FL is saving the communication cost of transmitting a large amount of training data. However, FL meets some new challenges. The training of the learning model is distributed to edge devices that may have non-indepen-



▲ Figure 1. Architecture of federated learning system.

dent and identically distributed (non-i.i.d.) training dataset^[13], which results in bad performance (such as low accuracy) of the learning model. Also, due to the different computation capabilities of devices, the FL system should consider the synchronization of the model updates from devices, and to address the straggler issues. In practical scenarios, the wireless resources of the FL system are usually limited, and thus the edge server may not be able to receive the local models from all the edge devices. To solve this problem, one direction of research is reducing the cost of transmitting the local model for every edge device, including model compression by quantization^[14] and only updating the model for the edge server when the models have significant improvement^[15]. Another research direction is the scheduling of devices, where the edge server needs to schedule a subset of edge devices to send the model update. The device scheduling can reduce the communication cost but may result in slower convergence rate of the model training. Given the constrained wireless resources, scheduling policies for FL are proposed to maximize the convergence rate^[16] of the learning model or to minimize the energy consumption^[17] of the whole system.

There are some existing surveys on FL and edge machine learning^[18–21]. In Ref. [18], the authors provide a general overview on FL and its challenges in implementation, but do not consider specific issues of deploying FL in wireless networks. The architecture of deep learning and the process of training and inference in the context of edge computing are studied in Ref. [19]. However, the authors of Ref. [19] place more emphasis on optimizing the FL algorithm itself rather than the scheduling policies for FL. The authors of Ref. [20] focus on communication-efficient FL in mobile edge computing platforms, rather than the scheduling policies that maximize the convergence rate of FL under resource constraints. In Ref. [21], the authors discuss potential FL applications in mobile edge computing, the resource allocation problems and data privacy problems in FL. Nevertheless, the authors of Ref. [21] have not provided an in-depth survey on the scheduling policies according to the model aggregation technique of FL in wireless networks, which can greatly affect the design of the scheduling policies.

In summary, none of the existing work has studied the FL in wireless networks from a scheduling perspective. Therefore, we provide a taxonomy on the aggregation methods used in FL, and discuss scheduling policies that can optimize the training performance under resource constraints for both digital-aggregation-based and analog-aggregation-based FL. The rest of this paper is organized as follows. In Section 2, we first introduce FL systems with analog-transmission-based aggregation, and then several scheduling policies designed for the analog-aggregation-based FL are discussed. The scheduling policies designed for the digital-aggregation-based FL are introduced in Section 3. Section 4 gives the conclusion of this paper and the future directions of federated learning.

2 Analog Aggregation

In a conventional wireless system, a base station needs to decode (deliver) the individual information from (to) each user. Accordingly, digital communications and orthogonal multiple access techniques have been developed and widely used. However, a key difference in the FL system is that, while aggregating the local models, the server is not interested in the individual parameters of edge devices, but their average. Note that the waveform-superposition property adds all the signals in a wireless multiple access channel, using analog transmission for global model aggregation, which is a more communication-efficient strategy^[22–27]. Edge devices synchronize with each other and transmit their local models concurrently. Then the wireless channel carries out the summation over the air, and the server receives the desired values, i.e., the average of the local models, after dividing the received signal by the number of devices involved. Analog aggregation is also called over-the-air computation and it can further support more flexible functions such as weighted summation via power allocation, so that the server can receive the weighted average of local parameters. Some recent papers are summarized in Table 1.

2.1 Device Scheduling for Analog Aggregation

A key issue of analog aggregation is how to schedule devices based on their channel states and power constraints. In the t -th round, each device n observes the channel state $h_{n,t}$, and then aligns the transmission power $p_{n,t}$, to ensure that the server can receive its desired value. The power alignment equation is given by

$$p_{n,t} = \begin{cases} \frac{a_t}{h_{n,t}}, & |h_{n,t}|^2 > h_{th} \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

▼ Table 1. Summary of recent papers on analog aggregation

Technology	Highlights	Related Works
	• Fundamental tradeoffs under Rayleigh fading channel	Ref. [23]
Power alignment	• Online energy-aware dynamic device scheduling policy	Ref. [24]
	• Device scheduling for multi-antenna analog aggregation	Ref. [25]
Sparsification and error accumulation	• Gradient sparsification and error accumulation • Device scheduling policy under average power constraint	Refs. [26–27]
Data redundancy	• Introducing data redundancy to deal with non-independent and identically distributed (non-i.i.d.) data	Ref. [24]

where a_i is a power scalar that determines the received SNR at the server side, as well as the energy consumption at the device side. Parameter h_{th} is called the power-truncation threshold, i.e., a device can be scheduled only if its current channel state is better than the threshold. Parameters a_i and h_{th} should be carefully selected in order to optimize the training performance for FL.

In Ref. [23], two fundamental tradeoffs, namely the SNR-truncation tradeoff and reliability-quantity tradeoff, are rigorously characterized. Under the assumption of Rayleigh fading, the relation between the received SNR and power-truncation threshold is studied. The SNR-truncation tradeoff is then revealed: increasing h_{th} can improve the received SNR at the server, at the cost of truncating more devices which cannot satisfy the channel quality requirement. Moreover, the received SNR is limited by the furthest device with largest path-loss. Followed by this observation, a cell-interior scheduling policy is proposed, where only the devices within a distance threshold r_{th} can be scheduled in each communication round. Parameter r_{th} balances the tradeoff between communication reliability and data quantity: larger r_{th} enables the server to schedule more devices and exploit more data for training, while it degrades the received SNR and leads to a noisier version of the average of local models. An alternating scheduling policy is proposed, where the server alternates between the cell-interior scheduling policy and all-included scheduling policy. Finally, theoretical analysis indicates that the communication latency of analog aggregation can be reduced by $O(N/\log_2 N)$ compared to its digital counterpart, where N is the number of devices.

Removing the Rayleigh fading constraint, an online energy-aware dynamic device scheduling policy is proposed in Ref. [24]. Since the explicit mapping between the loss function of the FL task and the set of devices scheduled in each round remains unknown, an alternative objective function that maximizes the average number of scheduled devices is considered. The long-term average energy constraint (which is equivalent to power constraint) of each device is transformed to a virtual energy deficit queue based on Lyapunov optimization. In each communication round, each device acquires the current channel state $h_{n,t}$ and decides whether to update its local model individually by considering the value of the virtual queue and the required energy consumption. The proposed device scheduling policy works in an online fashion, without requiring any information of the channel states in the future. It also works well if the channel states are non-i.i.d across time.

A multi-antenna analog aggregation FL system is considered in Ref. [25], where the number of scheduled devices is maximized under the mean-square-error (MSE) constraint. Satisfying the MSE requirement can limit the transmission error, and thus it guarantees the accuracy of the aggregated learning model parameters. In order to improve the efficiency of the device scheduling policy, a sparse and low-rank approach is in-

troduced.

2.2 Sparsification and Error Accumulation

The neural networks to be trained for FL tasks usually have huge dimensions, with thousands to millions of parameters. However, the wireless bandwidth is in general limited, and thus the communication latency scales up with the dimension of local models. To further reduce the communication cost for model aggregation, gradient sparsification techniques are introduced in Refs. [26] and [27]. Note that transmitting local gradients rather than local models can improve the power efficiency of analog aggregation, because all the power is used to transmit the information unknown to the server. Therefore, all the devices update their gradients rather than the up-to-date models.

To reduce the dimension of local gradients, a random linear projection is first employed, inspired by compressive sensing. In particular, each local model is multiplied by a random matrix, where each entry follows Gaussian distribution. The random matrix is shared by the devices and the server. Then each device only retains k entries with largest absolute values, which can be regarded as the most important parameters of the gradients, while setting all the other gradients to zero. Here, k is a design parameter which balances the tradeoff between communication reliability and distortion: with smaller k , each entry can be transmitted in a higher power, so that the SNR at the server is higher. However, more information of the local gradient is lost due to the sparsification, degrading the accuracy of the neural network as well as the convergence rate of training.

Instead of discarding all the lost information due to sparsification, a more efficient way is to do error accumulation at the device side. In particular, in each round, the device calculates the differences between the sparse gradients and the original gradients, and adds these differences to the gradients obtained in the next round before employing sparsification. In this way, the error due to sparsification is accumulated by workers, and the training accuracy can be improved according to the experimental results.

Device scheduling policies are also designed for analog aggregation with gradient sparsification and error accumulation. In Ref. [26], additive white Gaussian noise (AWGN) channel is considered, and both equal and unequal power allocation policies are designed. The unequal policy puts more power to the initial rounds, motivated by the fact that the variance of the gradients diminishes across time. Ref. [27] further considers Rayleigh fading channels. Extensive experiments show that compared to the digital aggregation, analog aggregation can improve the convergence rate of training, particularly at low bandwidth and stringent power regimes.

2.3 Non-IID Training Data

The non-i.i.d. data, i.e., the different distributions of data

samples at devices, is also a major bottleneck for FL. It is shown in Ref. [28] that high non-i.i.d. data reduces the accuracy of the neural network by 11% under the Modified National Institute of Standards and Technology (MNIST) dataset, and by over 50% under CIFAR-10 dataset. The non-i.i.d. level of data refers to the difference of local data distribution and global data distribution, which can be characterized by the earth mover's distance, a measurement of the distance between two distributions. To reduce the non-i.i.d. level and thus improve the training accuracy, the server collects some sharable data samples from the devices and disseminates the data to the whole FL system.

Non-i.i.d. data is still a key issue in analog aggregation FL systems. In Ref. [24], data redundancy is introduced to reduce the non-i.i.d. level of data samples, which can be obtained by exchanging data between a group of devices or collecting data with overlapped coverage in IoT networks. **Fig. 2** illustrates the analog aggregation for FL systems with data redundancy. Workers 1 and 2, 3 and 4 exchange their local datasets with each other, and the redundancy level of the system, i.e., how many devices store each data sample, is two. The experiment results with non-i.i.d. data using MNIST dataset are shown in **Fig. 3**, where \bar{E} is the average energy constraint (in J), "dyn" is the proposed online energy-aware dynamic device scheduling policy, and "myopic" is a benchmark policy where devices can use as much energy as \bar{E} in each round. Parameter r denotes the redundancy level, and $\bar{E} = \infty$ refers to the case where devices have infinite energy, so that all of them can be scheduled in each round. We can see that the proposed dy-

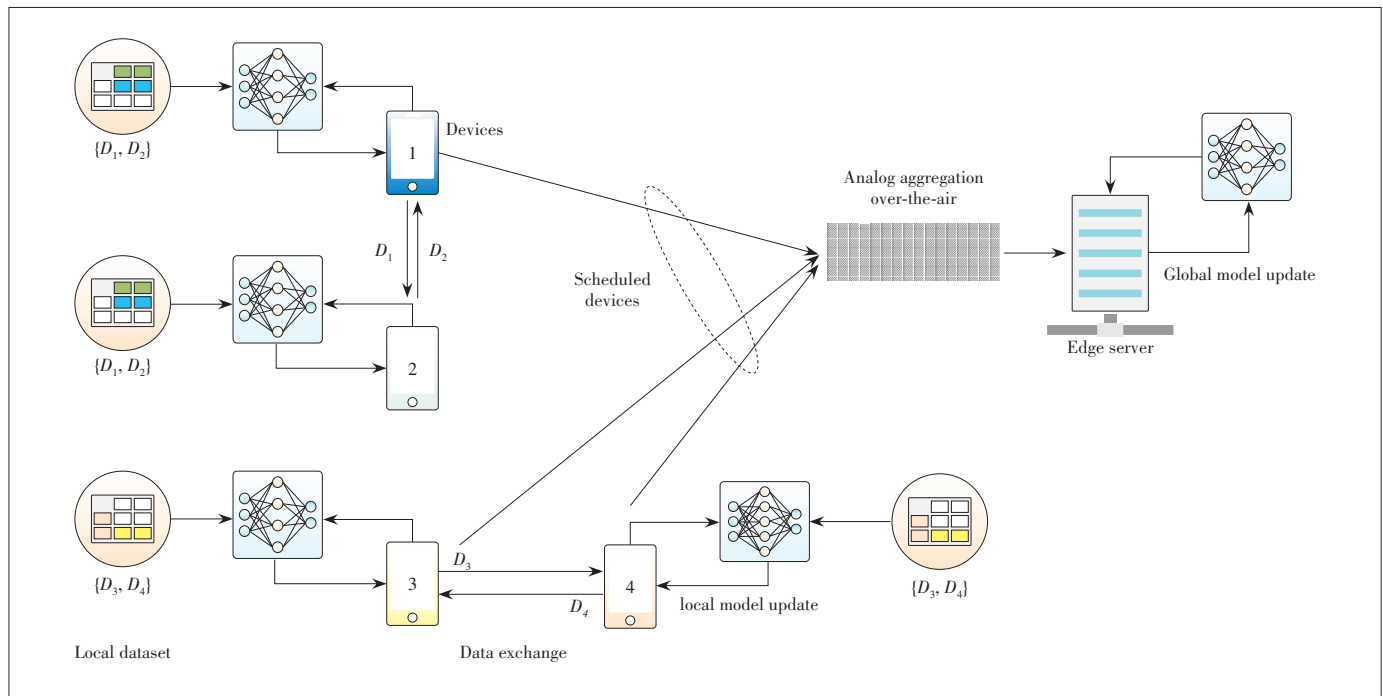
namic device scheduling policy outperforms the myopic benchmark, and data redundancy can improve the training accuracy significantly. In particular, when $\bar{E} = 5$, increasing redundancy from $r = 1$ to $r = 2$ can achieve an improvement of 10% in training accuracy.

3 Digital Aggregation

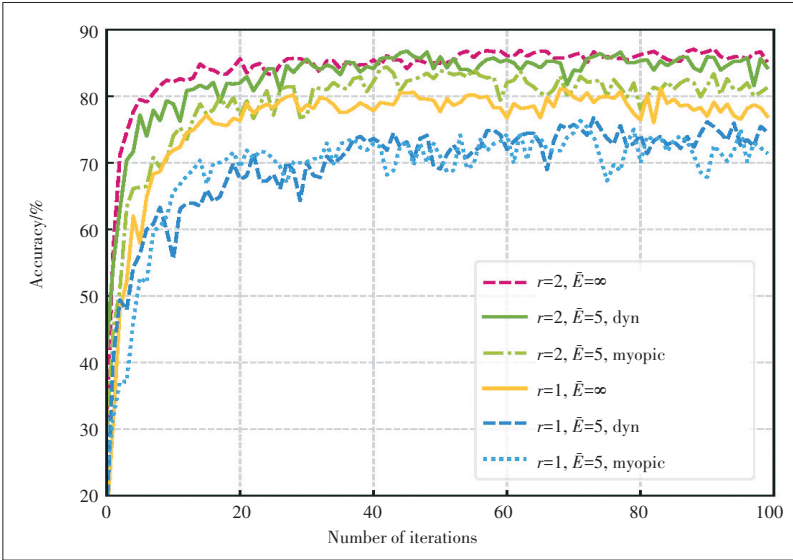
In many other studies, the FL systems are deployed in existing wireless networks (e.g., cellular network or Wi-Fi network), where orthogonal-access schemes such as orthogonal frequency division multiple access (OFDMA) are used for model aggregation. To distinguish them from analog aggregation approaches, we categorize these approaches into digital aggregation. In digital aggregation, the participating devices need to share the scarce wireless bandwidth to upload the updated local models, making the global aggregation very time-consuming. Further, the limited energy and computing resources of participating devices make it more challenging to deploy FL in real wireless networks. Therefore, various scheduling policies have been proposed to address these challenges. These scheduling policies can be divided into the following three categories: aggregation frequency adaptation, local accuracy tuning, and device scheduling. **Table 2** summarizes the highlights of recent papers on digital aggregation.

3.1 Aggregation Frequency Adaptation

In FL, the local update consumes computing resources of devices and the global aggregation consumes the bandwidth resources. Since FL iterates between local updates and global



▲ Figure 2. Analog aggregation for federated learning with data redundancy.



▲ Figure 3. Training accuracy of dynamic device scheduling policy in Ref. [24] under independent and identically distributed (i.i.d.) and non-i.i.d. data.

▼ Table 2. Summary of recent papers on digital aggregation

Technology	Highlights	Related Works
Aggregation frequency adaption	• Global aggregation frequency adaption under given resource constraints.	Ref. [29]
	• Extending Ref. [29] into a client-edge-cloud hierarchical FL system	Ref. [30]
Local accuracy tuning	• Tuning local model accuracy to balance the tradeoff between local update and global aggregation	Refs. [31 – 32]
	• Energy- and convergence-aware resource allocation	
Device scheduling	• Energy- and convergence-aware joint scheduling and resource allocation	Ref. [17]
	• Consider unreliable wireless transmissions	Refs. [35 – 36]
	• Maximize the convergence rate with respect to time	Refs. [16] and [37]

aggregations, the frequency of global aggregations (i.e., the reciprocal of the number of local updates between two adjacent global aggregations) should be carefully tuned to balance the consumption of computing and bandwidth resources. In Ref. [29], the authors first analyze the convergence bound of FL with respect to (w.r.t) the number of local updates between two adjacent global aggregations. The bound shows that a higher global aggregation frequency can speed up the FL convergence, while the drawback is consuming more wireless resources for global aggregation. Then a scheduling policy that adapts the frequency of global aggregations in real time to maximize the convergence rate of FL is derived based on the derived convergence bound. The proposed scheduling policy is applicable to non-i.i.d. data distributions and heterogeneous resource constraints of participating devices. Their simulation

results show that adaptively adjusting the global aggregation frequency can greatly improve the convergence rate of FL, compared with fixed global aggregation frequency counterparts. Further, the authors of Ref. [30] extend the scheduling policy proposed by Ref. [29] into a client-edge-cloud hierarchical system. In the client-edge-cloud hierarchical FL system, each edge server is allowed to perform partial aggregation that aggregates the updated local models of the edge devices within its communication range. While for the cloud-based global aggregation, the partially aggregated models at edge servers are aggregated through the backbone network by the centralized cloud server. The aggregation frequencies of two levels of model aggregation (i.e., edge-based partial aggregation and cloud-based global aggregation) are optimized to minimize the global loss function value under a constrained number of total local updates.

3.2 Local Accuracy Tuning

The tradeoff between computation and communication is balanced through optimizing the aggregation frequency in aggregation frequency adaptation. Alternatively, some researchers balance this tradeoff via tuning the accuracy level of the local models. In general, increasing local model accuracy requires more computation, while fewer communication rounds are needed for more accurate local models to achieve a fixed global accuracy.

In Ref. [31], the authors refer to an upper bound for the number of communication rounds w.r.t. global accuracy and local accuracy, which is applied to strong convex loss functions for designing the scheduling policy. They adopt time division multiple access (TDMA) for media access control (MAC) layer and dynamic voltage and frequency scaling (DVFS) for devices' CPUs. Thus the frequencies of devices' CPUs, the communication latency of local model uploading and the local accuracy are jointly optimized to minimize the weighted sum of training latency and device energy consumption. As a result, both the computation-communication tradeoff and the device energy consumption-FL training latency tradeoff can be characterized. The overall non-convex optimization problem is decoupled into convex sub-problems, and the closed-form optimal solutions to the sub-problems are illustrated by extensive numerical results. While in Ref. [32], the authors consider a similar FL system but with frequency division multiple access (FDMA). Therefore, the bandwidth allocated to each devices should be jointly optimized with the communication latency, the CPU frequency and the local accuracy. Due to the complicated nature of the problem, the authors of Ref. [32] proposed an iterative algorithm. Their simulation results show that up to 25.6% latency reduction and 37.6% energy reduction can be achieved compared to conventional FL.

3.3 Device Scheduling

Due to the limited wireless resources and stringent training delay budget, only a portion of devices are allowed to upload local models in each round in real FL systems^[33]. Thus the device scheduling policy is critical to FL and will affect the convergence performance in the following two aspects. On one hand, the server needs to wait until all scheduled devices have finished updating and uploading their local models in each round. Therefore, scheduling more devices per round can significantly slow down the model aggregation, because of the reduced bandwidth allocated to each device and the higher probability of having a straggler device (i.e., the device with limited computation capabilities or bad wireless channel). On the other hand, scheduling more devices per round increases the convergence rate (w.r.t. the number of rounds)^[34] and can potentially reduce the number of rounds required to attain the same accuracy. To this end, the scheduling policy should carefully balance the latency and learning efficiency per round.

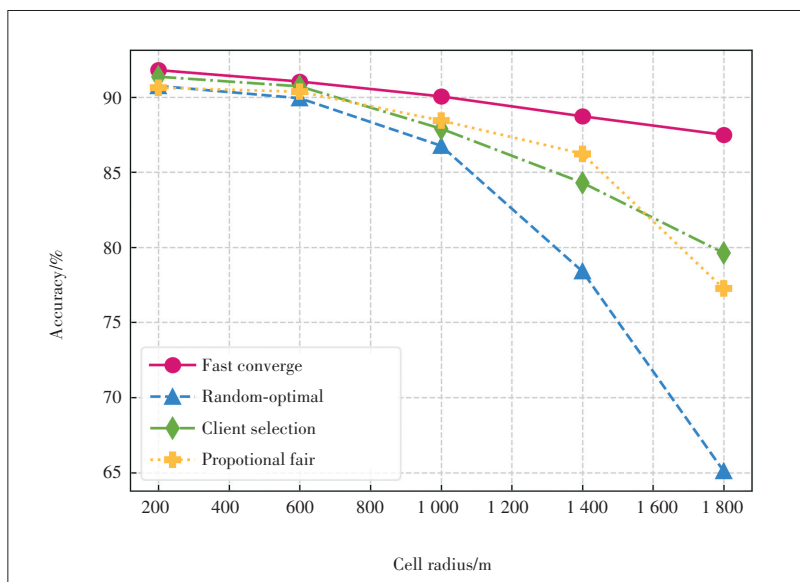
Recently, device scheduling problems in FL have received many research efforts. The authors of Ref. [17] consider a joint scheduling and radio resource allocation problem for FL. In Ref. [17], OFDMA is used for model uploading, where bandwidth allocation can be optimized to reduce the energy consumption. To further characterize the convergence performance, they assume that the convergence rate linearly increases with the number of scheduled devices. Therefore, the optimization objective is set to be the weighted sum of the energy consumption and the number of scheduled devices with a predetermined tradeoff factor, so as to balance the energy consumption and convergence rate. After relaxing the integer constraint for the device scheduling as the real-value constraint, the optimization problem is solved by iteratively solving the bandwidth allocation and scheduling sub-problems.

Furthermore, some recent studies consider the unreliable wireless transmissions. In Ref. [35], the authors propose to deploy FL in cellular networks where inter-cell interference can affect the transmissions of model aggregation. For the transmission quality, only if the received signal-to-interference-plus-noise ratio (SINR) exceeds a threshold, the received local models can be successfully decoded. The convergence rate of FL under such settings, accounting for effects from both scheduling and interference, is then derived. Furthermore, three basic scheduling policies, namely the random scheduling, round-robin and proportional fair, are compared in terms of FL convergence rate. Their results show that the proportional fair policy performs better under a high SINR threshold, while round-robin is suitable for a low SINR threshold. However, the authors of Ref. [36] consider OFDMA for model aggregation and use the packet error rate to capture the unreliability of the wireless transmission. In Ref. [36], a convergence rate bound w.r.t. packet errors is first derived, given the transmitting power of devices, OFDMA resource block allocation and device

scheduling policy. Then, the authors formulate an optimization problem to maximize the convergence rate by jointly optimizing the transmitting power allocation, resource block allocation and scheduling policy. The optimization problem is solved in a two-step manner: first obtaining the optimal transmitting power of each device given the device scheduling and resource block allocation; then using the Hungarian algorithm to find the optimal device scheduling and resource block allocation. As shown by simulations, the proposed method can reduce up to 10% and 16% loss function value, compared to: 1) optimal device scheduling with random resource allocation; 2) random device scheduling and random resource allocation, respectively.

However, the convergence rate w.r.t. time, which is critical for real-world FL applications, has not been addressed by aforementioned works. To accelerate the FL training, the authors of Ref. [37] propose to maximize the number of scheduled devices in a given time budget for each round, while the stragglers are discarded to avoid slowing down the model aggregation. The proposed greedy scheduling policy iteratively schedules the device that consumes the least time in model updating and uploading, until reaching the time budget. Although the proposed scheduling policy is simple, their experiments show that it is efficient and applicable to both non-i.i.d. data distributions and heterogeneous devices.

Nevertheless, the time budget is chosen through experiments and can hardly be adjusted under highly-dynamic FL systems. To overcome this drawback, Ref. [16] proposes a joint scheduling and resource allocation policy with fast convergence for FL. Specifically, a latency-optimal bandwidth allocation policy for local model updating and uploading is first derived. Then given the set of scheduled devices and the latency-optimal bandwidth allocation, based on a known upper bound of the number of required rounds to attain a fixed global accuracy, an upper bound of the time required to attain a fixed global accuracy is derived. Finally, an iterative scheduling policy is proposed that iteratively schedules the device that minimizes the approximate time upper bound until the approximate upper bound begins to increase (i.e., scheduling more devices makes the convergence time longer). **Fig. 4** shows the highest achievable accuracy within a total training time budget that equals to 300 seconds under different scheduling policies, including fast converge scheduling policy^[16], random scheduling policy with empirically optimal number of scheduled devices (random-opt), client selection policy^[37], and proportional fair policy^[35]. The experiments are conducted using non-i.i.d. distributed MNIST dataset, and it is assumed that all devices are randomly located in a cell. With different cell radius, the simulation results show that the fast converge scheduling policy always outperforms other scheduling policies in terms of the convergence rate w.r.t. time, and is applicable to non-i.i.d. data.



▲ Figure 4. Highest achievable accuracy under different scheduling policies v.s. the radius of device distributed area.

4 Conclusions and Future Directions

This paper presents a brief introduction of FL in wireless networks and in particular an overview on the scheduling policies for wireless FL. Firstly, the motivation of deploying FL in wireless networks and the fundamentals of FL systems are introduced. Then, a series of works in the FL systems with analog aggregation are discussed, including device scheduling, model sparsification and data redundancy. Afterwards, we provide an overview on another series of works in FL systems with digital aggregation, including aggregation frequency adaptation, local accuracy tuning and device scheduling. However, apart from the aforementioned works, there are still some challenges and future research directions in deploying FL in wireless networks:

1) Delayed CSI: In the existing works on analog aggregation, power alignment is based on perfect CSIs of devices. While in practice, the server only has delayed CSIs of devices, and how to align the transmission power of devices to minimize the distortion of the aggregated model under delayed CSI remains an open problem. To address this challenge, using the recurrent neural network to predict instantaneous CSI according to the historical CSI estimations may be a future direction.

2) Non-i.i.d. data distribution: Since the data distributions of different devices are usually non-i.i.d. in practical wireless FL applications, it is crucial to design non-i.i.d. data distribution-aware scheduling policies. Although the non-i.i.d. issue in FL systems with digital aggregation has been considered in Refs. [16], [29 – 30] and [37], none of them has proposed any method to alleviate the accuracy degradation caused by non-i.i.d. data. In the future, the data redundancy introduced in Ref. [24] and the communication-efficient data exchange technologies between different devices can be con-

sidered in FL systems with digital aggregation to address the non-i.i.d. issue.

3) Convergence guarantee: FL is actually a distributed optimization algorithm that cannot always guarantee to converge. Although most FL algorithms empirically converge and several existing works have provided convergence analysis for FL with convex or strongly convex loss functions. Theoretical analysis and evaluations on the convergence of FL with generally non-convex loss functions are still open problems.

References

- [1] CHIANG M, ZHANG T. Fog and IoT: an overview of research opportunities [J]. IEEE internet of things journal, 2016, 3(6): 854 – 864. DOI: 10.1109/JIOT.2016.2584538
- [2] ZHU G, LIU D, DU Y, et al. Towards an intelligent edge: wireless communication meets machine learning [EB/OL]. (2018-09-02)[2020-01-31]. <https://arxiv.org/abs/1809.00343>
- [3] PARK J, SAMARAKOON S, BENNIS M, et al. Wireless network intelligence at the edge [J]. Proceedings of the IEEE, 2019, 107(11): 2204 – 2239. DOI: 10.1109/JPROC.2019.2941458
- [4] LIM W Y, LUONG N C, HOANG D T, et al. Federated learning in mobile edge networks: a comprehensive survey [EB/OL]. (2019-09-26)[2020-01-31]. <https://arxiv.org/abs/1909.11875>
- [5] BRISIMI T S, CHEN R, MELA T, et al. Federated learning of predictive models from federated electronic health records [J]. International journal of medical informatics, 2018, 112: 59 – 67. DOI: 10.1016/j.ijmedinf.2018.01.007
- [6] MELIS L, SONG C, CRISTOFARO E DE, et al. Exploiting unintended feature leakage in collaborative learning [EB/OL]. (2018-05-10)[2019-11-01]. <https://arxiv.org/abs/1805.04049>
- [7] ABADI M, CHU A, GOODFELLOW I, et al. Deep learning with differential privacy [C]//Proceedings of 2016 ACM SIGSAC Conference on Computer and Communications Security. New York, USA: ACM, 2016: 308 – 318. DOI: 10.1145/2976749.2978318
- [8] GEYER R C, KLEIN T, NABI M. Differentially private federated learning: a client level perspective [EB/OL]. (2018-03-01)[2020-01-31]. <https://arxiv.org/abs/1712.07557>
- [9] SHOKRI R, SHMATIKOV V. Privacy-Preserving Deep Learning [C]//Proceedings of 22nd ACM SIGSAC Conference on Computer and Communications Security. New York, USA: ACM, 2015: 1310 – 1321. DOI: 10.1145/2810103.2813687
- [10] LIU Y, MA Z, MA S, et al. Boosting privately: privacy-preserving federated extreme boosting for mobile crowdsensing [EB/OL]. (2019-07-24) [2020-01-31]. <https://arxiv.org/abs/1907.10218>
- [11] AONO Y, HAYASHI T, WANG L, et al. Privacy-preserving deep learning via additively homomorphic encryption [J]. IEEE transactions on information forensics and security, 2017, 13(5): 1333 – 1345. DOI: 10.1109/TIFS.2017.2787987
- [12] HAO M, LI H W, XU G W, et al. Towards efficient and privacy-preserving federated deep learning [C]//IEEE International Conference on Communications (ICC). Shanghai, China, 2019: 1 – 6. DOI: 10.1109/icc.2019.8761267
- [13] KONEČNÝ J, MCMAHAN B, RAMAGE D. Federated optimization: distributed optimization beyond the datacenter [EB/OL]. (2015-11-11)[2020-01-31]. <https://arxiv.org/abs/1511.03575>
- [14] AJI A F, HEAFIELD K. Sparse communication for distributed gradient descent [C]//Proceedings of 2017 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, USA: Association for Computational Linguistics, 2017. DOI: 10.18653/v1/d17-1045

- [15] CHEN T Y, GIANNAKIS G B, SUN T, et al. LAG: Lazily aggregated gradient for communication-efficient distributed learning [C]//Advances in Neural Information Processing Systems 31 (NeurIPS 2018). Montreal, Canada, 2018: 5055 – 5065
- [16] SHI W, ZHOU S, NIU Z. Device scheduling with fast convergence for wireless federated learning [EB/OL]. (2019-11-03)[2020-01-31]. <https://arxiv.org/abs/1911.00856>
- [17] ZENG Q, DU Y, LEUNG K K, et al. Energy-efficient radio resource allocation for federated edge learning [EB/OL]. (2019-07-13)[2020-01-31]. <https://arxiv.org/abs/1907.06040>
- [18] LI T, SAHU A K, TALWALKAR A, et al. Federated learning: challenges, methods, and future directions [EB/OL]. (2019-8-21)[2020-01-31]. <https://arxiv.org/abs/1908.07873>
- [19] WANG X, HAN Y, LEUNG V C M, et al. Convergence of edge computing and deep learning: a comprehensive survey [J]. IEEE communications surveys & tutorials, 2020. DOI: 10.1109/COMST.2020.2970550
- [20] SHI Y, YANG K, JIANG T, et al. Communication-efficient edge AI: algorithms and systems [EB/OL]. (2020-02-22). <https://arxiv.org/abs/2002.09668>
- [21] LIM W Y B, LUONG N C, HOANG D T, et al. Federated learning in mobile edge networks: a comprehensive survey [EB/OL]. (2020-02-28). <https://arxiv.org/abs/1909.11875>
- [22] GUNDUZ D, DE KERRET P, SIDIROPOULOS N D, et al. Machine learning in the air [J]. IEEE journal on selected areas in communications, 2019, 37(10): 2184 – 2199. DOI: 10.1109/JSAC.2019.2933969
- [23] ZHU G X, WANG Y, HUANG K B. Broadband analog aggregation for low-latency federated edge learning [J]. IEEE transactions on wireless communications, 2020, 19(1): 491 – 506. DOI: 10.1109/TWC.2019.2946245
- [24] SUN Y, ZHOU S, GUNDUZ D. Energy-aware analog aggregation for federated learning with redundant data [EB/OL]. (2019-11-01)[2020-01-31]. <https://arxiv.org/abs/1911.00188>
- [25] YANG K, JIANG T, SHI Y, et al. Federated learning via over-the-air computation [EB/OL]. (2019-02-17)[2020-01-31]. <https://arxiv.org/abs/1812.11750>
- [26] AMIRI M M, GUNDUZ D. Machine learning at the wireless edge: distributed stochastic gradient descent over-the-air [C]//IEEE International Symposium on Information Theory (ISIT). Paris, France, 2019: 1432 – 1436. DOI: 10.1109/isit.2019.8849334
- [27] AMIRI M M, GUNDUZ D. Federated learning over wireless fading channels [EB/OL]. (2019-07-23)[2020-01-31]. <https://arxiv.org/abs/1907.09769>
- [28] ZHAO Y, LI M, LAI L, et al. Federated learning with non-iid data [EB/OL]. (2018-06-02)[2020-01-31]. <https://arxiv.org/abs/1806.00582>
- [29] WANG S Q, TUOR T, SALONIDIS T, et al. Adaptive federated learning in resource constrained edge computing systems [J]. IEEE journal on selected areas in communications, 2019, 37(6): 1205 – 1221. DOI: 10.1109/jsac.2019.2904348
- [30] LIU L, ZHANG J, SONG S H, et al. Edge-assisted hierarchical federated learning with non-iid data [EB/OL]. (2019-10-31)[2020-01-31]. <https://arxiv.org/abs/1905.06641>
- [31] TRAN N H, BAO W, ZOMAYA A, et al. Federated learning over wireless networks: optimization model design and analysis [C]//IEEE Conference on Computer Communications. Paris, France, 2019: 1387 – 1395. DOI: 10.1109/IN-FOCOM.2019.8737464
- [32] YANG Z, CHEN M, SAAD W, et al. Energy efficient federated learning over wireless communication networks [EB/OL]. (2019-11-6)[2020-01-31]. <https://arxiv.org/abs/1911.02417>
- [33] BONAWITZ K, EICHNER H, GRIESKAMP W, et al. Towards federated learning at scale: system design [EB/OL]. (2019-3-22)[2020-01-31]. <https://arxiv.org/abs/>
- [34] STICH S U. Local SGD converges fast and communicates little [EB/OL]. (2019-05-03)[2020-01-31]. <https://arxiv.org/abs/>
- [35] YANG H H, LIU Z Z, QUEK T Q S, et al. Scheduling policies for federated learning in wireless networks [J]. IEEE transactions on communications, 2020, 68(1): 317 – 333. DOI: 10.1109/tcomm.2019.2944169
- [36] CHEN M, YANG Z, SAAD W, et al. A joint learning and communications framework for federated learning over wireless networks [EB/OL]. (2019-9-17)[2020-01-31]. <https://arxiv.org/abs/1909.07972>
- [37] NISHIO T, YONETANI R. Client selection for federated learning with heterogeneous resources in mobile edge [C]//ICC 2019-2019 IEEE International Conference On Communications (ICC). Shanghai, China, 2019: 1-7. DOI: 10.1109/ICC.2019.8761315
- [38] NISHIO T, YONETANI R. Client selection for federated learning with heterogeneous resources in mobile edge [C]//IEEE International Conference on Communications (ICC). Shanghai, China, 2019: 1 – 7. DOI: 10.1109/icc.2019.8761315

Biographies

SHI Wenqi received his B.S. degree in electronic engineering from Tsinghua University, China in 2017. He is pursuing his Ph.D. degree in electronic engineering with Tsinghua University. His research interests include edge computing, machine learning and machine learning applications in wireless communications.

SUN Yuxuan received her B.S. degree in telecommunications engineering from Tianjin University, China, in 2015. She is currently working toward the Ph.D. degree in electronic engineering with Tsinghua University. Her research interests include mobile edge computing, vehicular cloud computing and distributed machine learning.

HUANG Xiufeng received his B.S. degree in electronic engineering from Tsinghua University, China, in 2018. He is currently a Ph.D. student in electronic engineering with Tsinghua University. His research interests include machine learning, edge computing and performance optimization for machine learning applications in wireless networks.

ZHOU Sheng (sheng.zhou@tsinghua.edu.cn) received his B.S. and Ph.D. degrees in electronic engineering from Tsinghua University, China, in 2005 and 2011, respectively. He is currently an associate professor of Electronic Engineering Department, Tsinghua University. His research interests include cross-layer design for multiple antenna systems, vehicular networks, mobile edge computing and green wireless communications.

NIU Zhisheng graduated from Beijing Jiaotong University, China, in 1985, and got his M.E. and D.E. degrees from Toyohashi University of Technology, Japan, in 1989 and 1992, respectively. During 1992 – 1994, he worked for Fujitsu Laboratories Ltd., Japan, and in 1994, he joined in Tsinghua University, China, where he is now a professor at the Department of Electronic Engineering. His major research interests include queueing theory, traffic engineering, mobile Internet, radio resource management of wireless networks, and green communication and networks.

Joint User Selection and Resource Allocation for Fast Federated Edge Learning



JIANG Zhihui, HE Yinghui, YU Guanding

(College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, China)

Abstract: By periodically aggregating local learning updates from edge users, federated edge learning (FEEL) is envisioned as a promising means to reap the benefit of local rich data and protect users' privacy. However, the scarce wireless communication resource greatly limits the number of participated users and is regarded as the main bottleneck which hinders the development of FEEL. To tackle this issue, we propose a user selection policy based on data importance for FEEL system. In order to quantify the data importance of each user, we first analyze the relationship between the loss decay and the squared norm of gradient. Then, we formulate a combinatorial optimization problem to maximize the learning efficiency by jointly considering user selection and communication resource allocation. By problem transformation and relaxation, the optimal user selection policy and resource allocation are derived, and a polynomial-time optimal algorithm is developed. Finally, we deploy two commonly used deep neural network (DNN) models for simulation. The results validate that our proposed algorithm has strong generalization ability and can attain higher learning efficiency compared with other traditional algorithms.

Keywords: data importance; federated edge learning; learning accuracy; learning efficiency; resource allocation; user selection

DOI: 10.12142/ZTECOM.202002004

<https://kns.cnki.net/kcms/detail/34.1294.TN.20200612.1311.003.html>, published online June 12, 2020

Manuscript received: 2020-01-31

Citation (IEEE Format): Z. H. Jiang, Y. H. He, and G. D. Yu, "Joint user selection and resource allocation for fast federated edge learning," *ZTE Communications*, vol. 18, no. 2, pp. 20 - 30, Jun. 2020. doi: 10.12142/ZTECOM.202002004.

1 Introduction

With the explosive growth of data generated by mobile devices and the remarkable breakthroughs made in artificial intelligence (AI) in recent years, the combination of AI and wireless networks is attracting more and more interests^[1]. To leverage the abundant data, which are unevenly distributed over a large number of

edge devices, and to train a high quality prediction model, the traditional scheme is to do centralized learning by transmitting the raw data to the data center. However, this scheme has two drawbacks. On the one hand, the privacy of users may be divulged when the data center suffers from malicious attacks. On the other hand, the communication latency is long since the volume of data is large and the communication resource is limited. To overcome these two issues, a new framework, namely federated edge learning (FEEL), has been recently proposed in Ref. [2]. This framework makes a collaboration of the distributed learning framework, named federated learning (FL)

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61671407.

[3] and mobile edge computing (MEC)^[4], which not only ensures users' privacy but also exploits the computing resource of both edge devices and edge servers.

In the FEEL system, edge devices need to interact with the edge server constantly to train a global model. Thus, communication cost is one of the major constraints of model training since the wireless communication resource is limited. Recently, several works have investigated accelerating the training task by reducing the communication overhead^[5-6]. To achieve a low-latency FEEL system, the authors in Ref. [5] propose a broadband analog aggregation scheme by exploiting over-the-air computation and derive two communication-and-learning tradeoffs. In Ref. [6], the authors propose a new protocol to reduce the communication overhead and improve the training speed by selecting devices as many as possible based on their channel state information (CSI). Besides, energy-efficient FL over wireless networks has been investigated in Refs. [7] and [8]. In Ref. [7], energy-efficient strategies are proposed for joint bandwidth allocation and energy-and-learning aware scheduling with less energy consumption. The authors in Ref. [8] propose an iterative algorithm to achieve the tradeoff between latency and energy consumption for FL. Moreover, several recent works focus on the problem of user selection for FL over wireless networks^[9-12]. In Ref. [9], the authors derive a tradeoff between the number of scheduled users and subchannel bandwidth under fixed amount of available spectrum. To improve the running efficiency of FL, the authors in Ref. [10] propose a scheduling policy by exploiting the CSI, i.e., the instantaneous channel qualities. In Ref. [11], the authors consider a user selection problem based on packet errors and the availability of wireless resources, and a probabilistic user selection scheme is proposed to reduce the convergence time of FL in Ref. [12].

However, the aforementioned works ignore the fact that the process of model training is time-consuming as well. According to Ref. [13], different training samples are not equally important in a training task. Therefore, faced with the massive data, the topic of selecting important data to further accelerate the training task deserves to be studied. Several recent works have studied on this topic. In Refs. [14] and [15], data importance is quantified by the signal-to-noise ratio (SNR) and data uncertainty measured by the distance to the decision boundary. Based on this, the authors propose a data importance aware retransmission protocol and a user scheduling algorithm, respectively.

As we have mentioned before, some works have already investigated the acceleration of the training task based on data importance. However, this topic has not been investigated in the FEEL system yet, which is a distributed edge learning system. Inspired by this, we consider an FEEL system, where the learning efficiency of the system is improved by user selection based on data importance. First, we analyze the relation between the loss decay and the learning update information

(LUI), i.e., the squared norm of the gradient, and derive an indicator to quantify the data importance. Then, an optimization problem to maximize the learning efficiency of the FEEL system is formulated by joint user selection and communication resource allocation. The closed-form solution for optimal user selection policy and communication resource allocation is derived by problem transformation and relaxation. Based on this, we develop a polynomial-time algorithm to solve this mixed-integer programming problem. Finally, we verify the generalization ability and the performance improvement of our proposed algorithm by extensive simulation.

The rest of this paper is organized as follows. In Section 2, we introduce the FEEL system and establish the deep neural network (DNN) model and communication model. In Section 3, we propose an indicator to quantify the data importance, analyze the end-to-end latency in each communication round, and formulate the optimization problem to maximize the learning efficiency. The optimal solution and the optimal algorithm are developed in Section 4. Simulation results are presented in Section 5 and the whole paper is concluded in Section 6.

2 System Model

In this section, we will first introduce the FEEL system model. Then, both the DNN model and communication model are introduced.

2.1 Federated Edge Learning System

We consider an FEEL system as shown in **Fig. 1**, which comprises an edge server and K distributed users, denoted by $\mathcal{K} = \{1, 2, \dots, K\}$. Each user utilizes its local dataset to train the local DNN model. Let $\mathcal{D}_k = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_{N_k}, y_{N_k})\}$ denote the local dataset of user k , where \mathbf{x}_i is the training sample, y_i is the size of the corresponding ground-true label, and N_k is the size of dataset. During each communication round, users first upload their gradients to the edge server. Then, the edge server collects the local gradients from users and aggregates them as the global gradient. Users update their local models by the global gradient broadcast by the edge server. Ultimately, users are supposed to collaborate with each other in training a shared global model. Therefore, users' privacy is protected since the raw data are not transmitted to the edge server. However, due to the limited wireless communication resource, the number of users participated in the training task is restricted. To tackle this issue, we intend to propose a user selection policy by jointly considering the LUI and CSI of each user. During each communication round, users' data are not of equal importance. So we only select part of users to upload their local gradients based on data importance and channel data rate. The following seven steps are defined as a communication round.

1) Calculate local gradient. In the n -th communication round, each user utilizes its local dataset to train its local mod-

5) **Aggregate global gradient.** The edge server receives the local gradients of all selected users and then aggregates them as the global gradient, which can be expressed as

where $a_k \in \{0, 1\}$ indicates whether user k is selected, i.e.,

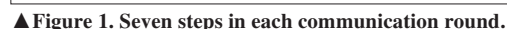
7) Update local model. After the global gradient is received, each user updates its local model, as

where $\xi[n]$ is the learning rate of the n -th communication round.

The above seven steps are periodically performed until the global model converges. During the training process, the local gradient and the CSI of users are different in each communication round. Therefore, the edge server should run the optimal algorithm to select users in each communication round.

In this work, all users adopt the same DNN model for training. To evaluate the error between the learning output and the ground-true label y_i , we define the loss function of training samples as $l(\mathbf{0}, \mathbf{x}_i, y_i)$. Thus, the local loss function of user k and the global loss function can be represented as

$$L_k(\boldsymbol{\theta}, \mathcal{D}_k) = \frac{1}{|\mathcal{D}_k|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_k} l(\boldsymbol{\theta}, \mathbf{x}_i, \mathbf{y}_i), \quad (3)$$



$$L(\boldsymbol{\theta}) = \frac{1}{\left| \bigcup_k a_k \mathcal{D}_k \right|} \sum_{k=1}^K a_k \left| \mathcal{D}_k \right| L_k(\boldsymbol{\theta}, \mathcal{D}_k), \quad (4)$$

respectively. In the course of training, the global loss function $L(\boldsymbol{\theta})$ is the objective function to be minimized. In our scheme, we aim to accelerate the training task and train a high quality global model. Without loss of generality, we utilize stochastic gradient descent (SGD) as the optimal algorithm. Then, the local gradient vector of user k is given by

$$\mathbf{G}_k^0 = \nabla L_k(\boldsymbol{\theta}, \mathcal{D}_k), \quad (5)$$

where ∇ implies the gradient operator.

2.3 Communication Model

As described above, distributed users and the edge server need to exchange data from each other in each communication round. In our scheme, two frequently-used approaches of data transmission are adopted, named TDMA and broadcasting.

First, those selected users upload their local gradients to the edge server via the TDMA method. Specifically, a time frame is divided into n time slots. Each user transmits its data on its own time slot. According to Ref. [16], the length of each time frame in LTE standards is 10 ms. Actually, the transmission delay of the gradients is on the scale of second, which is far larger than the length of a time frame^[17]. Therefore, we can use the average uplink channel capacity, rather than the instantaneous channel capacity, to evaluate the data rate of user k ^[18], which can be expressed as

$$R_k^U = W \mathbb{E}_h \left\{ \log_2 \left(1 + \frac{p_k^U |h_k^U|^2}{N_0} \right) \right\}, \quad (6)$$

where h_k^U is the uplink channel power gain of user k , p_k^U is the corresponding transmission power, \mathbb{E}_h is the expectation over the uplink channel power gain, W is the system bandwidth, and N_0 is the noise power.

After the global gradient aggregation is finished, the BS will broadcast the global gradient to all users. In this way, all users are able to receive the global gradient synchronously. Let h_k^D denote the downlink channel power gain of user k and p^D denote the transmission power for all users. Thus, the downlink data rate is given by

$$R^D = W \min_{k \in \mathcal{K}} \left\{ \mathbb{E}_h \left\{ \log_2 \left(1 + \frac{p^D |h_k^D|^2}{N_0} \right) \right\} \right\}. \quad (7)$$

3 Problem Formulation

In this section, we will first propose an indicator to quantify the data importance of users. Then, we analyze the end-to-end latency in each communication round and formulate the opti-

mization problem to maximize the lower bound of the system learning efficiency.

3.1 Importance Analysis

In each communication round, only part of users is selected to participate in the training task because of the limited wireless communication resource. According to Ref. [13], different training samples do not equally contribute to the model training. Consequently, we intend to select users based on the level of data importance as well as the channel data rate. To quantify the data importance, we define the loss decay function as

$$\Delta L[n] = L(\boldsymbol{\theta}[n-1]) - L(\boldsymbol{\theta}[n]). \quad (8)$$

The loss decay function $\Delta L[n]$ indicates the decrease of the loss in the n -th communication round. From Eq. (8), in the same period of time, the larger the loss decays, the faster the training speed is. In other words, the loss decay reflects the data importance to some extent.

According to Ref. [19], the loss decay is proportional to the squared norm of the gradient. Thus, the lower bound of the loss decay in the n -th communication round is given as

$$\Delta L[n] \geq \beta \|\mathbf{G}^0[n]\|_2^2, \quad (9)$$

where β is a constant determined by the learning rate and the specific DNN model. Therefore, we can further link the data importance with the squared norm of the gradient vector. With the above discussions, we can quantify the data importance of user k by the squared norm of its local gradient, which can be represented as

$$\rho_k = \beta \|\mathbf{G}_k^0[n]\|_2^2, \forall k \in \mathcal{K}. \quad (10)$$

Therefore, the lower bound of the global loss decay in a communication round can be expressed as

$$\Delta L = \sum_{k=1}^K a_k \rho_k. \quad (11)$$

3.2 End-to-End Latency Analysis

As mentioned before, our goal is to improve the learning efficiency of the FEEL system. Thus, the end-to-end latency of one communication round should be optimized. The detailed analysis of latency in one communication round is given as follows.

1) Calculate local gradient. The latency of local training for user k is denoted by T_k^L .

2) Upload local gradient of selected users. As we mentioned before, only those selected users upload their local gradients to the edge server via TDMA. So the average transmission delay of user k can be expressed by

$$T_k^T = a_k \frac{V}{\tau_k R_k^U}, \forall k \in \mathcal{K}, \quad (12)$$

where τ_k is the proportion of the time slot for user k in a time frame and V is the volume of the gradient, which is a constant for all users.

3) Broadcast global gradient. For all users, the latency of downloading the global gradient is given by

$$T^D = \frac{V}{R^D}. \quad (13)$$

4) Update local model. Let us denote T_k^U as the delay of model updating for user k .

Since the squared norm of local gradient and the value of a_k are small enough, the corresponding transmission delay can be neglected. Besides, the edge server has powerful computing capacity in general. Therefore, the aggregation delay can also be neglected.

Then we provide further analysis to obtain the whole latency of one communication round. Note that all users receive the global gradient and start to update the local model synchronously. However, the delay of model updating and training varies since users may have different computing power. Hence, users are only allowed to upload the squared norm of local gradient to the edge server until they all finish model updating and training. In addition, the edge server should begin to aggregate the global gradient until those selected users have uploaded their local gradients. Based on the above analysis, the end-to-end latency of the FEEL system in one communication round is given by

$$T = \max_{k \in \mathcal{K}} \{T_k^U + T_k^L\} + \max_{k \in \mathcal{K}} T_k^T + T^D. \quad (14)$$

3.3 Problem Formulation

In this work, we aim to improve the learning efficiency of the FEEL system by jointly considering user selection and communication resource allocation. According to Ref. [20], we adopt the following criterion to evaluate the training performance of the FEEL system.

Definition 1: The learning efficiency of the FEEL system can be defined as

$$E = \frac{\Delta L}{T}. \quad (15)$$

Remark 1: The definition of the learning efficiency implies the decay rate of the global loss in a given time period T . The improvement of the learning efficiency means the acceleration of the training task. Therefore, it is appropriate to evaluate the training performance of the FEEL system by the learning efficiency. In our work, we aim to reduce the communication delay of each communication round. Besides, we maximize the lower bound of the system learning efficiency. Consequently,

the learning efficiency of the FEEL system can be improved.

Based on the above analysis, the optimization problem can be mathematically formulated as

$$\mathcal{P}_1: \max_{\{a_k, \tau_k, T\}} E = \frac{\Delta L}{T} = \frac{\sum_{k=1}^K a_k \rho_k}{T}, \quad (16a)$$

$$s.t. \max_{k \in \mathcal{K}} \{T_k^U + T_k^L\} + T_k^T + T^D \leq T, \forall k \in \mathcal{K}, \quad (16b)$$

$$\sum_{k=1}^K \tau_k \leq 1, \quad (16c)$$

$$a_k \in \{0, 1\}, \forall k \in \mathcal{K}, \quad (16d)$$

$$\tau_k, T \geq 0, \forall k \in \mathcal{K}, \quad (16e)$$

where the constraint (16b) indicates that the end-to-end latency of each user in one communication round is no more than the end-to-end latency of the FEEL system and the constraint (16c) represents the uplink communication resource limitation. For description convenience, we rewrite $\max_{k \in \mathcal{K}} \{T_k^U + T_k^L\} + T^D$ as T^C in the following sections.

4 Optimal Solution

4.1 Problem Transformation

It is evident that the optimization problem \mathcal{P}_1 is a mixed-integer programming problem. Since the objective function of \mathcal{P}_1 is non-convex, it is rather challenging to directly solve it. Combining Eqs. (12) and (16b), we notice that T is relevant to a_k and τ_k . When a_k and τ_k are fixed, the variable T must be minimized to maximize the learning efficiency. Therefore, the optimal solution to problem \mathcal{P}_1 can be obtained when “ \leq ” in the constraint (16b) is set to “ $=$ ”, i.e. $\tau_k = a_k V / R_k^U (T - T^C)$.

However, problem \mathcal{P}_1 is still hard to solve due to the integer constraint (16d). Therefore, we relax the integer constraint $a_k \in \{0, 1\}$ to the real-value constraint $a_k \in [0, 1]$. Problem \mathcal{P}_1 can then be relaxed into problem \mathcal{P}_2 , which is given by

$$\mathcal{P}_2: \max_{\{a_k, T\}} \frac{\sum_{k=1}^K a_k \rho_k}{T}, \quad (17a)$$

$$s.t. \sum_{k=1}^K \frac{a_k V}{R_k^U} \leq T - T^C, \quad (17b)$$

$$a_k \in [0, 1], \forall k \in \mathcal{K}, \quad (17c)$$

$$T \geq 0. \quad (17d)$$

In the following sections, we first obtain the optimal solution to problem \mathcal{P}_2 with fixed T . Then, we continue to solve the problem \mathcal{P}_2 with varying T , and the optimal solution to problem \mathcal{P}_1 is finally derived.

4.2 Optimal User Selection

We now solve the problem \mathcal{P}_2 . When T is given, problem \mathcal{P}_2 can be converted to a standard convex optimization problem since the objective function is concave and all constraints are convex. Thus, we can derive the optimal solution to \mathcal{P}_2 with fixed T .

Theorem 1: The optimal solution to problem \mathcal{P}_2 with fixed T is given as follows.

- 1) If $\rho_k R_k^U < \lambda^*$, $a_k^* = 0$;
- 2) If $\rho_k R_k^U > \lambda^*$, $a_k^* = 1$;
- 3) If $\rho_k R_k^U = \lambda^*$, $0 \leq a_k^* \leq 1$,

where λ^* is the optimal value of the Lagrange multiplier satisfying the constraint (17b). Particularly, the real-value of a_k^* depends on the constraint (17b) if $\rho_k R_k^U = \lambda^*$.

Proof: See Appendix A.

Remark 2 (Optimal user selection policy): According to Theorem 1, λ^* can be regarded as the threshold which determines whether to select the user. Besides, the selection priority of user k depends on the product of its data importance ρ_k and the uplink data rate R_k^U . On the one hand, a user with more important data contributes more to the global model training. On the other hand, the transmission delay can be shortened by selecting users with higher uplink data rates. Thus, the system prefers to select users with larger values of $\rho_k R_k^U$. By doing so, the learning efficiency of the FEEL system can be improved.

4.3 Optimal System Latency and Communication Resource Allocation

In this part, we proceed to obtain the optimal system latency and develop the optimal communication resource allocation to further improve the learning efficiency of the FEEL system. So far, we have obtained the optimal user selection strategy when the system latency is invariant. Based on this, the optimal system latency must be obtained when “ \leq ” in the constraint (17b) is set to “ $=$ ”, i.e., $T = \sum_{k=1}^K a_k V / R_k^U + T^C$. In order to develop the optimal T and τ_k , we introduce the following theorem.

Theorem 2: The optimal solutions to problem \mathcal{P}_2 and problem \mathcal{P}_1 are exactly the same.

Proof: See Appendix B.

Remark 3 (Optimal system latency and communication resource allocation): Theorem 2 indicates that the optimal solu-

tion of a_k to problem \mathcal{P}_2 must be an integer solution. Based on this, the range of feasible solutions to problem \mathcal{P}_2 can be reduced greatly. Thus, we only need to compare the learning efficiency of the FEEL system when the total number of selected users varies. Here, users in the system are selected by the optimal user selection policy as aforementioned. So T^* that achieves the maximum learning efficiency is the optimal system latency to both problems \mathcal{P}_2 and \mathcal{P}_1 , which can be expressed as

$$T^* = \sum_{k=1}^K \frac{a_k^* V}{R_k^U} + T^C. \quad (18)$$

As we have indicated before, when “ \leq ” in the constraint (16b) is set to “ $=$ ”, the solution must be the optimal solution of problem \mathcal{P}_1 . Consequently, we can obtain the optimal communication resource allocation by simple mathematical calculation, as

$$\tau_k^* = a_k^* \frac{V}{R_k^U (T^* - T^C)}. \quad (19)$$

The result in Eq. (19) shows that a less time slot is allocated for the user with a higher uplink data rate.

4.4 Optimal Algorithm for Problem \mathcal{P}_1

Thus far, we have obtained the optimal solution to problem \mathcal{P}_1 . In this part, we intend to develop an optimal algorithm for problem \mathcal{P}_1 based on the above analysis. As mentioned before, in order to obtain the optimal solution to problem \mathcal{P}_1 , all selection cases should be compared. However, this would become very time-consuming as the number of users increases. Therefore, a low computational complexity algorithm is required. We define E_M , $M \in \{1, 2, \dots, K\}$ as the learning efficiency of the FEEL system when M users are selected. To better fit the practical systems, we have the following theorem.

Theorem 3: E_M increases first and then decreases with the increase of M .

Proof: See Appendix C.

Remark 4: Theorem 3 indicates that the learning efficiency E_M has only one global optimal. Therefore, we can select users successively by the optimal user selection policy until the learning efficiency of the FEEL system begins to decrease. By doing so, we are able to find the optimal solution to problem \mathcal{P}_1 . According to the above analysis, the optimal algorithm for problem \mathcal{P}_1 is shown in **Algorithm 1**. We can easily find that the computational complexity of this algorithm is determined by the sort operation. Therefore, the computational complexity is $\mathcal{O}(K \log K)$. With regard to mixed-integer programming problems, it is acceptable to find the optimal solution with a polynomial-time complexity, indicating that this algorithm can be applied to practical systems.

Algorithm 1: The optimal algorithm for problem \mathcal{P}_1

```

1: Calculate  $\rho_k R_k^U, \forall k \in \mathcal{K}$ .
2: Sort  $\rho_k R_k^U$  in descending order.
3: Select user successively by  $\rho_k R_k^U$  and calculate the learning efficiency  $E_M$  of the FEEL system.
4: For  $M = 1$  to  $K$ , do
5:   if  $M = 1$ , then
6:      $E_{\max} = E_M$ .
7:   else
8:     if  $E_M < E_{\max}$ , then
9:       break.
10:  else
11:     $E_{\max} = E_M$ .
12: End
13: Calculate the corresponding  $\{a_k^*, T^*, \tau_k^{U*}\}$  with  $E_{\max}$ .
14: Output the optimal solution  $\{a_k^*, T^*, \tau_k^{U*}\}$ .

```

5 Simulation Results

In this section, we test the performance of the proposed algorithm by simulation and validate the performance improvement by comparing with other traditional algorithms.

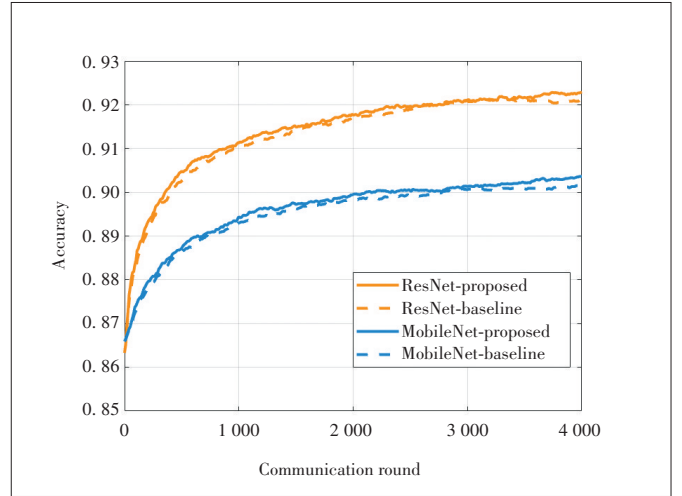
5.1 Simulation Settings

In the FEEL system, K users are stochastically distributed over the coverage of the BS. The coverage area of the BS is a circle with a radius of 500 m. All users are connected with the BS by wireless channels. The channel gains are generated by the pass loss model, $128.1 + 37.6 \log(d \text{ [km]})$, while the small-scale fading obeys the Rayleigh distribution with uniform variance. The noise power spectral density is -174 dBm/Hz and the system bandwidth is 5 MHz. The uplink and downlink transmit powers are both 24 dBm.

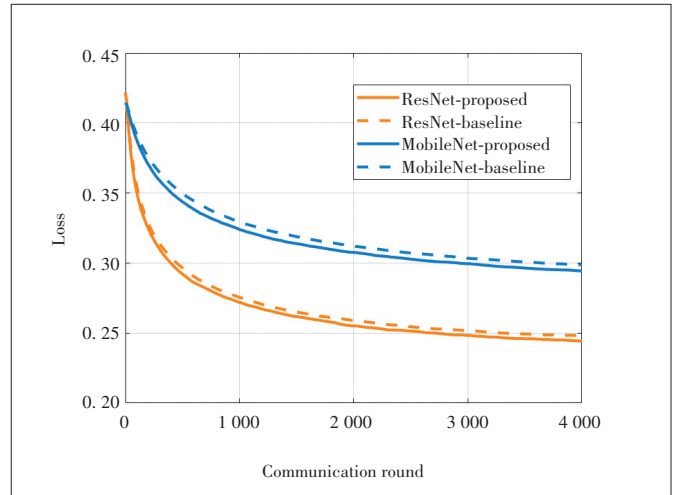
We utilize the dataset CIFAR-10 as the local dataset of all users to train model. The dataset is composed of 60 000 32×32 color images in 10 classes, which includes 50 000 training images and 10 000 test images. We shuffle all training samples first, divide them into K parts equally and then distribute them to all users, respectively. Two common DNN models, MobileNetV2 and ResNet18, are deployed for image classification. Since it is time-consuming to restart training, we utilize the pre-trained model to reduce the model convergence time.

5.2 Tests of Generalization Ability

The generalization ability refers to the adaptability of algorithms to different DNN models. To test the generalization ability of our proposed algorithm, we implement it on the two DNN models as mentioned before when there are $K = 14$ users in the FEEL system. Meanwhile, we make comparisons with the performance of proposed algorithm and the *baseline algorithm* where all users are selected with equal communication resource allocation. The simulation results of the test accuracy and the global training loss are shown in **Figs. 2 and 3**,



▲ **Figure 2.** The test accuracy versus communication round.



▲ **Figure 3.** The global training loss versus communication round.

respectively. From the figures, the proposed algorithm can achieve a high learning accuracy and a fast convergence rate for different DNN models. The result shows that our proposed algorithm has excellent generalization ability and can be widely implemented in practical systems. Moreover, the performance of our proposed algorithm is similar to that of the *baseline algorithm* with the increase of communication round rather than training time. It is reasonable since our proposed algorithm aims to reduce the communication delay in each communication round, rather than the number of communication rounds. Besides, this result demonstrates that our proposed algorithm can achieve the similar training speed by only selecting partial users in the FEEL system.

5.3 Performance Comparison Among Different Algorithms

In this part, we compare the performance of our proposed algorithm with other conventional algorithms to verify its superiority. The two benchmark algorithms are described as follows.

- *Baseline algorithm:* In each communication round, all users

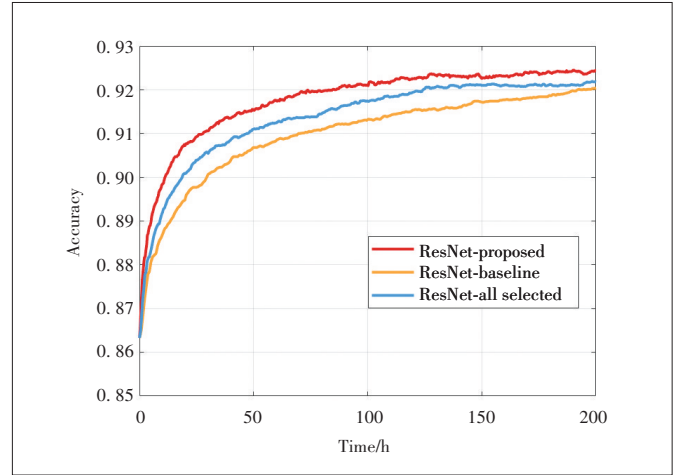
in the FEEL system participate in the training task with equal communication resource allocation, i.e., $\tau_k = 1/K, \forall k \in \mathcal{K}$.

- *All selected algorithm*: In each communication round, all users in the FEEL system participate in the training task with optimal communication resource allocation based on Eq. (19).

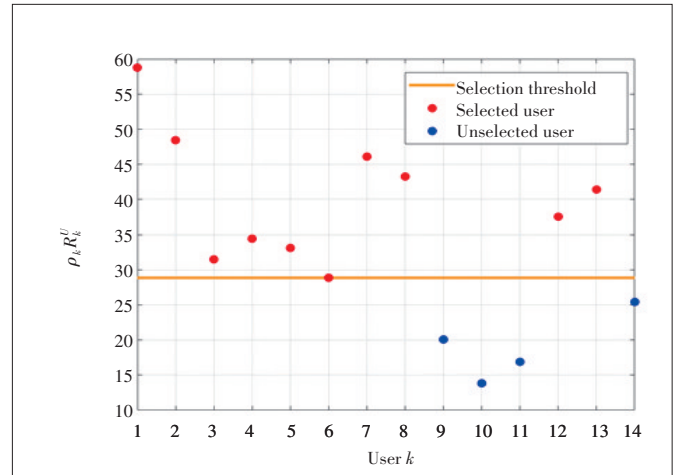
Here we use the pre-trained ResNet18 model to test the performance of the three algorithms in an FEEL system with $K = 14$ users. The test accuracy versus training time with different algorithms is shown in **Fig. 4**. From the figure, it can be seen that our proposed algorithm achieves the highest test accuracy among all algorithms. The reason is that our proposed algorithm not only selects users based on data importance but also makes the optimal communication resource allocation. By doing so, only users with more important data and higher uplink data rate participate in the training task. Thus, the communication latency is reduced and the global loss decay rate increases, which eventually improves the learning efficiency of the system. The gap between the *baseline algorithm* and the *all selected algorithm* demonstrates the gain obtained by the optimal communication resource allocation. The gap between the *all selected algorithm* and the *proposed algorithm* demonstrates the gain obtained by the optimal user selection. In conclusion, our proposed algorithm accelerates the training task and improves the learning efficiency of the FEEL system by jointly considering user selection and communication resource allocation.

To further verify the applicability and effectiveness of our proposed algorithm, we select one communication round randomly to obtain more simulation results. **Figs. 5** and **6** illustrate the results of user selection and communication resource allocation for our proposed algorithm in the communication round we selected, respectively. From **Fig. 5**, we can observe that user k is selected only when the product of its data importance and uplink data rate, i.e., $\rho_k R_k^U$, is no less than the selection threshold, which is consistent with Theorem 1. Moreover, in order to clearly present the relationship between the communication resource allocation and the uplink data rate, we plot the corresponding uplink data rate for all users in **Fig. 7**. Combining **Fig. 6** with **Fig. 7**, it can be observed that a selected user with a higher uplink data rate is allocated with less communication resource, which is consistent with Eq. (19).

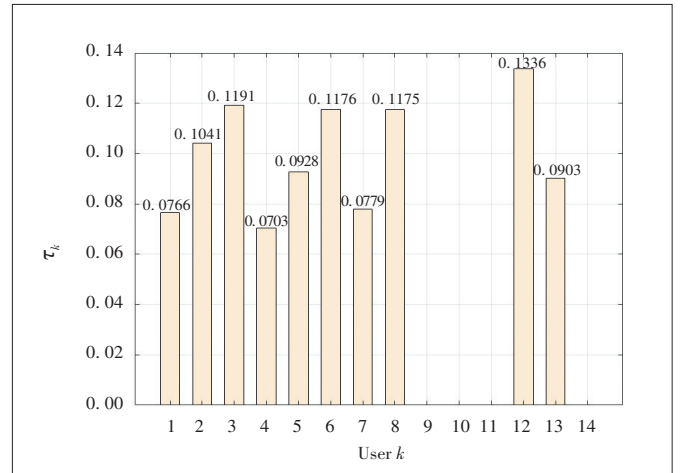
In the end, we further study how the number of users impacts the training performance of the FEEL system. The test accuracy versus training time with different numbers of users is shown in **Fig. 8**. From the figure, it can be seen that our proposed algorithm achieves the highest system learning efficiency when $K = 6$. The reasons can be explained as follows. The number of time slot allocated to the selected user is large when the number of users is small. Consequently, the communication latency greatly reduces, and the learning efficiency of the FEEL system significantly improves in this scenario. Moreover, the number of selected users is limited by the scarce



▲ **Figure 4.** The test accuracy versus training time with different algorithms.

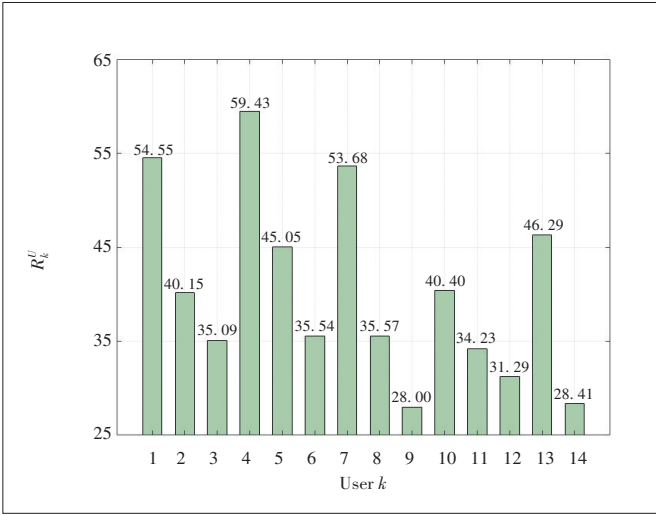


▲ **Figure 5.** User selection for the proposed algorithm.

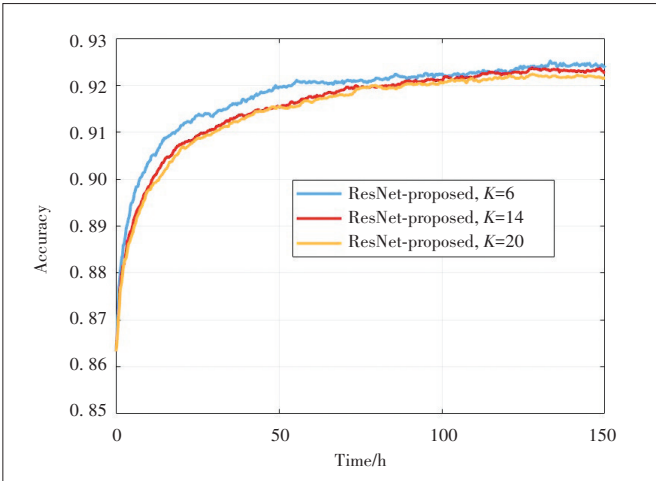


▲ **Figure 6.** Communication resource allocation for the proposed algorithm.

wireless communication resource when the number of users is too large. Therefore, the learning efficiency of the FEEL system does not improve with user number when too many users in the system.



▲ Figure 7. Corresponding uplink data rate.



▲ Figure 8. Test accuracy versus training time with different numbers of users.

6 Conclusions

In this paper, we aim to accelerate the training task and improve the learning efficiency of the FEEL system by proposing an optimal user selection policy based on data importance and CSI. After analyzing the data importance of users and the end-to-end latency of the FEEL system, we formulate an optimization problem to maximize the learning efficiency of the FEEL system. By problem transformation and relaxation, we first develop the optimal user selection policy. Based on this, the optimal communication resource allocation is developed in closed-form. We further develop a polynomial-time algorithm to solve this mixed-integer programming problem and prove its optimality. Finally, the simulation results show that our proposed algorithm has strong generalization ability and can significantly improve the learning efficiency of the FEEL system.

Our work has demonstrated that the learning efficiency of the FEEL system can be further improved by user selection

based on data importance and wireless resource allocation. However, some assumptions have been made to gain insightful results. In the future, we will make further investigation to better fit the practical systems. First, we have assumed that there is no inter-cell interference in the uplink. In the future, the FEEL system with inter-cell interference deserves further investigation. Second, the local gradient received by the edge server may contain data errors, which may affect the training performance of the FEEL system. Therefore, our future work can further study the impact of those errors. Last but not the least, it is meaningful to extend our proposed algorithm to the FEEL system, where orthogonal frequency-division multiple access (OFDMA) is adopted for data transmission.

Appendix A

Proof of Theorem 1

We apply the Lagrangian method to obtain the optimal solution to problem $\mathcal{P}2$ with fixed T since it is a convex optimization problem. The Lagrangian function is defined as

$$L = -\frac{\sum_{k=1}^K a_k \rho_k}{T} + \lambda \left(\sum_{k=1}^K \frac{a_k V}{R_k^U} - T + T^c \right), \quad (20)$$

where λ is the Lagrange multiplier related with the constraint (17b). By applying the Karush-Kuhn-Tucker (KKT) conditions and simple calculation, we can draw the following necessary and sufficient conditions, as

$$\frac{\partial L}{\partial a_k} = -\frac{\rho_k}{T} + \lambda^* \frac{V}{R_k^U} \begin{cases} \geq 0, & a_k^* = 0, \\ = 0, & 0 \leq a_k^* \leq 1, \forall k \in \mathcal{K}, \\ \leq 0, & a_k^* = 1, \end{cases} \quad (21)$$

$$\lambda^* \left(\sum_{k=1}^K \frac{V a_k^*}{R_k^U} + T^c - T \right) = 0, \quad \lambda^* \geq 0. \quad (22)$$

With simple mathematical calculation, we can derive the optimal user selection policy as shown in Theorem 1, which ends the proof.

Appendix B

Proof of Theorem 2

According to Theorem 1, users are selected by the descending order of $\rho_k R_k^U$. Hence, we can assume that $a_k = 1$ when $k = 1, 2, \dots, M$ and $a_k = 0$ when $k = M+2, M+3, \dots, K$. Moreover, it is not clear whether $a_{M+1} = 0$ or $a_{M+1} = 1$. Then, we denote $E^{(1)}$ as the objective function of problem $\mathcal{P}2$, which can be expressed as

$$E^{(1)} = \frac{\sum_{k=1}^M \rho_k + a_{M+1} \rho_{M+1}}{\sum_{k=1}^M \frac{V}{R_k^U} + a_{M+1} \frac{V}{R_{M+1}^U} + T^C}. \quad (23)$$

So the derivative of $E^{(1)}$ with respect to a_{M+1} is given by

$$\frac{\partial E^{(1)}}{\partial a_{M+1}} = \frac{\rho_{M+1} \left(\sum_{k=1}^M \frac{V}{R_k^U} + T^C \right) - \frac{V}{R_{M+1}^U} \sum_{k=1}^M \rho_k}{\left(\sum_{k=1}^M \frac{V}{R_k^U} + a_{M+1} \frac{V}{R_{M+1}^U} + T^C \right)^2}. \quad (24)$$

It shows that the sign of derivative is consistent with the sign of the numerator of Eq. (24). However, the value of the numerator of Eq. (24) is independent of a_{M+1} . Therefore, $E^{(1)}$ is monotone when $a_{M+1} \in [0, 1]$. That is, the maximum value of $E^{(1)}$ must be obtained either when $a_{M+1} = 0$ or when $a_{M+1} = 1$. In conclusion, the optimal solution of a_k to \mathcal{P}_2 must be an integer solution. Hence, this solution must be the feasible solution to problem \mathcal{P}_1 as well. Moreover, after relaxation, the maximum value of the objective function is non-decreasing. Thus, the optimal solutions to problem \mathcal{P}_2 and \mathcal{P}_1 are exactly the same, which ends the proof.

Appendix C

Proof of Theorem 3

According to Theorem 2, we know that the optimal solutions to problem \mathcal{P}_2 and \mathcal{P}_1 are exactly the same. Thus, we only consider the integer solutions here. When no user is selected, the learning efficiency is zero obviously. The learning efficiency must increase first with the number of selected users. In other words, at least one user is selected. Then we consider the following condition.

Denote $T_M = \sum_{k=1}^M V/R_k^U + T^C, M \in \{1, 2, \dots, K\}$ as the system latency when M users are selected. Assume that the following formulas exist

$$E_M - E_{M-1} = \frac{\rho_M \sum_{k=1}^{M-1} \frac{V}{R_k^U} + T^C \rho_M - \frac{V}{R_M^U} \sum_{k=1}^{M-1} \rho_k}{\left(\sum_{k=1}^M \frac{V}{R_k^U} + T^C \right) \left(\sum_{k=1}^{M-1} \frac{V}{R_k^U} + T^C \right)} > 0, \quad (25)$$

$$E_{M+1} - E_M = \frac{\rho_{M+1} \sum_{k=1}^M \frac{V}{R_k^U} + T^C \rho_{M+1} - \frac{V}{R_{M+1}^U} \sum_{k=1}^M \rho_k}{\left(\sum_{k=1}^{M+1} \frac{V}{R_k^U} + T^C \right) \left(\sum_{k=1}^M \frac{V}{R_k^U} + T^C \right)} < 0. \quad (26)$$

From Eqs. (25) and (26), we can obtain the following inequalities.

$$\rho_M R_M^U \left(\sum_{k=1}^{M-1} \frac{V}{R_k^U} + T^C \right) > V \sum_{k=1}^{M-1} \rho_k, \quad (27)$$

$$\rho_{M+1} R_{M+1}^U \left(\sum_{k=1}^M \frac{V}{R_k^U} + T^C \right) < V \sum_{k=1}^M \rho_k. \quad (28)$$

According to Eq. (27), we can derive the recurrence formula as

$$\begin{aligned} & \rho_{M-1} R_{M-1}^U \left(\sum_{k=1}^{M-2} \frac{V}{R_k^U} + T^C \right) - V \sum_{k=1}^{M-2} \rho_k = \\ & \rho_{M-1} R_{M-1}^U \left(\sum_{k=1}^{M-1} \frac{V}{R_k^U} + T^C \right) - V \sum_{k=1}^{M-1} \rho_k > \\ & \rho_M R_M^U \left(\sum_{k=1}^{M-1} \frac{V}{R_k^U} + T^C \right) - V \sum_{k=1}^{M-1} \rho_k > 0, \end{aligned} \quad (29)$$

which implies $E_{M-2} < E_{M-1}$. Then we can obtain the conclusion recursively, as

$$E_1 < E_2 < \dots < E_M. \quad (30)$$

Similar to the above analysis, we have the following conclusion, as

$$E_1 < E_2 < \dots < E_M > E_{M+1} > E_{M+2} > \dots > E_K. \quad (31)$$

Based on the above analysis, E_M first increases and then decreases with M , which ends the proof.

References

- [1] ZHOU Z, CHEN X, LI E, et al. Edge intelligence: paving the last mile of artificial intelligence with edge computing [J]. *Proceedings of the IEEE*, 2019, 107 (8): 1738 – 1762. DOI: 10.1109/jproc.2019.2918951
- [2] ZHU G, LIU D Z, DU Y Q, et al. Towards an intelligent edge: wireless communication meets machine learning [EB/OL]. (2018-09-02)[2019-12-05]. <https://arxiv.org/abs/1809.00343>
- [3] MCMAHAN B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data [C]//20th International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, USA, 2017: 1273 – 1282.
- [4] ETSI. Mobile edge computing-introductory technical white paper [R]. 2014
- [5] ZHU G X, WANG Y, HUANG K B. Broadband analog aggregation for low-latency federated edge learning (extended version) [EB/OL]. (2018-10-30)[2019-01-16]. <https://arxiv.org/abs/1812.11494>
- [6] NISHIO T, YONETANI R. Client selection for federated learning with heterogeneous resources in mobile edge [C]//IEEE International Conference on Communications (ICC). Shanghai, China, 2019: 18852422. DOI: 10.1109/icc.2019.8761315
- [7] ZENG Q S, DU Y Q, LEUNG K K, et al. Energy-efficient radio resource allocation for federated edge learning [EB/OL]. (2019-07-13)[2019-12-20]. <https://arxiv.org/abs/1907.06040>

Biographies

JIANG Zhihui (zhihui.jiang@zju.edu.cn) received the B.E. degree in information engineering from Zhejiang University, China in 2020, where she is currently pursuing the master's degree with the College of Information Science and Electronic Engineering. Her research interests mainly include federated learning and edge learning.

HE Yinghui received the B.E. degree in information engineering from Zhejiang University, China in 2018, where he is currently pursuing the master's degree with the College of Information Science and Electronic Engineering. His research interests mainly include mobile edge computing and device-to-device communications.

YU Guanding received the B.E. and Ph.D. degrees in communication engineering from Zhejiang University, China in 2001 and 2006, respectively. He joined Zhejiang University, in 2006, where he is currently a full professor with the College of Information and Electronic Engineering. From 2013 to 2015, he was a visiting professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology, USA. His research interests include 5G communications and networks, mobile edge computing, and machine learning for wireless networks. Dr. YU received the 2016 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award. He regularly chairs the technical program committee boards of prominent IEEE conferences, such as ICC, GLOBECOM, and VTC. He also serves as a symposium co-chair for IEEE Globecom 2019 and the track chair for IEEE VTC 2019' Fall. He has served as a guest editor for the *IEEE Communications Magazine* special issue on full-duplex communications, an editor for the *IEEE Journal on Selected Areas in Communications* Series on green communications and networking, a leading guest editor for the *IEEE Wireless Communications Magazine* special issue on LTE in unlicensed spectrum, and an editor for the *IEEE Access*. He serves as an editor for the *IEEE Transactions on Green Communications and Networking* and the *IEEE Wireless Communications Letters*.

- [8] YANG Z H, CHEN M Z, SAAD W, et al. Energy efficient federated learning over wireless communication networks [EB/OL]. (2019-11-06)[2019-12-10]. <https://arxiv.org/abs/1911.02417>
- [9] YANG H H, LIU Z Z, QUEK T Q S, et al. Scheduling policies for federated learning in wireless networks [J]. *IEEE transactions on communications*, 2020, 68(1): 317 – 333. DOI: 10.1109/tcomm.2019.2944169
- [10] YANG H H, ARAFA A, QUEK T Q S, et al. Age-based scheduling policy for federated learning in mobile edge networks [C]//IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Barcelona, Spain, 2020. DOI: 10.1109/icassp40776.2020.9053740
- [11] CHEN M Z, YANG Z H, SAAD W, et al. A joint learning and communications framework for federated learning over wireless networks [EB/OL]. (2019-09-17)[2020-01-10]. <https://arxiv.org/abs/1909.07972>
- [12] CHEN M Z, POOR H V, SAAD W, et al. Convergence time optimization for federated learning over wireless networks [EB/OL]. (2020-01-22)[2020-03-25]. <https://arxiv.org/abs/2001.07845>
- [13] KATHAROPOULOS A, FLEURET F. Not all samples are created equal: deep learning with importance sampling [EB/OL]. (2018-03-02)[2019-10-28]. <https://arxiv.org/abs/1803.00942>
- [14] LIU D Z, ZHU G X, ZHANG J, et al. Wireless data acquisition for edge learning: data-importance aware retransmission [EB/OL]. (2018-10-05)[2019-03-19]. <https://arxiv.org/abs/1812.02030>
- [15] LIU D Z, ZHU G X, ZHANG J, et al. Data-importance aware user scheduling for communication-efficient edge machine learning [EB/OL]. (2019-03-19)[2019-10-05]. <https://arxiv.org/abs/1910.02214>
- [16] ETSI. LTE; evolved universal terrestrial radio access (E-UTRA); physical channels and modulation (3GPP TS 36.211 version 15.6.0 release 15); ETSI TS 136 211 V15.6.0 [S]. ETSI, 2019
- [17] LIN Y J, HAN S, MAO H Z, et al. Deep gradient compression: reducing the communication bandwidth for distributed training [EB/OL]. (2017-10-05)[2018-02-05]. <https://arxiv.org/abs/1712.01887>
- [18] REN J K, YU G D, CAI Y L, et al. Latency optimization for resource allocation in mobile-edge computation offloading [J]. *IEEE transactions on wireless communications*, 2018, 17(8): 5506 – 5519. DOI: 10.1109/twc.2018.2845360
- [19] CHEN T Y, GIANNAKIS G B, SUN T, et al. LAG: lazily aggregated gradient for communication-efficient distributed learning [EB/OL]. (2018-05-25)[2019-12-02]. <https://arxiv.org/abs/1805.09965>
- [20] REN J K, YU G D, and DING G Y. Accelerating DNN training in wireless federated edge learning system [EB/OL]. (2019-05-23)[2020-03-28]. <https://arxiv.org/abs/1905.09712>

◀ From Page 01

future directions. The third paper “Joint User Selection and Resource Allocation for Fast Federated Edge Learning” by JIANG et al. presents a new policy for joint user selection and communication resource allocation to accelerate the training task and improve the learning efficiency.

Edge learning includes both edge training and edge inference. Due to the stringent latency requirements, edge inference is particularly bottlenecked by the limited computation and communication resources at the network edge. The fourth paper “Communication-Efficient Edge AI Inference over Wireless Networks” by YANG et al. identifies two communication-efficient architectures for edge inference, namely, on-device distributed inference and in-edge cooperative inference, thereby achieving low latency and high energy efficiency. The fifth paper “Knowledge Distillation for Mobile Edge

Computation Offloading” by CHEN et al. introduces a new computation offloading framework based on deep imitation learning and knowledge distillation that assists end devices to quickly make fine-grained offloading decisions so as to minimize the end-to-end task inference latency in MEC networks. By considering edge inference in MEC-enabled UAV systems, the last paper “Joint Placement and Resource Allocation for UAV-Assisted Mobile Edge Computing Networks with URLLC” by ZHANG et al. jointly optimizes the UAV's placement location and transmitting power to facilitate ultra-reliable and low-latency round-trip communication from sensors to UAV servers to actuators.

We hope that the aforementioned six papers published in this special issue stimulate new ideas and innovations from both the academia and industry to advance this exciting area of edge learning.



Communication-Efficient Edge AI Inference over Wireless Networks

YANG Kai, ZHOU Yong, YANG Zhanpeng, SHI Yuanming

(School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China)

Abstract: Given the fast growth of intelligent devices, it is expected that a large number of high-stakes artificial intelligence (AI) applications, e.g., drones, autonomous cars, and tactile robots, will be deployed at the edge of wireless networks in the near future. Therefore, the intelligent communication networks will be designed to leverage advanced wireless techniques and edge computing technologies to support AI-enabled applications at various end devices with limited communication, computation, hardware and energy resources. In this article, we present the principles of efficient deployment of model inference at network edge to provide low-latency and energy-efficient AI services. This includes the wireless distributed computing framework for low-latency device distributed model inference as well as the wireless cooperative transmission strategy for energy-efficient edge cooperative model inference. The communication efficiency of edge inference systems is further improved by building up a smart radio propagation environment via intelligent reflecting surface.

Keywords: communication efficiency; cooperative transmission; distributed computing; edge AI; edge inference

DOI: 10.12142/ZTECOM.202002005

<https://kns.cnki.net/kcms/detail/34.1294.TN.20200611.1444.003.html>, published online June 11, 2020

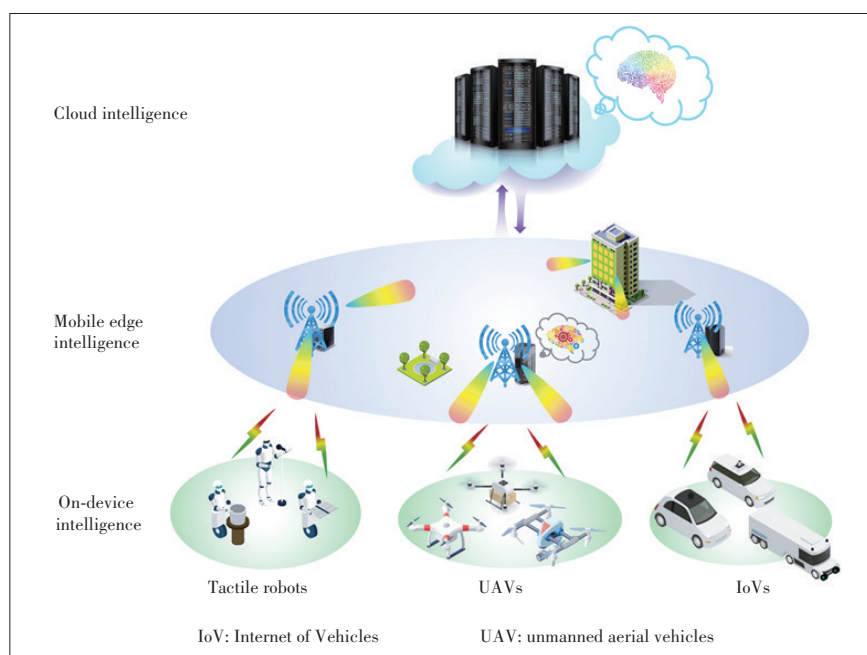
Manuscript received: 2020-02-10

Citation (IEEE Format): K. Yang, Y. Zhou, Z. P. Yang, et al., "Communication-efficient edge AI inference over wireless networks," *ZTE Communications*, vol. 18, no. 2, pp. 31 – 39, Jun. 2020. doi: 10.12142/ZTECOM.202002005.

1 Introduction

The past few decades have witnessed a rapidly growing interest in the area of artificial intelligence (AI), which has contributed to the astonishing breakthroughs in image recognition, speech processing, etc. With the advancement of mobile edge computing^[1], it becomes increasingly attractive to push the AI engine from the cloud center to the network edge. Such a transition makes AI proximal to the end devices and has the potential to mitigate the privacy and latency concerns. This novel area is termed as "edge AI", including both edge training and edge inference, which is envisioned to revolutionize the future mobile networks and enable

the paradigm shift from "connected things" to "connected intelligence"^[2]. Edge AI can provide various AI services, such as Internet of Vehicles (IoV), unmanned aerial vehicles (UAVs) and tactile robots, as illustrated in **Fig. 1**. By deploying AI models and performing inference tasks at network edge, edge inference is the main focus of this article and faces the following three major challenges. First, the large size of AI models makes it difficult to be deployed at the network edge. Second, the inference latency is severely bottlenecked by the limited computation and communication resources at the network edge. Third, edge devices are usually battery-powered



▲ Figure 1. Illustration of edge AI.

with limited energy budget and computing power.

It is generally impractical to deploy the entire AI models on a single resource-constrained end device. Fortunately, a recently proposed edge inference architecture, termed as “on-device distributed AI inference”, is capable of pooling the computing resources on a large number of distributed devices to perform inference tasks requested by each end device^[3]. For popular distributed computing structures such as MapReduce^[4], the dataset (i.e., AI model for inference) is split and deployed on end devices during the phase of dataset placement. Each end device computes the intermediate values of all tasks locally with the map functions. After exchanging the intermediate values, each device obtains all map function values for its inference task and performs the reduce function to yield the desired inference result. However, the communication efficiency of the intermediate value exchange is the main performance bottleneck of distributed edge inference systems^[5]. To this end, we shall propose a communication-efficient data shuffling strategy for on-device distributed AI inference based on cooperative transmission and interference alignment.

For computation-intensive inference tasks, it is beneficial to deploy the AI models at the edge servers, e.g., access points (APs), followed by uploading the input dataset to the proximal edge servers. This helps to perform the inference tasks and return the inference results to the end devices through downlink transmission. On the other hand, cooperative transmission^[6] is a well-known approach that can mitigate co-channel interference as well as improve the reliability and energy efficiency of downlink transmission. These facts motivate us to propose the in-edge cooperative AI inference architecture by performing

each task at multiple edge servers and enabling cooperative transmission to improve the quality of service (QoS) and reliability for the delivery of inference results. However, performing each inference task by multiple edge servers leads to a higher computation power consumption. We thus propose a joint task allocation and downlink coordinated beamforming approach to achieve energy-efficient in-edge cooperative AI inference through minimizing the total power consumption consisting of both transmit and computation power consumptions under the target QoS constraints.

Although our joint computation and communication designs can greatly improve the communication efficiency for on-device distributed AI inference and in-edge cooperative AI inference, the achievable low-latency and energy efficiency are still fundamentally limited by the radio propagation environment. We thus resort to an emerging technology,

i.e., intelligent reflecting surface (IRS)^[7], to actively control the wireless propagation environment. In particular, we propose to utilize the IRS for further enhancing the communication efficiency of edge AI inference systems, thereby providing low-latency and energy-efficient AI services. By dynamically adjusting the phase shifts of the IRS, our proposed strategy improves the feasibility of the interference alignment conditions for the data shuffling of on-device distributed AI inference systems, as well as reduces the energy consumption of in-edge cooperative AI inference systems.

2 Overview of Edge AI Inference

In this section, we present the architectures, key performance metrics, and promising applications of edge AI inference.

2.1 Architecture

In the conventional cloud-based AI systems, a large amount of data collected/generated by the end devices is required to be delivered to the central cloud center for AI model training. Such cloud-based AI systems are generally limited by scarce spectrum resources and susceptible to data modification attacks. With the increase of the computing power and storage capability of edge servers (e.g., APs and base stations) and end devices (vehicles and robots), there is a trend of pushing AI engines from the cloud center to the network edge^[8–9]. Therefore, edge AI emerges as a promising research area that performs the training and inference tasks at the network edge. In this article, we go beyond that and focus on a much broader scope of edge AI to fully leverage the distributed computation and storage resources at the network

edge across end devices, edge servers and cloud centers to provide low-latency and energy-efficient AI services, such as IoVs, UAVs and tactile robots.

According to Ref. [8], the edge AI can generally be classified into six levels, including cloud-edge co-inference, in-edge co-inference, on-device inference, cloud-edge co-training, all in-edge, and all on-device. The training of AI models can be performed on end devices, in edge servers, or with the collaboration of the cloud center and edge nodes, which are out of the scope of this article. We in this article mainly focus on the model inference of edge AI, also known as edge inference. The major architectures of edge inference are listed as follows:

- **Device-based edge inference:** Deploying AI models directly on end devices can reduce the communication cost due to information exchange. However, this poses stringent requirements on the storage capability, computing power, energy budget of each end device. To this end, a promising structure of device-based edge inference is to enable cooperation among multiple devices via a distributed computing framework^[3], i.e., on-device distributed AI inference.

- **Edge-based edge inference:** The end devices offload the dataset to the neighboring edge servers, which perform the inference tasks and return the inference results to the end users. This inference architecture has the potential to perform computation-intensive inference tasks. However, the limited channel bandwidth is the main performance-limiting factor of this edge inference architecture. To address this issue, it is promising to enable cooperation among multiple edge servers^[10-11] to facilitate in-edge cooperative AI inference.

- **Others:** In addition to the device-based and edge-based edge inference architectures, there are also other promising edge inference architectures. The device-edge architecture with model partition proposed in Refs. [12] and [13] can enhance the energy efficiency and reduce the latency of edge inference systems. Moreover, the inference tasks can also be accomplished by adopting the edge-cloud collaborative architecture, which is particularly suitable for end devices with highly constrained resources.

This paper emphasizes on two promising system architectures, i.e., on-device distributed AI inference and in-edge cooperative AI inference, which pool the computation and communication resources across multiple end devices and edge servers, respectively. In such distributed systems, the communication efficiency is a critical issue in determining the performance of edge inference systems. We thus focus on designing communication-efficient on-device distributed AI inference and in-edge cooperative AI inference strategies for computation-intensive inference tasks, thereby achieving low latency and high energy efficiency.

2.2 Key Performance Metrics

The communication efficiency of edge inference systems can be measured by the following metrics:

- **Latency:** In edge inference systems, latency is a crucial performance metric that measures how fast the inference results can be obtained, which in turn determines the timeliness of the inference results. The latency is generally composed of the computation and communication latency. Achieving low latency is challenging as it depends on various factors, including channel bandwidth, transmission strategy, and channel conditions.

- **Energy efficiency:** As performing inference tasks are generally energy consuming, the energy efficiency is a critical performance metric of edge inference systems. The energy consumption typically consists of both communication and computation energy consumptions, which depend on the type of the inference tasks and the size of the dataset.

- **Others:** There are also other indicators that can describe the performance of edge inference. For example, privacy is a major concern in edge inference systems for various high-stake AI applications such as IoVs and UAVs. For such applications, it is also critical to ensure that the inference results are received at the end devices with a high level of reliability.

2.3 Applications

Efficient edge inference is envisioned to be capable of supporting various low-latency AI services, including IoVs, UAVs, and tactile robots, as shown in Fig. 1.

- **Internet of Vehicles:** IoV is a network system that integrates networking and intelligence for promoting the efficiency of transportation and improving the quality of life^[4], as well as emphasizes the interaction of humans, vehicles, and roadside units. Numerous AI models are necessary for IoV such as the advanced driver-assistance system (ADAS) for the detection of vehicles, pedestrians, lane lines, etc. It is generally impractical to deploy all AI models on the resource-constrained vehicles. As a result, to achieve low-latency and energy-efficient inference for a large number of AI models, it is critical to pool the distributed computation and storage resources of the vehicles and edge servers at the network edge.

- **Unmanned aerial vehicles:** There has been a fast-growing interest in UAVs^[15] for the transportation of cargo, monitoring, relaying, etc. Although the UAVs are battery-powered with limited energy budget, they are deployed to accomplish a variety of intelligent computation tasks. As it is energy inefficient for UAVs to communicate with the remote cloud center, enabling cooperative inference on the devices or in the edge is a promising solution that can achieve low-latency and energy-efficient processing of inference tasks, as well as enhance the data privacy.

- **Tactile robots:** As remote representatives of human beings, smart robots are envisioned to be capable of achieving physical interaction by enabling haptic capabilities, leading to the new field of tactile robots^[16]. The greatly improved capability of processing tactile sensation and the connectivity of a large number of robots make tactile robots a representa-

tive embodiment of the tactile Internet. Exploring the potential of edge inference for tactile robots is able to provide integrated intelligence for agriculture, manufacture, health care, traffic, etc.

3 Wireless Distributed Computing System for On-Device Distributed AI Inference

In this section, we shall present a communication-efficient data shuffling strategy in the wireless distributed computing system for on-device distributed AI inference.

3.1 MapReduce-Based Distributed Computing System

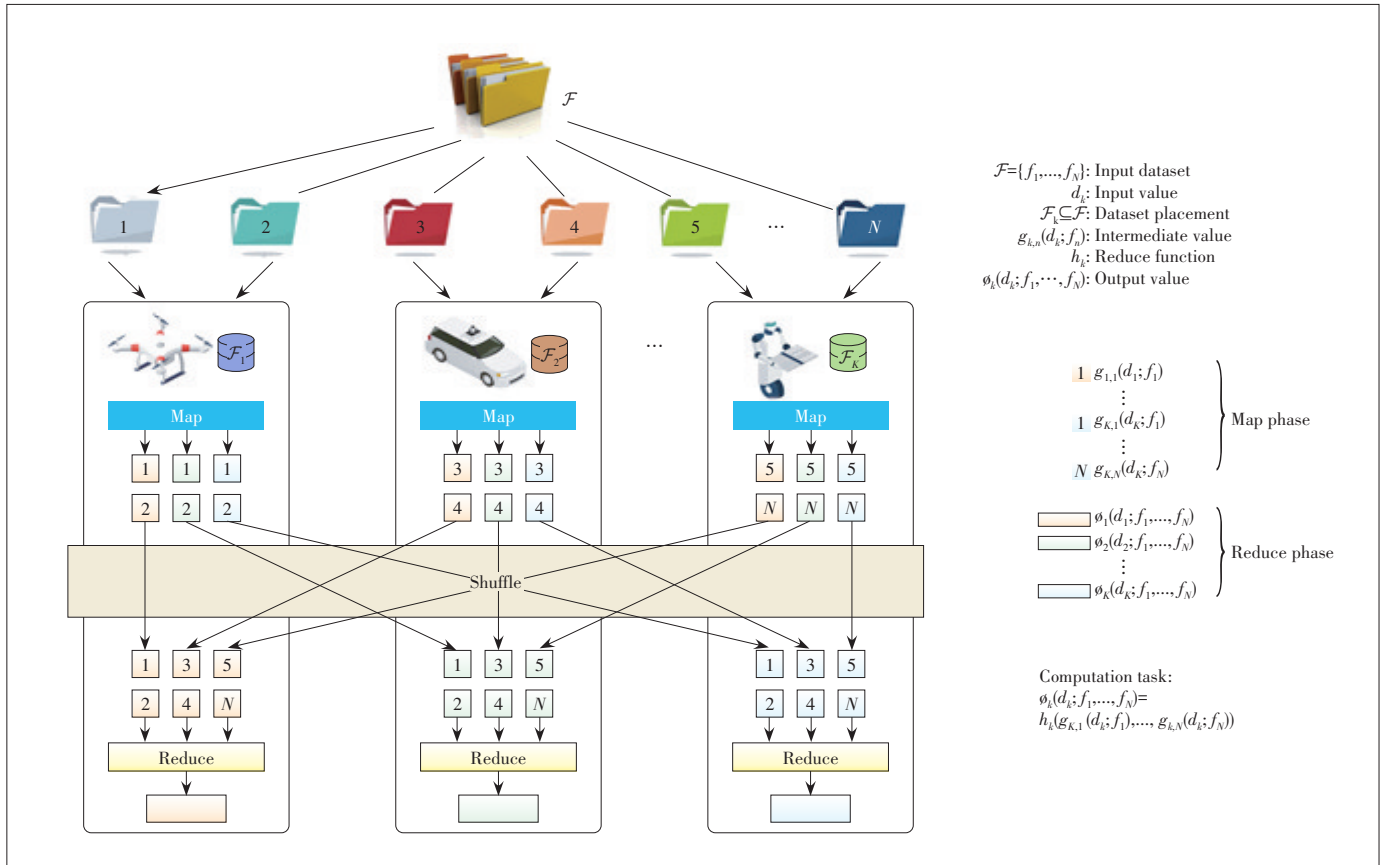
MapReduce is a ubiquitous distributed computing framework that processes tasks with a large amount of data across multiple distributed devices^[4]. For a computing task with the MapReduce-like structure, the target function is decomposed as the “reduce” function value of a number of map functions, which can be computed in a parallel manner. Hence, the MapReduce-based distributed computing system is capable of pooling the computation and storage resources of multiple devices to enable on-device distributed AI inference.

For a wireless distributed computing system consisting of multiple mobile devices, the inference result (e.g., a machine

learning model) to be obtained by each device depends on the entire input dataset. Supposing that each computation task for inference fits the MapReduce computation structure (**Fig. 2**), K mobile devices cooperatively accomplish the inference tasks through the following four phases

- **Dataset placement:** In this phase, the entire dataset is partitioned into N portions and each mobile device is allocated a subset of the entire dataset before inference.
- **Map function:** With the allocated local data, each mobile device computes the map function values with respect to all the input data, which yields the intermediate values for itself and other devices.
- **Shuffling:** As each mobile device does not have enough information for inference, the intermediate values computed by each device shall be transmitted to the corresponding devices over radio channels in this phase.
- **Reduce function:** Finally, based on the collected N intermediate values, each mobile device calculates the reduce function to obtain the corresponding inference result.

With limited radio resources, the shuffling of intermediate values among multiple mobile devices leads to significant communication overhead and is the main performance-limiting factor for on-device distributed AI inference systems.



▲ Figure 2. Illustration of computing model of MapReduce-based distributed computing framework.

3.2 Communication-Efficient Data Shuffling

As data shuffling over radio channels is the major bottleneck of MapReduce-based distributed computing systems, it is necessary to propose a communication-efficient data shuffling strategy for a given dataset placement. We take a wireless communication system consisting of multiple mobile devices and an AP as an example (**Fig. 3**). The basic idea for achieving low-latency data shuffling is to explore the opportunity of concurrent transmission, detailed as follows.

- Uplink multiple access: After computing the intermediate values with map functions, each mobile device transmits its precoded intermediate values to the AP over the multiple access channel.

- Downlink broadcasting: The AP broadcasts the received signal of uplink transmission to each device, which decodes its desired intermediate values.

The output of each computation task depends on both the locally computed intermediate values at each device based on its own dataset and intermediate values computed by other devices. By treating each intermediate value as an independent message, the data shuffling procedure is indeed a message delivery problem. The AP first receives a mixed signal from all mobile devices in the uplink, and then simply broadcasts the mixed signal to all mobile devices in the downlink. By studying the input-output relationship from all mobile devices to all mobile devices after the uplink and downlink transmissions, the proposed data shuffling strategy can be equivalently modeled as a data delivery problem over the K -user interference channel with side information available at both the transmitters and the receivers. Note that the AP behaves like a two-way relay^[17] and simply transmits an amplified version of the received signal. The side information refers to the available intermediate values at each device. As a result, the goal becomes the transceiver design for maximizing the communication efficiency of data shuffling. It has been demonstrated that the linear coding schemes are effective for the transceiver design because of their optimality in terms of the degree of freedoms (DoFs) for interference alignment as well as low implementation complexity. Note that DoF is a first-order characterization of channel capacity, which is thus chosen as the performance metric for data shuffling. With interference alignment, the solutions meeting interference alignment conditions yield transceivers that are able to simultaneously preserve the desired signal and cancel the co-channel interference.

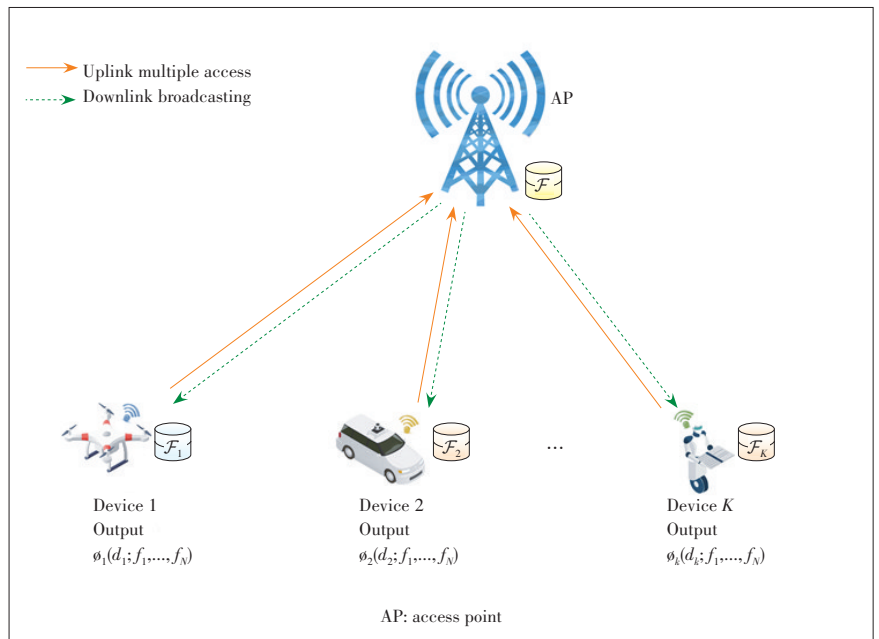
The problem of finding solutions to the interference alignment conditions with a maximum achievable DoF can be tackled by de-

veloping an efficient algorithm based on a low-rank optimization approach^[3]. This is achieved by defining the product of the aggregated precoding matrix and the aggregated decoding matrix as a new matrix variable, based on the following two key observations:

- The interference alignment conditions can be represented as affine constraints in terms of the newly defined matrix variable and the global channel state information.
- The rank of the matrix is inversely proportional to the achievable DoF.

Therefore, the maximum achievable DoF can be obtained via minimizing the matrix rank, subjecting to the affine constraints. For the nonconvex low-rank optimization problem, the traditional nuclear norm minimization approach yields unsatisfactory performance, which motivates us to propose a novel computationally efficient difference of convex functions (DC)^[18] algorithm to achieve considerable performance enhancement.

With limited radio resources, the scalability of the data shuffling strategy is also critical to the wireless distributed computing framework. We prefer a data shuffling strategy if the communication cost (which can be measured by achievable DoF) does not increase too much with more involved mobile devices. We present simulation results to demonstrate the effectiveness of the proposed algorithm for data shuffling. In simulations, we consider a single-antenna system, where the dataset is evenly split into five files and each device stores up to two files locally. With the uniform dataset placement strategy, each file is stored by $2K/5$ mobile devices. The achievable DoFs averaged over 100 channel realizations are illustrated in **Fig. 4**. Interestingly, the achievable DoF of the proposed DC



▲ **Figure 3.** Illustration of communication model for data shuffling of on-device distributed AI inference systems.

approach remains almost unchanged as the number of devices increases, while the nuclear norm relaxation approach suffers from a severe DoF deterioration. This demonstrates the scalability of the proposed DC approach. The main intuition is that the collaboration opportunities are increased as each file can be stored at more devices, although more intermediate values are requested with more involved devices.

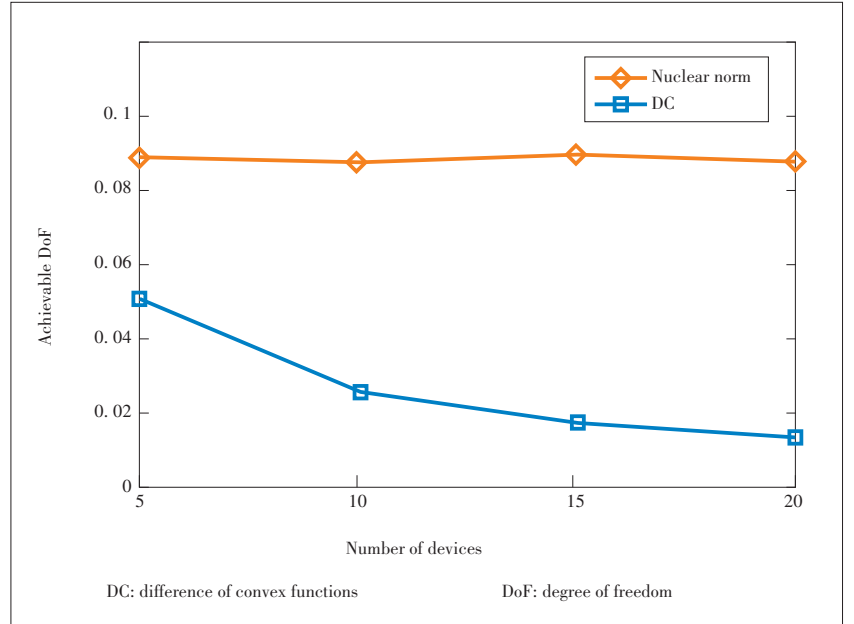
4 Edge Processing System for In-Edge Cooperative AI Inference

In this section, we present a cooperative wireless transmission approach for energy-efficient edge processing of computational intensive inference tasks at edge servers.

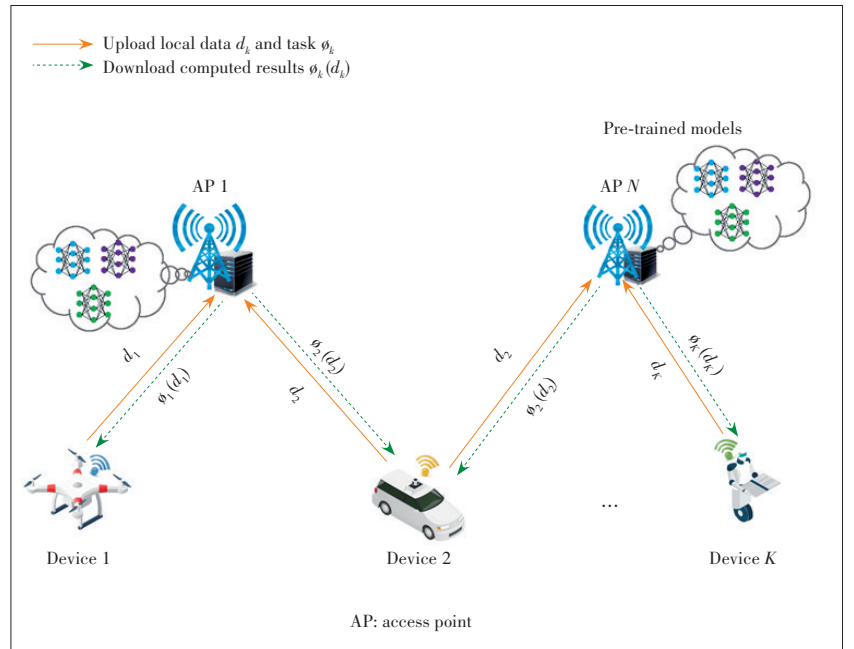
Due to the strong capability of capturing data representation, machine learning techniques, in particular deep learning^[19], have been widely used for achieving greatly improved performance in distilling intelligence from images, videos, texts, etc. However, the deep learning model are usually large and complex, and processing deep neural networks (DNNs) is a computation-intensive task. For resource-constrained mobile devices equipped with limited storage, computation power, and energy budget, such as drones and robots, a promising solution of performing computation-intensive inference tasks is to enable edge processing at the APs of mobile networks. With more powerful computing power than the resource-constrained mobile devices, APs have the potential of efficiently performing the inference tasks and transmitting the inference results to mobile users^[20]. The design target is to enable cooperative transmission among multiple APs to provide higher QoS for reliably delivering the inference results, while minimizing the total power consumption consisting of the computation power of inference tasks and the transmission power at APs. The computation power of each task at an AP can be determined via estimating the energy consumption of processing DNNs^[21] and computation time.

We consider a typical edge processing system consisting of N APs served as edge processing nodes and K mobile users, as demonstrated in Fig. 5. Each mobile user has an inference task to be accomplished. The inference results can be obtained by uploading the input of each mobile user to the APs, processing a subset of inference tasks at each AP, and cooperatively transmitting the inference results to the

corresponding mobile user. The pre-trained models can be downloaded from the cloud center and deployed at each AP in advance to facilitate edge inference. For example, the inference task can be the GauGan AI system by Nvidia, where the inputs are rough doodles and the outputs are photorealistic landscapes. Although the cooperative edge inference is able to deliver the reliable inference results to mobile users, the energy efficiency becomes critical as a huge amount of computation is required for processing DNNs at multiple APs.



▲ Figure 4. Achievable DoF of algorithms versus the number of devices for the data shuffling of on-device distributed AI inference systems.



▲ Figure 5. Illustration of in-edge cooperative AI inference systems.

There exists a tradeoff between communication and computation in an edge processing system by enabling cooperation among APs. In particular, if each AP performs more inference tasks, the inference results can be delivered with better QoS via cooperative downlink transmission. However, more computation power is consumed at the APs for processing the DNNs. To balance the tradeoff, we thus propose to minimize the total power consumption, consisting of computation power and communication power, under the target QoS constraints. This problem involves the joint design of the task allocation strategy across APs and the downlink beamforming vectors. Interestingly, if an inference task is not performed at one AP, the corresponding beamforming vector could be set as zero. This intrinsic connection between the task allocation strategy and the group sparsity structure of the downlink beamforming vectors allows us to reformulate the total power minimization problem under target QoS constraints as a group-sparse beamforming problem with QoS constraints. The group sparse structure can be induced with a well-recognized mixed $\ell_{1,2}$ norm, which results in a convex second-order cone program (SOCP) problem that can be efficiently solved. We leave simulation results in **Fig. 6** in Section 5.3 to evaluate the total power consumption of the proposed approach as well as the intelligent reflecting surface empowered in-edge cooperative AI inference.

5 IRS for Enhancing Communication Efficiency of Edge Inference Systems

In this section, we introduce the novel IRS^[7] technique for improving the signal propagation conditions of wireless environment, which is able to further enhance the communication efficiency for on-device distributed AI inference and in-edge cooperative AI inference.

5.1 Principles of IRS

An IRS is a low-cost two-dimensional surface of electromagnetic (EM) materials and composed of structured passive scattering elements^[22]. The structural parameters determine how the incident radio waves are transformed at the IRS. The specially designed scattering elements introduce a shift of the resonance frequency and a change of boundary conditions, resulting in phase changes of both the reflected and diffracted radio waves. The scattering elements on IRS are reconfigurable by imposing external stimuli to alter their physical parameters, which can be exploited to fully control the phase shift of each element at the IRS.

Although the communication efficiency of data shuffling for on-device distributed AI inference and wireless cooperative transmission for in-edge AI inference can be greatly im-

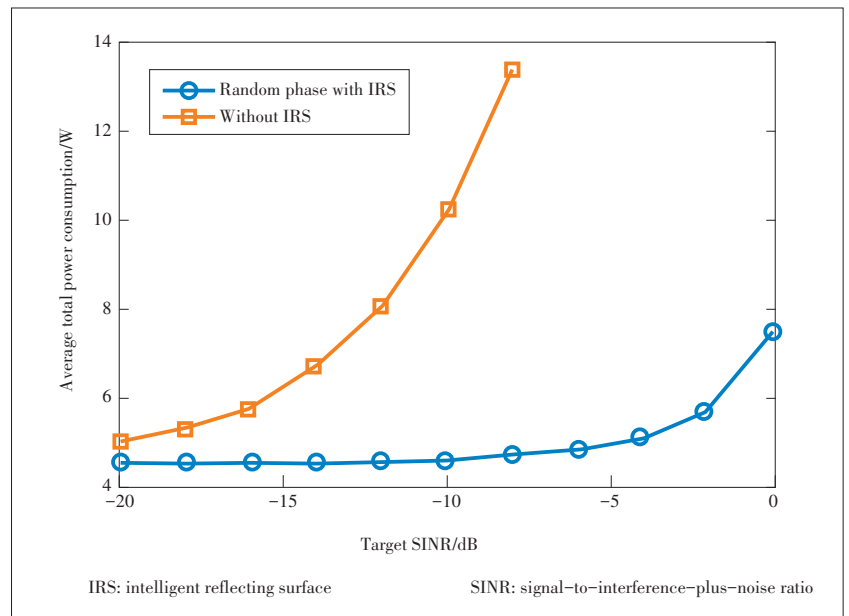
proved by our novel communication strategy and algorithm design, it is still fundamentally limited by the wireless propagation environments. To this end, we resort to IRS that is capable of building a smart radio environment to address this issue.

5.2 IRS-Empowered Data Shuffling for On-Device Distributed AI Inference

IRS with real-time reconfigurability is capable of controlling the signal propagation environments, thereby improving the spectral efficiency and reducing the energy consumption of wireless networks. The controllable phase shifts to the incident signals make IRS possible for further improving the achievable DoFs for data shuffling in Section 3. In particular, by actively reconfiguring the radio propagation environment, the feasibility of interference alignment conditions can be achieved. As a result, IRS is a promising technology for providing low-latency on-device distributed AI inference services for a wide range of applications. Note that we can still use the communication scheme and interference alignment technique provided in Section 3.2, and model the data shuffling problem as a side information aided message delivery problem in interference channel, while the channel coefficients could be adjusted by the phase shifts of IRS. The additional dimension provided by the phase shifts at RIS is able to further enhance the desired signals while nulling interference.

5.3 IRS-Empowered In-Edge Cooperative AI Inference

To further reduce the power consumption of in-edge cooperative inference in Section 4, it is promising to combat the unfavorable channel conditions by actively adjusting the phase shifts of IRS, rather than only adapting to the wireless propagation environments. By dynamically configuring the phase



▲ **Figure 6.** Average total power consumption comparison between edge processing systems with and without IRS.

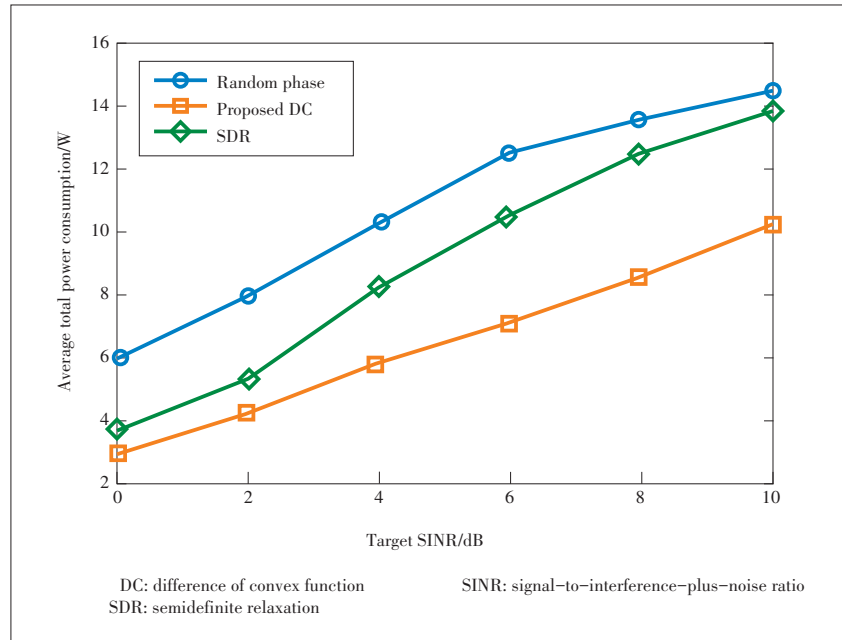
shifts of the IRS, a desired channel response can be achieved at the mobile devices, which in turn improves the signal power. Therefore, under the same QoS requirements, IRS can be utilized to further reduce the total power consumption of the edge processing system.

However, it calls for the joint design of the task allocation strategy, the downlink beamformers and the phase shifts of IRS. Exploiting the group structure of beamformers yields a highly nonconvex group-sparse optimization problem with coupled optimization variables in the QoS constraints, i.e., the downlink beamforming vectors at the APs and the phase shifts at the IRS. An alternating optimization framework can be adopted to decouple the highly nonconvex QoS constraints, for which updating the downlink beamforming vector is exactly the same as that in Section 4.1. The update for phase shifts at the IRS can be transformed to a homogeneous quadratically constrained quadratic program (QCQP) problem with nonconvex unit modulus constraints. To tackle the nonconvex constraints, the problem is further reformulated as a rank-one constrained optimization problem by leveraging the matrix lifting technique. The resulting optimization problem can then be solved with a DC algorithm by minimizing the difference between trace norm and spectral norm of the matrix variable.

For illustration purpose, we consider an edge processing system with three 5-antenna APs and ten single-antenna mobile users that are uniformly located in the square area of $[0, 200]_{\text{m}} \times [0, 200]_{\text{m}}$. An IRS equipped with 25 reflecting elements is deployed at the center of the square area. In simulations, the power consumption of performing an inference task at the AP is 0.45 W and the maximum transmit power of AP is 1 W. Fig. 6 shows the average total power consumption versus the target signal-to-interference-plus-noise ratio (SINR) for edge processing systems without and with an IRS, where a simple random phase shift strategy is adopted. Simulation demonstrate that the power consumption can be significantly reduced by leveraging the advantages of IRS. We then compare the proposed DC approach with the semidefinite relaxation (SDR) approach as well as the random phase shifts strategy in Fig. 7. It demonstrates that the proposed approach is able to achieve the least total power consumption among others.

6 Conclusions

In this article, we presented the communication-efficient designs for edge inference. We identified two representative system architectures for edge inference, i.e., on-device distribut-



▲ Figure 7. Average total power consumption comparison for different algorithms in edge processing systems.

ed AI inference and in-edge cooperative AI inference. For on-device distributed AI inference, we proposed a low-latency data shuffling strategy, followed by developing a low-rank optimization method to maximize the achievable DoFs. We also proposed a group-sparse beamforming approach to minimize the total power consumption of in-edge cooperative AI inference. In addition, we explored the potential of deploying IRS to further enhance the communication efficiency by combating the detrimental effects of wireless fading channels. Our proposals are capable of achieving low-latency and high energy efficiency for edge AI inference.

References

- [1] MAO Y Y, YOU C S, ZHANG J, et al. A survey on mobile edge computing: the communication perspective [J]. IEEE communications surveys & tutorials, 2017, 19 (Fourthquarter): 2322 – 2358. DOI: 10.1109/COMST.2017.2745201
- [2] LETAIEF K B, CHEN W, SHI Y M, et al. The roadmap to 6G: AI empowered wireless networks [J]. IEEE communications magazine, 2019, 57(8): 84 – 90. DOI: 10.1109/MCOM.2019.1900271
- [3] YANG K, SHI Y M, DING Z. Data shuffling in wireless distributed computing via low-rank optimization [J]. IEEE transactions on signal processing, 2019, 67 (12): 3087 – 3099. DOI:10.1109/tsp.2019.2912139
- [4] LI S Z, MADDAH-ALI M A, YU Q, et al. A fundamental tradeoff between computation and communication in distributed computing [J]. IEEE transactions on information theory, 2018, 64(1): 109 – 128. DOI: 10.1109/TIT.2017.2756959
- [5] LI S Z, MADDAH-ALI M A, AVESTIMEHR A S. Coding for distributed fog computing [J]. IEEE communications magazine, 2017, 55(4): 34 – 40. DOI: 10.1109/mcom.2017.1600894
- [6] GESBERT D, HANLY S, HUANG H, et al. Multi-cell MIMO cooperative net-

- works: a new look at interference [J]. *IEEE journal on selected areas in communications*, 2010, 28(9): 1380 – 1408. DOI: 10.1109/jsac.2010.101202
- [7] YUAN X J, ZHANG Y-J, SHI Y M, et al. Reconfigurable-intelligent-surface empowered 6G wireless communications: challenges and opportunities [EB/OL]. (2020-01-02). <https://arxiv.org/abs/2001.00364>
- [8] ZHOU Z, CHEN X, LI E, et al. Edge intelligence: paving the last mile of artificial intelligence with edge computing [J]. *Proceedings of the IEEE*, 2019, 107(8): 1738 – 1762. DOI: 10.1109/jproc.2019.2918951
- [9] PARK J, SAMARAKOON S, BENNIS M, et al. Wireless network intelligence at the edge [J]. *Proceedings of the IEEE*, 2019, 107(11): 2204 – 2239. DOI: 10.1109/jproc.2019.2941458
- [10] YANG K., SHI Y M, YU W, et al. Energy-efficient processing and robust wireless cooperative transmission for edge inference [J]. *IEEE internet of things journal*, 2020. DOI: 10.1109/IJOT.2020.2979523
- [11] HUA S, ZHOU Y, YANG K, et al. Reconfigurable intelligent surface for green edge inference [EB/OL]. (2019-12-02). <https://arxiv.org/abs/1912.00820>
- [12] LI E, ZENG L K, ZHOU Z, et al. Edge AI: On-demand accelerating deep neural network inference via edge computing [J]. *IEEE transactions on wireless communications*, 2020, 19(1): 447 – 457. DOI: 10.1109/twc.2019.2946140
- [13] ESHRATIFAR A E, ABRISHAMI M S, PEDRAM M. JointDNN: an efficient training and inference engine for intelligent mobile cloud computing services [J]. *IEEE transactions on mobile computing*, 2019: 1. DOI: 10.1109/tmc.2019.2947893
- [14] ZHANG J, LETAIEF K B. Mobile edge intelligence and computing for the internet of vehicles [J]. *Proceedings of the IEEE*, 2020, 108(2): 246 – 261. DOI: 10.1109/jproc.2019.2947490
- [15] ZENG Y, WU Q Q, ZHANG R. Accessing from the sky: a tutorial on UAV communications for 5g and beyond [J]. *Proceedings of the IEEE*, 2019, 107(12): 2327 – 2375. DOI: 10.1109/jproc.2019.2952892
- [16] HADDADIN S, JOHANNMEIER L, LEDEZMA F D. Tactile robots as a central embodiment of the tactile Internet [J]. *Proceedings of the IEEE*, 2019, 107(2): 471 – 487. DOI: 10.1109/JPROC.2018.2879870
- [17] LIU K Q, TAO M X. Generalized signal alignment: on the achievable DoF for multi-user MIMO two-way relay channels [J]. *IEEE transactions on information theory*, 2015, 61(6): 3365 – 3386. DOI: 10.1109/tit.2015.2420100
- [18] TAO P D, AN L T H. Convex analysis approach to DC programming: theory, algorithms and applications [J]. *Acta mathematica vietnamica*, 1997, 22(1): 289 – 355
- [19] LECUN Y, BENGIO Y, HINTON G. Deep learning [J]. *Nature*, 2015, 521(7553): 436 – 444. DOI: 10.1038/nature14539
- [20] LI K K, TAO M X, CHEN Z Y. Exploiting computation replication for mobile edge computing: a fundamental computation - communication tradeoff study [EB/OL]. (2019-03-26). <https://arxiv.org/abs/1903.10837>
- [21] YANG T-J, CHEN Y-H, SZE V. Designing energy-efficient convolutional neural networks using energy-aware pruning [C]/*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, USA, 2017: 5687 – 5695. DOI: 10.1109/cvpr.2017.643
- [22] GONG S M, LU X, HOANG D T, et al. Towards smart radio environment for wireless communications via intelligent reflecting surfaces: a comprehensive survey [EB/OL]. (2019-12-17). <https://arxiv.org/abs/1912.07794>

Biographies

YANG Kai received the B.S. degree in electronic engineering from Dalian University of Technology, China in 2015. He is currently working toward the Ph.D. degree with the School of Information Science and Technology, ShanghaiTech University, China, also with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, China, and also with the University of Chinese Academy of Sciences, Beijing, China. His research interests include data processing and optimization for mobile edge artificial intelligence.

ZHOU Yong (zhouyong@shanghaitech.edu.cn) received the B.S. and M.Eng. degrees from Shandong University, China in 2008 and 2011, respectively, and the Ph.D. degree from The University of Waterloo, Canada in 2015. From November 2015 to January 2018, he worked as a postdoctoral research fellow in the Department of Electrical and Computer Engineering, The University of British Columbia, Canada. Since March 2018, he has been with the School of Information Science and Technology, ShanghaiTech University, where he is currently an assistant professor. His research interests include 5G and beyond, IoT, and edge AI.

YANG Zhanpeng will receive his B.S. degree from Xidian University, China on July 2020. He will join the School of Information Science and Technology, ShanghaiTech University, in Fall 2020. He mainly focuses on developed reconfigurable intelligence surface based 6G wireless technologies for mobile edge AI systems.

SHI Yuanming received the B.S. degree in electronic engineering from Tsinghua University, China in 2011. He received the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology (HKUST), China in 2015. Since September 2015, he has been with the School of Information Science and Technology, ShanghaiTech University, China, where he is currently a tenured associate professor. He visited University of California, Berkeley, USA from October 2016 to February 2017. Dr. SHI is a recipient of the 2016 IEEE Marconi Prize Paper Award in Wireless Communications and the 2016 Young Author Best Paper Award by the IEEE Signal Processing Society. He is an editor of *IEEE Transactions on Wireless Communications*. His research areas include optimization, statistics, machine learning, signal processing, and their applications to 6G, IoT, AI and FinTech.

Knowledge Distillation for Mobile Edge Computation Offloading



CHEN Haowei, ZENG Liekang, YU Shuai, CHEN Xu

(School of Data and Computer Science, Sun Yat-sen University, Guangzhou, Guangdong 510006, China)

Abstract: Edge computation offloading allows mobile end devices to execute compute-intensive tasks on edge servers. End devices can decide whether the tasks are offloaded to edge servers, cloud servers or executed locally according to current network condition and devices' profiles in an online manner. In this paper, we propose an edge computation offloading framework based on deep imitation learning (DIL) and knowledge distillation (KD), which assists end devices to quickly make fine-grained decisions to optimize the delay of computation tasks online. We formalize a computation offloading problem into a multi-label classification problem. Training samples for our DIL model are generated in an offline manner. After the model is trained, we leverage KD to obtain a lightweight DIL model, by which we further reduce the model's inference delay. Numerical experiment shows that the offloading decisions made by our model not only outperform those made by other related policies in latency metric, but also have the shortest inference delay among all policies.

Keywords: mobile edge computation offloading; deep imitation learning; knowledge distillation

DOI: 10.12142/ZTECOM.202002006

<http://kns.cnki.net/kcms/detail/34.1294.TN.20200529.1853.002.html>, published online May 29, 2020

Manuscript received: 2019-12-01

Citation (IEEE Format): H. W. Chen, L. K. Zeng, S. Yu, et al., "Knowledge distillation for mobile edge computation offloading," *ZTE Communications*, vol. 18, no. 2, pp. 40 - 48, Jun. 2020. doi: 10.12142/ZTECOM.202002006.

1 Introduction

Nowadays more and more end devices are running compute-intensive tasks, such as landmarks recognition apps in smartphones^[1], vehicles detection apps used for traffic monitoring in cameras^[2], and augmented reality apps in Google Glass. The advantages of executing compute-intensive tasks on end devices are twofold. On the one hand, most data, such as images, audios and videos, are generated at end devices. Compared with sending these data to the

cloud server, processing data locally on end devices can avoid time-consuming data transmission and reduce heavy bandwidth consumption. On the other hand, some tasks are sensitive to latency and the execution result can be out of date if being late. In some cases (e.g., face recognition applications), high latency can result in poor user experience. If computation tasks are offloaded to the cloud, the unreliable and delay-significant wide-area connection can be problematic. Hence, executing compute-intensive tasks on end devices is a potential solution to lower end-to-end latency.

However, compared with cloud servers, the computing resources of end devices are very limited. Even a smartphone's computing capability is far weaker than a cloud server, not to mention the Google Glass and cameras. It turns out that exe-

This work was supported in part by the National Science Foundation of China under Grant No. 61972432 and the Program for Guangdong Introducing Innovative and Entrepreneurial Teams under Grant No. 2017ZT07X355.

cuting compute-intensive tasks on end devices may result in high computation latency. In addition, end devices often have energy consumption restrictions; for example, most smartphone users do not want a single app to consume too much power. Thus, it is unwise to execute tasks on end devices indiscriminately.

Recently, edge computing has emerged as a new paradigm different from local execution and cloud computing, and has attracted more and more attention. The European Telecommunications Standards Institute provided a concept of multi-access edge computing (MEC)^[3]. In the MEC architecture, distributed edge servers are located at the network edge to provide computing capabilities and IT services with high bandwidth and real-time processing. Edge servers become the third offloading location of compute-intensive tasks in addition to end devices and cloud. However, due to edge servers' restricted computing capability, they cannot completely take place of cloud servers. Many factors, including available computation and communication resources, should be taken into consideration when making offloading decisions. To tackle this challenge, in this paper, we design a computation offloading framework which jointly considers computation and communication and dynamically makes optimal offloading decisions to minimize the end-to-end execution latency.

Recent advances in deciding offloading strategies focus on learning-based methods. YU et al.^[4] propose to "imitate" the optimal decisions of traditional methods by deep imitation learning (DIL), where DIL^[5] uses instances generated from human's behaviors to learn the decision strategies in specific environments. DIL enjoys two advantages compared with traditional methods^[6] and deep reinforcement learning methods^[7]. First, inference delay of DIL is much shorter than that of traditional methods especially when the amount of input data is large (as shown in our experiment in Section 5). Second, DIL has higher accuracy in imitating optimal offloading decisions compared with approaches based on deep-reinforcement-learning (DRL).

However, the DIL model is built upon deep neural network (DNN), which is compute-intensive and typically requires high inference latency. On this issue, model compression^[8] is proposed, and knowledge distillation (KD) is one of the solutions^[9]. The idea behind KD is similar to transfer learning. KD not only effectively reduces the size of the neural network and improves the inference efficiency, but also improves the accuracy in the case where training samples are insufficient and unbalanced, which may appear in DIL training phase. Hence, we believe that applying KD can benefit the deployment of DIL model.

In this article, we leverage the emerging edge computing paradigm and propose a framework based on DIL and KD, which jointly considers available computation and communication resources and makes fine-grained offloading decisions for end devices. The objective of the proposed framework is to minimize the end-to-end latency of compute-intensive tasks on end devices.

We use offloading decision instances to train our DIL model offline and compress the model to a lightweight one by KD online for quickly making near-optimal offloading decisions.

The rest of this article is organized as follows. We briefly review related works in Section 2. We explain how to build a DIL model and use it in computation offloading decisions in Section 3. Then we describe how to use KD to further optimize the performance of the DIL model in Section 4. Numerical experiment results are shown in Section 5. At last we discuss some future directions and conclude in Section 6.

2 Related Work

2.1 Computation Offloading Strategies

To achieve lower latency or energy, mobile end devices usually choose to offload tasks to the cloud or edge servers. However, due to the complexity of network conditions in practice, for different devices at different times, the optimal computation offloading decisions are different. It is difficult to find this optimal decision in real time. Traditional computation offloading strategies are mostly based on mathematical modeling. Researchers in Ref. [6] study the computation offloading problem in multi-user MEC environment. They firstly prove that finding the best offloading strategies in multi-channel and multi-user condition is NP-hard. Then they model this problem as an offloading game and design a distributed approach to reach the Nash equilibrium. The authors in Ref. [10] study offloading video objects detection tasks to cloud server. In Ref. [10], a big YOLO model is deployed in cloud while a lite YOLO model is deployed at end devices. Many factors such as bit rate, resolution and bandwidth are considered and the offloading problem is formulated into a multi-label classification problem. A near-optimal solution is found by an iteration approach and it successfully achieves higher accuracy in video objects detection. The main disadvantage of mathematical modeling methods is that they are so complicated that they may cause non-negligible inference delays and are difficult to be deployed in MEC network.

One of the typical compute-intensive tasks is DNN inference, on which many researchers study specialized computation offloading strategies. KANG et al.^[11] propose Neurosurgeon framework for DNN offloading. Neurosurgeon divides DNN into two parts. One part runs at end devices and the other runs at the cloud. This method reduces the calculation at end devices, and tries to find a balanced point between computation and transmission. Neurosurgeon evaluates the latency of each DNN layer by regression models offline, and uses these models to calculate the best divided point online tailored to end devices' performance and bandwidth.

Recently, some researchers introduce DRL to find computation offloading strategies. In this case, the latency or energy consumption serves as agents' reward. The authors in Ref. [7]

consider a condition of vehicular networks based on software defined network and jointly optimize networking, caching, and computer resource by a double-dueling Deep-Q-Network. The main drawback of DRL-based approaches in computation offloading is that the offline training and online inference takes many overheads. To tackle this challenge, we propose to utilize DIL for computation offloading, the training cost and inference latency of which are significantly lower than those of DRL.

2.2 Deep Imitation Learning and Knowledge Distillation

DIL refers to training agents to imitate human's behaviors by a number of demos. Compared with DRL, training and inference time of DIL is much shorter. The authors in Ref. [4] build an edge computation offloading framework based on DIL. However, since DIL is based on DNN, if the size of DNN grows too large, it may still result in high inference delay. On this issue, we use Knowledge Distillation to compress the DIL model.

KD is firstly proposed in Ref. [9], where the authors show that small DNNs can achieve approximately high accuracy as large DNNs with relatively less inference latency. This motivates us to compress the models to reduce inference delay with tiny accuracy loss. In KD, a large DNN is trained on a large training set and a lite DNN is trained on a small training set whose labels are the output of large DNN after "softened".

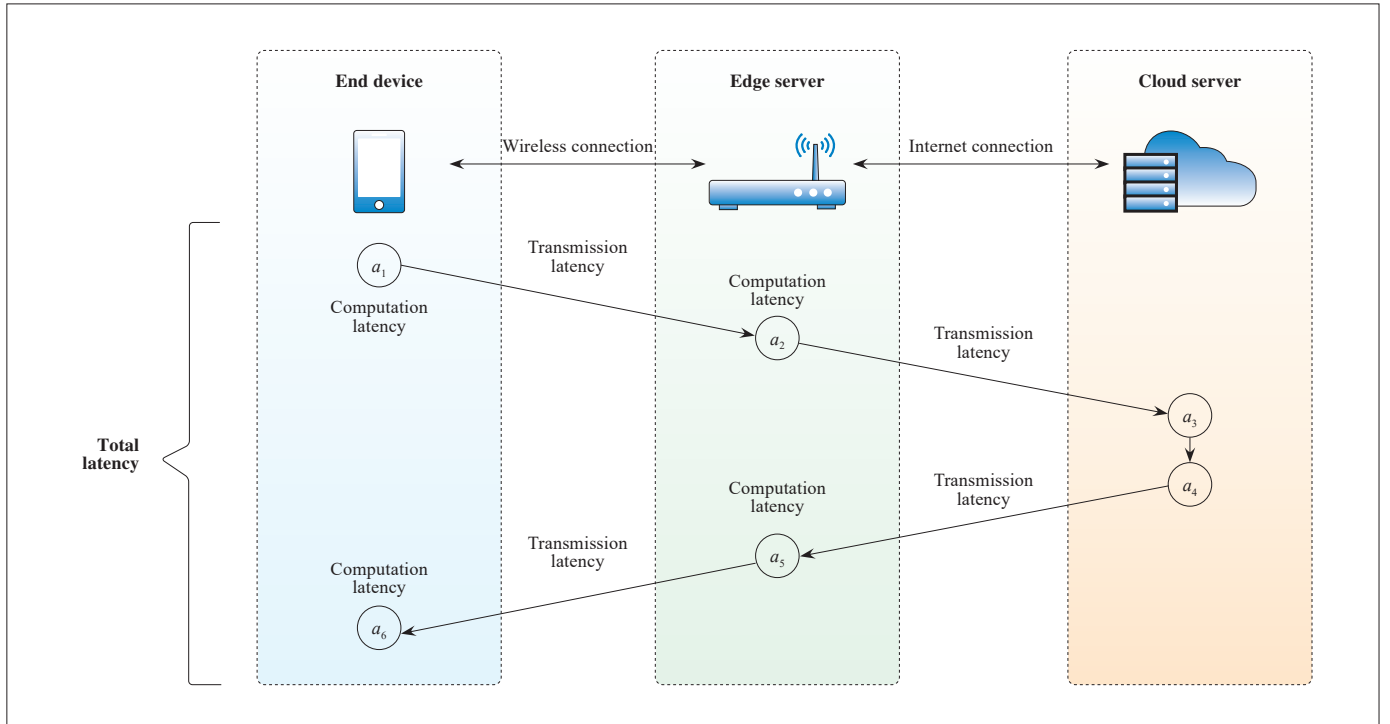
In our work, we compress our DIL model through KD to further reduce the inference delay, and improve the model's performance when training samples are missing and unbalanced.

3 Edge Computation Offloading by Deep Imitation Learning

3.1 System Model

We study the problem of making fine-grained offloading decisions for a single end device user. A compute-intensive task A on end device needs to be executed. We firstly split task A into some subtasks, following Ref. [12]. Each subtask can be denoted by a tuple $a_i = (t, \varepsilon_i, d_i, d_{i+1})$. Task A can be seen as a set of all subtasks a_i . And ε_i represents the computation complexity of the t -th subtask (usually in central processing unit (CPU) cycles). All of the computation complexity forms a set $E = \{\varepsilon_i | i \in [0, |A|]\}$. The d_i denotes the size of input data of the t -th subtask (usually in bytes). When $t=0$, d_0 represents the size of input data of task A . d_{i+1} denotes the size of output data of the t -th subtask, and is also the input data size of the $(t+1)$ -th subtask. When $t=|A|$, $d_{|A|}$ represents output size data of task A . Sizes of all data flow jointly form the set $D = \{d_i | i \in [0, |A| + 1)\}$.

As shown in **Fig. 1**, during the runtime of the mobile end device, a wireless connection with an edge server is established, and the edge server maintains a connection with the cloud server through the Internet. When a computation task in the end device needs to be executed, it will be divided into some subtasks. Each subtask can choose to be executed locally on end device or sent to the edge server. When the edge server receives a requirement of execution of a subtask, it can decide whether to execute it locally on edge server or further send it to cloud server. Execution of a subtask leads to compu-



▲ Figure 1. Subtasks are offloaded to end device, edge server and cloud server respectively.

tation latency, which depends on the profile of end device and edge server and the computation complexity of subtasks E . If two adjacent subtasks are offloaded to different locations, transmission latency will also occur, which mainly depends on the bandwidth between end device, edge server and cloud server and transmission data size D . In this paper, due to the strong computing capability of the cloud server, cloud computation latency is far less than the transmission latency. Hence, when the subtask is offloaded to cloud server, the computation latency can be ignored and only the transmission latency is concerned.

3.2 Problem Formulation

When a computation task needs to be executed, end devices split it into some subtasks and evaluate computation complexity E and transmission data sizes D of all subtasks. We can leverage the method introduced in Ref. [12] to evaluate E and D . Then all subtasks, E , D and the computing capability of the end device (denoted by p_1) are sent to edge server; p_1 can be measured in CPU frequency (in Hz). The edge server measures the bandwidth between the end device and edge server (denoted by b_1) and the bandwidth between the edge server and cloud server (denoted by b_2). Factors mentioned above and the computing capability of edge server (denoted by p_2) jointly form the description of current offloading requirement $S = (E, D, p_1, p_2, b_1, b_2)$. The edge server is responsible for making offloading decisions of each subtask according to S .

For each subtask a_i , its offloading decision is represented by $I_i \in \{0, 1, 2\}$. $I_i = 0, 1, 2$ indicates that subtask a_i is executed at end device, edge server or cloud server respectively. Offloading decision of the whole task A is given by $I = \{I_i | i \in [0, |A|]\}$. Obviously, $|I| = 3^{|A|}$. The offloading problem turns into finding the offloading decision I with the shortest end-to-end latency according to given S .

Now we compute the end-to-end latency of a specific I . As we have discussed, end-to-end latency can be divided into computation latency and transmission latency. Let L_{exec}^t denote the computation latency of t -th subtask. When $I_t = 0, 1$, the subtask is executed at end device or edge server, hence $L_{exec}^t = \varepsilon_t/p_1$ or $L_{exec}^t = \varepsilon_t/p_2$, respectively. When $I_t = 2$, as mentioned in Section 3.1, computation latency at cloud server is ignored, hence $L_{exec}^t = 0$. Given S and offloading decision I , computation latency of the whole task A is:

$$L_{exec}(S, I) = \sum_{t=0}^{|A|-1} L_{exec}^t. \quad (1)$$

Let L_{trans}^t represent the data flow size between t -th and $(t-1)$ -th subtask. When data are transmitted between end device and edge server, $L_{trans}^t = d_t/b_1$, and when data is transmitted between edge server and cloud server, $L_{trans}^t = d_t/b_2$. Note that the data at the beginning of the whole task are input by the end device, and the final output destination is also the end device, thus we can assume that L_{-1} and $L_{|A|}$ are always 0. Given S and

offloading decision I , transmission latency of the whole task A is:

$$L_{trans}(S, I) = \sum_{t=0}^{|A|} L_{trans}^t. \quad (2)$$

Our goal is to find the offloading decision I^* with the shortest end-to-end latency, which is:

$$I^* = \operatorname{argmin}_I (L_{exec}(S, I) + L_{trans}(S, I)). \quad (3)$$

So far, we have formulated computation offloading problem to an end-to-end latency minimization problem. By changing the parameter of argmin to energy, we can switch optimization objective to the energy consumption. Let S represent the description of offloading requirement, I represent the offloading decision, $R_{exec}(S, I)$ be the energy consumption of computation and $R_{trans}(S, I)$ be the energy consumption of transmission. Then the best offloading decision I^* is: $I^* = \operatorname{argmin}_I (R_{exec}(S, I) + R_{trans}(S, I))$. If it is required to optimize latency and energy simultaneously, we can set the parameter of argmin to a weighted sum of latency and energy.

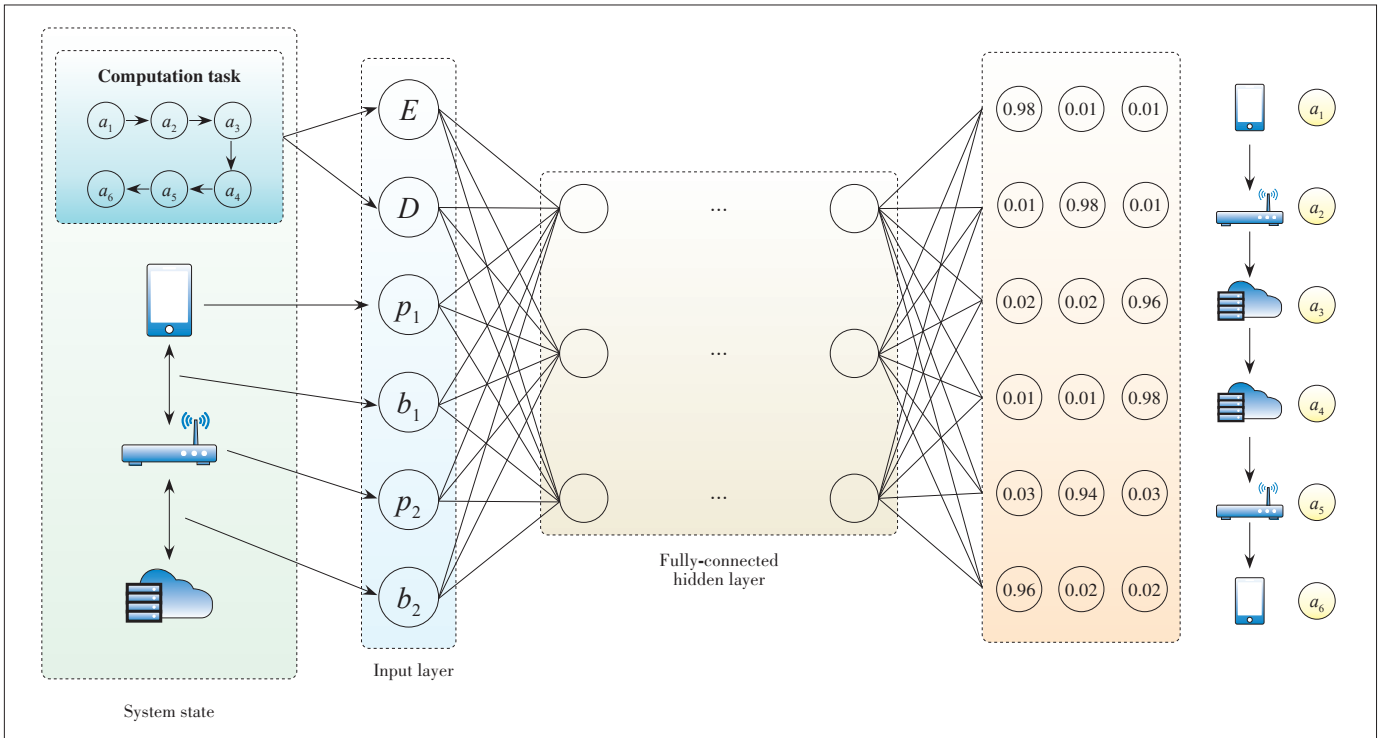
3.3 Deep Imitation Learning for Offloading

The above minimization problem can be considered as a combinatorial optimization problem. Existing technologies such as traditional offloading algorithms or reinforcement learning are difficult to solve such problems efficiently. Hence, we apply DIL to deal with it. Finding the best offloading decision I^* can be formulated to a multi-label classification problem^[13]. Decision I is a set of $|A|$ labels and the three values of I_i corresponding to three classes. The idea of DIL is to use a DNN to learn the mapping from S to the best offloading decision I^* . To this end, offloading requirement S can serve as features of input samples and I^* serves as the real labels of samples, as shown in Fig. 2.

DIL for offloading consists of three phases described as follows:

1) Generate training samples offline. DIL is supervised learning and it needs a number of features labels pair (S, I^*) . The feature S can be obtained by collecting the actual offloading task requirement, or randomly generating features based on the distribution of various parameters in the actual offloading task requirement. Since labels I^* are generated offline, some expensive non-real-time algorithms can be applied. In addition, performance of our DIL model is limited by the quality of labels, and only the labels with high accuracy can ensure highly accurate DIL model. Note that the size of decision space is $3^{|A|}$. In summary, when $|A|$ is small, we can use an exhaustive approach to obtain the optimal offloading decision by searching the whole decision space. When $|A|$ is large, we solve this problem as integer programming problem by existing efficient solvers such as CPLEX.

2) Train DIL model offline. We train a DNN model to learn



▲ Figure 2. Deep imitation learning model for edge computing offloading. Given the offloading requirement $S=(E, D, p_p, b_p, p_s, b_s)$ as the input, the deep imitation learning model can output the offloading decision $I^*=(a_p, a_s, a_s, a_p, a_s, a_s)$.

the mapping from S to I^* . In this multi-label classification problem, the output of DNN consists of predictions of $|A|$ labels. Each prediction has three possibilities corresponding to three values of I_r . Hence the output layer of DNN has $3 \times |A|$ neurons and the activation function is SoftMax. All hidden layers are full connected layers.

3) Make offloading decisions online. After our DIL model is trained, it is deployed to edge server to make offloading decisions online. Experiment shows that the efficiency of DIL model inference is higher than baseline models.

DIL is based on learning. DIL's performance is closely related to the training samples. If the training samples are diverse, DIL model can deal with more conditions, i.e., it becomes more robust. If training samples contain offloading requirement under the conditions with fluctuation of wireless channels, DIL model can learn how to make a good decision under these conditions. In practice, training samples are from actual offloading requirement. The fluctuation of the wireless channels is also covered.

After the DIL model is trained, we should consider where the DIL model is deployed for online inference. Same as the computation tasks, DIL model can be deployed on end devices, edge server or cloud server. However, if DIL model is deployed on the cloud server, the wide-area connection will become an unstable factor. To ensure model's performance, we expect that the inference result of DIL model can be obtained with a low and predictable delay. Hence, even though the com-

puting capability of cloud server is much stronger, it is not recommended to deploy DIL model on cloud server. In addition, since having all model inference workload on end device may lead to high energy consumption, we believe that edge server is a better place for DIL model deployment.

4 Knowledge Distillation for Model Compression

Since our DIL model is based on compute-intensive DNN execution, the inference latency could be high due to the limited computing capability of edge servers. We hope that the DIL model running on the edge server is lightweight and the model inference delay is minimized. Towards that, a potential solution is to put the three phases mentioned above into edge server to train a DIL model based on small DNN locally on edge server. However, it raises two problems. First, limited by the number of parameters, the learning capability of a small DNN is insufficient. Compared with large DNN, it may cause loss of accuracy and make performance worse. Second, in the phase of generated demo offline, training samples are obtained by collecting the actual offloading task requirement or randomly generated based on distribution of various parameters in the actual offloading task. However, the service area of an edge server is highly limited. Compared with the samples collected by cloud server, samples collected by edge server may be not enough and unbalanced. This further incurs the accuracy and

performance of small DNN. To this end, directly training a lightweight DIL model on edge server is not practical^[15].

The authors in Ref. [9] proposed KD, which can be used for DNN compression. This technology helps us transfer the knowledge from a large DNN to a small DNN. When the training samples are inadequate and unbalanced, accuracy of the DNN trained by KD is higher than that of the DNN directly trained on samples. Large DNN is called the “teacher” and small DNN is called the “student”. Back to our offloading problem, we can leverage the strong computing capability of cloud server and a number of samples to train a large DNN with high accuracy to serve as the teacher, and then transfer the knowledge learned by large DNN to small DNN which is deployed to edge server by KD, achieving low inference delay and small scale with tiny loss of accuracy, as shown in Fig. 3.

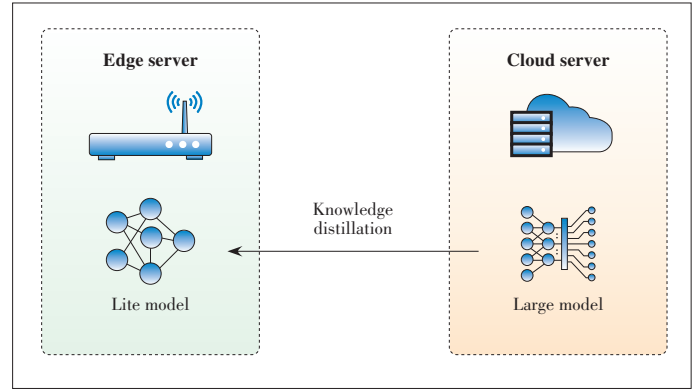
KD can be applied to any neural networks whose output layer is activated by SoftMax; in other words, the networks are used for solving classification problem. In KD, we train two networks, the teacher network and the student network. Training the teacher network is the same as training conventional network, and training the student network is also similar. The only difference is that the initial labels of student network before training are from the teacher network’s trained labels, rather than from the training dataset.

In some cases, teacher network’s trained labels may be very small and close to zero (e.g., $< 10^{-3}$), which is nearly the same as the original one-hot encoded labels and remains difficult for student network to learn the differences between labels. To alleviate this problem, we amplify the differences by further “softening” the labels. Let p_i be the probability of the i -th class predicted by the teacher, and q_i is the softened probability corresponding to p_i . We slightly change the form of the softening formula in Ref.[9] to compute q_i :

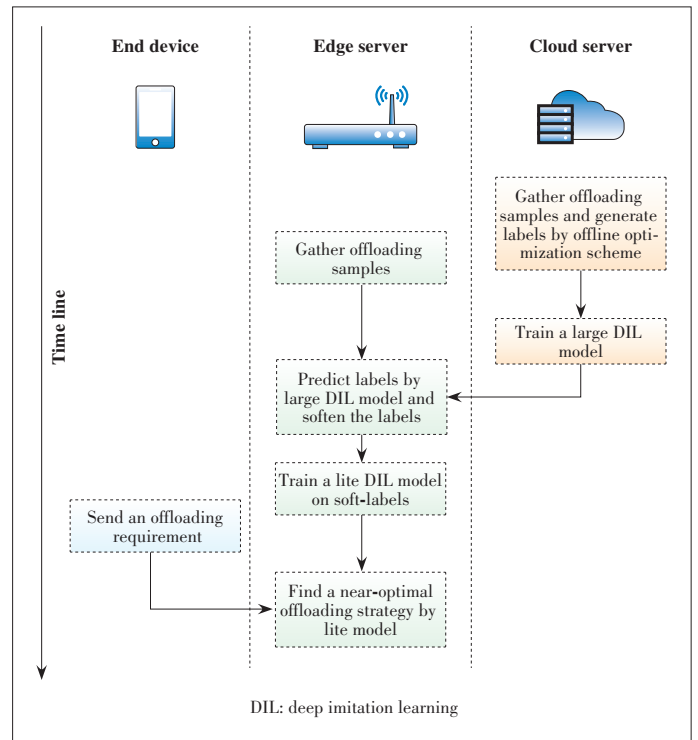
$$q_i = \frac{\exp\left(\frac{\ln(p_i)}{T}\right)}{\sum_{j=1}^C \exp\left(\frac{\ln(p_j)}{T}\right)}, \quad (4)$$

where C is the total number of classes, in our offloading problem $C=3$. T is a tunable hyper-parameter with the constraint $T \geq 1$. If $T=1$, $q_i = p_i$. The labels will be softer with higher T . For instance, if original label is $(0.999, 2 \times 10^{-4}, 3 \times 10^{-6})$, when $T=5$, the softened label will be $(0.71, 0.20, 0.09)$; when $T=10$, the softened label will be $(0.53, 0.28, 0.19)$. In the following experiment we set $T=5$. Back to the offloading problem, we use a teacher network trained at cloud server to predict labels of the training set obtained by edge server. Then soften these labels by the formula mentioned above and train student network by softened labels at edge server.

We show the complete flowchart of our DIL offloading framework with KD in Fig. 4.



▲ Figure 3. Compress model by knowledge distillation to get a lightweight model deployed to edge server.



▲ Figure 4. Complete flowchart of our edge offloading framework based on DIL and KD.

5 Evaluation

5.1 Evaluate Large DIL Model Performance

In this section, we set up a numerical experiment to evaluate the performance of DIL model described in Section 3. We consider that an MEC network consists of an end device user and an edge server connected by wireless connection, meanwhile the edge server connects to cloud server via the Internet^[16]. We assume that the compute-intensive task A on end device is divided into 6 subtasks, which is $|A|=6$. If the number of subtasks of some computation tasks is not 6, we can merge some subtasks or insert empty subtasks to make the number of subtasks 6. The computation complexity of each

subtasks ε_i (measured in CPU cycles) is in the interval of $[0, 2000] \times 10^6$, following uniform distribution. Sizes of data transmission between subtasks follow uniform distribution with $d_i \in [0, 10]$ MB, like the setting in Ref.[14]. In addition, we assume that the computing capability of end device and edge server (both measured by CPU frequency in Hz) is in the intervals of $[100, 1000]$ MHz and $[500, 5000]$ MHz respectively, both following the uniform distribution. The bandwidth between end device and edge server and the bandwidth between edge server and cloud server are uniformly distributed in $b_1 \in [0, 2]$ MB/s and $b_2 \in [0, 3]$ MB/s respectively. We randomly generate 100 000 samples offline to train DIL model and 10 000 testing samples for testing.

Our DIL model is based on a DNN with 5 hidden layers. All hidden layers are full connected layers and consist of 256 neurons. The number of parameters in the whole DNN is 1.6 million. Activation function of hidden layers is RELU and output layer is activated by SoftMax. To evaluate the performance of our DIL based offloading framework, we consider some baseline frameworks listed follow:

1) Optimal. Exhaustive method: For each sample, search the whole 3^{41} decision space, compute the latency described in Section 3.2 and choose the offloading decision with minimal latency. Note that this minimal latency is the lower bound in the decision space. Hence, this decision is bound to be optimal.

2) Greedy. For each sample, find the offloading location one by one for each subtask to minimize the computation and transmission latency of current subtask.

3) DRL. Offload framework based on deep reinforcement learning. Features of samples serve as environment and offloading decisions serve as actions. The opposite number of latency acts as reward. The deep Q network is similar to that in Ref. [7].

4) Others. Local: The whole task is executed on end device, which is for any t , $I_t = 0$; Edge: All subtasks are executed on edge server, which means $I_t = 1$; Cloud: All subtasks are offloaded to cloud server, which is $I_t = 2$; Random: Randomly choose offloading location for each subtask, that is to say, I_t are randomly chosen from $\{0, 1, 2\}$.

Fig. 5 shows the normalized latency of the DIL model and baseline frameworks with the latency of optimal decision are normalized to 1.0, and then the latency of decision made by our DIL model is 1.095, with an increase less than 10%. Experiment results show that our model outperforms other base-

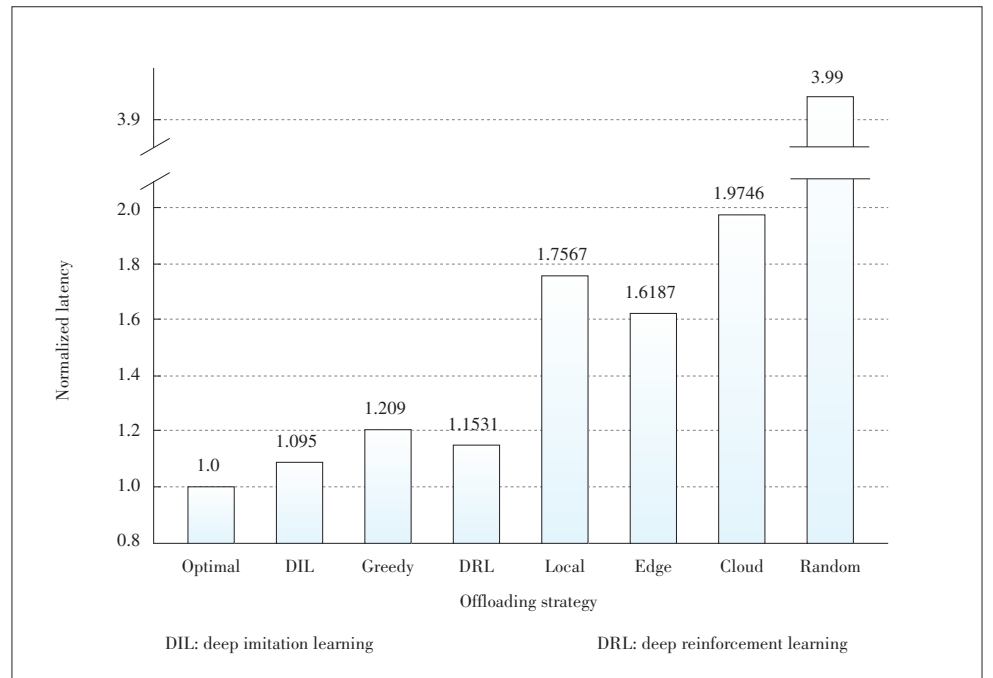
line frameworks. Note that latency of “Edge” is less than “Local” and “Cloud”, which indicates that edge server can certainly improve the compute-intensive tasks in end-to-end latency. At last, latency of “Random” is far higher than others, this is because randomly choosing offloading location will cause high transmission latency, which is expectable.

5.2 Evaluate Knowledge Distillation Performance

As mentioned in Section 4, we should compress our DIL model before deploying it to edge server and deal with the situation in which training samples on edge server are insufficient and unbalanced. We call our compressed model “KD-DIL” for short. In this section, we assume the CPU cycles of subtasks are uniformly distributed in $\varepsilon_i \in [500, 1500] \times 10^6$. Sizes of transmission data between subtasks are in $d_i \in [3, 8]$ MB, following uniform distribution. The distribution range of ε_i and d_i is reduced by half compared with that in Section 5.1. Distributions of other parameters remain the same. In order to simulate the case in which training samples are insufficient, we only generate 1 000 samples for training in this section, reduced by 99% compared with that in Section 5.1. Testing samples remain the same as that in Section 5.1.

Our KD-DIL model is still based on DNN consisting of full connect layers. There are only 2 hidden layers in DNN with 32 neurons in each layer. The number of parameters of the whole DNN is about 10 000, reduced by 99.375% compared with that in Section 5.1. The following baseline models are used for evaluating the performance of our KD-DIL model.

1) Baseline DIL: This DIL model is based on the DNN which is same as that in KD-DIL. The difference is that Baseline DIL



▲ Figure 5. Normalized end-to-end latency of offloading decisions made by our DIL model and baselines.

is directly trained on the training set described above without applying KD described in Section 4.

2) DRL: Deep reinforcement learning based on DQN. The difference between this and DRL model in Section 5.1 is that it is trained on training set with 1 000 samples described above instead of that with 100 000 samples described in Section 5.1.

3) Greedy: Same as Greedy in Section 5.1.

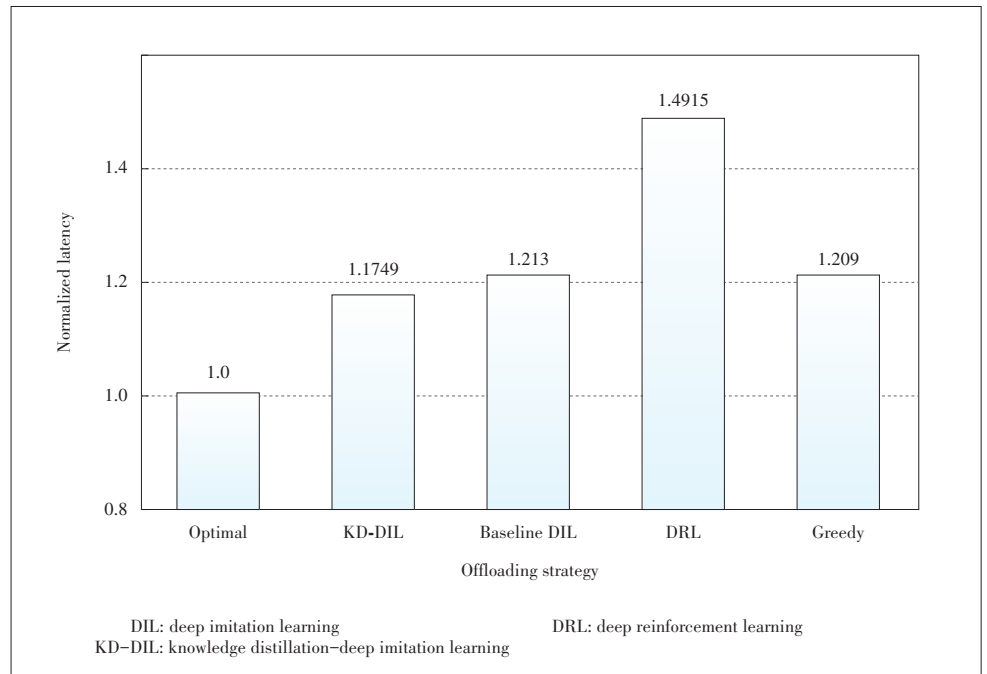
Fig. 6 shows the normalized latency of KD-DIL models and baseline models. Again, the latency of optimal decision is normalized to 1.0. It shows that our KD-DIL model still outperforms baseline models. Note that the performance of DRL has a sharp decreasing compared that in Section 5.1 because of the change of training set. It is further shown that when the number and distribution of training samples are changed, the accuracy loss of our KD-DIL model is relatively small.

At last, **Table 1** shows the normalized inference delay of all models with delay of “Greedy” being normalized to 1.00, since the greedy method is the typical method for computation offloading. We measure the delay of making 100 000 decisions of all the models, and divide this delay by 100 000 to get the average delay of each decision. As shown in Table 1, compared with the large DIL model, the inference delay of KD-DIL model decrease by 63% (0.17/0.51). Table 1 shows that the inference delay of the Greedy approach is slightly higher than DIL model. As described in Section 5.1, the Greedy approach finds deployment place for each subtask by iterations. The number of iterations equals to that of subtasks. In practice, the number of subtasks may be much higher than 6, so the inference delay of the Greedy approach may become correspondingly higher.

Lastly, the inference of the optimal approach and DRL is hundreds of times that of our DIL models. Because optimal apply exhaustive method, high inference delay is expectable. While making decisions by DRL, we treat each strategy as an action and end-to-end latency as reward. We calculate each action’s reward to find the highest reward, which needs many times of DNN inference. Hence, the delay of DRL inference is much higher than DIL.

6 Future Work and Conclusions

Flowcharts of subtasks can be represented by directed acy-



▲ **Figure 6.** Normalized KD-DIL model and baselines when using a small training set.

▼ **Table 1.** Inference delay of all models

Model Name	KD-DIL	Large DIL	DRL	Greedy	Optimal
Normalized Delay	0.17	0.51	119.72	1.00	122.54

KD-DIL: knowledge distillation-deep imitation learning

DIL: deep imitation learning
DRL: deep reinforcement learning

clic graph (DAG) known as computation graph. In computation graph, nodes denote subtasks, edges denote data flow and directions of edge represent data transmission directions. DNN can also be regarded as a computation graph. In many programming frameworks dedicated to deep learning, such as TensorFlow, the concept of computation graph is applied. Offloading a computation graph in MEC network to optimize end-to-end latency is a difficult problem. The subtasks flow-chart studied in this article has a list structure. In our future work we will focus on how to modify our work to adapt to DAG.

In this article, we have studied fine-grained edge computing offloading framework. In the situation in which an end device wirelessly connects to an edge server, compute-intensive tasks can choose to be executed at end device, edge server or cloud server. We first review existing edge offloading framework including mathematic model method (game theory) and reinforcement learning. Then we provide model of computing task and describe the execution process of a task. Offloading problem is formulated into a multi-label classification problem and is solved by a deep imitation learning model. Next, in order to deal with the insufficient and unbalanced training sample, we apply knowledge distillation to get a lightweight model with tiny accuracy loss, making it easier to be deployed to edge serv-

er. Numerical experiment shows that the offloading decisions made by our model have the lowest end-to-end latency and the inference delay of our model is the shortest, and after knowledge distillation we successfully reduce the inference delay by 63% with tiny accuracy loss. At last we briefly discuss some future directions of edge computation offloading.

References

- [1] YAP K H, CHEN T, LI Z, et al. A comparative study of mobile-based landmark recognition techniques [J]. *IEEE intelligent systems*, 2010, 25(1): 48 – 57. DOI: 10.1109/mis.2010.12
- [2] JIANG J C, ANANTHANARAYANAN G, BODIK P, et al. Chameleon: scalable adaptation of video analytics [C]//2018 Conference of the ACM Special Interest Group on Data Communication. Budapest, Hungary, 2018: 253 – 266. DOI: 10.1145/3230543.3230574
- [3] Multi-access edge computing-standards for MEC [EB/OL]. (2019-11-04)[2020-01-05]. <https://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing>
- [4] YU S, CHEN X, YANG L, et al. Intelligent edge: leveraging deep imitation learning for mobile edge computation offloading [J]. *IEEE wireless communications*, 2020, 27(1): 92 – 99. DOI:10.1109/mwc.001.1900232
- [5] ZHANG T H, MCCARTHY Z, JOW O, et al. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation [C]//2018 IEEE International Conference on Robotics and Automation (ICRA). Brisbane, Australia, 2018: 1 – 8. DOI:10.1109/icra.2018.8461249
- [6] CHEN X, JIAO L, LI W Z, et al. Efficient multi-user computation offloading for mobile-edge cloud computing [J]. *ACM transactions on networking*, 2016, 24(5): 2795 – 2808. DOI:10.1109/tnet.2015.2487344
- [7] HE Y, ZHAO N, YIN H X. Integrated networking, caching, and computing for connected vehicles: a deep reinforcement learning approach [J]. *IEEE transactions on vehicular technology*, 2018, 67(1): 44 – 55. DOI: 10.1109/tvt.2017.2760281
- [8] BA L J, CARUANA R. Do deep nets really need to be deep? [EB/OL]. (2014-10-11) [2020-01-05]. <https://arxiv.org/abs/1312.6184>
- [9] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network [EB/OL]. (2015-03-09)[2020-01-10]. <https://arxiv.org/abs/1503.02531>
- [10] RAN X, CHEN H, ZHU X, et al. Deep decision: A mobile deep learning framework for edge video analytics [C]//IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE, 2018: 1421 – 1429
- [11] KANG Y P, HAUSWALD J, GAO C, et al. Neurosurgeon: collaborative intelligence between the cloud and mobile edge [J]. *ACM SIGARCH computer architecture news*, 2017, 45(1): 615 – 629. DOI:10.1145/3093337.3037698
- [12] CUERVO E, BALASUBRAMANIAN A, D-KCHO, et al. MAUI: Making smartphones last longer with code offload [C]//Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services. San Francisco, USA, 2010: 49 – 62
- [13] TSOUMAKAS G, KATAKIS I. Multi-label classification [J]. *International journal of data warehousing and mining*, 2007, 3(3): 1 – 13. DOI: 10.4018/jdwm.2007070101
- [14] YOU C S, ZENG Y, ZHANG R, et al. Asynchronous mobile-edge computation offloading: energy-efficient resource management [J]. *IEEE transactions on wireless communications*, 2018, 17(11): 7590 – 7605. DOI: 10.1109/twc.2018.2868710
- [15] ZHOU Z, CHEN X, LI E, et al. Edge intelligence: paving the last mile of artificial intelligence with edge computing [EB/OL]. (2019-05-24)[2020-01-05]. DOI:10.1109/JPROC.2019.2918951
- [16] CHEN X, PU L, GAO L, et al. Exploiting massive D2D collaboration for energy-efficient mobile edge computing [J]. *IEEE wireless communications*, 2017, 24(4): 64 – 71. DOI:10.1109/MWC.2017.1600321

Biographies

CHEN Haowei received the B.S. degree in computer science from the School of Data and Computer Science, Sun Yat-sen University (SYSU), China in 2020. He is working towards the master's degree in the School of Data and Computer Science, SYSU. His research interests include mobile deep computing, edge intelligence and deep learning.

ZENG Liekang received the B.S. degree in computer science from the School of Data and Computer Science, Sun Yat-sen University, China in 2018. He is currently pursuing the master's degree with the School of Data and Computer Science, Sun Yat-sen University. His research interests include mobile edge computing, deep learning, and distributed computing.

YU Shuai received the Ph.D. degree from Pierre and Marie Curie University (now Sorbonne Université), France, in 2018, the M.S. degree from Beijing University of Post and Telecommunications, China, in 2014, and the B.S. degree from Nanjing University of Post and Telecommunications, China, in 2009. He is now a post-doctoral Research Fellow at the School of Data and Computer Science, Sun Yat-sen University. His research interests include wireless communications, mobile computing and machine learning.

CHEN Xu (chenxu35@mail.sysu.edu.cn) is a full professor in Sun Yat-sen University, China, and the vice director of National and Local Joint Engineering Laboratory of Digital Home Interactive Applications. He received the Ph.D. degree in information engineering from The Chinese University of Hong Kong in 2012, and worked as a post-doctoral research associate at Arizona State University, USA from 2012 to 2014, and a Humboldt Scholar Fellow at Institute of Computer Science of University of Goettingen, Germany from 2014 to 2016. He is currently an area editor of *IEEE Open Journal of the Communications Society*, an associate editor of the *IEEE Transactions Wireless Communications*, *IEEE Internet of Things Journal* and *IEEE Journal on Selected Areas in Communications (JSAC) Series on Network Softwareization and Enablers*.



Joint Placement and Resource Allocation for UAV-Assisted Mobile Edge Computing Networks with URLLC

ZHANG Pengyu¹, XIE Lifeng¹, XU Jie²

(1. School of Information Engineering, Guangdong University of Technology, Guangzhou, Guangdong 510006, China;
2. Future Network of Intelligence Institute and School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong 518172, China)

Abstract: This paper investigates an unmanned aerial vehicle (UAV) assisted mobile edge computing (MEC) network with ultra-reliable and low-latency communications (URLLC), in which a UAV acts as an aerial edge server to collect information from a set of sensors and send the processed data (e.g., command signals) to the corresponding actuators. In particular, we focus on the round-trip URLLC from the sensors to the UAV and to the actuators in the network. By considering the finite block-length codes, our objective is to minimize the maximum end-to-end packet error rate (PER) of these sensor-actuator pairs, by jointly optimizing the UAV's placement location and transmitting power allocation, as well as the users' block-length allocation, subject to the UAV's sum transmitting power constraint and the total block-length constraint. Although the maximum-PER minimization problem is non-convex and difficult to be optimally solved, we obtain a high-quality solution to this problem by using the technique of alternating optimization. Numerical results show that our proposed design achieves significant performance gains over other benchmark schemes without the joint optimization.

Keywords: UAV; MEC; URLLC; placement optimization; resource allocation

DOI: 10.12142/ZTECOM.202002007

<http://kns.cnki.net/kcms/detail/34.1294.tn.20200522.1450.004.html>, published online May 25, 2020

Manuscript received: 2019-04-27

Citation (IEEE Format): P. Y. Zhang, L. F. Xie, J. Xu, et al., "Joint placement and resource allocation for UAV-assisted mobile edge computing networks with URLLC," *ZTE Communications*, vol. 18, no. 2, pp. 49 – 56, Jun. 2020. doi: 10.12142/ZTECOM.202002007.

1 Introduction

Recent advances in artificial intelligence (AI) and Internet of Things (IoT) are envisioned to enable various new intelligent applications such as augmented reality (AR), virtual reality (VR), and unmanned aerial ve-

hicles (UAVs). Towards this end, billions of IoT devices (e.g., smart sensors and actuators) will be deployed in future wireless networks to collect information from the environments and take physical actions, and machine learning functionalities will be incorporated into wireless networks to analyze and acquire knowledge from these data for making decisions. In this case, how to provide real-time sensing, communication, and control among a large number of sensors and actuators, and how to implement real-time machine learning in the loop are challenging issues in the design of beyond-fifth-generation (B5G) or sixth-generation (6G) cellular networks towards a vision of network intelligence.

Mobile edge computing (MEC)^[1-7] and learning^[8-10] have

This work was supported in part by the Key Area R&D Program of Guangdong Province with grant No. 2018B030338001, by the National Key R&D Program of China with grant No. 2018YFB1800800, by Natural Science Foundation of China with grant Nos. 61871137 and 61629101, by the Guangdong Province Basic Research Program (Natural Science) with grant No. 2018KZDXM028, by Guangdong Zhujiang Project No. 2017ZT07X152, and by Shenzhen Key Lab Fund No. ZDSYS201707251409055.

emerged as important techniques to deal with the above issues, by pushing the cloud-like computation and storage capabilities, and the machine learning functionality at the network edge, e.g., base stations (BSs) and access points (APs). Accordingly, the edge servers at BSs/APs can help end users remotely execute the computation-intensive applications in a swift way, and quickly acquire knowledge from the locally generated data at IoT devices for making quick decisions. However, wireless communications among end devices and BSs/APs are becoming the performance bottleneck for such systems, as the wireless channels connecting them may fluctuate over time and be unstable. Prior works have investigated the joint communication and computation design for mobile edge computing^[1–4] and for training in mobile edge learning^[8–10], respectively. Besides the joint design of communication and computation, the ultra-reliable and low-latency round-trip communications from sensors to edge servers and to actuators are another crucial issue for successfully implementing the machine edge learning with critical latency requirements. For instance, consider the inference phase in mobile edge learning, where trained machine learning models are deployed at the edge server. In this case, IoT devices^[11] (e.g., sensors) first send the sensed information to the edge server; after receiving such information, the edge server implements the inference process and sends the inference results (e.g., such as command signals) back to the same or other IoT devices (e.g., actuators) for taking actions. In this scenario, the round-trip ultra-reliable and low-latency communications (URLLC) for the “sensors-edge-server-actuators” flow is important and thus is the main focus of this paper.

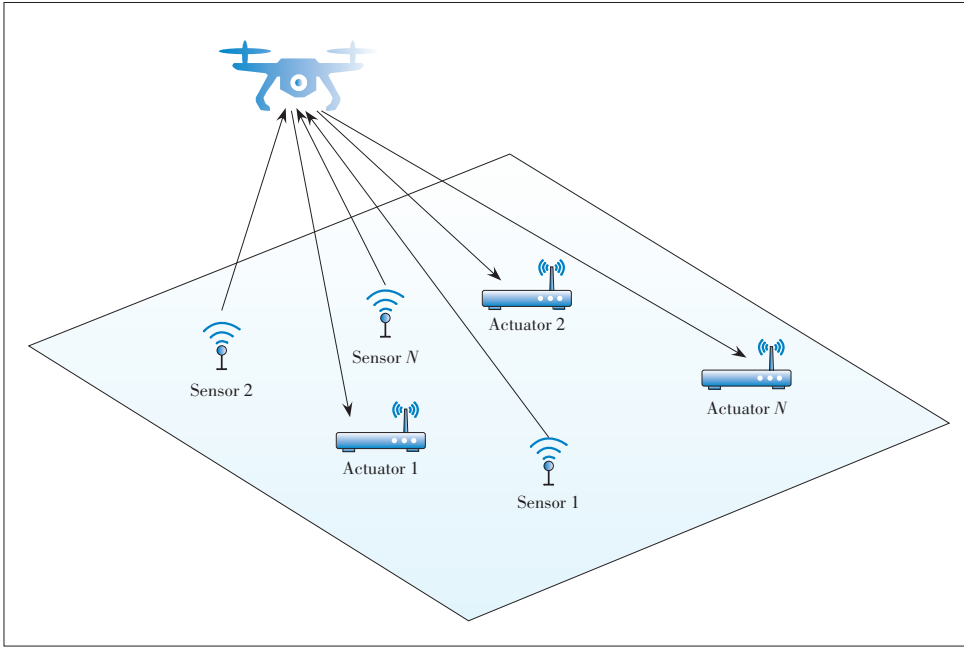
Furthermore, UAV-assisted wireless platforms^[12–14] are promising techniques towards 5G. UAV-assisted wireless platforms can provide flexible wireless services to on-ground devices by deploying wireless transceivers (such as BSs and APs) at UAVs that can fly freely over the three-dimensional (3D) space. Compared with conventional terrestrial wireless infrastructures, UAV-enabled BSs/APs are advantageous due to their deployment flexibility, strong line of sight (LoS) channels with on-ground users, and highly controllable mobility^[15–18]. By exploiting the controllable mobility, the UAVs can fly closer to intended on-ground devices and fly farther away from unintended ones to help enhance the communication performance. By integrating UAVs with MEC, UAV-enabled MEC has attracted a lot of recent research interests, in which the UAV is deployed as dedicated aerial MEC server to support the communication and computation of end users on the ground. Prior works have investigated the computation offloading design in the UAV-assisted MEC, in which wireless devices (such as smartphones) offload their own computation tasks to the UAV for enhancing the performance of task execution^{[13], [19–24]}. For instance, Refs. [22] and [23] aim to minimize the energy consumption of the UAV while ensuring the quality of service (QoS) requirements at users, by jointly optimizing

the UAV’s flight and wireless resource allocation. Refs. [19] and [24] optimize the flight trajectory and communication wireless resource allocation at the UAV, so as to maximize the UAV’s endurance time or communication rate.

Different from prior works, this paper focuses on the round-trip URLLC in mobile edge networks, in which the UAV-enabled edge server is employed to improve the round-trip communication performance from on-ground sensors to the UAV and to the actuators. This may practically correspond to a delay-sensitive inference scenario in mobile edge learning, where the machine learning models are deployed at the UAV for remote control. To our best knowledge, the problem of round-trip URLLC under this scenario has not been addressed yet. This problem, however, is challenging to be dealt with. First, for achieving URLLC, the delivered packets (e.g., the sensed information by the sensors and the command signals sent from the MEC server to the actuators) are generally with small block lengths, and as a result, the conventional Shannon capacity under the assumptions of infinite block length and zero decoding error is not applicable. Therefore, we must take into account the effect of finite block-length codes, under which new performance metrics characterizing the relations among the communication rate, packet error rate (PER), and block-length should be considered^[25–26]. Next, there generally exist a large number of sensors and actuators over IoT networks. It is thus very important to efficiently design wireless resource allocation among these sensor-actuator pairs. This, however, is technically very difficult due to the new performance metrics considered. Last but not least, the UAVs’ mobility can be exploited via trajectory control^[24] or deployment optimization^[27] for optimizing the MEC performance. How to jointly design the UAVs’ deployment optimization or trajectory control together with the wireless resource allocation is also a new problem to be tackled for URLLC.

Notice that Ref. [27] studies the UAV-enabled relaying system with URLLC, in which the UAV’s deployment location and the block-length allocation are jointly optimized, for the purpose of minimizing the end-to-end PER from the ground source node to the ground destination node. In contrast to Ref. [27] that focused on the relaying scenario with only one single source-destination pair, this paper studies a different UAV-enabled MEC scenario with multiple sensor-actuator pairs, for which both the transmitting power allocation at the UAV and the block-length allocation are considered, together with the UAV’s deployment optimization.

This paper investigates a UAV-assisted MEC network with URLLC as shown in **Fig. 1**, in which a single UAV acts as an aerial edge server to collect information sent from multiple sensors, analyze such information (via, e.g., machine learning), and then send the processed data (e.g., command signals) to their respective actuators. We focus our study on the round-trip URLLC by assuming the time and resource consumption for information processing at the UAV which is given and thus ig-



▲ Figure 1. Unmanned aerial vehicle (UAV) assisted mobile edge computing (MEC) network with one UAV acting as an MEC server to serve multiple sensors and actuators on the ground.

nored. Furthermore, we consider the quasi-stationary UAV scenario¹, in which the UAV hovers at an optimized location during the whole communication period of our interest. The main results of this paper are summarized as follows.

- Under the above setup, we aim to minimize the maximum end-to-end PER of these sensor-actuator pairs, by jointly optimizing the UAV's placement location and wireless resource allocation, subject to the UAV's sum transmitting power constraint and the total block-length constraint.
- The formulated problem is non-convex and thus is difficult to be solved optimally. To tackle this difficulty, we propose an alternating-optimization-based algorithm to obtain a high-quality solution, in which the UAV's placement location and transmitting power allocation and the users' block-length allocation are optimized in an alternating manner.
- Numerical results are provided to validate the performance of our proposed UAV-enabled round-trip URLLC among multiple sensor-actuator pairs. It is shown that our proposed design achieves much lower PER than other benchmark schemes without such joint optimization. It is also shown that when the transmitting power at the UAV becomes large, proper wireless resource allocation among different sensor-actuator pairs is crucial to enhance the maximum PER performance.

The remainder of this paper is organized as follows. Section 2 introduces the system model of the UAV-assisted MEC network with URLLC, and formulates the maximum-PER minimization problem of our interest. Section 3 proposes an efficient algorithm to obtain a high-quality solution to the formulated problem by using the alternating optimization and the Lagrange duality method. Section 4 presents numerical results to validate

the performance of our proposed approaches. Finally, Section 5 concludes this paper.

2 System Model

As shown in Fig. 1, a UAV-assisted MEC network, in which a UAV is dispatched as an aerial MEC server to serve N pairs of sensors and actuators, is considered. We use $\mathcal{N} = \{1, \dots, N\}$ to denote the set of sensors or actuators. In particular, the UAV collects information sent from the N sensors in the uplink and then transmits the processed data (or command signals) to the respective actuators in the downlink. Suppose that the sensor $i \in \mathcal{N}$ and actuator $i \in \mathcal{N}$ on the ground have fixed locations $(\hat{x}_i, \hat{y}_i, 0)$ and $(\tilde{x}_i, \tilde{y}_i, 0)$ in a 3D Cartesian coordinate system, where $\hat{\mathbf{w}}_i = (\hat{x}_i, \hat{y}_i)$ and $\tilde{\mathbf{w}}_i = (\tilde{x}_i, \tilde{y}_i)$ are defined as their horizontal coordinates, respectively. The locations of sensors and actuators are assumed to be a-priori known by the UAV to facilitate the placement location optimization and wireless resource allocations.

The UAV is assumed to stay at a fixed altitude H above the ground, and the horizontal coordinate of the UAV is denoted by $\mathbf{q} = (x, y)^2$. Therefore, the distance from the UAV to sensor i and actuator i are respectively given as:

$$\hat{d}_i = \sqrt{H^2 + \|\mathbf{q} - \hat{\mathbf{w}}_i\|^2}, \forall i \in \mathcal{N}, \quad (1)$$

$$\tilde{d}_i = \sqrt{H^2 + \|\mathbf{q} - \tilde{\mathbf{w}}_i\|^2}, \forall i \in \mathcal{N}, \quad (2)$$

where $\|\cdot\|$ denotes the Euclidean norm of a vector.

It is assumed that the wireless channels from the UAV to ground sensors or actuators are dominated by LoS links. Thus, the channel power gained from the UAV to sensor i and actuator i follows the free-space path loss model, which is expressed as:

$$\hat{h}_i(\mathbf{q}) = \rho_0 \hat{d}_i^{-2} = \frac{\rho_0}{H^2 + \|\mathbf{q} - \hat{\mathbf{w}}_i\|^2}, \forall i \in \mathcal{N}, \quad (3)$$

¹ There is another scenario, namely the fully-mobile UAV scenario, in which the UAV can fly around over the communication period and thus the trajectory control becomes crucial. Note that in our considered setup, the on-ground sensors and actuators are at fixed locations. Therefore, we only consider the quasi-stationary UAV scenario by optimizing the deployment location only.

² In this paper, we assume that the UAV hovers at an unchanged location during the whole flight period.

$$\tilde{h}_i(\mathbf{q}) = \rho_0 \tilde{d}_i^{-2} = \frac{\rho_0}{H^2 + \|\mathbf{q} - \tilde{\mathbf{w}}_i\|^2}, \forall i \in \mathcal{N}, \quad (4)$$

where ρ_0 denotes the channel power gained at the reference distance of $d_0 = 1$ m.

In the uplink, each sensor adopts constant power Q to send messages to the UAV. In this case, the correspondingly received signal-to-noise-ratio (SNR) at the UAV can be expressed as:

$$\hat{\gamma}_i(\mathbf{q}) = \frac{Q \hat{h}_i(\mathbf{q})}{\sigma^2}, \forall i \in \mathcal{N}. \quad (5)$$

In the downlink, the UAV adopts transmitting power $p_i, i \in \mathcal{N}$ to send the processed data to actuator i . Thus, the correspondingly received SNR at actuator i can be expressed as:

$$\tilde{\gamma}_i(\mathbf{q}, p_i) = \frac{p_i \tilde{h}_i(\mathbf{q})}{\sigma^2}, \forall i \in \mathcal{N}, \quad (6)$$

where σ^2 denotes the power of the additive white Gaussian noise (AWGN) at the receiver. Suppose that the UAV's downlink transmission power is P_{sum} . Then we have $\sum_{i \in \mathcal{N}} p_i \leq P_{\text{sum}}$.

We consider the time-division multiple access (TDMA) transmission protocol, in which the uplink transmission from each sensor to the UAV and the downlink transmission from the UAV to each actuator are implemented over the same frequency band and orthogonal time instants. Suppose that the size of the packet generated by sensor i is denoted as \hat{k}_i and that desired by actuator i is denoted as \tilde{k}_i , which are generally different. Accordingly, let \tilde{m}_i and \hat{m}_i denote the allocated block-length during the uplink and downlink transmission for the i -th sensor-actuator pair, $i \in \mathcal{N}$, respectively. Thus, we have $\sum_{i=1}^N (\hat{m}_i + \tilde{m}_i) \leq M$, where M denotes the total block-length.

In order to process the uploaded data from sensors, the UAV needs to consume certain time and energy for implementing the inference task. Let f and κ denote the CPU frequency and the effective capacitance for computing at the UAV, C denote the total CPU cycles required for accomplishing the task. Then the energy required for executing the inference task is approximated $P_{\text{comp}} = \kappa C f^2$ and the time duration for computation is given as T_{comp} ^[13]. Suppose that δ is the symbol length for wireless communication and T_{total} denotes the total end-to-end delay for the inference task. Then we have $\delta M = T_{\text{total}} - T_{\text{comp}}$. In this paper, we assume that the computation delay T_{comp} and energy consumption P_{comp} are given and thus are not considered in the optimization of our interest.

Based on the achievable rate formula of finite block-length codes^[25], it follows that to transmit a short packet within finite symbols, the PERs (within (0, 0.5)) of the uplink and downlink transmission for the i -th sensor-actuator pairs are approximat-

ed as the following two formulas, respectively^[25].

$$\hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) = Q \left(\frac{\hat{m}_i \ln(1 + \hat{\gamma}(\mathbf{q})) - \hat{k}_i \ln 2}{\sqrt{\hat{m}_i} \sqrt{1 - (1 + \hat{\gamma}(\mathbf{q}))^{-2}}} \right), \quad (7)$$

$$\tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) = Q \left(\frac{\tilde{m}_i \ln(1 + \tilde{\gamma}(\mathbf{q})) - \tilde{k}_i \ln 2}{\sqrt{\tilde{m}_i} \sqrt{1 - (1 + \tilde{\gamma}(\mathbf{q}))^{-2}}} \right), \quad (8)$$

where $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$.

As a result, we define the end-to-end PER of the i -th sensor-actuator pair as the rate when the packet error occurs at either the uplink or downlink transmission, which is denoted as ε_i and given by

$$\varepsilon_i = 1 - (1 - \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i))(1 - \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i)) = \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) - \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) \times \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i). \quad (9)$$

In general, under our URLLC consideration, the sensor-actuator pairs should work at the regime when the PERs are generally very small, i.e., it should hold that $\hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) \leq 10^{-1}$, $\tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \leq 10^{-1}, i \in \mathcal{N}$. In this case, we have $\hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \gg \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) \times \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i)$, and accordingly, it follows that $\varepsilon_i \approx \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i), \forall i \in \mathcal{N}$ ^[27].

Our objective is to minimize the maximum PER of the N pairs, by jointly optimizing the UAV's placement location and transmitting power allocation, and the users' block-length, subject to the total block-length constraint and the sum transmitting power constraint at the UAV. For notational convenience, we denote that $\mathbf{m} \triangleq \{\hat{m}_i, \tilde{m}_i\}$, $\mathbf{p} \triangleq \{p_i\}$. Therefore, the maximum end-to-end PER minimization problem of our interest can be formulated as

$$(P1): \min_{\mathbf{q}, \mathbf{m}, \mathbf{p}} \max_{i \in \mathcal{N}} \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i)$$

$$\text{s.t. } \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) < 10^{-1}, \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) < 10^{-1}, \forall i \in \mathcal{N} \quad (10a)$$

$$\sum_{i \in \mathcal{N}} (\hat{m}_i + \tilde{m}_i) \leq M \quad (10b)$$

$$\sum_{i \in \mathcal{N}} p_i \leq P_{\text{sum}} \quad (10c)$$

$$p_i \geq 0, \forall i \in \mathcal{N}, \quad (10d)$$

where Eq. (10a) corresponds to the constraints for the approximation of objective function to be accurate, Eq. (10b) denotes the total block-length constraint and Eq. (10c) denotes the sum transmitting power constraint at the UAV. As the objective function in (P1) is a non-convex function in general, the problem (P1) is a non-convex problem that is generally difficult to be optimally solved.

3 Proposed Solution to Problem (P1)

In this section, we propose an efficient algorithm to obtain a high-quality solution to the problem (P1). Towards this end, we first introduce an auxiliary variable ε , and equivalently reformulate the problem (P1) as

$$(P2): \min_{q, m, p, \varepsilon} \varepsilon \quad (11)$$

$$\text{s.t. } \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \leq \varepsilon, \forall i \in \mathcal{N}$$

(10a), (10b) and (10c).

However, the problem (P2) is still non-convex. To tackle this challenge, we propose an algorithm to solve the problem (P2) or (P1) by using the alternating optimization technique, in which the block-length allocation, the transmitting power allocation, and the deployment location are optimized in an alternating manner, by considering the others to be given, towards a converged solution.

3.1 Block-Length Allocation

Under any given UAV's location \mathbf{q} and power allocation \mathbf{p} , the block-length allocation problem is formulated as

$$(P2.1): \min_{m, \varepsilon} \varepsilon \quad (12a)$$

$$\text{s.t. } \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \leq \varepsilon, \forall i \in \mathcal{N}$$

$$\sum_{i \in \mathcal{N}} (\hat{m}_i + \tilde{m}_i) \leq M. \quad (12b)$$

Since the error rate functions $\varepsilon(k, \mathbf{q}, m)$ in the constraint (12a) are convex with respect to $m^{[26]}$, the problem (P2.1) is a convex optimization problem. Therefore, the strong duality holds between (P2.1) and its Lagrange dual problem. As a result, we can optimally solve (P2.1) by using the Lagrange duality method^[28].

Let $\lambda_i \geq 0, \forall i \in \mathcal{N}$ and $\mu \geq 0$ denote the dual variables associated with the i -th constraint in Eqs. (12a) and (12b), respectively. Then we define $\boldsymbol{\lambda} \triangleq [\lambda_1, \dots, \lambda_N]$. Let \mathcal{X} denote the set $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ specified by the constraints in the dual problem of (P2.1). The Lagrangian of problem (P2.1) is given by

$$\mathcal{L}_1(\varepsilon, \mathbf{m}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = (1 - \sum_{i \in \mathcal{N}} \lambda_i) \varepsilon + \sum_{i \in \mathcal{N}} \lambda_i (\hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i)) + \mu \sum_{i \in \mathcal{N}} (\hat{m}_i + \tilde{m}_i) - \mu M. \quad (13)$$

Accordingly, the dual function of (P2.1) is

$$g(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \min_{m, \varepsilon} L(\mathbf{m}, \varepsilon, \boldsymbol{\lambda}, \boldsymbol{\mu}) \quad (14)$$

$$\text{s.t. (12a) and (12b).}$$

As a result, the dual problem is given by

$$(D2.1): \max_{\boldsymbol{\lambda}, \boldsymbol{\mu}} g(\boldsymbol{\lambda}, \boldsymbol{\mu}) \quad (15)$$

$$\text{s.t. } \sum_{i=1}^N \lambda_i = 1$$

$$\mu \geq 0, \lambda_i \geq 0, \forall i \in \mathcal{N}.$$

First, we obtain the dual function under any given $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ by solving Eq. (14). Towards this end, we obtain the optimal solution $\{\tilde{m}_i^*\}$ and $\{\hat{m}_i^*\}$ via solving Eqs. (16) and (17) by using a bisection search.

$$\frac{\partial \mathcal{L}_1}{\partial \tilde{m}_i} = \lambda_i \frac{-(\tilde{a}\tilde{m}_i + \tilde{b})}{2\sqrt{2\pi} \tilde{m}_i^{3/2}} e^{-(\tilde{a}\sqrt{\tilde{m}_i} - \tilde{b}/\sqrt{\tilde{m}_i})^2/2} + \mu = 0, \quad (16)$$

$$\frac{\partial \mathcal{L}_1}{\partial \hat{m}_i} = \lambda_i \frac{-(\hat{a}\hat{m}_i + \hat{b})}{2\sqrt{2\pi} \hat{m}_i^{3/2}} e^{-(\hat{a}\sqrt{\hat{m}_i} - \hat{b}/\sqrt{\hat{m}_i})^2/2} + \mu = 0, \quad (17)$$

$$\text{where } \tilde{a} = \frac{\ln(1 + \tilde{\gamma}_i)}{\sqrt{1 - (1 + \tilde{\gamma}_i)^{-2}}} > 0 \text{ and } \tilde{b} = \frac{k \ln 2}{\sqrt{1 - (1 + \tilde{\gamma}_i)^{-2}}} > 0.$$

Then, we obtain the optimal $\boldsymbol{\lambda}^{\text{opt}}$ and $\boldsymbol{\mu}^{\text{opt}}$ via solving the dual problem (D2.1) by using sub-gradient based method^[29], such as the ellipsoid method. With $\boldsymbol{\lambda}^{\text{opt}}$ and $\boldsymbol{\mu}^{\text{opt}}$ at hand, we can obtain the optimal solution $\{\tilde{m}_i^{\text{opt}}\}$ and $\{\hat{m}_i^{\text{opt}}\}$ by replacing $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ in Eqs. (16) and (17) as $\boldsymbol{\lambda}^{\text{opt}}$ and $\boldsymbol{\mu}^{\text{opt}}$. Therefore, problem (P2.1) is solved.

3.2 Power Allocation

For any given UAV's location \mathbf{q} and block-length allocation \mathbf{m} , the power allocation problem is formulated as:

$$(P2.2): \min_{p, \varepsilon} \varepsilon \quad (18a)$$

$$\text{s.t. } \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \leq \varepsilon, \forall i \in \mathcal{N}$$

$$\sum_{i \in \mathcal{N}} p_i \leq P_{\text{sum}} \quad (18b)$$

$$p_i \geq 0, \forall i \in \mathcal{N}. \quad (18c)$$

We have the following lemma for solving the problem.

Lemma: For any given UAV's location \mathbf{q} and latency allocation \mathbf{m} , the error rate ε is convex in \mathbf{p} under the mild condition $\varepsilon(\gamma, m) < 0.5$.

Proof: See Appendix.

Since the error rate functions $\varepsilon(k, \mathbf{q}, m)$ in the constraint (18a) are convex with respect to p , the problem (P2.2) is a convex optimization problem. Therefore, the strong duality also holds between (P2.2) and its Lagrange dual problem. As a result, we can optimally solve (P2.2) by using Lagrange duality method.

Let $\zeta_i \geq 0, \nu_i \geq 0, \forall i \in \mathcal{N}$ and $\eta \geq 0$ denote the dual variables associated with the constraints (18a), (18c) and (18b), re-

spectively. Then we define $\zeta \triangleq [\zeta_1, \dots, \zeta_N]$ and $\nu \triangleq [\nu_1, \dots, \nu_N]$. Let γ denote the set of ζ , η and ν specified by the constraints in the dual problem of (P2.2). The Lagrangian of the problem (P2.2) is given by

$$\mathcal{L}_2(p, \varepsilon, \zeta, \nu, \eta) = (1 - \sum_{i \in \mathcal{N}} \zeta_i) \varepsilon + \sum_{i \in \mathcal{N}} \zeta_i (\hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i)) + \eta \sum_{i \in \mathcal{N}} p_i - \sum_{i \in \mathcal{N}} \nu_i p_i - \eta P_{\text{sum}}. \quad (19)$$

Accordingly, the dual function is given as:

$$g(\zeta, \nu, \eta) = \min_{p, \varepsilon} L(p, \varepsilon, \zeta, \nu, \eta) \quad (20)$$

s.t. (18a), (18b) and (18c).

As a result, the dual problem of (P2.2) is expressed as

$$(D2.2): \max_{\zeta, \nu, \eta} g(\zeta, \nu, \eta) \quad (21)$$

s.t. $\sum_{i=1}^N \zeta_i = 1$
 $\eta \geq 0, \zeta_i \geq 0, \nu_i \geq 0, \forall i \in \mathcal{N}$.

First, we obtain the dual function of Eq. (20) under any given ζ , η and ν by solving the problem of Eq. (22). In particular, we can obtain the optimal solution $\{p_i^*\}$ via solving Eq. (22) by the bisection search.

$$\frac{\partial \mathcal{L}_2}{\partial p_i} = \frac{\zeta_i}{\sqrt{2\pi}} A_d e^{(-A^2)/2} + \eta - \nu_i = 0, \quad (22)$$

where

$$A_d \triangleq \frac{\frac{h}{\sigma^2} m^{3/2} \sqrt{1-(1+\gamma_i)^{-2}} - \sqrt{m} \frac{h}{\sigma^2} (m \ln(1+\gamma_i) - k \ln 2)}{(1+\gamma_i)^3 \sqrt{1-(1+\gamma_i)^{-2}}}, \quad (23)$$

$$A \triangleq \frac{m \ln(1+\gamma_i) - k \ln 2}{\sqrt{m} \sqrt{1-(1+\gamma_i)^{-2}}}. \quad (24)$$

Then we obtain the optimal ζ^{opt} , η^{opt} and ν^{opt} via solving the dual problem (D2.2) by using the ellipsoid method^[28]. With ζ^{opt} , η^{opt} and ν^{opt} obtained, we can determine the optimal solution $\{p_i^{\text{opt}}\}$ by replacing ζ , η and ν in Eq. (22) as ζ^{opt} , η^{opt} and ν^{opt} . Therefore, problem (P2.2) is finally solved.

3.3 UAV Placement Optimization

Finally, under any given UAV's transmitting power allocation p and users' block-length allocation m , we optimize the UAV placement location, for which the optimization problem is formulated as

$$(P2.3): \min_{x, y, \varepsilon} \varepsilon$$

$$\text{s.t. } \hat{\varepsilon}_i(\hat{k}_i, \mathbf{q}, \hat{m}_i) + \tilde{\varepsilon}_i(\tilde{k}_i, \mathbf{q}, \tilde{m}_i) \leq \varepsilon, \forall i \in \mathcal{N}. \quad (25)$$

We solve the problem (P2.3) by adopting a two-dimensional (2D) exhaustive search over the region $[\underline{x}, \bar{x}] \times [\underline{y}, \bar{y}]$, where $\underline{x} = \min_{i \in \mathcal{N}}(\hat{x}_i, \tilde{x}_i)$, $\bar{x} = \max_{i \in \mathcal{N}}(\hat{x}_i, \tilde{x}_i)$, $\underline{y} = \min_{i \in \mathcal{N}}(\hat{y}_i, \tilde{y}_i)$, $\bar{y} = \max_{i \in \mathcal{N}}(\hat{y}_i, \tilde{y}_i)$.

In summary, we optimize the UAV's placement location \mathbf{q} and the wireless resource allocation m and p in an alternating manner. It is worth noting that the objective value (i.e., the achieved maximum end-to-end PER value) is monotonically non-increasing after each update. As a result, the alternating-optimization-based approach eventually converges to a converged solution to (P2) or (P1), as the maximum PER value is lower bounded by zero. It is also worth noting that the proposed algorithm can be employed offline before the UAV is launched for helping perform the inference task, which can thus be implemented efficiently in practice and will not affect the low latency requirement of the online computation task.

4 Numerical Results

In this section, we present numerical results to evaluate the performance of the proposed design. In the simulation, we randomly generate sensors and actuators' positions in a 2D area within $100 \times 100 \text{ m}^2$. We set $\hat{k}_i = 100$ bit and $\tilde{k}_i = 80$ bit, and $\forall i \in \mathcal{N}$ for uplink and downlink communications. The reference channel power gain is set as $\rho_0 = -40$ dB and the receiver noise power is $\sigma^2 = -90$ dBm. The transmitting power of all sensors is $Q = 1$ W. The UAV flies at a fixed altitude of $H = 120$ m. We consider the following reference schemes for performance comparison.

- **Benchmark scheme:** In this scheme, the UAV hovers above a fixed location (i.e., the middle point of the area) and wireless resources are allocated equally among all sensor-actuator pairs (i.e., $\frac{M}{2N}$ for all the sensor-actuator pairs' block-length and $\frac{P_{\text{sum}}}{N}$ for the UAV's transmitting power to actuators).

- **Placement optimization only:** In this scheme, we consider equal block-length and power allocations (i.e., wireless resources are allocated to all the sensor-actuator pairs evenly). Under this design, the UAV hovers at an optimized location, which can be obtained by solving the problem (P2.3) under given UAV's transmitting power allocation p and users' block-length allocation m .

- **Resource allocation only:** The UAV hovers above the middle point of the area with optimal wireless resource allocations, which can be obtained via solving the problems (P2.1) and (P2.2).

Fig. 2 shows the maximum end-to-end PER versus the total block-length M , where we set $P_{\text{sum}} = 36$ dBm. It is observed

that our proposed design outperforms other reference schemes and the performance gain becomes more significant when the total block-length becomes larger. It is also observed that the resource allocation only scheme significantly outperforms the placement optimization only scheme. This shows the importance of the joint uplink and downlink resource allocation.

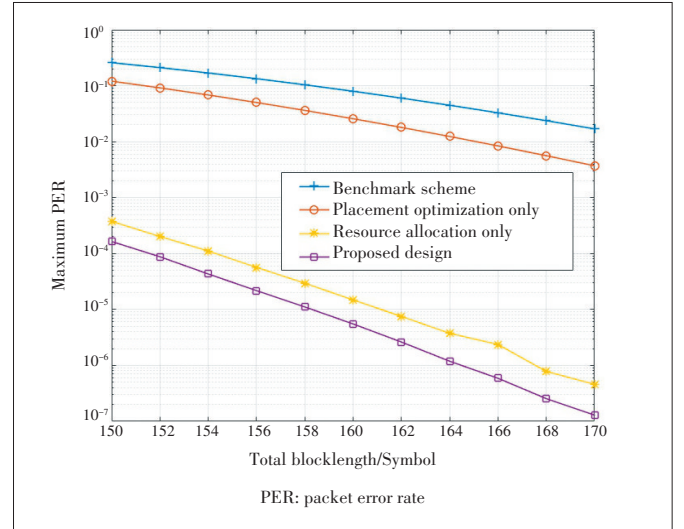
Fig. 3 shows the maximum end-to-end PER versus the total transmitting power P_{sum} , where we set $M = 150$. It is observed that the scheme with placement optimization only and the benchmark scheme both lead to a PER error floor when $P_{\text{sum}} > 33$ dBm. It is also observed that in the low transmitting power regime, the placement only scheme slightly outperforms the resource allocation only scheme. By contrast, when the transmitting power becomes high, the placement only scheme and the benchmark scheme result in unchanged maximum PER values, which is due to the fact that the PER performance is fundamentally limited by the uplink because of the lacking of resource allocations. In this case, the resource allocation only scheme and the proposed design lead to monotonically decreasing maximum PER values as transmitting power increases. Over all transmitting power regimes, the proposed design with both resource allocation and UAV placement optimization is observed to always achieve the best performance, and the performance gain becomes more evident when the transmitting power becomes larger.

5 Conclusions

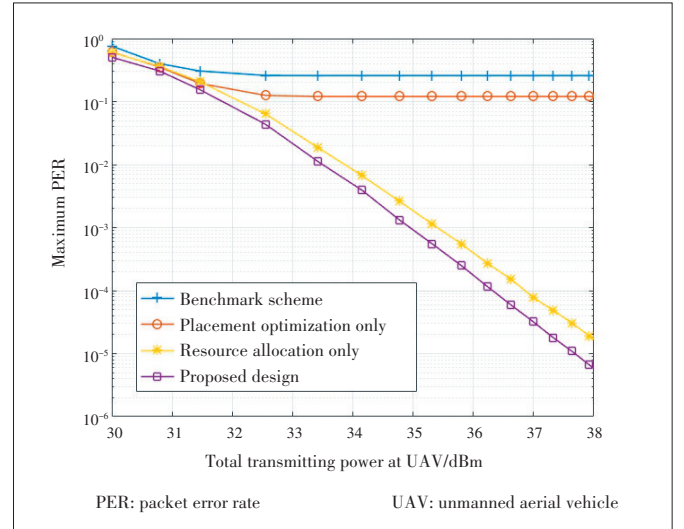
In this paper, we study a new UAV-assisted MEC network with URLLC, in which a UAV is deployed at an optimizable location for serving multiple pairs of sensors and actuators. We minimize the maximum end-to-end PER of these sensor-actuator pairs by jointly optimizing the UAV's placement location and transmitting power allocation, and the block-length allocation among these sensor-actuator pairs. We propose an effective algorithm based on the alternating optimization technique to obtain a high-quality solution to this challenging problem. Numerical results show that the proposed algorithm achieves better performance than other benchmark schemes. Due to the space limitation, there have been some other interesting issues that are not addressed in this paper, which are discussed in the following to motivate future work.

We consider the quasi-stationary UAV scenario by only optimizing the UAV's deployment location. In some other scenarios (e.g., the IoT devices have intermittent traffics that happen at different time instants), it may be feasible to exploit the UAV's mobility over time for further enhancing the round-trip URLLC performance. In this case, how to optimize the UAV's trajectory optimization (instead of placement location only) and the wireless resource allocation to maximize the system performance is an interesting and challenging problem.

We only consider the round-trip URLLC among the sensor-actuator pairs by ignoring the computation or information pro-



▲ **Figure 2.** The maximum end-to-end PER versus the number of total available block-length M .



▲ **Figure 3.** The maximum end-to-end PER versus the UAV's maximum transmitting power P_{sum} .

cessing at the UAV. Eventually, the computation-communication tradeoff in MEC and mobile edge learning systems can also be exploited for enhancing the latency performance. How to optimize the performance of UAV-enabled MEC systems for various edge machine learning applications is an interesting direction for future investigation.

Appendix

Since $Q(x)$ is strictly decreasing and convex in x when $Q(x) < 0.5$, it suffices to show the convexity of $\varepsilon(\gamma, m)$ in γ by proving

$$f(m, \gamma) \triangleq \frac{m \ln(1 + \gamma) - N \ln 2}{\sqrt{m} \sqrt{1 - (1 + \gamma)^{-2}}}, \quad (26)$$

which is strictly concave in γ for any given m .

Let $t = 1 + \gamma > 1$, and then we have

$$\bar{f}(m, t) \triangleq f(m, \gamma) \triangleq \frac{mt \ln(t) - tN \ln 2}{\sqrt{m(t^2 - 1)}}. \quad (27)$$

Thus, we have

$$\frac{\partial \bar{f}(m, t)}{\partial t} = \frac{m(t^2 - \ln t - 1) + N \ln 2}{(t^2 - 1)\sqrt{m(t^2 - 1)}} = \frac{\sqrt{m}(t^2 - \ln t - 1)}{(t^2 - 1)^{\frac{3}{2}}} + \frac{N \ln 2}{\sqrt{m}} \frac{1}{(t^2 - 1)^{\frac{3}{2}}}. \quad (28)$$

Let $A = \sqrt{m}$, $B = \frac{N \ln 2}{\sqrt{m}}$, we have

$$\begin{aligned} \frac{\partial^2 \bar{f}(m, t)}{\partial t^2} &= \\ A((2t - \frac{1}{t})(t^2 - 1)^{-\frac{3}{2}} - 3Bt(t^2 - 1)^{-\frac{5}{2}} - 3t(t^2 - \ln t - 1)(t^2 - 1)^{-\frac{5}{2}}) &= \\ (2t - \frac{1}{t})(t^2 - 1)^{-\frac{3}{2}} - 3t(t^2 - \ln t - 1) - 3t \frac{B}{A} &= \\ (2 - \frac{1}{t^2})(t^2 + 1) - 3(t^2 - \ln t - 1) - 3 \frac{B}{A} &= \\ -t^2 + \frac{1}{t^2} + 3 \ln t - 3 \frac{B}{A} < 0, \end{aligned} \quad (29)$$

where \triangleq means both sides have the same sign. Therefore, $\bar{f}(m, t)$ is concave with respect to (w.r.t.) t . Since t is the affine transformation of p , it follows that $\bar{f}(m, t)$ is also concave w.r.t. p . Due to the convexity rule of compound function^[29], $\varepsilon(\gamma, m)$ is strictly convex w.r.t. p under a mild condition of $\varepsilon(\gamma, m) < 0.5$.

References

- [1] MAO Y Y, YOU C S, ZHANG J, et al. A survey on mobile edge computing: the communication perspective [J]. IEEE communications surveys & tutorials, 2017, 19(4): 2322–2358. DOI:10.1109/comst.2017.2745201
- [2] DING Z G, XU J, DOBRE O A, et al. Joint power and time allocation for NOMA-MEC offloading [J]. IEEE transactions on vehicular technology, 2019, 68(6): 6207–6211. DOI:10.1109/tvt.2019.2907253
- [3] SUN H J, ZHOU F H, HU R Q. Joint offloading and computation energy efficiency maximization in a mobile edge computing system [J]. IEEE transactions on vehicular technology, 2019, 68(9): 3052–3056. DOI:10.1109/tvt.2019.2893094
- [4] BAI T, WANG J J, REN Y, et al. Energy-efficient computation offloading for secure UAV-edge-computing systems [J]. IEEE transactions on vehicular technology, 2019, 68(6): 6074–6087. DOI:10.1109/tvt.2019.2912227
- [5] WANG F, XU J, DING Z G. Multi-antenna NOMA for computation offloading in multiuser mobile edge computing systems [J]. IEEE transactions on communications, 2019, 67(3): 2450–2463. DOI:10.1109/tcomm.2018.2881725
- [6] LIU J, MAO Y Y, ZHANG J, et al. Delay-optimal computation task scheduling for mobile-edge computing systems[C]/2016 IEEE International Symposium on Information Theory. Barcelona, Spain, 2016. DOI:10.1109/isit.2016.7541539
- [7] WANG F, XU J, WANG X, et al. Joint offloading and computing optimization in

- wireless powered mobile-edge computing systems [J]. IEEE transactions on wireless communications, 2018, 17(3): 1784–1797. DOI: 10.1109/twc.2017.2785305
- [8] ZHU G X, LIU D Z, DU Y Q, et al. Toward an intelligent edge: wireless communication meets machine learning [J]. IEEE communications magazine, 2020, 58(1): 19–25. DOI:10.1109/mcom.001.1900103
- [9] MO X P, XU J. Energy-efficient federated edge learning with joint communication and computation design [EB/OL]. (2020–2–29) [2020–4–1]. <https://arxiv.org/abs/2003.00199>
- [10] CUI Q M, GONG Z Z, NI W, et al. Stochastic online learning for mobile edge computing: learning from changes [J]. IEEE communications magazine, 2019, 57(3): 63–69. DOI:10.1109/mcom.2019.1800644
- [11] MOTLAGH N H, BAGAA M, TALEB T. UAV-Based IoT Platform: a crowd surveillance use case [J]. IEEE Communications Magazine, 2017, 55(2): 128–134. DOI:10.1109/mcom.2017.1600587cm
- [12] ZENG Y, XU J, ZHANG R. Energy minimization for wireless communication with rotary-wing UAV [J]. IEEE transactions on wireless communications, 2019, 18(4): 2329–2345. DOI:10.1109/twc.2019.2902559
- [13] CAO X W, WANG F, XU J, et al. Joint computation and communication cooperation for energy-efficient mobile edge computing [J]. IEEE Internet of Things journal, 2019, 6(3): 4188–4200. DOI:10.1109/jiot.2018.2875246
- [14] ZENG Y, ZHANG R, LIM T J. Wireless communications with unmanned aerial vehicles: opportunities and challenges [J]. IEEE communications magazine, 2016, 54(5): 36–42. DOI:10.1109/mcom.2016.7470933
- [15] ZENG Y, ZHANG R, LIM T J. Throughput maximization for UAV-enabled mobile relaying systems [J]. IEEE transactions on communications, 2016, 64(12): 4983–4996. DOI:10.1109/tcomm.2016.2611512
- [16] XU J, ZENG Y, ZHANG R. UAV-enabled wireless power transfer: trajectory design and energy optimization [J]. IEEE transactions on wireless communications, 2018, 17(8): 5092–5106. DOI:10.1109/twc.2018.2838134
- [17] WU Q Q, ZENG Y, ZHANG R. Joint trajectory and communication design for multi-UAV enabled wireless networks [J]. IEEE transactions on wireless communications, 2018, 17(3): 2109–2121. DOI:10.1109/twc.2017.2789293
- [18] XIE L F, XU J, ZHANG R. Throughput maximization for UAV-enabled wireless powered communication networks [J]. IEEE Internet of Things journal, 2019, 6(2): 1690–1703. DOI:10.1109/jiot.2018.2875446
- [19] ZHOU F, WU Y, HU R Q, QIAN Y. Computation rate maximization in UAV-enabled wireless powered mobile-edge computing systems [J]. IEEE journal on selected areas in communications, 2018, 36(9): 1927–1941. DOI: 10.1109/JSAC.2018.2864426A
- [20] GARG S, SINGH A, BATRA S, et al. UAV-empowered edge computing environment for cyber-threat detection in smart vehicles [J]. IEEE network, 2018, 32(3): 42–51. DOI:10.1109/mnet.2018.1700286
- [21] PU L J, CHEN X, MAO G Q, et al. Chimera: an energy-efficient and deadline-aware hybrid edge computing framework for vehicular crowdsensing applications [J]. IEEE Internet of Things journal, 2019, 6(1): 84–99. DOI:10.1109/jiot.2018.2872436
- [22] DU Y, WANG K Z, YANG K, et al. Energy-efficient resource allocation in UAV based MEC system for IoT devices [C]/2018 IEEE Global Communications Conference (GLOBECOM). Abu Dhabi, United Arab Emirates, 2018: 9–13. DOI:10.1109/glocom.2018.8647789
- [23] JEONG S, SIMEONE O, KANG J. Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning [J]. IEEE transactions on vehicular technology, 2018, 67(3): 2049–2063. DOI: 10.1109/tvt.2017.2706308
- [24] ZHOU F H, WU Y P, SUN H J, et al. UAV-enabled mobile edge computing: offloading optimization and trajectory design [C]/2018 IEEE International Conference On Communications (ICC). Kansas City, USA, 2018: 20–24. DOI: 10.1109/icc.2018.8422277
- [25] DURISI G, KOCH T, POPOVSKI P. Toward massive, ultrareliable, and low-latency wireless communication with short packets [J]. Proceedings of the IEEE, 2016, 104(9): 1711–1726. DOI:10.1109/jproc.2016.2537298
- [26] SHEN C, CHANG T H, XU H Q, et al. Joint uplink and downlink transmission design for URLLC using finite blocklength codes [C]/2018 15th International Symposium on Wireless Communication Systems (ISWCS). Lisbon, Portugal, 2018: 28–31. DOI:10.1109/iswcs.2018.8491069

➔ To Page 82



Adaptive and Intelligent Digital Signal Processing for Improved Optical Interconnection

SUN Lin¹, DU Jiangbing¹, HUA Feng²,
TANG Ningfeng², HE Zuyuan¹

(1. State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China;

2. State Key Laboratory of Mobile Network and Mobile Multimedia Technology, ZTE Corporation, Shenzhen, Guangdong 518055, China)

DOI: 10.12142/ZTECOM.202002008

[http://kns.cnki.net/kcms/detail/34.1294.](http://kns.cnki.net/kcms/detail/34.1294.TN.20200612.1034.001.html)

TN.20200612.1034.001.html, published online June 12, 2020

Manuscript received: 2019-03-04

Abstract: In recent years, explosively increasing data traffic has been boosting the continuous demand of high speed optical interconnection inside or among data centers, high performance computers and even consumer electronics. To pursue the improved interconnection performance of capacity, energy efficiency and simplicity, effective approaches are demonstrated including particularly advanced digital signal processing (DSP) methods. In this paper, we present a review about the enabling adaptive DSP methods for optical interconnection applications, and a detailed summary of our recent and ongoing works in this field. In brief, our works focus on dealing with the specific issues for short-reach interconnection scenarios with adaptive operation, including signal-to-noise-ratio (SNR) limitation, level nonlinearity distortion, energy efficiency consideration and the decision precision.

Keywords: optical interconnection; digital signal processing; advanced modulation formats

Citation (IEEE Format): L. Sun, J. B. Du, F. Hua, et al., "Adaptive and intelligent digital signal processing for improved optical interconnection," *ZTE Communications*, vol. 18, no. 2, pp. 57 – 73, Jun. 2020. doi: 10.12142/ZTECOM.202002008.

1 Introduction

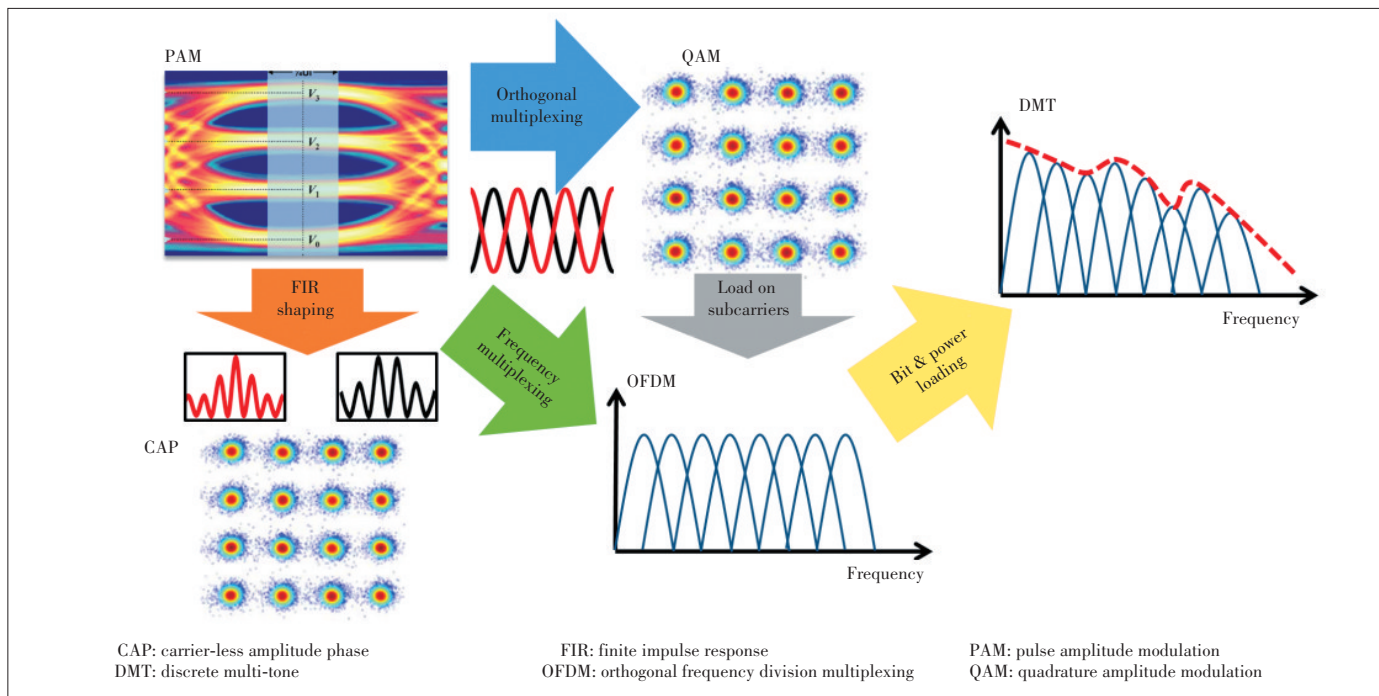
In the past decade, the data traffic has been explosively increasing due to applications such as 4K/8K display, cloud computing, 4G/5G and augmented reality/virtual reality (AR/VR), urgently driving the demand of high-capacity data communications. Typically, the very severe data traffic for communication and interaction occurs in data centers with single-lane data rate over 100 Gbit/s. For this scenario, traditional electrical interconnects find its bottlenecks in the perspective of power consumption, available bandwidth and implementation density. In contrast, optical approach exhibits great advantages of high capacity, high density, and perfect robustness to ambient environment as well as improved energy efficiency. With these excellent features, optical intensity-modulation and direct-detection (IM-DD) solution is adopted by IEEE 802.3bs Task Force, for over 100 Gbit/s interconnects with distance ranging from 100 m to 10 km^[1]. The ongoing trend of standardization is utilizing advanced modulation formats and suitable optical devices for IM-DD systems.

In the perspective of modulation formats, advanced solu-

tions enable improved spectrum efficiency (SE) for short-reach application including pulse amplitude modulation (PAM)^[2–4], quadrature amplitude modulation (QAM)^[5–7], orthogonal frequency division multiplexing (OFDM)^[8–10], discrete multi-tone (DMT) modulation^[11–14] and carrier-less amplitude phase (CAP) modulation^[15–18], as shown in **Fig. 1**. PAM, which exhibits the advantages of simplicity and easy synchronization is currently the most suitable candidate for IM-DD optical interconnects with a distance below 10 km. By comparison, QAM and OFDM based on orthogonal multiplexing supporting coherent detection with improved receiver sensitivity, are applicable for interconnection distance over 10 km. DMT is one special solution of channel-adaptive OFDM, by loading modulations with different bit numbers on individual subcarriers with reference to the channel's signal-to-noise-ratio (SNR) response. Consequently, DMT performs better in channels constrained and fluctuated by bandwidth. While CAP improves its SE by means of Nyquist shaping, it is more bandwidth efficient than PAM. Moreover, the finite impulse response (FIR) filters can be implemented on digital circuits with ignorable latency (related to FIR's tap number), making CAP more promising for short-reach optical interconnection.

To get the best use of these formats, hardware parts of an IM-DD system should be carefully selected to find the best

This work was supported by National Natural Science Foundation of China (NSFC) under Grant Nos. 61935011, 61875124 and 61875049.



▲ Figure 1. Advanced modulation formats for short-reach optical interconnections.

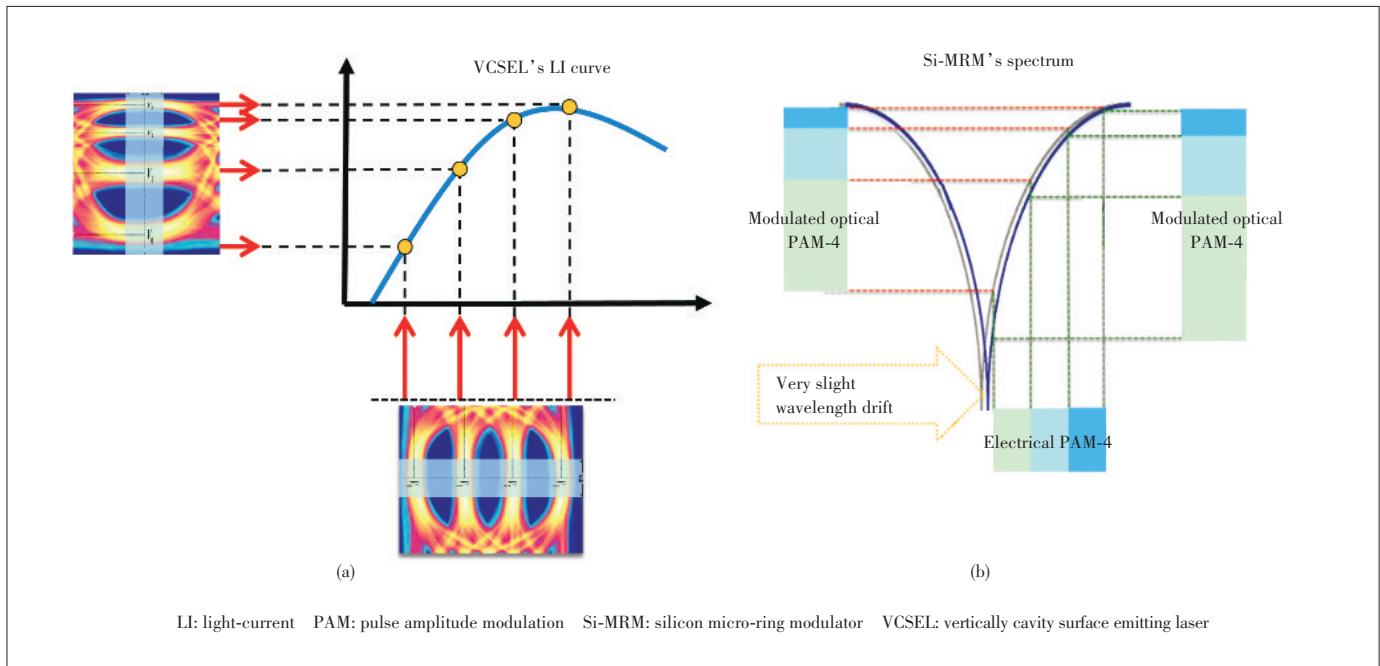
trade-off among available bandwidth, fabrication density and power consumption. For interconnection distance below 1 km, vertically cavity surface emitting laser (VCSEL) combined with multi-mode fiber (MMF) is the most power efficient solution, because power cost is mostly concerned in this circumstance. While for longer interconnection distance where severe dispersion bothers, single-mode fiber (SMF) transmission assisted with silicon modulator exhibits simultaneous advantages of large-scale integration and high capacity. Typically, silicon micro-ring modulator (Si-MRM) has the attractive feature of compact footprint, high modulation speed, and low energy consumption, thus it is quite suitable for this scenario.

Consequently, above-mentioned modulation formats, combined with suitable optical hardware, can improve the capacity whilst maintaining the simplicity and reliability. However, advanced modulation formats always show comparably weak robustness to noise and signal distortion during modulation and transmission. In detail, at the same baud rate, PAM-4 suffers severer eye closure than on-off keying (OOK) due to the bandwidth limitation^[19]. More level numbers (constellation numbers for QAM) of PAM lead to more sensitivity to SNR limitation. Furthermore, modulation nonlinearity always occurs, which severely degrades signal quality for PAM-4 (much severer for PAM-8). As for VCSEL-based links, modulation nonlinearity is mainly caused by the nonlinear light-current (LI) response of VCSEL, as shown in Fig. 2a^[20]. Moreover, due to the temperature-sensitive feature of VCSEL, the corresponding LI response can be easily deteriorated by temperature change, leading to more difficulties for linear operation. While for Si-MRM based SMF links, modulation nonlinearity is

mainly induced due to the free carrier dispersion effect and the Lorentz spectral shape of the Si-MRM^[21–22]. A small wavelength drift of MRM's spectra may lead to severe nonlinearity, as shown in Fig. 2b. So for VCSEL and Si-MRM based systems, the modulation nonlinearity is a specific distortion of PAM and other formats. In addition, power cost is also a critical concern for short-reach interconnection. As a result, how to pursue the cost-effective signaling is quite important. Therefore, for short-reach optical interconnection utilizing advanced modulation formats based on VCSEL and Si-MRM, the main issues are: 1) how to improve the performance under limited SNRs; 2) how to mitigate the modulation nonlinearity; 3) how to further enhance the energy efficiency.

At the same time, improved digital signal processing (DSP) technologies have been extensively investigated due to the desire of high-capacity transmission and low power consumption for optical interconnection. Widely known methods such as maximum likelihood sequence estimator (MLSE)^[23–25], decision feedback equalizer (DFE)^[26–28] and feed-forward equalizer (FFE)^[29–30] have been utilized in attempt to improve the system's robustness to inter-symbol interference (ISI).

In this paper, we take a review about the current works as to how to deal with above-mentioned three specific issues by DSP methods for adaptive operation, including our finished and ongoing works as well. This paper is composed of four parts. Section 1 is the introduction of short-reach optical interconnections. Section 2 is DSP methods at the transmitting side, mainly introducing probabilistic shaping. Section 3 is DSP in the receiver, mainly about machine-learning assisted techniques. Section 4 is the conclusions.



▲ Figure 2. Modulation nonlinearity caused by: (a) nonlinear LI curve of vertically cavity surface emitting laser VCSEL; (b) nonlinear spectra of Si-MRM as well as wavelength drift.

2 Adaptive Probabilistic Shaping at the Transmitter

Very recently, probabilistic shaping (PS) as a coding method has been rapidly developed in the field of coherent optical communications^[31–39]. PS can bring two benefits simultaneously: 1) improved achievable information rate (AIR) at the low SNR condition^[40]; 2) reduced average power due to PS when voltage of peak-to-peak (V_{pp}) is fixed^[41]. Consequently, PS is a quite useful approach to improve the transmission performance at the SNR-constrained condition, and enhance the energy efficiency at the same time. In the field of direct detection system, the theoretical AIR gain of 0.19 bit at 16.2 dB SNR has been reported by using the exponential distributions of 6 level PAM signals, and a corresponding experiment is presented by external modulation at 1550 nm for single mode fiber transmission^[42]. Meanwhile, an entropy loading scheme of DMT is reported to fit channel SNR response based on PS^[43–45]. In Ref. [44], the entropy loading is employed after water-filling algorithm to get rid of extra power reallocation for coherent optical system. Because the subcarriers' power is previously adapted to water filling, only limited numbers of shaped distribution are demonstrated. In Ref. [45], an entropy-loaded DMT without power allocation is proposed by using Maxwell-Boltzmann (MB) distribution in visible light communication system, with an AIR of 204 Mbit/s. However, above-stated researches do not focus on the short-reach interconnection.

As discussed in the introduction part, VCSEL exhibits excellent features of lower power cost, high fabrication density and large electrical bandwidth. It is quite suitable for short-

reach interconnections applications. Generally, the SNR and bandwidth for VCSEL-MMF links are more limited compared with coherent systems which require high SNR for high-level QAM and large bandwidth for achieving large capacity. It can thus be expected that PS coding will lead to improved AIR which is particularly desired for the VCSEL-MMF links. Most non-uniform distributions (such as the Maxwell-Boltzmann distribution and the exponential distribution) require distribution matchers and complex code words to code independent and identically distributed sequences into non-uniform ones with the desired distributions^[36], and the induced complexity is not suitable for the cost-sensitive short reach optical interconnections. As a result, the objective of this section is to investigate the power-efficient and low-complexity PS methods for cost-effective VCSEL-MMF optical interconnections.

2.1 Dyadic Probabilistic Shaping for PAM

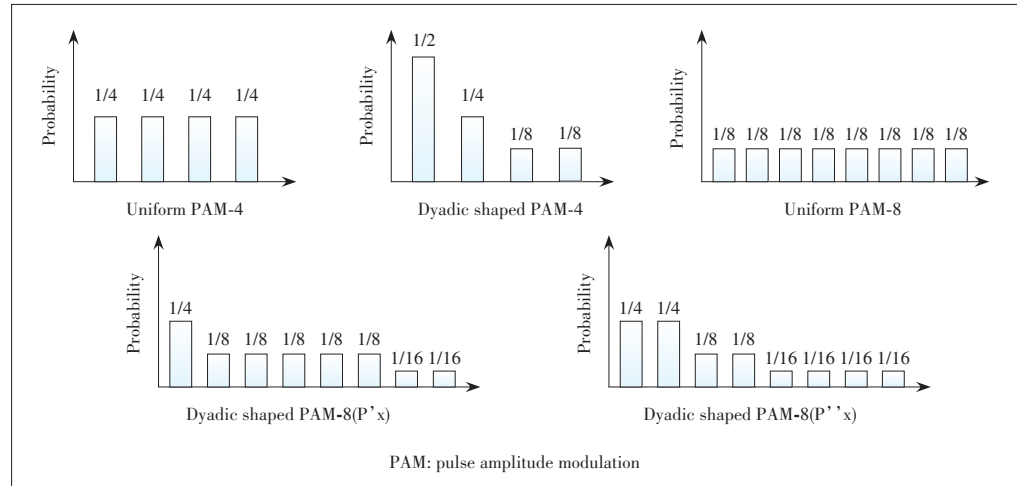
Dyadic PS is advantageous with simple implementation, which makes it particularly suitable for the cost-effective short reach applications. Moreover, PS coding can reallocate majority distributions to lower levels (near direct current) of the PAM signal, resulting in reduced average power along with improved energy efficiency. The VCSEL-MMF solution is currently dominating the sub-hundred-meter-distance optical interconnections with very large volume. The SNR and bandwidth for VCSEL-MMF links are particularly constrained. Therefore, we believe the proposed dyadic PS method is an opportune solution for the cost-sensitive VCSEL-MMF links due to its simplicity of coding with shaping gain and power reduction. To obtain dyadic distributions, binary mapping of M bits generates symbols with probability of 2^{-M} . However, such

variable-length coding will induce synchronization complexity at the receiver, when the noisy sequences are processed. One practical solution is to insert ambiguity bits to maintain the code words in the same length. The probability distributions of PAM-4 and PAM-8, and the shaped signals are shown in **Fig. 3**.

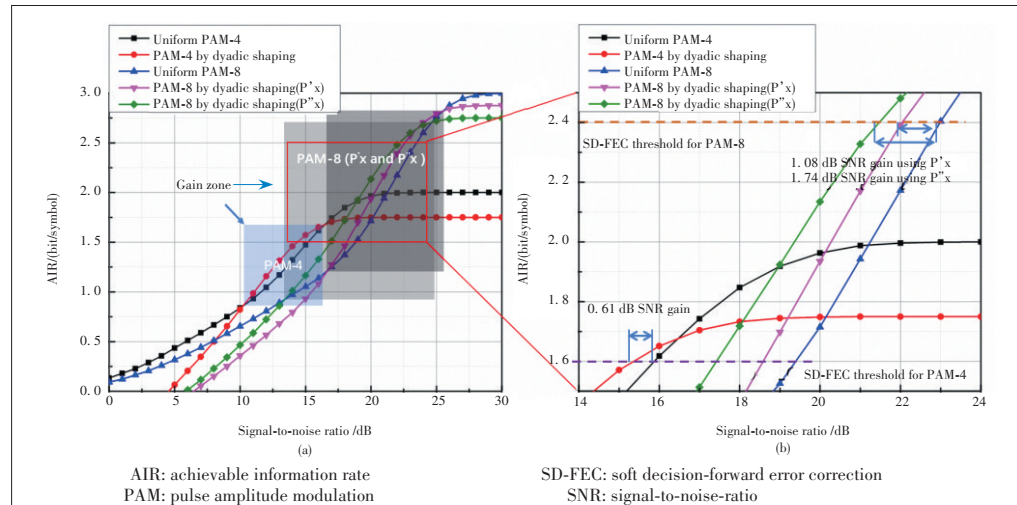
Then the theoretical AIR values are calculated for evaluating the performance of above distributions. As shown in **Fig. 4a**, AIR goes saturated along with the increase of SNR. The saturated AIR (maximum AIR) is reduced after dyadic PS, compared with the uniform distribution. However, at a certain SNR region, such as 10.3 – 16.57 dB for PAM-4 and 16.6 – 25.5 dB for PAM-8 ($P'X$), AIR values after dyadic PS become larger compared with those before shaping. It indicates that dyadic PS can increase the AIR of PAM- N system at the condition of constrained SNR. The zoom-in plot in **Fig. 4b** shows the AIR performances of the PAM- N modulations for SNR ranging from 14 dB to 24 dB. For PAM-4, the SNR requirement is reduced by 0.61 dB to achieve the 20% soft decision-forward error correction (SD-FEC) threshold (AIR=1.6 bit/symbol) after dyadic PS. As for PAM-8, 1.08 dB and 1.74 dB, SNR gains are obtained by dyadic PS to achieve 2.4 bit/symbol AIR, by using distributions of $P'X$ and $P''X$ respectively.

Experiments have been carried out directly for PAM-8 modulation over a VCSEL-MMF optical interconnection link to verify the performance of dyadic PS. Due to the inserted ambiguity bits of PS PAM-8, the code rate is 13/16. Consequently, to maintain the same data rate of 60 Gbit/s, the symbol rate of PS PAM-8 increases to 25 Gbaud (30 Gbaud to

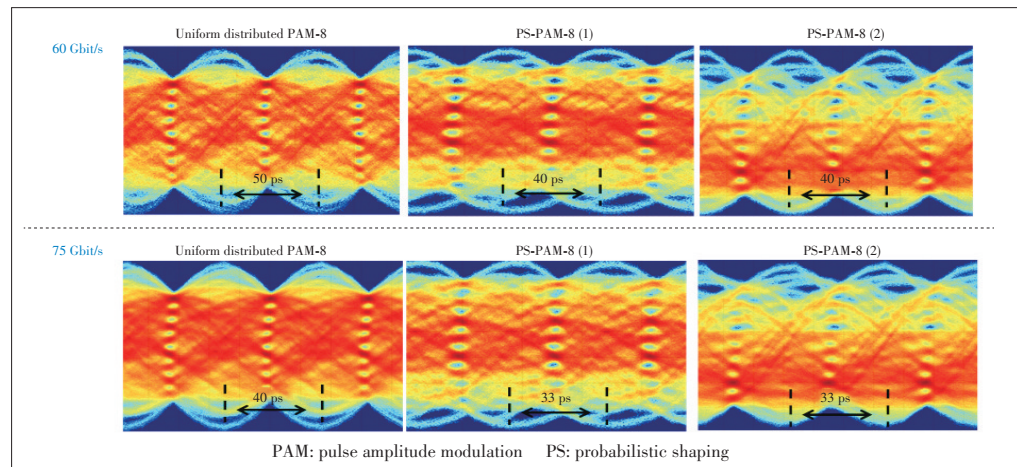
achieve the net rate of 75 Gbit/s). The optical back-to-back (B2B) eye-diagrams of the 60 Gbit/s and 75 Gbit/s PAM-8, before and after PS, are plotted in **Fig. 5**. The opening of the eye-



▲ **Figure 3.** Dyadic probability distributions of PAM-4, PAM-8, dyadic shaped PAM-4, dyadic shaped PAM-8($P'X$) and dyadic shaped PAM-8($P''X$).



▲ **Figure 4.** AIR curves for PAM-4 and PAM-8(a) with and (b) without dyadic PS (probabilistic shaping).



▲ **Figure 5.** Optical eye diagrams of 60 Gbit/s and 75 Gbit/s PAM-8 signals, with and without dyadic shaping.

diagrams has been improved (the sub-eye-diagrams are clearer with larger eye-height and eye-width) compared with that before PS. The bit error ratio (BER) curves of 75 Gbit/s PAM-8 signals are plotted in **Fig. 6**. Due to the increased symbol rates, the limited bandwidth of the optical channel shrinks the signal spectrum more severely. Consequently, the shaping gain of dyadic PS PAM-8 signals is smaller than theoretical one under this circumstance. Despite this, 0.88 dB and 0.3 dB SNR gain is obtained for optical B2B case and 100 m OM3 fiber transmission at 75 Gbit/s.

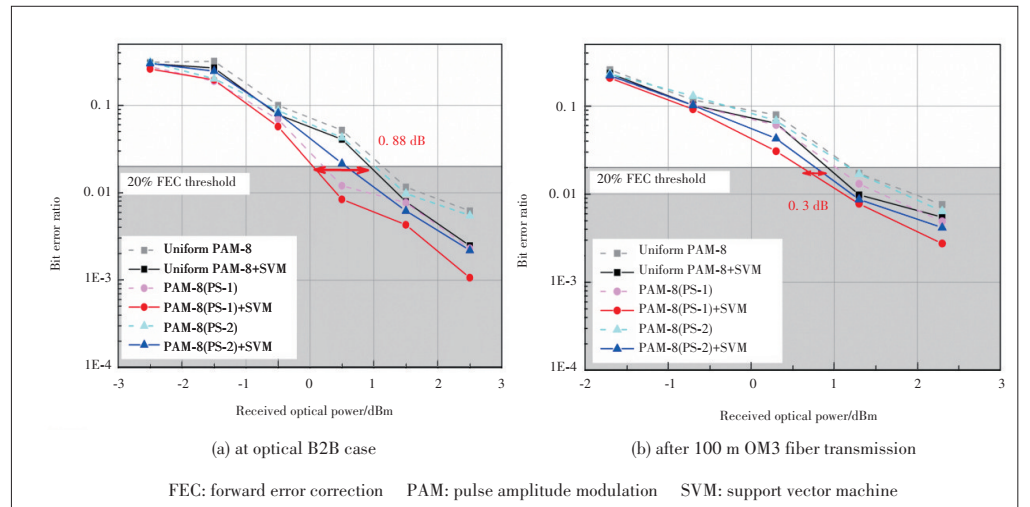
2.2. Maxwell-Boltzmann and Dyadic Probabilistic Shaping for DMT

The dyadic shaping has been demonstrated in PAM system. While for most practical IM-DD channels, frequency response is usually uneven with SNR fluctuating in the frequency domain due to fiber dispersion. A DMT modulation is proposed as a way to address this problem, by loading modulations with different bit-numbers on individual subcarriers with reference to the channel's SNR response. However, for conventional DMT, the constellations of individual subcarriers are all equal-probability distributed. Here we demonstrate a frequency-resolved adaptive probabilistic shaping method which refers to channel frequency response, for the 112 Gbit/s DMT-modulated IM-DD optical interconnection system. The continuously bit (in terms of entropy) loading is realized by adapted probability distributions, allowing for better fitting to channel frequency response with simultaneous shaping gain.

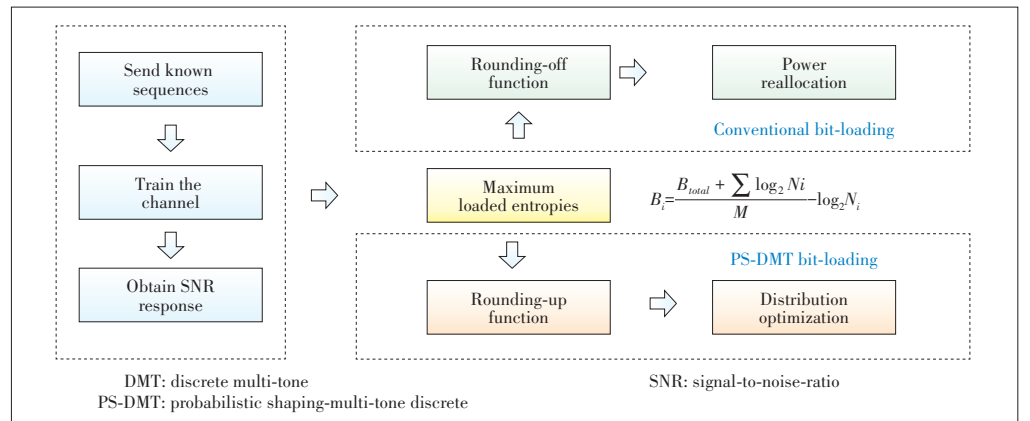
The proposed bit-loading metrology for probabilistic shaping-discrete multi-tone (PS-DMT) is illustrated in **Fig. 7**. A rounding-up function is performed for B_i to obtain standard constellations for QAM- N noted as B_i^* , and N denotes the constellation number. The corresponding bit-to-symbol mapping is performed as $N = 2^{B_i^*}$. Finally, the optimization problem must be solved to search for the distribution of the 1D PAM signals. The optimization object is to obtain the 1D probability distribution with entropy that is most close to $B_i/2$. These distributions are subject to the MB equation $P =$

$e^{-\lambda x}$ with a variable. In addition, the MB distributions require large-length block to perform PS coding. Here, we use Geometric Huffman Coding (GHC) to match dyadic distributions to MB ones.

The experiment investigation of 112 Gbit/s optical interconnection is carried out using a VCSEL-MMF link. In the optical B2B link case, the bit-loading results for conventional DMT are shown as the red points in **Fig. 8**, while the bit-loading results for the proposed PS-DMT (MB) scheme are plotted as blue points. Entropies of dyadic shaped PS-DMT are marked by blue lines. The adapted SNR response by power reallocation is measured by sending multi-tone quadrature phase shift keying (QPSK) signals with reallocated power, plotted as pink line in Fig. 8. In fact, the adapted response is not precisely matched to loaded bit numbers for conventional DMT. Fortunately, PS-DMT can get rid of this deviation, by directly adapting entropies to channel response. The corresponding optimized distributions for 12th and 66th subcarriers are shown in Fig. 8a, respectively. It should be noted here that it is usually impractical to implement the ideal bit-loading number (typically a decimal) perfectly. Only specific PS coding can be



▲ **Figure 6.** Bit error ratio (BER) curves of 75 Gbit/s PAM-8 signals.

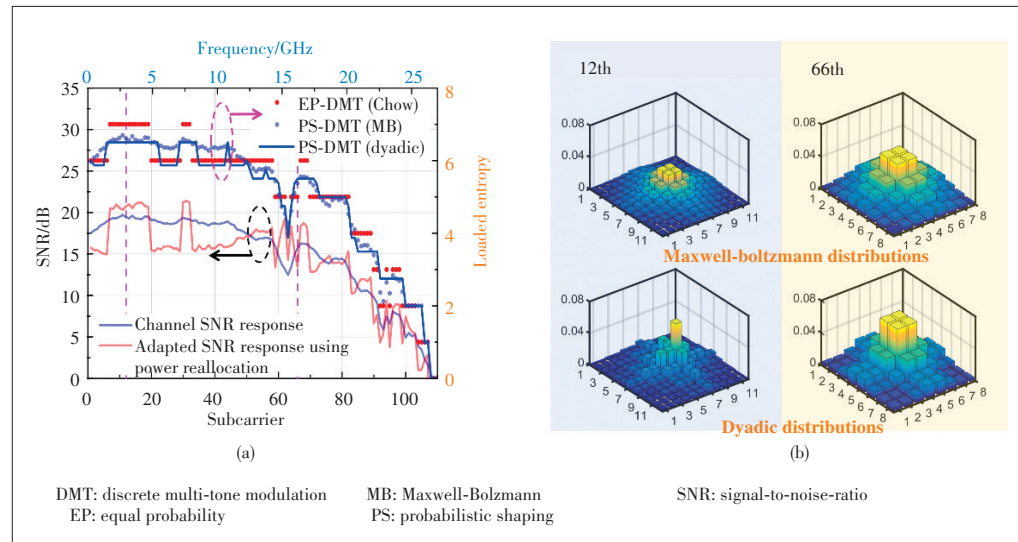


▲ **Figure 7.** Bit loading metrologies of conventional DMT and proposed PS-DMT schemes.

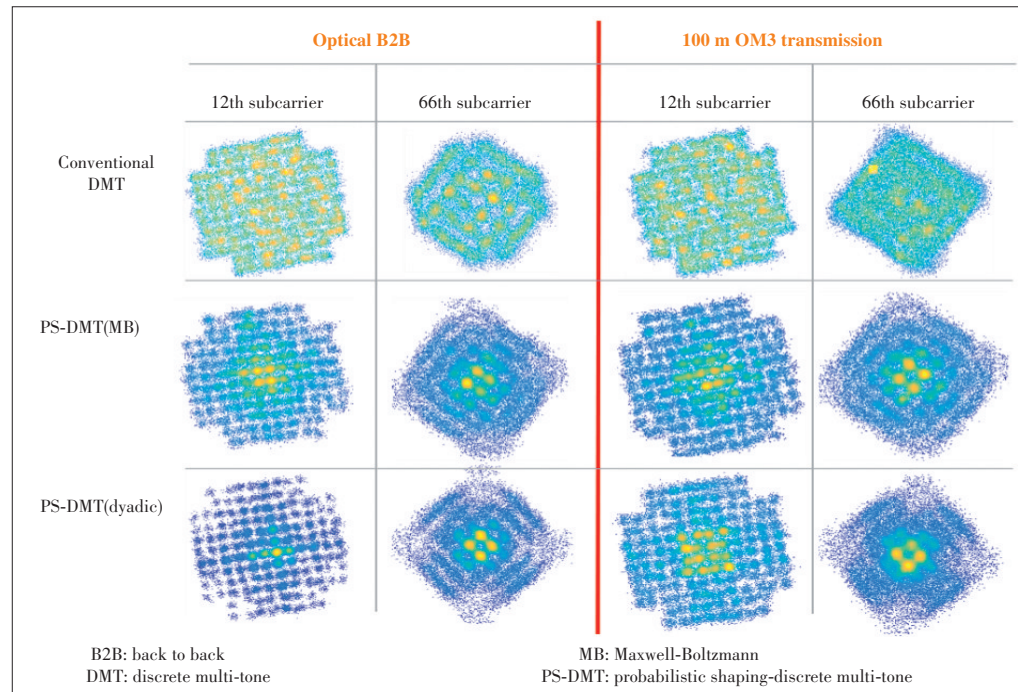
used to approach it maximally. A distribution matcher (DM) using dyadic distributions is used to approach the ideal bit-loading number approximately. The matched dyadic distributions are also inserted in Fig. 8b.

With an optical power of 3.5 dBm for the optical B2B case, the demodulated constellations for both DMT and PS-DMT at the receiver are plotted as shown in Fig. 9. Constellations are drawn for two selected typical subcarriers (12th and 66th). In addition, the right side of Fig. 9 shows the constellations of these typical subcarriers after 100 m OM3 fiber transmission with a received optical power of 3.3 dBm. The improvement in the signal quality related to the PS shaping gain can be observed visually by the clearer constellations that appear after PS when compared with those obtained before PS. It can also be seen that the constellations become clearer after PS. This occurs because the Euclidean distances between the symbols are broadened, with more symbols being gathered at the center, when the average power is fixed. After 100 m OM3 fiber transmission, however, the channel-efficient bandwidth is reduced, and more bits are loaded on the lower-frequency subcarriers. As a result, the constellations are much noisier after the 100 m transmission. Additionally, the shaped constellations are obviously clearer, with more bits being allocated at their centers. However, after transmission, the received constellations always rotate because of the fluctuating phase response.

The total general mutual information (GMI) values of a DMT symbol are then calculated with varying received optical power values, as shown in Fig. 10. Without noise and signal distortion, the total GMI value equals to the loaded bit numbers of a DMT symbol (552 in this experiment). As the optical power de-

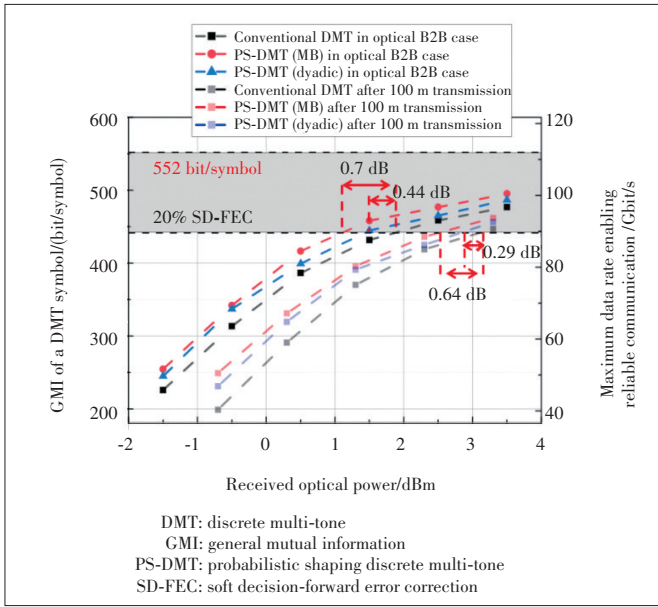


▲ Figure 8. Experimental bit loading results for DMT and PS-DMT in the optical B2B case: (a) bit-loading results for conventional DMT, PS-DMT and PS-DMT (dyadic); (b) shaped probability distributions of two typical subcarriers (22th and 66th subcarriers) for PS-DMT.



▲ Figure 9. Constellations of 12th and 66th subcarriers for optical B2B case (3.5 dBm received power) and after 100 m OM3 fiber transmission (3.3 dBm received power).

creases, the total GMI also decreases rapidly because of the low SNR. In the optical B2B case, with an optical power of more than 1 dBm, the SNR is sufficient to keep the GMI value stable. However, for optical powers below 1 dBm, any reduction in the SNR also reduces the GMI values of high-density constellations, which results in sharply reduced GMI values. After 100 m OM3 fiber transmission, more bits are allocated to the lower-frequency subcarriers because the bandwidth is more constrained. In this case, when the optical power decreases, the higher-density constellations with larger bit-loading numbers suffer greater



▲ Figure 10. Experimental GMI values and data rate of reliable communication under various received optical powers.

GMI reduction than those in the optical B2B case. To enable error-free signaling at 112 Gbit/s with 20% SD-FEC, power sensitivity gains of 0.7 dB and 0.64 dB can be obtained for the optical B2B case and the 100 m OM3 fiber transmission case, assisted by PS-DMT (MB). For PS-DMT (dyadic), power sensitivity gains are 0.44 dB and 0.29 dB for optical B2B and 100 m transmission. The corresponding enabling data rate for reliable transmission is calculated by the multiplication of GMI value and the symbol rate.

3 Advanced DSP at the Receiver

DSP embedded at the receiver side mainly includes post equalization, decoding and decision. Equalization aims to recover signals from severe ISI and noise. Decoding is always performed to correct bit errors, combined with pre-coding at the transmitter. Then advanced decision methods are in attempts to obtain reduced BER when nonlinear distortion occurs. Apart from the tradition DSP methods like FFE, DFE and MLSE, machine-learning assisted DSP recently exhibits improved performance. It can be used for formats identification^[46–47], system monitoring^[48–49] as well as optical signal receiving. For signal receiving technologies, machine learning can be utilized in equalization and decision for achieving distinguished performance^[50–55].

For the equalization part, the progressive support vector machine (SVM) algorithm has been applied to reduce nonlinear inter-subcarrier intermixing in coherent optical OFDM^[50]. Moreover, artificial neural network can also be embedded in equalization part, to reduce the error vector magnitude (EVM) of constellations^[51]. As for the decision part, machine-learning techniques can adaptively learn the optimal decision line for

obtaining the lowest BER^[52–55]. Related works mainly focus on QAM signals, and utilized SVMs to mitigate the phase noise. Therefore, the ML method is quite a viable solution for solving the nonlinearity problem (Kerr nonlinearity or modulation nonlinearity), which is the key issue in optical communications. For short reach optical communications, modulation nonlinearity becomes very serious for advanced modulation formats like QAM, PAM, CAP and DMT which are all sensitive to the linearity, both for direct modulation of VCSEL and for external modulation of silicon modulator.

3.1 SVM for QAM Decision

High-order QAM is an efficient format for increasing the transmission capacity due to its high spectral efficiency. However, such dense constellations make QAM signals very sensitive to nonlinear distortion. When nonlinear distortion bothers, the decision boundary can no longer be a simple straight line for obtaining a better BER performance. To deal with it, we propose several SVM multi-classification methods to obtain adaptive nonlinear decision boundary for QAM decision, based on one versus rest (OvR) and binary tree (BT) structure. Different classification methods have different performance in terms of classification precision and complexity.

The OvR SVM is to generate a hyper plane between a class of samples and the remaining multi-class samples, and to realize multi-class recognition. Therefore, if it is an N classification problem, then N SVMs ($N>2$) are required to perform classification. For example, if the OvR SVM scheme is used for deciding QAM-8 sequences, the data are divided into two categories for every SVM classifier. Consequently, it requires eight SVMs to decide eight symbols of QAM-8 signal.

On the other hand, BT structure can be used to reduce the number of SVM classifier for QAM- N signal decision. Starting from the root node, the category is divided into two subclasses, and then the two subclasses are further divided, until the subclass contains only one category. Here, we employ three different BT-based SVM classification schemes including the binary encoding (BE), the constellation rows (CR) and columns, and the in-phase and quadrature components (IQC). Since the QAM constellation mapping can be realized through binary encoding, the multi-classification based on BE can be performed to decide every bit information of QAM- N . Fig. 11a shows the training model for 16-QAM, where the hyper plane generated by each SVM does not take all the training set data into consideration, which can effectively reduce the training time. When testing, only four SVMs is required to perform decision, as shown in Fig. 11b. Another BT-based decision scheme can be realized by rows-and-columns classification. Most of the constellations are rectangle, except for 32-QAM, 128-QAM, etc., thus the label feature according to rows and columns is reasonable. Besides, according to the IQC, the QAM signal can be regarded as two PAM signals, which means splitting a binary tree training model into two binary trees to reduce the SVM numbers.



In terms of complexity during training and decision processes, OvR-based SVM is the worst, with about 6 to 8 times the number of support vectors to others. While, the number of SVMs for decision is similar for the remaining three methods. In detail, the IQ-based decision method only requires about one-third of the SVM number compared with other methods, which benefits simple implementation. It is worth mentioning that one should carefully evaluate the requirement regarding different application scenarios (particularly



Complexity	OvR	BE	CR	IQC
SVM number for training	2 294	2 177	2 177	804
Support vector number ($\times 10^5$)	11.61	1.367	1.376	1.997
Average SVM number for testing	2 294	492	492	492

BE: binary encoding
OvR: one versus rest
IQC: in-phase and quadrature components
CR: constellation rows

different modulation formats and different nonlinear distortions) when choosing a specific classification method for SVM machine learning detection.

3.2 SVM for PAM Decision

As discussed above, BT-based SVM decision can efficiently improve transmission performance, along with considerable complexity. In this section, BT-based SVM decision is employed to mitigate modulation nonlinearity in PAM-modulated systems, specifically including VCSEL-MMF as well as Si-MRM optical links. In detail, as for VCSEL-based links, modulation nonlinearity is mainly caused by the nonlinear LI response of VCSEL. While for SMF links based on Si-MRM, modulation nonlinearity is mainly induced due to the Lorentz spectral shape of the Si-MRM. When processing nonlinearly-distorted PAM signals, SVM-based decision scheme can generate adaptive boundaries for the obtained improved decision performance.

3.2.1 Optical Interconnection Link Based on VCSEL-MMF

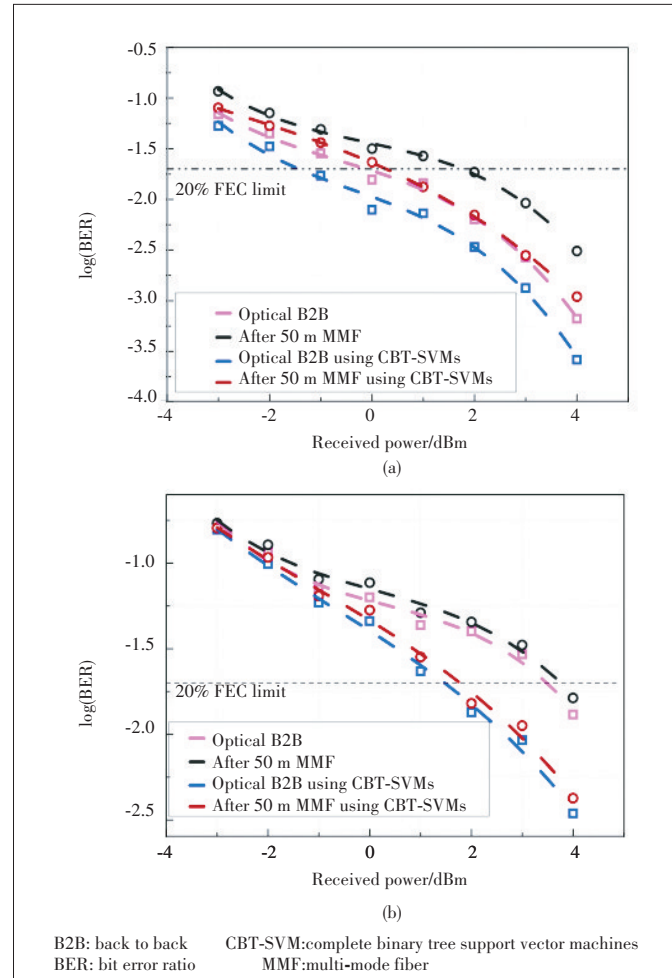
To evaluate the performance of binary tree-support vector machines (BT-SVM) decision for PAM signals, the experiment based on VCSEL-MMF system has been carried out at a bit rates of 60 Gbit/s for PAM-4 and PAM-8. With the received optical power being manually attenuated, BER curves of the PAM-4 and PAM-8 signals are plotted in **Fig. 13**, where 20%-overhead FEC is assumed for investigating the receiver sensitivity. Consequently, there are about 1 dB and 2 dB receiver sensitivity improvements with the use of complete binary tree-support vector machines (CBT-SVMs) classifier, respectively for PAM-4 and PAM-8 signals. Improvement for PAM-8 is clearly better compared with PAM-4 due to its doubled modulation levels, which makes it more sensitive to modulation nonlinearity distortion as we expected.

Moreover, it is essential to quantitatively analyze the machine learning performance of CBT-SVMs classifier for PAM signals under different modulation nonlinearities. **Fig. 14** shows the SVM machine learning performance of CBT-SVMs classifier with the increase of eye-linearity (increase of modulation nonlinearity distortion). Here we use 7% overhead FEC threshold for investigating receiver sensitivity. The sensitivity changes almost linearly with eye-linearity. Smaller power (receiver sensitivity) can be obtained by using BT-SVMs classifier, which has a smaller slope as shown in **Fig. 14**. The smaller slope means an increased sensitivity gain with the increase of eye-linearity. The very severely distorted eye diagram with an eye-linearity of 1.72 is also presented in **Fig. 14**. A sensitivity gain of 2.5 dB is obtained by the proposed CBT-SVMs at eye-linearity of 1.72. One can expect larger gain for larger eye-linearity.

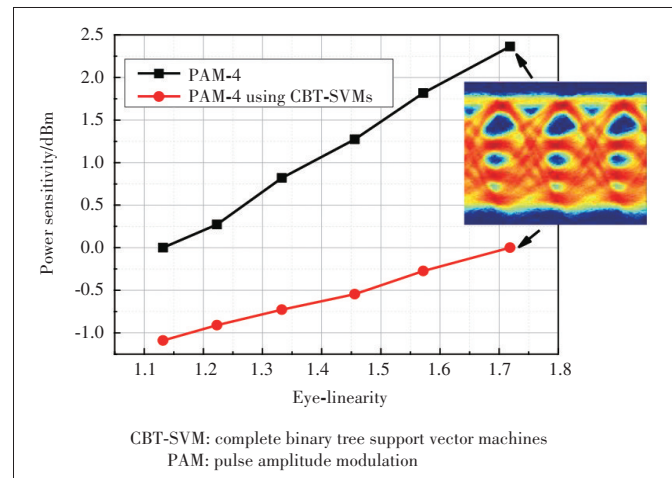
3.2.2 Optical Interconnection Link Based on Si-MRM-SMF

Besides VCSEL-based optical link, the Si-MRM will induce severe modulation nonlinearity. Si-MRM exhibits the attractive features of compact footprint, high modulation speed, and low energy consumption. However, the high Q factor of the Si-MRM makes it very sensitive to resonance drift, which means it may cause serious damage to the signal. Here, we model a PAM-4

modulated optical interconnections system, with different Si-MRM resonator wavelengths as shown in **Fig. 15**. 100 Gbit/s PAM-4 signals are generated with a bandwidth of 50 GHz. The



▲ **Figure 13.** BER curves by using conventional hard decision and proposed CBT-SVM for (a) PAM-4; (b) PAM-8.



▲ **Figure 14.** Optical power sensitivity versus different eye-linearity value by using conventional hard decision and proposed CBT-SVM.

bandwidths of digital-to-analogue-converter and analogue-to-digital-converter (DAC/ADC) as well as optical link in the simulation system are set at 40 GHz. A PD with 1 A/W sensitivity is used to detect the optical signal. The thermal noise of the PD is set to be 10^{-12} A/ $\sqrt{\text{Hz}}$.

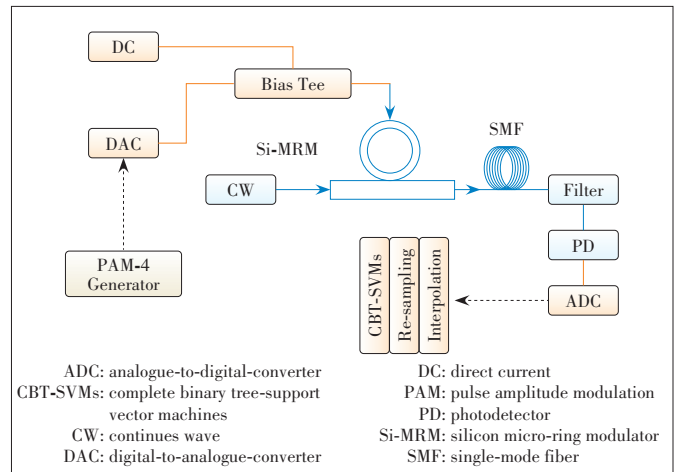
In the simulation, PAM-4 signals are biased at 0.7 V, and V_{pp} of PAM-4 signals is 1 V. The Si-MRM linear operation range is very narrow, and the temperature drift will affect the Si-MRM transmission which leads to the degradation of the PAM-4 modulation. To quantitatively analyze the influence of wavelength drift for the PAM-4 signals, we use a term of level-deviation (LD).

Fig. 16 denotes the LD as a function of the Si-MRM resonant wavelength, which refers to the wavelength drift. It can be seen that even very slight wavelength drift will lead to deteriorated LD. **Fig. 17** shows the sensitivity gain as a function of LD. The power sensitivity gain means the reduction of the received power requirement for achieving error free (assuming 7% overhead FEC) by using CBT-SVMs with respect to hard decision. The black line in Fig. 17 represents the forward wavelength drift and the red one refers to the reverse wavelength drift. From Fig. 17, generally, the sensitivity gain increases along with the increase of the LD (absolute value) which means larger gain due to machine learning detection for larger modulation nonlinearity distortion. The largest sensitivity gain is about 2.7 dB. Fig. 17 also gives the sensitivity of CBT-SVMs at different LDs as shown by the blue and yellow dash-dot curves. The sensitivity powers for all the cases with different LDs are comparably stable with less than 3 dB fluctuation. This also indicates the very useful capability of machine learning detection for stabilized PAM-4 modulation without wavelength drift control at the transmitter side.

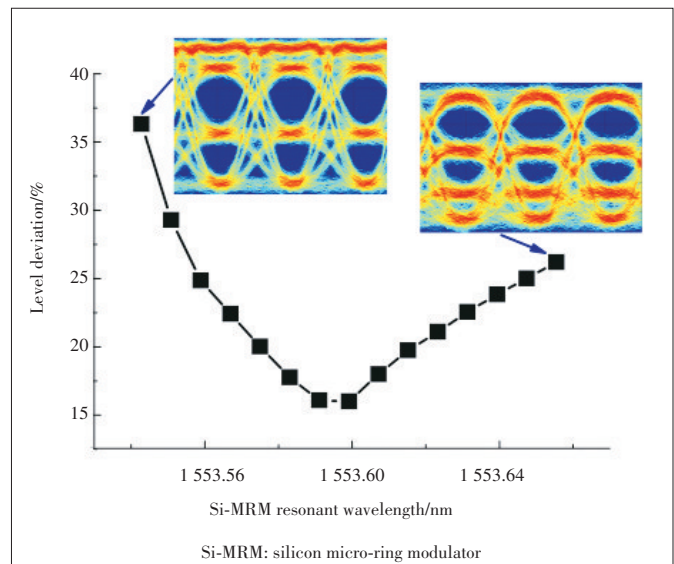
3.3 Recurrent Neural Network (RNN) for PAM

For PAM modulated optical interconnections system, another distortion degrading performance is eye skewing. **Fig. 18** illustrates the eye-diagrams of PAM-4 signal before and after VCSEL modulation. The eye-skewing always occurs when the signal is modulated directly on laser. The potential reason behind the skewed eye is the variant rise times with different amplitudes.

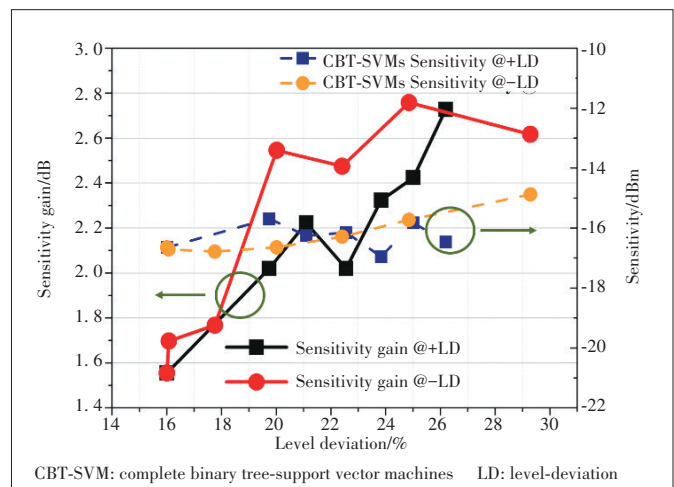
Therefore, we propose an RNN-based demodulator to deal with the problem of system performance degradation caused by eye skewing in VCSEL-based PAM system. Compared with other neural network methods, RNN adds a feedback mechanism to the network architecture, which may adaptively learn the skewing during VCSEL modulation. Here, we employ long short-term memory (LSTM) scheme to decode eye-skewing PAM-4 signal. The structure of LSTM is indicated in **Fig. 19**. LSTM is split into four parts: unit status, forgetting gates, input gates, and output gates. The unit state of LSTM is primarily used to transmit information from the previous unit to the next unit. The main function of the forgetting gate is to receive the information of the previous neuron and the input information of current neuron, and at the same time determine how



▲ **Figure 15.** Simulation setup of Si-MRM based optical interconnects system.



▲ **Figure 16.** The level-deviation curve as a function of the wavelength drift. Inserted pictures are the eye diagrams at different wavelengths.

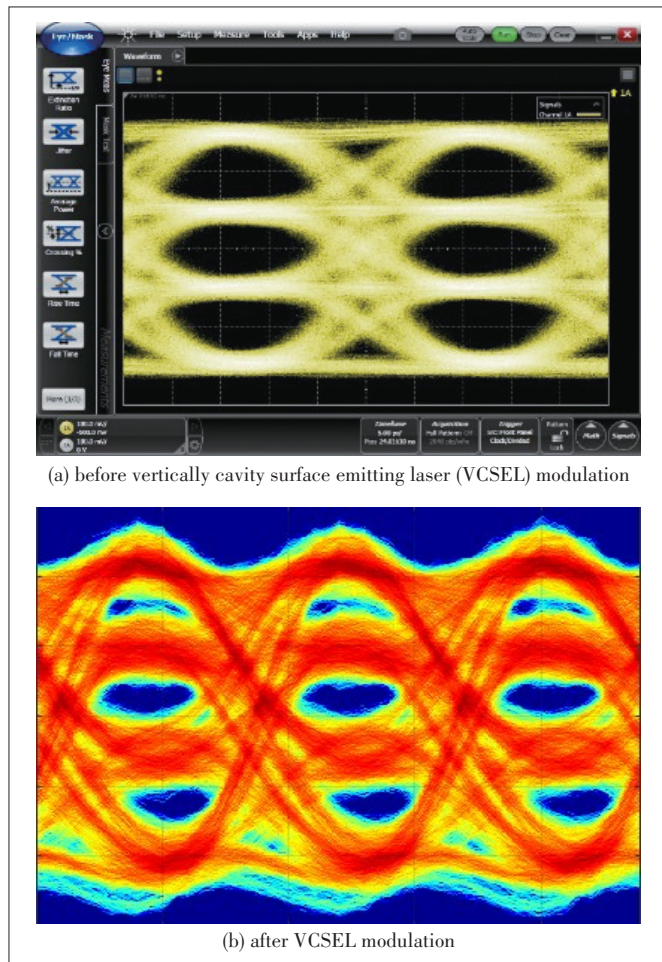


▲ **Figure 17.** Sensitivity gain (solid curves) and receiver sensitivity power (dashed curves) for the machine learning detection at different LDs.

much the state of the last neuron is forgotten. The function of the input gate is mainly to cooperate with a tanh function to control the input of new information. The output gate activates the neuron state information through the tanh function to obtain the output result. The sigmoid in the structure produces coefficients that control the amount of information filtered. In this way, RNN completes the function of information transmission, forgetting, and memory.

In this experiment, the RNN we used was a 10-layer neural network. The number of data for this trial was 20 000. We used 20% as the training set for the neural network, and the remaining 80% was used to test the performance of the RNN in the system.

Fig. 20 shows the BER curves of PAM-4 by using conventional hard decision (HD), CBT-SVM decision and LSTM. BER curves by using conventional HD and CBT-SVM exhibit a linear increasing trend as the optical power decreases, but RNN shows different phenomena around 1 dBm optical power. For RNN by means of sequence mining, the BER performance of the system is stabilized. In general, compared with hard decisions, RNN can bring about 2 dB power sensitivity improvement to the system, slightly better than CBT-SVMs.

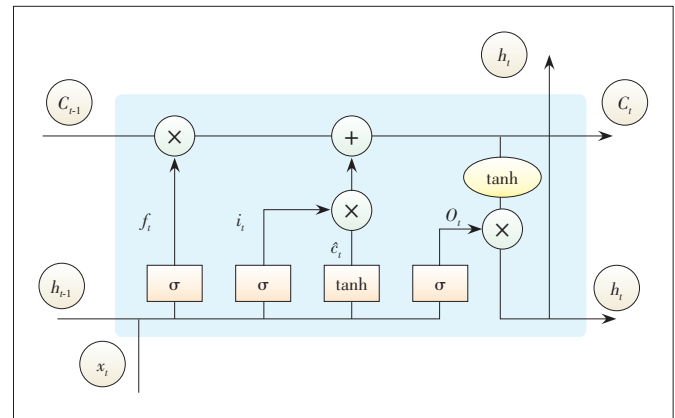


▲ Figure 18. PAM-4 eye-diagram.

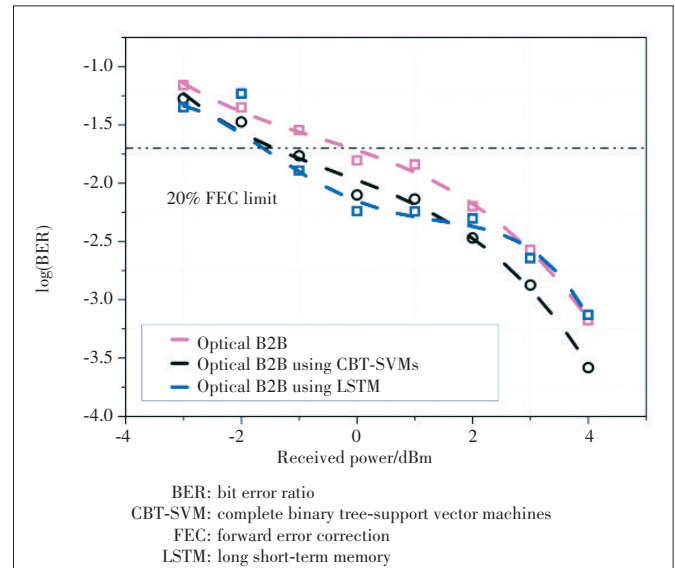
3.4 K-Means Clustering for Soft Decision in PAM System

Above mentioned classification methods require training the system through prior-known sequences, resulting in enhanced complexity. To deal with it, a K-means machine learning method assisted signal receiver including both equalization and soft decision is proposed for VCSEL-based PAM-4 optical interconnection. Mean values and noise variances of four levels can be obtained through K-means clustering, without training using prior-known sequences. According to the learned level means, least mean square (LMS) equalization based on the corresponding level-adapted sequences is expected with improved performance. Moreover, based on learned level means and variances, the precision of log-likelihood ratio (LLR) estimation can be enhanced, which leads to the improved performance of soft decision (SD).

The schematic diagram of proposed receiver is plotted as **Fig. 21**. K-means clustering is employed after resampling, to adaptively learn the mean values and noise variances for individual levels, denoted as L_n and σ_n . The upper blue box is the



▲ Figure 19. Long short-term memory.



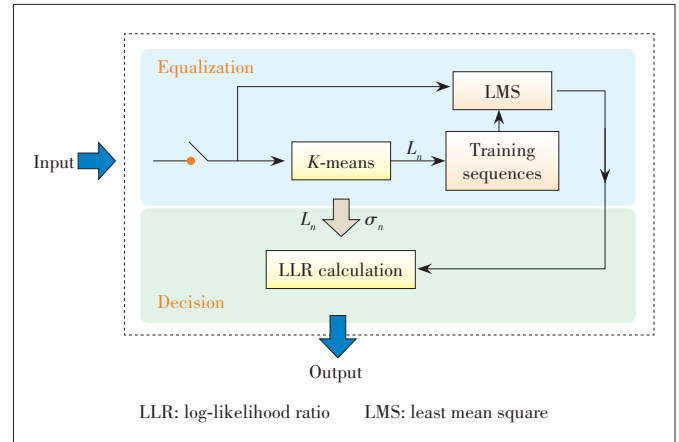
▲ Figure 20. BER curves.

equalization part. Tap coefficients are trained through LMS process. Different from conventional LMS, the levels of training sequence are altered adaptively based on learned level means through K-means approach. The lower box is the decision part. With learned level L_n ($n=1, 2, 3, 4$) and noise variances σ_n , the corresponding LLRs can be calculated. Because it takes level nonlinearity (affecting L_n) and level-dependent noise (affecting σ_n) into consideration, the proposed SD is expected with improved decision precision.

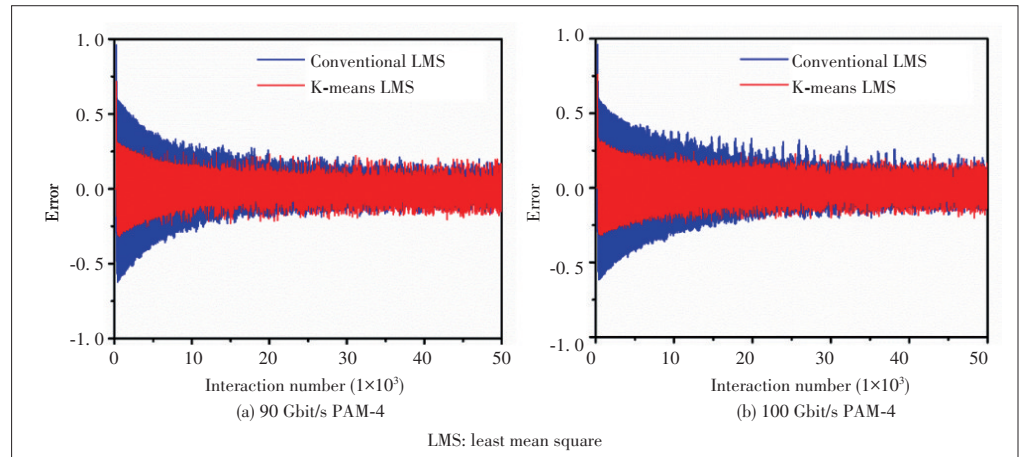
Experiments have been carried out for PAM-4 modulation over a VCSEL-MMF optical interconnection link to verify the performance of the proposed K-means adaptive receiver. To realize 90 Gbit/s and 100 Gbit/s PAM-4 signaling, the sampling rate of AWG is set at 45 Gsa/s and 50 Gsa/s, respectively. The DC bias current is fixed at 15 mA in the experiment. To process the PAM-4 sequences, the sampled signals by DSO have to be re-sampled to 1 sample/symbol.

Then the samples are sent into LMS for equalization, and the normalized errors are recorded for every interaction. Error convergence curves for 90 Gbit/s and 100 Gbit/s PAM-4 in optical B2B case are shown in Figs. 22a and 22b respectively, with received optical power of 3 dBm. The residual errors are mainly induced by random noise and residual ISI. And after 100 m transmission, residual errors are higher than the case of optical B2B. It can also be seen that, for K-means LMS (red line), errors converge faster than conventional LMS (blue line). Because for conventional LMS, the levels are mismatched between training sequences and received samples, which deteriorates the convergence performance. This result indicates that lower numbers of interaction can be achieved for K-means LMS.

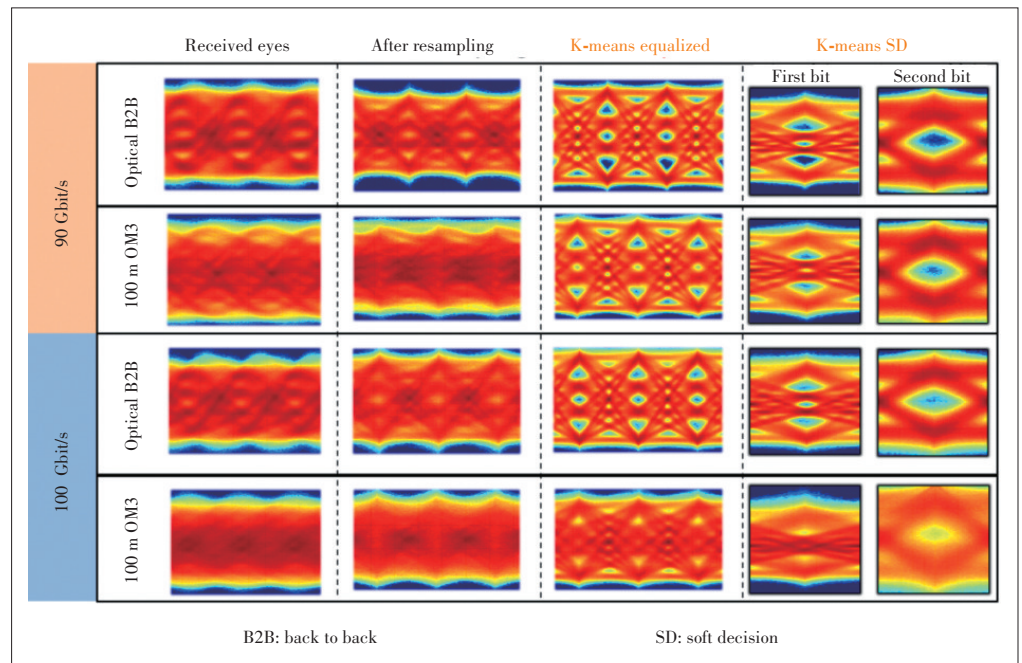
The eye-diagrams of original sequences, equalized ones as well as LLRs are depicted in Fig. 23. It can be seen that 90 Gbit/s PAM-4 in optical B2B case shows clearer eyes than others, which is because less ISI occurs in this circumstance. While for



▲ Figure 21. Proposed K-means receiver including equalization and soft decision.



▲ Figure 22. LMS errors during interaction for optical B2B case.



▲ Figure 23. Results of K-means equalization and SD for 90 Gbit/s and 100 Gbit/s PAM-4.

100 Gbit/s signaling after 100 m OM3 fiber transmission, the eye severely closes with undistinguished four levels. Resampling is performed to obtain 1 sample/symbol sequences, whose eyes are depicted in the second column in Fig. 9. By using K-means equalization, the corresponding eyes are opened obviously, with observable four levels, as shown in the third column. While in the case of 100 Gbit/s and 100 m transmission, the corresponding eye is still noisy, which is because the residual ISI cannot be effectively eliminated. At the fourth column, the two LLR tributaries are demonstrated by means of eye diagram.

3.5 K-Nearest Neighbor (K-NN) for CAP Decision

Apart from PAM, CAP also shows weak tolerance to modulation nonlinearity. Modulation nonlinearity mainly results in irregular constellations for CAP. Consequently, conventional hard decision with the straight decision line cannot obtain considerable BER without considering the modulation nonlinearity. Here, we experimentally investigate the machine learning for nonlinearity mitigation by using K-NN algorithm in 32 Gbit/s CAP system. The basic principle of K-NN algorithm can be intuitively understood from Fig. 24 (right picture). Firstly, the training signal has to be sampled, shown as blue scatters in constellation diagram. Then when the signal (red point) is detected, its distances to training signals are required to be calculated. And the shortest K distances with responding training samples are selected, to get the constellation label which contains the majority of these training samples (L3 in Fig. 24). It is intuitive that when linear distortion like Gauss white noise occurs, the K-NN algorithm cannot decrease BER compared with the hard decision. However, when constellation is distorted nonlinearly, K-NN is desired to have better performance than the hard decision. The constellation of 32 Gbit/s CAP signal in optical B2B with received optical power at -2 dBm is shown in Figs.

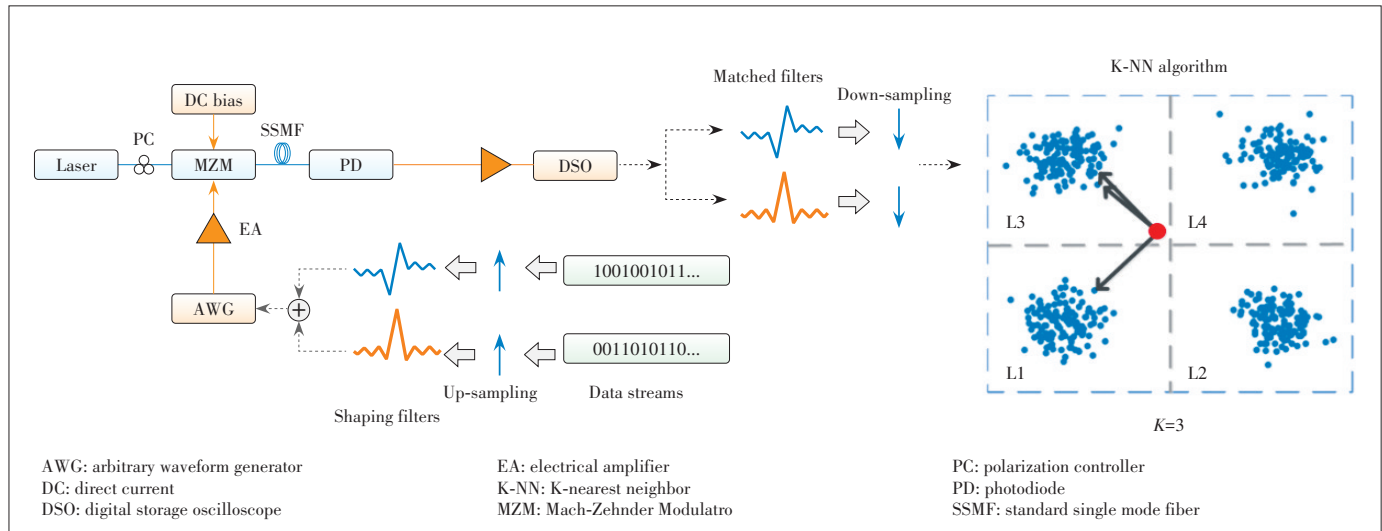
25a and 25b. And the constellation of CAP after 7 km transmission with the same received power is noisier than B2B case, as shown in Fig. 25c.

The algorithm complexity of K-NN mainly comes from the selection of K shortest distances. The BER curves of 32 Gbit/s CAP is shown in Fig. 26. The K-NN can reduce BER by more than 20 dB for the optical CAP signal, when signal is mainly distorted nonlinearly. However, after 7 km transmission, the signal quality improvement of K-NN is not so obvious when processing signal with low SNR, which is because signal quality is mainly influenced by linear noise.

3.6 SVM for DMT Carrier-by-Carrier Decision

For DMT modulation, nonlinear distortion behaves differently for different subcarriers due to the different bit allocation and SNR. Consequently, every subcarrier suffers different distortions, even for those subcarriers who are loaded by same-order QAM modulation. Thus, adaptive decision is required to perform for every subcarrier. As mentioned above, the SVM-based decision method has the advantage of adaptive decision boundary. Thus, efficient mitigation of the nonlinear distortion for DMT system can be expected through SVM-assisted carrier-by-carrier decision.

An experiment of 112 Gbit/s DMT signaling is performed on VCSEL-MMF optical link. At transmitting side, bit loading and power reallocation are realized according to channel frequency response. The measured channel responses in the case of optical B2B and 100 m OM3 transmission are plotted in Fig. 27, as well as corresponding bit loading results. At the receiving side, demodulation of every subcarrier is performed, as well as SVM decision. Constellations of some typical subcarriers are depicted in Fig. 28. A total of 40 320 DMT symbols are sent in the experiment and 20% of the data has been used for training. The decoded signal in binary sequence after



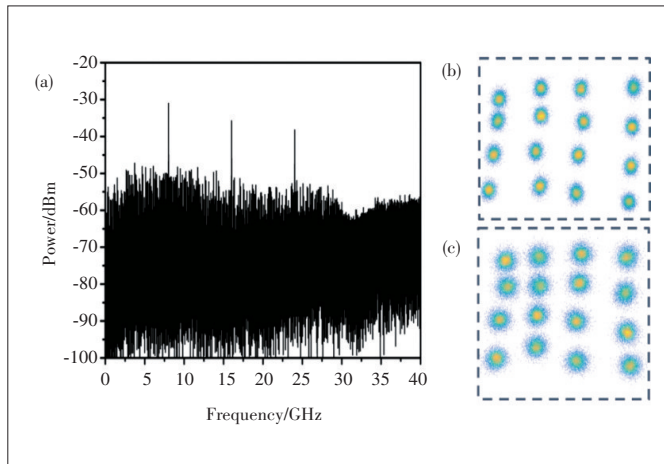
▲ Figure 24. Experimental setup of optical carrier-less amplitude phase modulation (CAP) transmission system (left) and principle of K-nearest neighbour algorithm (right).

de-mapping is then off-line processed for the BER measurement.

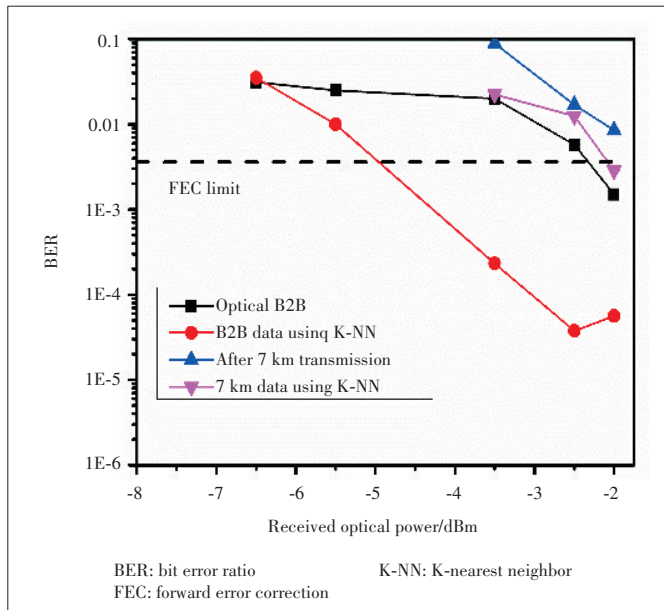
Assisted with SVM decision, BER result of DMT system is shown in **Fig. 29**, where significant reduction of BER has been achieved by using SVM detection compared with that using convention detection. Error-free operation has been achieved for B2B case at 7% FEC and 100 m MMF transmission case at 20% FEC.

4 Conclusions

The above mentioned is our current works about advanced DSP methods for optical interconnection systems, mainly in-



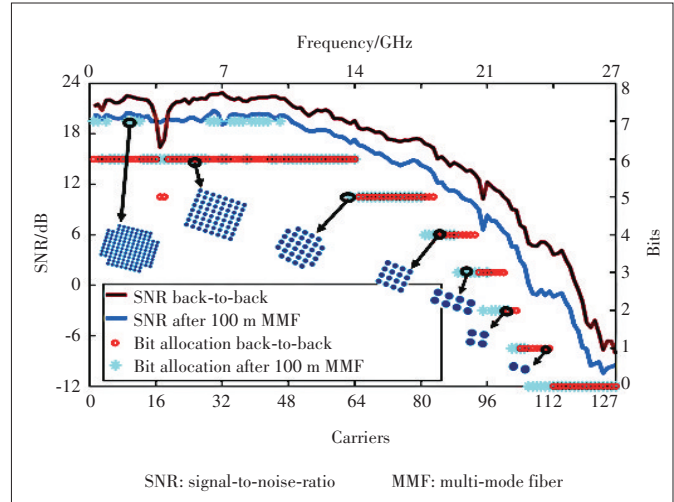
▲ **Figure 25.** (a) Electrical spectrum of 32 Gbit/s carrier-less optical amplitude phase (CAP) modulation signal; (b) constellation of 32 Gbit/s CAP in optical B2B case; (c) constellation of 32 Gbit/s CAP after 10 km transmission.



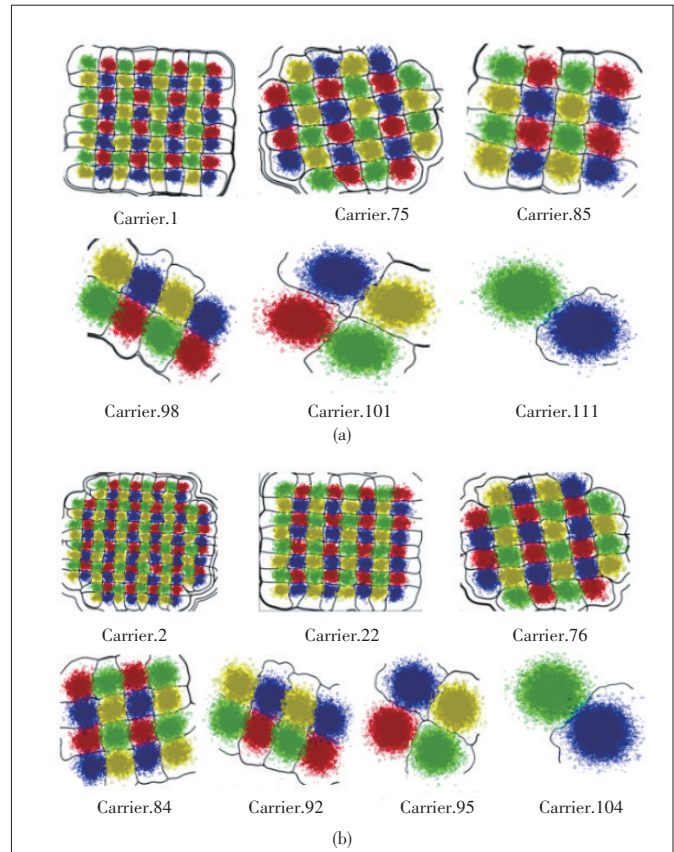
▲ **Figure 26.** BER curves of 32 Gbit/s optical carrier-less amplitude phase modulation (CAP) by using conventional hard decision and K-NN decision.

cluding PS coding for transmitter side and machine-learning based DSP for receiver side.

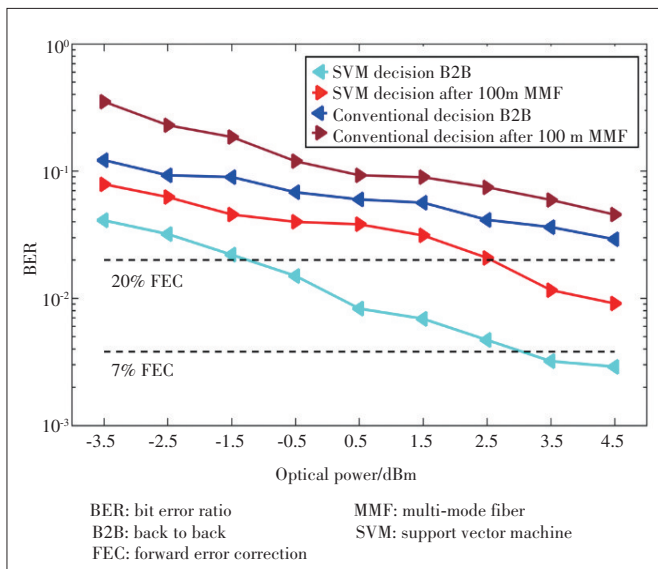
In the perspective of PS, we proposed dyadic shaping for PAM system. Up to 1.74 dB SNR gain can be achieved theoretically by dyadic PS for PAM-8. Proof-of-concept experi-



▲ **Figure 27.** SNR response and bit-allocation of multi-tone modulation (DMT) for back to back (B2B) and 100 m MMF.



▲ **Figure 28.** Constellations of some typical subcarriers for multi-tone modulation (DMT) as well as SVM-based decision boundaries for (a) optical B2B case and (b) 100 m MMF transmission.



▲ Figure 29. BER comparison between using SVM decision and conventional hard decision.

ments have been carried out over a VCSEL-MMF link. Assisted by SVM classifier, 100 m MMF transmission of PS-PAM-8 signals at 75 Gbit/s has been achieved. Energy efficient optical interconnection with 16% reduction of power consumption has been obtained by PS coding. On the other hand, channel-adaptive PS method has been proposed by using DMT modulation, combined with MB shaping. The proposed PS-DMT scheme can significantly improve the signaling capacity since two significant benefits are simultaneously utilized: 1) the shaping gain of PS for carriers with limited SNR; 2) the frequency-resolved continuous entropy loading for better fitting the channel frequency response. Proof-of-concept investigations have been carried out via both simulations and experiments, for MB and dyadic distributions. Data rate improvement of 5.68 Gbit/s was obtained theoretically using the PS-DMT with dyadic distributions. In addition, optical signaling was realized experimentally using a commercial VCSEL at 112 Gbit/s data rate. The 0.44 dB and 0.29 dB power gains have been achieved for optical B2B and 100 m OM3 fiber transmissions, by using dyadic shaping of DMT.

While for DSP methods in the receiver, we mainly employ the machine-learning algorithm to realize improved signal detection. In detail, for QAM signaling, we proposed a SVM multi-classification method to obtain adaptive nonlinear decision boundary. Different classification methods have been evaluated in terms of classification precision and complexity including OvR and BT structure. Among them, the in-phase and quadrature classification method only requires about one-third of the SVM number compared with other methods, which is much simpler for implementation. One should carefully evaluate the requirement regarding different application scenarios (particularly different modulation formats and different

nonlinear distortions) when choosing a specific classification method for SVM machine learning detection.

Besides QAM decision, we also use SVM method to mitigate the modulation nonlinearity in PAM-modulated VCSEL-MMF as well as Silicon MRM-SMF optical links. Compared with the published works, we firstly come up with the CBT structure multi-classes SVMs which are more suitable for PAM modulation. For VCSEL-based PAM system, there are about 1 dB and 2 dB receiver sensitivity improvements respectively for PAM-4 and PAM-8 signals with the use of CBT-SVMs classifier. The improvement for PAM-8 is clearly better compared with PAM-4 due to its doubled modulation levels, which makes it more sensitive to modulation nonlinearity distortion as we have expected. For silicon MRM-SMF optical link, the sensitivity versus different LDs are comparably stable with less than 3 dB fluctuation by using proposed CBT-SVM. It indicates the very useful capability of machine learning detection for stabilized PAM-4 modulation without wavelength drift control at the transmitter side. Moreover, we also propose an RNN-based demodulator to mitigate eye skew in VCSEL-PAM system. Compared with other neural networks, RNN adds a feedback mechanism to the network architecture, which can comprehensively consider the association of information before and after the sequence. In detail, compared with hard decisions, RNN can bring about 2 dB power sensitivity improvement to the system, slightly better than CBT-SVMs. Subsequently, a K-means assisted receiver including both equalization and soft decision is proposed for VCSEL-based PAM-4 optical interconnects. Through self-learning of mean values and noise variances of four levels, performance of LMS equalization as well as LLR-based SD has been improved. Besides, we also investigate the performance of machine-learning methods in CAP and DMT system. For CAP system, K-NN based decision is realized for mitigating nonlinear distortion. For DMT system, carrier-by-carrier decision based on SVM is performed, obtained significant reduction of BER.

The above-mentioned are proposed DSP methods and corresponding results. In brief, PS embedded in transmitter (Tx) side and machine-learning decision in receiver (Rx) side are realized. With assistance of such intelligent DSP approach, improved performances in terms of BER and capacity are achieved. Moreover, these methods can be not only utilized for short-reach link, but also extended for long-haul transmission. The challenges for practical application are mainly the cost of complexity with respect to the sensitivity gain. Presently, the complexity is still large and applications for short reach optical interconnection are difficult. However, for long haul, which is not so sensitive to power consumption and cost, complex DSP can be acceptable as long as the gain is large enough. For data center interconnection (DCI) applications ranging from tens of kilometers to hundreds of kilometers, there would be a good balance.

References

- [1] XU K, SUN L, XIE Y Q, et al. Transmission of IM/DD signals at 2 μ m wavelength using PAM and CAP [J]. *IEEE photonics Journal*, 2016, 8(5): 1 – 7. DOI: 10.1109/jphot.2016.2602080
- [2] KANEDA N, LEE J, CHEN Y K. Nonlinear equalizer for 112-Gb/s SSB-PAM4 in 80 - km dispersion uncompensated link [C]//Optical Fiber Communication Conference. Los Angeles, USA, 2017. DOI: 10.1364/ofc.2017.tu2d.5
- [3] PANG X D, OZOLINS O, GAIARIN S, et al. Experimental study of 1.55- μ m EML-based optical IM/DD PAM-4/8 short reach systems [J]. *IEEE photonics technology letters*, 2017, 29(6): 523 – 526. DOI: 10.1109/lpt.2017.2662948
- [4] WINZER P J, GNAUCK A H, DOERR C R, et al. Spectrally efficient long-haul optical networking using 112-Gb/s polarization-multiplexed 16-QAM [J]. *Journal of lightwave technology*, 2010, 28(4): 547 – 556. DOI: 10.1109/jlt.2009.2031922
- [5] KOIZUMI Y, TOYODA K, YOSHIDA M, et al. 1024 QAM (60 Gbit/s) Single-carrier coherent optical transmission over 150 km [J]. *Optics express*, 2012, 20 (11): 12508 – 12514. DOI:10.1364/oe.20.012508
- [6] SEIMETZ M, NOELLE M, PATZAK E. Optical systems with high-order DPSK and star QAM modulation based on interferometric direct detection [J]. *Journal of lightwave technology*, 2007, 25(6): 1515 – 1530. DOI: 10.1109/jlt.2007.896810
- [7] ARMSTRONG J, LOWERY A J. Power efficient optical OFDM [J]. *Electronics letters*, 2006, 42(6): 370 – 372. DOI:10.1049/el:20063636
- [8] YI X W, SHIEH W, TANG Y. Phase estimation for coherent optical OFDM [J]. *IEEE photonics technology letters*, 2007, 19(12): 919 – 921. DOI: 10.1109/lpt.2007.897572
- [9] SHIEH W. PMD-Supported Coherent Optical OFDM Systems [J]. *IEEE photonics technology letters*, 2007, 19(3): 134 – 136. DOI: 10.1109/lpt.2006.889035
- [10] LEE S C J, RANDEL S, BREYER F, et al. PAM-DMT for intensity-modulated and direct-detection optical communication systems [J]. *IEEE photonics technology letters*, 2009, 21(23): 1749 – 1751. DOI: 10.1109/lpt.2009.2032663
- [11] TAKAHARA T, TANAKA T, NISHIHARA M, et al. Discrete multi-tone for 100 Gb/s optical access networks [C]//Optical Fiber Communication Conference. San Francisco, USA, 2014. DOI: 10.1364/OFC.2014.M2I.1
- [12] ZHONG K P, ZHOU X, GUI T, et al. Experimental study of PAM-4, CAP-16, and DMT for 100 Gb/s short reach optical transmission systems [J]. *Optics express*, 2015, 23(2): 1176. DOI: 10.1364/oe.23.001176
- [13] NADAL L, SVALUTO MOREOLO M, FABREGA J M, et al. DMT modulation with adaptive loading for high bit rate transmission over directly detected optical channels [J]. *Journal of lightwave technology*, 2014, 32(21): 4143 – 4153. DOI: 10.1109/jlt.2014.2347418
- [14] SUN L, DU J B, HE Z Y. Multiband three-dimensional carrierless amplitude phase modulation for short reach optical communications [J]. *Journal of lightwave technology*, 2016, 34(13): 3103 – 3109. DOI: 10.1109/jlt.2016.2559783
- [15] OLMEDO M I, ZUO T J, JENSEN J B, et al. Multiband carrierless amplitude phase modulation for high capacity optical data links [J]. *Journal of lightwave technology*, 2014, 32(4): 798 – 804. DOI: 10.1109/jlt.2013.2284926
- [16] INGHAM J D, PENTY R V, WHITE I H, et al. 40 Gb/s carrierless amplitude and phase modulation for low-cost optical data communication links [C]//Optical Fiber Communication Conference/National Fiber Optic Engineers Conference. Los Angeles, USA, 2011. DOI: 10.1364/ofc.2011.othz3
- [17] SUN L, DU J B, YOU Y, et al. 45-Gbit/s 3D-CAP transmission over a 16-GHz bandwidth SSMF link assisted by wiener filtering [J]. *Optics communications*, 2017, 389: 118 – 122. DOI: 10.1016/j.optcom.2016.11.055
- [18] SZCZERBA K, WESTBERGH P, KAROUT J, et al. 4-PAM for high-speed short-range optical communications [J]. *Journal of optical communications and networking*, 2012, 4(11): 885 – 894. DOI: 10.1364/jocn.4.000885
- [19] MENA P V, MORIKUNI J J, KANG S M, et al. A simple rate-equation-based thermal VCSEL model [J]. *Journal of lightwave technology*, 1999, 17(5): 865 – 872. DOI: 10.1109/50.762905
- [20] WANG L, QIU Y, XIAO X, et al. 24-Gb/s PAM-4 over 150-km SSMF using a driverless silicon microring modulator [C]//Asia Communications and Photonics Conference. Shanghai, China, 2014. DOI: 10.1109/jssc.2006.884342
- [21] BOGAERTS W, HEYN P D, VAERENBERGH T V, et al. Silicon microring resonators [J]. *Laser & photonics reviews*, 2012, 6(1): 47 – 73. DOI: 10.1002/lpor.201100017
- [22] RUBSAMEN M, WINZER P J, ESSIAMBRE R J. MLSE receivers for narrow-band optical filtering [C]//Optical Fiber Communication Conference & the National Fiber Optic Engineers Conference. Anaheim, USA, 2006. DOI: 10.1109/OFC.2006.215452
- [23] STOJANOVIC N, HUANG Y, HAUSKE F N, et al. MLSE-based nonlinearity mitigation for WDM 112 Gbit/s PDM-QPSK transmissions with digital coherent receiver [C]//Optical Fiber Communication Conference. Los Angeles, USA, 2011. DOI: 10.1364/OFC.2011.OWW6
- [24] RYLYAKOV A V, SCHOW C L, KASH J A. New ultra-high sensitivity A, low-power optical receiver based on a decision-feedback equalizer [C]//Optical Fiber Communication Conference/National Fiber Optic Engineers Conference, Los Angeles, USA, 2011. DOI: 10.1364/ofc.2011.othp3
- [25] BULZACCHELLI J F, MEGHELLI M, RYLOV S V, et al. A 10-Gb/s 5-Tap DFE/4-Tap FFE transceiver in 90-nm CMOS technology [J]. *IEEE journal of solid-state circuits*, 2006, 41(12): 2885 – 2900. DOI: 10.1109/jssc.2006.884342
- [26] ZHOU J, QIAO Y J, YU J J, et al. Interleaved single-carrier frequency-division multiplexing for optical interconnects [J]. *Optics express*, 2017, 25(9): 10586 – 10596. DOI: 10.1364/oe.25.010586
- [27] MAN J W, CHEN W, SONG X L, et al. A Low-cost 100GE optical transceiver module for 2km SMF interconnect with PAM4 modulation [C]//Optical Fiber Communication Conference, San Francisco, USA, 2014. DOI: 10.1364/ofc.2014.m2e.7
- [28] KANAZAWA S, YAMAZAKI H, NAKANISHI Y, et al. Transmission of 214-Gbit/s 4-PAM signal using an ultra-broadband lumped-electrode EADFB laser module [C]//Optical Fiber Communication Conference, Anaheim, USA, 2016. DOI: 10.1364/OFC.2016.Th5B.3
- [29] VERPLAETSE M, LIN R, KERREBROUCK J VAN, et al. Real-time 100 Gb/s transmission using three-level electrical duobinary modulation for short-reach optical interconnects [J]. *Journal of lightwave technology*, 2017, 35(7): 1313 – 1319. DOI: 10.1109/jlt.2016.2643778
- [30] FEHENBERGER T, LAVERY D, MAHER R, et al. Sensitivity gains by mismatched probabilistic shaping for optical communication systems [J]. *IEEE photonics technology letters*, 2016, 28(7): 786 – 789. DOI: 10.1109/lpt.2015.2514078
- [31] BEYGI L, AGRELL E, KAHN J M, et al. Rate-adaptive coded modulation for fiber-optic communications [J]. *Journal of lightwave technology*, 2014, 32(2): 333 – 343. DOI: 10.1109/jlt.2013.2285672
- [32] BUCHALI F, STEINER F, BÖCHERER G, et al. Rate adaptation and reach increase by probabilistically shaped 64-QAM: an experimental demonstration [J]. *Journal of lightwave technology*, 2016, 34(7): 1599 – 1609
- [33] YANKOV M P, ZIBAR D, LARSEN K J, et al. Constellation shaping for fiber-optic channels with QAM and high spectral efficiency [J]. *IEEE photonics technology letters*, 2014, 26(23): 2407 – 2410. DOI: 10.1109/lpt.2014.2358274
- [34] BÖCHERER G. Capacity-achieving probabilistic shaping for noisy and noiseless channels [EB/OL]. (2012) [2019]. https://www.researchgate.net/publication/268291605_Capacity_-_Achieving_Probabilistic_Shaping_for_Noisy_and_Noiseless_Channels
- [35] BÖCHERER G. On Joint design of probabilistic shaping and forward error correction for optical systems [C]//Optical Fiber Communication Conference, San Diego, USA, 2018. DOI:10.1364/ofc.2018.m4e.1
- [36] PAN C P, KSCHISCHANG F R. Probabilistic 16-QAM shaping in WDM systems [J]. *Journal of lightwave technology*, 2016, 34(18): 4285 – 4292. DOI: 10.1109/jlt.2016.2594296
- [37] RENNER J, FEHENBERGER T, YANKOV M P, et al. Experimental comparison of probabilistic shaping methods for unrepeatable fiber transmission [J]. *Journal of lightwave technology*, 2017, 35(22): 4871 – 4879. DOI: 10.1109/jlt.2017.2752243
- [38] SEMRAU D, XU T H, SHEVCHENKO N A, et al. Achievable information rates estimates in optically amplified transmission systems using nonlinearity compensation and probabilistic shaping [J]. *Optics Letters*, 2017, 42(1): 121 – 124. DOI: 10.1364/ol.42.000121
- [39] FEHENBERGER T, BÖCHERER G, ALVARADO A, et al. LDPC coded modulation with probabilistic shaping for optical fiber systems [C]//Optical Fiber Communication Conference, Los Angeles, USA, 2015. DOI: 10.1364/ofc.2015.th2a.23
- [40] ERIKSSON T A, CHAGNON M, BUCHALI F, et al. 56 Gbaud probabilistically shaped PAM8 for data center interconnects [C]//2017 European Conference on Optical Communication, Gothenburg, Sweden, 2017. DOI: 10.1109/

ecoc.2017.8346148

- [41] CHEN X, CHO J, CHANDRASEKHAR S, et al. Single-wavelength, single-polarization, single- photodiode Kramers-Kronig detection of 440-Gb/s entropy-loaded discrete multitone modulation transmitted over 100-km SSMF[C]//2017 IEEE Photonics Conference, Orlando, USA, 2017. DOI: 10.1109/pc2.2017.8283402
- [42] CHE D, SHIEH W. Approaching the capacity of colored-SNR optical channels by multicarrier entropy loading [J]. Journal of lightwave technology, 2018, 36(1): 68 – 78. DOI: 10.1109/jlt.2017.2778290
- [43] XIE C H, CHEN Z X, FU S N, et al. Achievable information rate enhancement of visible light communication using probabilistically shaped OFDM modulation [J]. Optics express, 2018, 26(1): 367. DOI: 10.1364/oe.26.000367
- [44] KHAN F N, ZHONG K P, AL-ARASHI W H, et al. Modulation format identification in coherent receivers using deep machine learning [J]. IEEE photonics technology letters, 2016, 28(17): 1886 – 1889. DOI: 10.1109/lpt.2016.2574800
- [45] WANG D S, ZHANG M, LI Z, et al. Modulation format recognition and OSNR estimation using CNN-based deep learning [J]. IEEE photonics technology letters, 2017, 29(19): 1667 – 1670. DOI: 10.1109/lpt.2017.2742553
- [46] THRANE J, WASS J, PIELS M, et al. Machine learning techniques for optical performance monitoring from directly detected PDM-QAM signals [J]. Journal of lightwave technology, 2017, 35(4): 868 – 875. DOI: 10.1109/jlt.2016.2590989
- [47] ANDERSON T B, KOWALCZYK A, CLARKE K, et al. Multi impairment monitoring for optical networks [J]. Journal of lightwave technology, 2009, 27(16): 3729 – 3736. DOI: 10.1109/jlt.2009.2025052
- [48] GIACOMIDIS E, MHATLI S, NGUYEN T, et al. Kerr-induced nonlinearity reduction in coherent optical OFDM by low complexity support vector machine regression-based equalization [C]//Optical Fiber Communication Conference, Anaheim, USA, 2016. DOI: 10.1364/ofc.2016.th2a.49
- [49] JARAJREH M A, GIACOMIDIS E, ALDAYA I, et al. Artificial neural network nonlinear equalizer for coherent optical OFDM [J]. IEEE photonics technology letters, 2015, 27(4): 387 – 390. DOI: 10.1109/lpt.2014.2375960
- [50] NGUYEN T, MHATLI S, GIACOMIDIS E, et al. Fiber nonlinearity equalizer based on support vector classification for coherent optical OFDM [J]. IEEE photonics journal, 2016, 8(2): 1 – 9. DOI: 10.1109/jphot.2016.2528886
- [51] CHEN G Y, SUN L, XU K, et al. Machine learning of SVM classification utilizing complete binary tree structure for PAM-4/8 optical interconnection [C]//IEEE Optical Interconnects Conference (OI), Santa Fe, USA, 2017. DOI: 10.1109/oic.2017.7965524
- [52] WANG D S, ZHANG M, FU M X, et al. Nonlinearity mitigation using a machine learning detector based on k-nearest neighbors [J]. IEEE photonics technology letters, 2016, 28(19): 2102 – 2105. DOI: 10.1109/lpt.2016.2555857
- [53] CHEN G Y, DU J B, SUN L, et al. Nonlinear distortion mitigation by machine learning of SVM classification for PAM-4 and PAM-8 modulated optical interconnection [J]. Journal of lightwave technology, 2018, 36(3): 650 – 657. DOI: 10.1109/jlt.2017.2763961
- [54] CHEN G Y, DU J B, SUN L, et al. Machine learning adaptive receiver for PAM-4 modulated optical interconnection based on silicon microring modulator [J]. Journal of light wave technology, 2018, 36(18): 4106 – 4113. DOI: 10.1109/jlt.2018.2861710
- [55] SUN L, DU J B, HE Z Y. Machine learning for nonlinearity mitigation in CAP modulated optical interconnect system by using k-nearest neighbour algorithm

[C]//Asia Communications and Photonics Conference 2016, Wuhan, China, 2016. DOI: 10.1364/acpc.2016.as1b.1

Biographies

SUN Lin received the bachelor's degree in electronic engineering from Sichuan University, China in 2014 and has developed great interest in optical fiber communication field. He is currently working toward the Ph.D. degree at Shanghai Jiao Tong University, China. His research activities and interests include fiber communications, optical signal processing and optical transmission and interconnection.

DU Jiangbing (dujiangbing@sjtu.edu.cn) received the bachelor's degree and the master's degree from the College of Physics and Institute of Modern Optics, Nankai University, China in 2005 and 2008, respectively, and the Ph.D. degree in electronic engineering from the Chinese University of Hong Kong, China, in 2011. He was with Huawei Technologies from 2011 to 2012. He joined Shanghai Jiao Tong University, China as an assistant professor since 2012, and became an associate professor since 2014. He is the author or coauthor of more than 140 journals and conference papers.

HUA Feng received the B.S. and M.S. degrees in optical instrument from Tianjin University, China in 1993 and 1996 respectively. She has worked with ZTE Corporation since 2000. Currently she is a senior engineer focusing on advanced research of cutting-edge optical communication technologies including silicon photonics, spatial division multiplexing and optical backplane. She has more than 10 patents.

TANG Ningfeng received the M.S. degree in testing engineering from Nanjing University of Aeronautics and Astronautics, China, in 1999. He has worked with ZTE Corporation since 1999. Currently he is an architecture engineer focusing on co-packaged optics and optical interconnection equipment. He has more than 10 patents.

HE Zuyuan received B.S. and M.S. degrees in electronic engineering from Shanghai Jiao Tong University, China in 1984 and 1987, respectively, and the Ph.D. degree in optoelectronics from the University of Tokyo, Japan, in 1999. He joined the Nanjing University of Science and Technology, China as a research associate in 1987, and became a lecturer in 1990. From 1995 to 1996, he was a research fellow in University of Tokyo. In 1999, he became a research associate with the University of Tokyo. In 2001, he joined CIENA Corporation, Maryland, USA, as a lead engineer leading the Optical Testing and Optical Process Development Group. He returned to the University of Tokyo as a lecturer in 2003, and became an associate professor in 2005 and a full professor in 2010. He is currently a chair professor of Shanghai Jiao Tong University. His research focuses on optical fiber sensing.



Crowd Counting for Real Monitoring Scene

LI Yiming¹, LI Weihua², SHEN Zan³,
NI Bingbing¹

(1. Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, Shanghai 200240, China;

2. Video Production Line, ZTE Corporation, Chongqing 401121, China;

3. Institute of Technology, Ping An Technology (Shenzhen) Co., Ltd., Shanghai 200120, China)

DOI: 10.12142/ZTECOM.202002009

<http://kns.cnki.net/kcms/detail/34.1294>.

TN.20191210.1718.002.html, published online December 11, 2019

Manuscript received: 2019-01-17

Abstract: Crowd counting is a challenging task in computer vision as realistic scenes are always filled with unfavourable factors such as severe occlusions, perspective distortions and diverse distributions. Recent state-of-the-art methods based on convolutional neural network (CNN) weaken these factors via multi-scale feature fusion or optimal feature selection through a front switch-net. L2 regression is used to regress the density map of the crowd, which is known to lead to an average and blurry result, and affects the accuracy of crowd count and position distribution. To tackle these problems, we take full advantage of the application of generative adversarial networks (GANs) in image generation and propose a novel crowd counting model based on conditional GANs to predict high-quality density maps from crowd images. Furthermore, we innovatively put forward a new regularizer so as to help boost the accuracy of processing extremely crowded scenes. Extensive experiments on four major crowd counting datasets are conducted to demonstrate the better performance of the proposed approach compared with recent state-of-the-art methods.

Keywords: crowd counting; density; generative adversarial network

Citation (IEEE Format): Y. M. Li, W. H. Li, Z. Shen, et al., "Crowd counting for real monitoring scene," *ZTE Communications*, vol. 18, no. 2, pp. 74 – 82, Jun. 2020. doi: 10.12142/ZTECOM.202002009.

1 Introduction

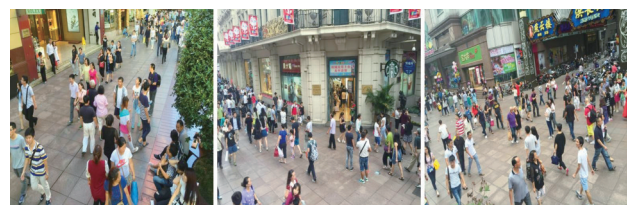
With the population density of major cities increasing in recent years, crowd scene analysis has already become an important safety index in the field of video surveillance, especially the crowd count and high-quality density map which has a wide range of applications in public safety, traffic monitoring, scene understanding and flow monitoring. However, predicting accurate crowd count while ensuring high-quality density map is a really challenging task, because complex crowd scenes are always accompanied with severe occlusions, perspective distortions and diverse distributions, and also put forward a great challenge to the algorithm model. Several typical still crowd images from the ShanghaiTech dataset^[1] are shown in **Fig. 1**.

In order to solve these problems in computer vision field, a great many algorithms have been proposed, which can be mainly divided into two categories, namely, the hand-crafted feature based regression and the convolutional neural network (CNN) based regression. Recent works^[1-3] indicate that the

CNN based regression has a more excellent performance. Such methods obtain the number of people from a still image by mapping the image to its density map through a CNN architecture. They have achieved significant improvements on



(a) pictures with dense crowd



(b) pictures with relatively sparse crowd

▲ **Figure 1.** Examples of crowd scene from the ShanghaiTech dataset^[1].

This work was supported by ZTE Industry-University-Institute Cooperation Funds.

count estimates, whereas the quality of their estimated density map is unfortunately poor due to the throng scene and self-defect of Euclidean loss.

In the past two years, generative adversarial networks (GANs)^[4] have become the most popular frameworks in all relevant fields of image generation. Some of its derivatives such as conditional GANs (cGANs)^[5] and information maximizing generative adversarial nets (InfoGANs)^[6] can generate extremely realistic images. Therefore, the key point is whether we can draw the advantages of GANs to generate high-quality and high-resolution density maps. Inspired by this, we propose a novel crowd counting model based on cGANs called Crowd Counting Network for Real Monitoring Scene.

The initial inspiration of Crowd Counting Network for Real Monitoring Scene derives from Ref. [7] which uses cGANs to realize pixel-to-pixel translation. Usually, most existing CNN-based approaches on crowd counting add several max-pooling layers in their networks, forcing them to regress on down-sampled density maps, and traditional Euclidean loss is employed to optimize their network parameters which will eventually lead to a relatively blurry result. While in our proposed approach, the generator of cGANs is designed to generate density maps having the same size as input images through a U-net^[8] structure with the same amount of convolutional and deconvolutional layers. In other words, it executes a pixel-wise translation from a crowd image to its estimated density map. Thanks to the combination of pixel-wise Euclidean loss, perceptual loss, inter-frame loss and the adversarial training loss provided by GANs, the density map predicted by the generator overcomes blurry results obtained by optimizing only over Euclidean loss and achieves higher quality than that of the previous methods. Besides, we innovatively propose a novel regularizer which provides a very strong regularization constraint on the consistency of parent-child-relationship density maps between different scales to excavate multi-scale consistent information. Unlike using different sizes of filters to extract multi-scale features, we care more about local and overall interrelation between adjacent image patches.

Contributions of this paper are summarized as follows.

- We propose a novel crowd counting framework based on cGANs, called Crowd Counting Network for Real Monitoring Scene. It implements end-to-end training. The use of adversarial training loss helps generate high-quality crowd density map.
- A novel regularizer is introduced to help solve perspective distortions and diverse distributions problems in crowd scenes by providing a very strong constraint on the consistency of parent-child-relationship patches to excavate multi-scale consistent information.
- An inter-frame loss is denoted for the crowd counting in video stream, which can improve the continuity of detection by constraining the number of people calculated by density map between adjacent frames. The loss can also enhance the stability of the network in predicting the density map of video

information.

- Our method obtains state-of-the-art performance on four major crowd counting datasets involving the ShanghaiTech dataset, WorldExpo'10 dataset, UCF_CC_50 dataset and UCSD dataset.

2 Related Work

A large number of algorithms have been proposed to tackle crowd counting task in computer vision. Early works estimate the number of pedestrians via head or body detection^[9-11]. Such detection based methods are limited by severe occlusions in extremely dense crowd scenes. Methods in Refs. [12-15] use regressors trained with low-level features to predict global counts, and Ref. [16] makes a fusion of hand-crafted features from multiple sources, including the histogram of oriented gradients (HOG), scale-invariant feature transform (SIFT), Fourier analysis, and detections. These methods cannot provide the distribution of crowd, and such low-level features are outperformed by features extracted from CNN which have better and deeper representations.

Several works focus on crowd counting in videos by trajectory-clustering. RABAUD et al.^[17] utilized a highly parallelized version of the Kanade-Lucas-Tomasi Tracking (KLT) tracker to extract a set of feature trajectories from videos. Fragmentation of trajectories is restrained by conditioning the trajectories spatially and temporally. BROSTOW et al.^[18] proposed an unsupervised data driven Bayesian clustering algorithm, which uses space-time proximity and trajectory for clustering. However, such tracking based methods are limited in crowd counting from arbitrary still image for lack of temporal information.

In recent years, crowd counting has entered the era of CNN. WANG et al.^[19] trained a classic Alexnet style CNN model to predict crowd counts. Regrettably, this model has limitation in crowd analysis as it does not provide the estimation of crowd distribution. ZHANG et al.^[3] proposed a deep convolutional neural network for crowd counting which is alternatively regressed with two related learning objectives: the crowd count and the density map. Such switchable objective-learning helps improve the performance of both objectives. However, the application of this method is limited as it requires perspective maps which are not easily available in practice during the process of both training and testing. Multi-column CNN is employed by Refs. [1] and [20]. Different CNN columns with varied receptive fields are designed to capture scale variations and perspectives, and then features from these columns are fused together by a 1×1 convolutional layer to regress crowd density. Switch-CNN^[2] based on the multi-column convolutional neural network (MCNN)^[1] is a patch-based switching architecture before the crowd patches go into multi-column regressors. The switch-net is trained as a classifier to choose the most appropriate regressor for a particular input patch, which

takes advantage of patch-wise variations in density within a single image. These methods have made great contributions to the progress of crowd counting by deep learning; at the same time, they add max-pooling layers in their networks and use L2 loss to optimize the whole model. Namely, they pay more attention to the accuracy of predicted crowd count, and neglect the quality of the regressed density map. The latest proposed contextual pyramid CNN (CP-CNN)^[21] is a contextual Pyramid CNNs for incorporating global and local contexts which are obtained by learning various density levels. This contextual information is fused with high dimensional feature maps extracted from a multi-column CNN^[1] by a fusion-CNN consisting of a set of convolutional and fractionally-strided layers. Adversarial loss is used to help generate high-quality density maps in the last fusion-CNN. Up to now, this approach acquires the lowest counting error on three major crowd datasets in addition to generating high-quality density maps.

The above methods utilize multi-scale features fusion or optimum feature selection to deal with crowd in varied scales, but to some extent they only consider crowd in different scales having different sensitivities to diverse convolutional kernel, which is a relatively local consideration. The latest one incorporates contextual information by classifying images or patches into five density levels independently, while ignoring the correlation between adjacent patches. In other words, none of them research on the statistical consistency of the crowd counts in multi-scale joint patches; for example, a patch is supposed to be equally divided into four sub-patches and the estimated crowd count of the patch ought to be equal to the sum of the estimated crowd counts of these four sub-patches. Such multi-scale consistency offers an effective and strong regularization constraint for crowd count and density estimation. Unfortunately, these methods do not take it into consideration.

3 Our Approach

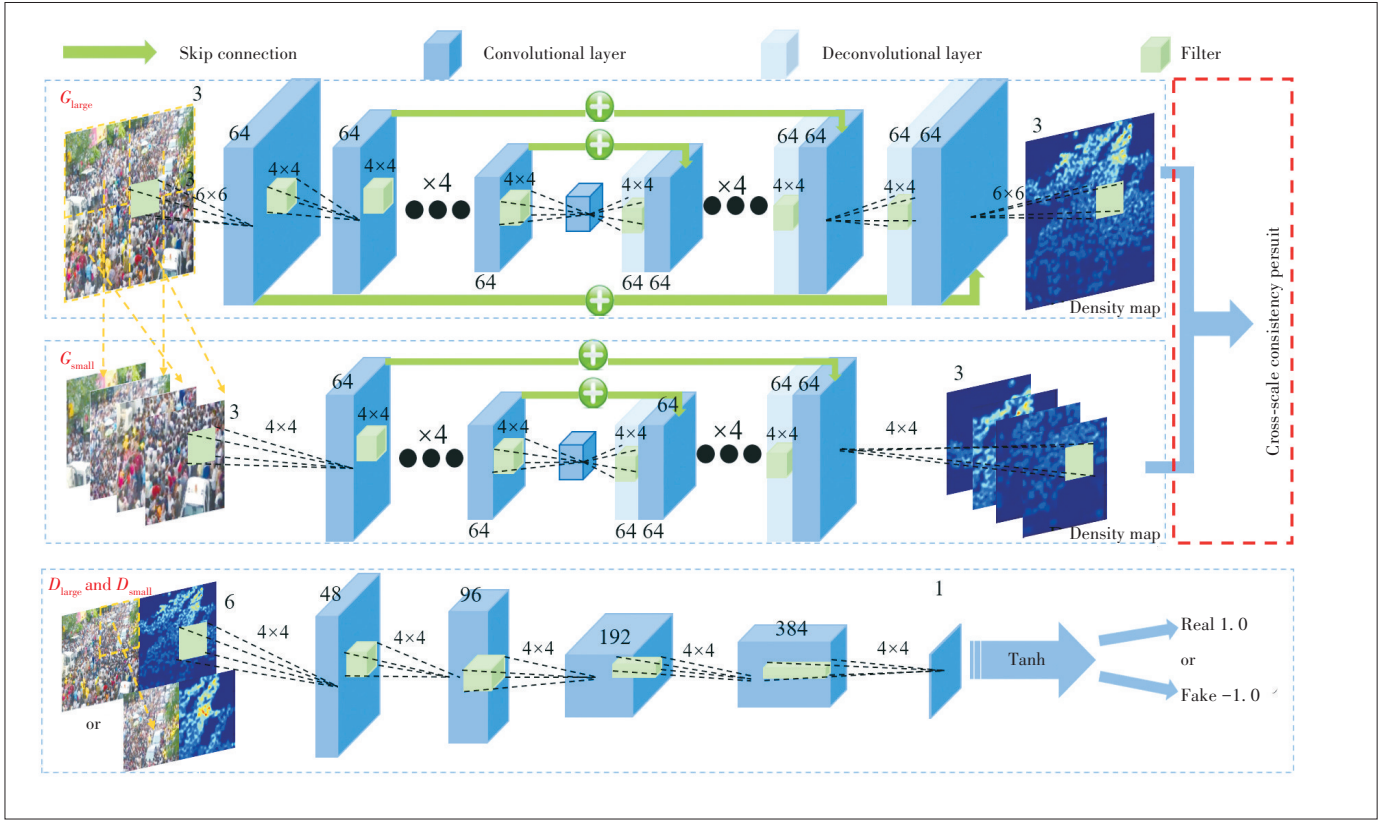
We proposed a novel GANs-based crowd counting framework called Real Monitoring Scene Network (RMSN) for Crowd Counting. Many of the previous state-of-the-art methods^[1-2] choose L2 loss to regress density map, which is widely acknowledged to result in low-quality and blurry results especially for image reconstruction tasks^{[7], [22]}. To overcome this flaw and generate high-quality and high-resolution density maps, we design a weighted combination of loss including: adversarial training loss, perceptual loss and pixel-wise Euclidean loss, and a new regularizer is proposed in our GANs-based model to excavate multi-scale consistent information. After generating the density map, we will get the density matrix information between -1 and 1 and then normalize it. The count number from density map can be obtained by summing the normalized matrix divided by a certain coefficient 0.12 .

3.1 Architecture

RMSN is based on the idea of pixel-to-pixel translation, and in order to leverage the proposed regularizer, our network architecture consists of two complementary conditional GANs: $\text{GAN}_{\text{large}}$ and $\text{GAN}_{\text{small}}$. A classic GAN architecture usually contains two models: a generator G trained to produce outputs and a discriminator D trained to distinguish the real target and fake outputs from G . In our method, the generator G learns an end-to-end mapping from input crowd image to its density map. **Fig. 2** shows the integral architecture of RMSN. The general structures of the two GANs are quite similar. Specific details are discussed below.

A general problem of pixel-to-pixel translation is the difficulty to efficiently map a high resolution input image to a high resolution output image. Fortunately, previous works^{[7], [23-24]} have provided an excellent solution by using an encoder-decoder network^[25]. In RMSN, a U-net^[8] structure is introduced to the generator G as an encoder-decoder. Let us start with the large GANs in our proposed architecture. In the generator G_{large} , eight convolutional layers along with batch normalization layers and LeakyReLU activation layers are stacked in the encoder part which serves as a feature extractor. Then, eight de-convolutional layers along with batch normalization layers and ReLU activation layers (except for the last one) are added in the decoder part, followed by a tanh function. Note that the de-convolutional layers are a mirrored version of the foregone convolutional layers. We set a stride of 2 in all layers, which means convolutions in the encoder down-sample by a factor of 2, whereas deconvolutions upsample by a factor of 2. In addition, three dropout layers are added behind the first three de-convolutional layers with dropout ratio set to 0.5 in order to alleviate over-fitting. Skip connections are also added between mirror-symmetry convolutional and de-convolutional layers to help improve the performance and efficiency, similar to Ref. [7]. The architecture of G_{large} can be depicted as: C(64, 6)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-DCD(64, 4)-DCD(64, 4)-DCD(64, 4)-DC(64, 4)-DC(64, 4)-DC(64, 4)-DC(64, 4)-DC(3, 6)-Tanh, where C is a Conv-BN-LReLU layer, DCD is a deConv-BN-Dropout-ReLU layer, DC is a de-Conv-BN-ReLU layer and the first number in every parenthesis represents the number of filters while the second number represents filter size.

The generator G_{small} which is similar to G_{large} contains 7 convolutional layers and 7 deconvolutional layers. 4×4 filters are used in all layers with a stride of 2. The architecture of generator G_{small} can be depicted as: C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-C(64, 4)-DCD(64, 4)-DCD(64, 4)-DC(64, 4)-DC(64, 4)-DC(64, 4)-DC(3, 4)-Tanh. The inputs of generator G_{large} are $240 \times 240 \times 3$ sized crowd patches, and the inputs of generator G_{small} are $120 \times 120 \times 3$ sized crowd patches equationally cropped from the input of the generator G_{large} without overlapping, as shown in the upper left corner of Fig. 2 Their outputs are of the same size as their inputs. That means the



▲ Figure 2. Architecture of the proposed Crowd Counting Network for Real Monitoring Scene (RMSN): The top level is the structure of generator G_{large} , the middle part is the structure of generator G_{small} , and the bottom part is the discriminators D_{large} and D_{small} that have the same structure.

density maps generated from our RMSN contain more details and have better characterization capabilities than previous density-map-based works^[1-3] as their density maps are always much smaller than the origin images.

The discriminators D_{large} and D_{small} have the same structure, displayed at the bottom of Fig. 2. Five convolutional layers along with batch normalization layers and LeakyReLU activation layers (except for the last one) act as a feature extractor. A tanh function is stacked at the end of these convolutional layers to regress a probabilistic score ranges from -1.0 to 1.0 . The architecture of discriminators D_{large} and D_{small} can be depicted as: C(48,4)-C(96,4)-C(192,4)-C(384,4)-C(1,4)-Tanh. The inputs of the discriminators D_{large} and D_{small} are $240 \times 240 \times 6$ and $120 \times 120 \times 6$ sized concatenated pairs of crowd patch and density map, respectively. The values of the output matrix indicate whether the input is real (close to 1.0) or fake (close to -1.0).

3.2 Loss Function

In our problem, motivated by recent success of GANs, we propose an adversarial loss of generating crowd density map from image patch. The adversarial loss involves a discriminator D and a generator G playing a two-player minimax game: D is trained to distinguish synthetic images from ground truth while G is trained to generate images to fool D . The adversari-

al loss is denoted as:

$$L_A(G, D) = \mathbb{E}_{x, y \sim P_{data}(x, y)} [\log D(x, y)] + \mathbb{E}_{x \sim P_{data}(x)} [\log (1 - D(x, G(x)))], \quad (1)$$

where x denotes a training patch and y denotes corresponding ground-truth density map. G tries to minimize this objective, whereas D tries to maximize it.

Due to the lack of direct constraint from ground truth, just using an adversarial loss may sometimes lead to aberrant spatial structure. Thus, we include two conventional losses to smooth and improve the solution, which is denoted as follows.

In our problem, l_2 loss $L_E(G)$ can force the generated estimated density map to fool D and be close to the ground truth in an L2 sense.

$$L_E(G) = \frac{1}{C} \sum_{c=1}^C \|p^G(c) - p^{GT}(c)\|_2^2, \quad (2)$$

where $p^G(c)$ represents the pixels in generated density map and $p^{GT}(c)$ represents the pixels in ground-truth density map, with $c=3$.

Perceptual loss is first introduced by JOHNSON et al.^[24] for image transformation and super resolution task. By minimizing the perceptual differences between the two images, the

synthetic image can be more semantically similar to the objective image. The perceptual loss is defined as:

$$L_p(G) = \frac{1}{C} \sum_{c=1}^C \|f^G(c) - f^{GT}(c)\|_2^2, \quad (3)$$

where $f^G(c)$ represents the pixels in high level perceptual features of generated density map and $f^{GT}(c)$ represents the pixels in high level perceptual features of ground-truth density map, with $c=128$.

Therefore, the integrated loss is expressed as:

$$L_1 = \arg \min_G \max_D L_A(G, D) + \lambda_e L_E(G) + \lambda_p L_p(G), \quad (4)$$

where λ_e and λ_p are predefined weights for Euclidean loss and perceptual loss. Suggested by previous works^[26], we set $\lambda_e = \lambda_p = 150$.

In our problem, we propose a new inter-frame loss for the prediction in video stream, which can improve the continuity of detection by constraining the number of people between adjacent frames and enhance the stability of the network in predicting the density map of video information. The loss is defined as the distance between two adjacent frames of generated density maps, which is denoted as:

$$L_i(G) = \frac{1}{N_{\text{pix}}} \|n^G(c) - n^{*G}(c)\|_2^2, \quad (5)$$

where N_{pix} represents the whole numbers of pixels in generated density maps, $n^G(c)$ represents the number of pedestrians calculated from the current frame in generated density map, and $n^{*G}(c)$ represents the number of pedestrians calculated from the previous frame.

Therefore, for video stream information, the integrated loss L_1 should be denoted as:

$$L_1 = \arg \min_G \max_D L_A(G, D) + \lambda_e L_E(G) + \lambda_p L_p(G) + \lambda_i L_i(G), \quad (6)$$

where $\lambda_i=150$ is predefined weights for inter-frame loss.

To restrain the cross-scale consistency of parent-child-relationship density maps, we propose a Cross-Scale Consistency Pursuit loss^[27] defined as the discrepancy/distance between P_{concat} and P_{parent} . The CSCP loss of a $W \times H$ density map with channels is defined as:

$$L_c(G) = \frac{1}{C} \sum_{c=1}^C \|p^{\text{prt}}(c) - p^{\text{cnt}}(c)\|_2^2, \quad (7)$$

where $p^{\text{prt}}(c)$ represents the pixels in density map P_{parent} and $p^{\text{cnt}}(c)$ represents the pixels in density map P_{concat} , with $c=3$.

As pointed out above, the four loss functions are weightedly combined to a final objective,

$$L_{\text{II}} = L_1 + \lambda_c L_c(G), \quad (8)$$

where $\lambda_c=10$ is the predefined weight for cross-scale consistency pursuit loss.

3.3 Training Details

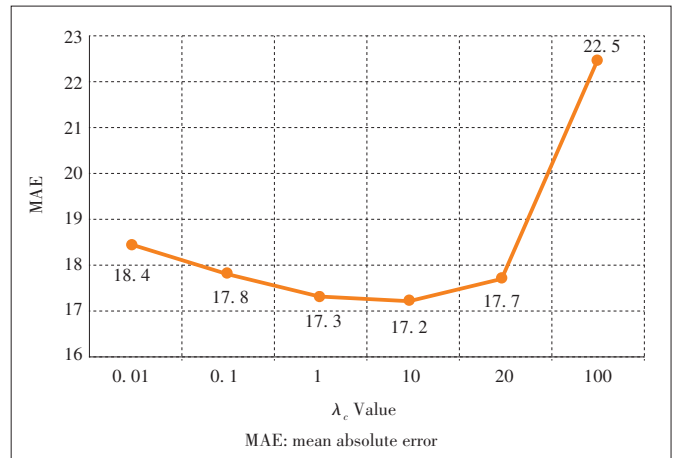
During training, the input is an image pair consisting of a crowd patch and its corresponding density map. Such an image pair is first input to the large-scale subnet G_{large} , and then evenly divided into four equidistant image pairs without overlapping and finally input to the small-scale subnet G_{small} . Both subnets are jointly trained. The RMS prop optimizer has a learning rate set to 0.00005 and is used to update the parameters of the network. We follow the update rule: in each iteration, G_{small} 's four updates are followed by a G_{large} .

To increase the training data, one of the general methods is to resize the input image pair to a larger size and randomly crop the image pair of a particular size. However, such data increases are not appropriate in our crowd counting tasks because image interpolation algorithms such as recent and bilinear algorithms inevitably change the number of people in the density map. Therefore, in our experiments, we use filled and flipped images to replace image size adjustments with a probability of 50% for data enhancement.

Our model requires approximately 300 periods of training to converge. In order to balance the training of the two sub-networks, in the first 100 periods, the predefined weight λ_c in Eq. (6) is set to 0, then it is adjusted to 10 and the training process is continued. Finally, the well-trained generator G_{large} is used to predict the density map of the test image. Training and testing of the proposed network is implemented on the Torch7 framework.

3.4 Parameter λ_c Study

We did comparative experiments performed on Part_B of the ShanghaiTech dataset to choose the optimum value of λ_c . As shown in **Fig. 3**, mean absolute error (MAE) decreases



▲ Figure 3. Comparisons of MAE for different λ_c values on Shanghai-Tech Part_B.

when the value of λ_c increases. The lowest MAE value is obtained at $\lambda_c=10$. After that, when the value of λ_c increases, the error rises rapidly, because the comparison of the weight of cross-scale consistency loss and L_1 loss becomes too significant. Therefore, we finally assign 10 to λ_c .

4 Experiments

We evaluate our method in four major crowd counting datasets, including the ShanghaiTech dataset, WorldExpo'10 dataset, UCF CC 50 dataset and UCSD dataset. Compared with the state-of-the-art methods, our method gains a superior or at least competitive performance in all datasets used for evaluation. Training and testing of the proposed network are implemented on Torch7 framework.

We use MAE and mean squared error (MSE) to evaluate the performance of our method on existing works.

Adversarial pursuit seeks to exploit adversarial loss, perceptual loss and U-net structured generator to improve the quality of generated density maps. It is worth noting that our predicted density map is better distributed than the MCNN population, with less blur and noise. In addition, comparative experiments were performed on the ShanghaiTech^[1] and WorldExpo'10^[3] datasets in **Table 1** above. It can be observed that training with additional adversarial loss and perceptual loss (i.e. LI) results in far less errors than training with Euclidean loss only.

4.1 ShanghaiTech

The ShanghaiTech dataset is created by ZHANG et al.^[1], which that consists of 1 198 annotated images. The dataset is divided into two parts. Part A contains 482 images downloaded from the Internet with extremely dense crowd, and Part B contains 716 images taken from the busy street in Shanghai with normal flow of crowd. Our model is trained and tested on the training and testing set split by author respectively. To augment the training data, we resize all the images to 720×720 and cropped patches from each image. Each patch is 1 size of origin image and is cropped from different locations. Ground-truth density maps are generated by geometry-adaptive Gaussian kernels. At the test time, a window of size 240×240 slides on the test image to crop patches with 50% overlapping as inputs of the well trained generator. Then, outputs from the generator are integrated to a weight-balanced density map which has the same size of the test image. Finally, the estimated crowd count of the image can be calculated by the sum of the density map. The proposed method is compared with four current state-of-the-art CNN-based approaches: a switchable objective-learning CNN^[3], MCNN^[1], Switch-CNN^[2] and CP-CNN^[21]. ZHANG et al.^[3] proposed a switchable objective-learning CNN which is alternatively regressed with two related learning objectives: crowd count and density map. This method is highly dependent on the perspective maps during

training and testing. ZHANG et al.^[1] employed a MCNN to extract multi-scale features and to fuse them to get a better representation. Switch-CNN^[2] trained a prepositive switch-net to intelligently choose the optimal regressor instead of multi-column feature fusion. CP-CNN^[21] incorporated global and local contextual information with fused multi-column features, and is trained in an end-to-end fashion using a combination of adversarial loss and pixel-level Euclidean loss. From **Table 2** we can see, on Part B of which images are closer to the real monitoring screens, the proposed approach obtains appreciable improvement in contrast to the best model CP-CNN at the time. On Part A, besides CP-CNN, our method has also achieved the best results, compared with the other three ones. In order to fairly evaluate the quality of the generated density map, we choose the same set of test images published in MCNN^[1] paper along with ground-truth and predicted density maps, shown in **Fig. 4**. It can be intuitively seen that our predicted density maps conform to the distribution of crowd much better than MCNN's with noticeable blur and noise, which benefits from our GANs-based architecture and new regularizer.

4.2 WorldExpo'10 Dataset

The WorldExpo'10 dataset is created by ZHANG et al.^[3] with 1 132 annotated video sequences captured by 108 surveillance cameras from Shanghai 2010 World Expo. A total of 199 923 pedestrians in 3 980 frames are labeled at the centers of their heads. In these frames, 3 380 frames are treated as the training set; the rest 600 frames are used as the test set, which are sampled from five different scenes, each containing 120 frames. The pedestrian number in the test scene ranges from 1 – 220. This dataset also provides perspective maps, the value of which represents the number

▼Table 1. Comparisons of errors for training with different losses

	Part A		Part B		WorldExpo'10
Objective	MAE	MSE	MAE	MSE	AMAE
L_E	95.8	149.4	24.1	36.4	9.95
L_I	83.2	131.3	18.4	28.8	8.48
L_{II}	75.7	102.7	17.2	27.4	7.5

AMAE: average mean absolute error MAE: mean absolute error MSE: mean squared error

▼Table 2. Comparison of RMSN with other three state-of-the-art CNN-based methods on ShanghaiTech dataset

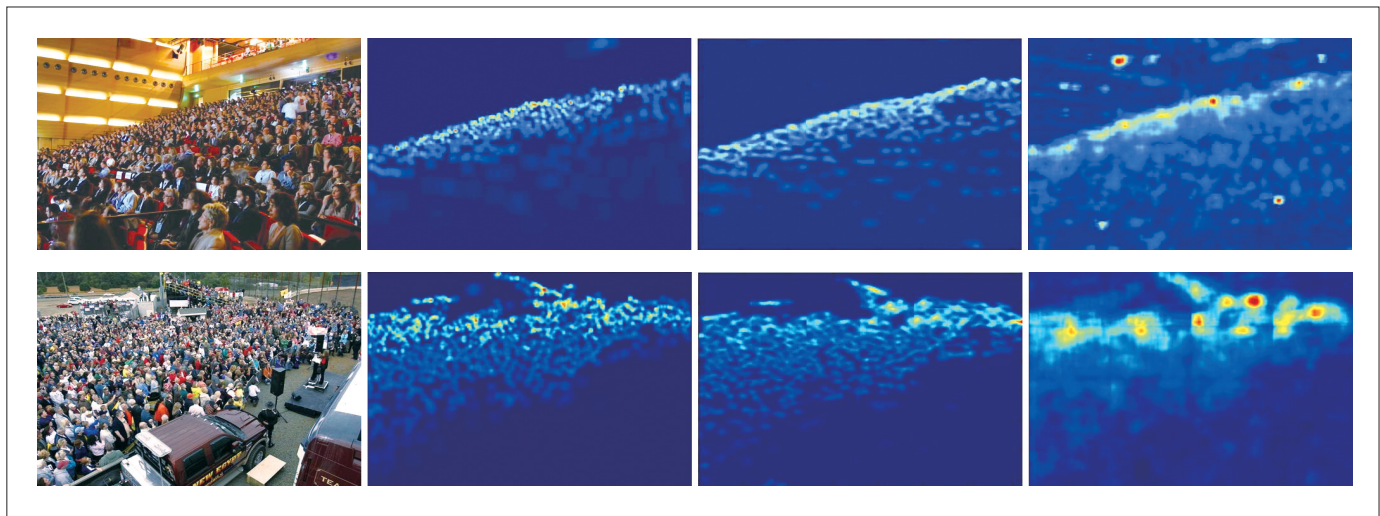
	Part A		Part B	
Methods	MAE	MSE	MAE	MSE
The approach in Ref. [3]	181.8	277.7	32.0	49.8
MCNN ^[1]	110.2	173.2	26.4	41.3
Switch-CNN ^[2]	90.4	135.0	21.6	33.4
The proposed RMSN	86.2	145.4	17.2	27.4

MAE: mean absolute error

MCNN: multi-column convolutional neural network

MSE: mean squared error

RMSN: real monitoring scene network



▲ Figure 4. Two test images sampled from the ShanghaiTech Part A dataset (From left to right, the four columns successively denote test images, ground-truth density maps, our estimated density maps and the multi-column convolutional neural network (MCNN) 's^[1] respectively).

of pixels in the image covering one square meter at real location. For fair comparison, we choose the crowd density distribution kernel introduced by Ref. [3], which contains two terms: a normalized Gaussian kernel as a head part and a bivariate normalized distribution as a body part, to generate density maps with perspective information. To follow the previous methods, only the crowd in region of interest (ROI) are taken into consideration. So we multiply predicted density map by specifying ROI mask, which means that the area out of ROI is set to zero. MAE is suggested by ZHANG et al.^[3] to evaluate the performance of crowd counting model on this dataset.

Table 3, in which MAE is used to evaluate the performance on each scene and the average result across scenes, reports the performance of our method on five different test scenes in comparison to other four state-of-the-art methods. Our method refreshes the scores of three scenes: Scene2, Scene3 and Scene5, while achieving comparable performance on the rest two scenes, and outperforms the leader CP-CNN^[21] by a margin of 0.41 points in terms of average MAE across scenes.

4.3 UCF_CC_50 Dataset

The UCF_CC_50 dataset, which is a very challenging dataset composed of 50 annotated crowd images with a large variance in crowd counts and scenes, is firstly introduced by IDREES et al.^[28]. The crowd counts range from 94 to 4 543. We follow Ref. [28] and use 5-fold cross-validation to evaluate the proposed method.

We compare our method with five existing methods on UCF_CC_50 dataset using MAE and MSE as metrics in **Table 4**. IDREES et al.^[28] proposed to use multi-source features like head detections, Fourier analysis and texture features. Our approach acquires the best MAE and comparable MSE among existing approaches.

▼ **Table 3. Comparison of RMSN with other four state-of-the-art CNN-based methods on the WorldExpo'10 dataset**

Methods	Scene 1	Scene 2	Scene 3	Scene 4	Scene 5	Average
The approach in Ref. [3]	9.8	14.1	14.3	22.2	3.7	12.9
MCNN ^[1]	3.4	20.6	12.9	13.0	8.1	11.6
Switch-CNN ^[2]	4.4	15.7	10.0	11.0	5.9	9.4
CP-CNN ^[21]	2.9	14.7	10.5	10.4	5.8	8.9
The proposed RMSN	4.1	14.05	9.6	11.8	2.9	8.49

CP-CNN: contextual pyramid convolutional neural network

MCNN: multi-column convolutional neural network

RMSN: real monitoring scene network

▼ **Table 4. Comparative results on the UCF_CC_50 dataset**

Methods	MAE	MSE
The approach in Ref. [28]	419.5	541.6
The approach in Ref. [3]	467.0	498.5
MCNN ^[1]	377.6	509.1
Switch-CNN ^[2]	318.1	439.2
CP-CNN ^[21]	295.8	320.9
The proposed RMSN	291.0	404.6

MAE: mean absolute error

MSE: mean squared error

MCNN: multi-column convolutional neural network

RMSN: real monitoring scene network

4.4 UCSD Dataset

We also evaluate our method on the single-scene UCSD dataset with video stream. This dataset consists of 2 000 labeled frames with size of 158×238. Ground truth is labeled at the center of every pedestrian and the largest number of people is under 46. The ROI and perspective map are provided as well. In order to cover the pedestrian contour, we choose a bivariate normalized distribution kernel shaped ellipse to generate density maps. We follow the same train-test setting in Ref. [13]. The 800 frames from 601 to 1 400 are treated as training set and the rest 1 200 frames as test set. At the test time, MAE

and MSE are used as evaluation metrics.

Table 5 exhibits the comparison of our method with other state-of-the-art methods on UCSD dataset. Crowd count is calculated within the given ROI. The first two methods^{[12], [14]} adopts hand-crafted features, while the rest three are CNN-based. All their results are relatively close due to the comparatively simple scene with low variation of crowd density. Nevertheless, our method outperforms most of the methods, which shows that our approach is also applicable in relatively sparse and single crowd scene.

Fig. 5 shows the application of our method under video information from UCSD dataset. In practical applications, we calculate the pedestrian flow and retention based on the density map. In the velocity map, we can see the small arrows around pedestrian area which represents the direction of pedestrian movement. In the retention map, we use the chromatic area of different colors near head to indicate the length of retention of the corresponding pedestrian, based on the residence time of the pedestrian in a certain place.

5 Conclusions

In this paper, we propose a GANs-based crowd counting network which takes full advantage of excellent performance of GANs in image generation. To better reduce errors caused

▼ **Table 5. Comparative results on the UCSD dataset**

Methods	MAE	MSE
Kernel Ridge Regression ^[12]	2.16	7.45
Cumulative Attribute Regression ^[14]	2.07	6.86
The approach in Ref. [3]	1.60	3.31
Switch-CNN ^[2]	1.62	2.10
The proposed RMSN	1.47	1.98

CNN: convolutional neural network
MAE: mean absolute error

MSE: mean squared error
RMSN: real monitoring scene network



▲ **Figure 5.** One test video information sampled from the UCSD dataset (from left to right and top to bottom, the four images successively denote real time source, density map, velocity map and retention map respectively).

by different scales of the crowd, we propose a novel regularizer which provides a strong regularization constraint on multi-scale crowd density estimation. Extensive experiments indicate that our method achieves the state-of-the-art performance on major crowd counting datasets used for evaluation.

References

- [1] ZHANG Y Y, ZHOU D S, CHEN S Q, et al. Single-image crowd counting via multi-column convolutional neural network [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, USA, 2016: 589 – 597. DOI: 10.1109/cvpr.2016.70
- [2] SAM D B, SURYA S, BABU R V. Switching convolutional neural network for crowd counting [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, USA, 2017. DOI:10.1109/cvpr.2017.429
- [3] ZHANG C, LI H, WANG X, et al. Cross-scene crowd counting via deep convolutional neural networks [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, USA, 2015: 833 – 841. DOI: 10.1109/cvpr.2015.7298684
- [4] GOODFELLOW I J, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial networks [J]. Advances in neural information processing systems, 2014 (3): 2672 – 2680
- [5] MIRZA M, OSINDERO S. Conditional generative adversarial nets [EB/OL]. (2014-11-06) [2018-10-12]. <https://arxiv.org/abs/1411.1784>
- [6] CHEN X, DUAN Y, HOUTHOOFT R, et al. InfoGAN: interpretable representation learning by information maximizing generative adversarial nets [C]//Conference and Workshop on Neural Information Processing Systems. Barcelona, Spain, 2016
- [7] ISOLA P, ZHU J-Y, ZHOU T, et al. Image-to-image translation with conditional adversarial networks [EB/OL]. (2016-09-21) [2018-10-12]. <https://arxiv.org/abs/1611.07004>
- [8] RONNEBERGER O, FISCHER P, BROX T. U-Net: convolutional networks for biomedical image segmentation [C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. Munich, Germany, 2015: 234 – 241. DOI: 10.1007/978-3-319-24574-4_28
- [9] LIN Z L, DAVIS L S. Shape-based human detection and segmentation via hierarchical part-template matching [J]. IEEE transactions on pattern analysis and machine intelligence, 2010, 32(4): 604 – 618. DOI: 10.1109/tpami.2009.204
- [10] WANG M, WANG X. Automatic adaptation of a generic pedestrian detector to a specific traffic scene [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Colorado Springs, USA, 2011. DOI: 10.1109/cvpr.2011.5995698
- [11] WU B, NEVATIA R. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors [C]//Tenth IEEE International Conference on Computer Vision (ICCV'05). Beijing, China, 2005: 90 – 97. DOI: 10.1109/iccv.2005.74
- [12] AN S J, LIU W Q, VENKATESH S. Face recognition using kernel ridge regression [C]//IEEE Conference on Computer Vision and Pattern Recognition. Minneapolis, USA, 2007: 1110 – 1116. DOI: 10.1109/cvpr.2007.383105
- [13] CHANA B, LIANG Z-S J, VASCONCELOS N. Privacy preserving crowd monitoring: counting people without people models or tracking [C]//IEEE Conference on Computer Vision and Pattern Recognition. Anchorage, USA, 2008: 1 – 7. DOI: 10.1109/cvpr.2008.4587569
- [14] CHEN K, LOY C C, GONG S, et al. Feature mining for localised crowd counting [C]//British Machine Vision Conference. Surrey, UK, 2012. DOI: 10.5244/c.26.21

- [15] KONG D, GRAY D, TAO H. A viewpoint invariant approach for crowd counting [C]//International Conference on Pattern Recognition. Hong Kong, China, 2006. DOI: 10.1109/icpr.2006.197
- [16] BANSAL A, VENKATESH K S. People counting in high density crowds from still images [EB/OL]. (2015 - 07 - 30) [2018 - 10 - 12]. <https://arxiv.org/abs/1507.08445v1>
- [17] RABAUD V, BELONGIE S J. Counting crowded moving objects [C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). New York, USA, 2006: 705 - 711. DOI: 10.1109/cvpr.2006.92
- [18] BROSTOW G J, CIPOLLA R. Unsupervised bayesian detection of independent motion in crowds [C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, USA, 2006: 594 - 601. DOI: 10.1109/cvpr.2006.320
- [19] WANG C, ZHANG H, YANG L, et al. Deep people counting in extremely dense crowds [C]//ACM International Conference on Multimedia. Brisbane, Australia, 2015. DOI: 10.1145/2733373.2806337
- [20] BOOMINATHAN L, KRUTHIVENTI S S, BABU R V. CrowdNet: a deep convolutional network for dense crowd counting [C]//ACM Conference on Multimedia. Vienna, Austria, 2016. DOI: 10.1145/2964284.2967300
- [21] SINDAGI V A, PATEL V M. Generating high-quality crowd density maps using contextual pyramid CNNs [C]//IEEE International Conference on Computer Vision. Venice, Italy, 2017
- [22] LI C, WAND M. Precomputed real-time texture synthesis with markovian generative adversarial networks [C]//European Conference on Computer Vision. Amsterdam, Netherlands, 2016: 702 - 716. DOI: 10.1007/978-3-319-46487-9_43
- [23] PATHAK D, KRAHENBUHL P, DONAHUE J, et al. Context encoders: feature learning by inpainting [C]//IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 2536 - 2544. DOI: 10.1109/cvpr.2016.278
- [24] JOHNSON J, ALAHI A, LI F F. Perceptual losses for real-time style transfer and super-resolution [C]//European Conference on Computer Vision. Amsterdam, Netherlands, 2016: 694 - 711. DOI: 10.1007/978-3-319-46475-6_43
- [25] HINTON G E, SALAKHUTDINOV R R. Reducing the dimensionality of data with neural networks [J]. Science, 2006, 313(5786): 504 - 507. DOI: 10.1126/science.1127647
- [26] ZHANG H, SINDAGI V, PATEL V M. Image de-raining using a conditional generative adversarial network [EB/OL]. (2017-01-21) [2018-10-12]. <https://arxiv.org/abs/1701.05957>
- [27] SHEN Z, XU Y, NI B B, et al. Crowd counting via adversarial cross scale consistency pursuit [C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, USA, 2018: 5245 - 5254. DOI: 10.1109/cvpr.2018.00550
- [28] IDREES H, SALEEMI I, SEIBERT C, et al. Multi-source multi-scale counting

in extremely dense crowd images [C]//IEEE Conference on Computer Vision and Pattern Recognition. Portland, USA, 2013: 2547 - 2554. DOI: 10.1109/cvpr.2013.329

Biographies

LI Yiming received the B.S. degree in information engineering from Shanghai Jiao Tong University, China in 2018. From 2018 to the present, he is pursuing his M.S. degree at the Institute of Image Communications and Network Engineering of Shanghai Jiao Tong University. His research interests include crowd counting in dense scenes and the image enhancement, segmentation and texture recognition technologies in materials science.

LI Weihua received the B.S. degree in information engineering from Southwest University, China in 1996. He is currently responsible for the VSS product planning at ZTE Corporation.

SHEN Zan received the B.S. and M.S. degrees in electronics and information engineering from Shanghai Jiao Tong University, China in 2016 and 2019 respectively. He once participated in the internship of Tencent Youtu Lab in 2018. After graduation, he works at Ping An Technology (Shenzhen) Co, Ltd. His research interests include but not limited to deep learning, computer vision, and machine learning. He has published one technical paper in IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

NI Bingbing (nibingbing@sjtu.edu.cn) received the B.S. degree in electronic engineering of Shanghai Jiao Tong University, China in 2005, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore in 2011. He is currently a special researcher, long-hired associate professor and doctoral supervisor at the Department of Electronics of Shanghai Jiao Tong University. His main research areas are computer vision, machine learning and multimedia computing, specializing in face recognition, video understanding, intelligent interactive creative media generation and intelligent medical treatment. He has published more than 100 papers in top international journals and conferences in the field of artificial intelligence/computer vision and more than 40 papers in CVPR, ICCV and other international top computer vision conferences.

← From Page 56

- [27] PAN C H, REN H, DENG Y S, et al. Joint blocklength and location optimization for URLLC-enabled UAV relay systems [J]. IEEE communications letters, 2019, 23(3): 498-501. DOI:10.1109/lcomm.2019.2894696
- [28] BOYD S, VANDENBERGHE L. Convex optimization [M]. Cambridge, U.K: Cambridge University Press, 2004. DOI:10.1017/cbo9780511804441
- [29] BOYD S. Convex optimization II [EB/OL]. (2013-09-11)[2019-10-20]. <http://www.stanford.edu/class/ee364b/lectures.html>

Biographies

ZHANG Pengyu received the B.E. degree from Guangdong University of Technology, China in 2017. He is pursuing his master degree in the School of Information Engineering, Guangdong University of Technology. His research interests include UAV communications, mobile edge computing, and ultra-reliable and low-latency communications.

XIE Lifeng received the B.E. degree from Guangdong University of Technology, China in 2016. He is currently a Ph.D. candidate in the School of Information Engineering, Guangdong University of Technology. His research interests include energy harvesting in wireless communications, wireless information and power transfer, and UAV communications.

XU Jie (xujie@cuhk.edu.cn) received the B.E. and Ph.D. degrees from University of Science and Technology of China in 2007 and 2012 respectively. From 2012 to 2014, he was a Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore. From 2015 to 2016, he was a post-doctoral Research Fellow with the Engineering Systems and Design Pillar, Singapore University of Technology and Design. From 2016 to 2019, he was a professor with the School of Information Engineering, Guangdong University of Technology, China. He is currently an associate professor with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China. His research interests include energy efficiency and energy harvesting in wireless communications, wireless information and power transfer, UAV communications, and mobile edge computing and learning.

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>

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
- Index of Copernicus
- Inspec
- Ulrich's Periodicals Directory
- Wanfang Data

ZTE COMMUNICATIONS

Vol. 18 No. 2 (Issue 70)

Quarterly

First English Issue Published in 2003

Supervised by:

Anhui Publishing Group

Sponsored by:

Time Publishing and Media Co., Ltd.

Shenzhen Guangyu Aerospace Industry Co., Ltd.

Published by:

Anhui Science & Technology Publishing House

Edited and Circulated (Home and Abroad) by:

Magazine House of ZTE Communications

Staff Members:

General Editor: WANG Xiyu

Editor-in-Chief: JIANG Xianjun

Executive Editor-in-Chief: HUANG Xinming

Editor-in-Charge: ZHU Li

Editors: REN Xixi, LU Dan, XU Ye, YANG Guangxi

Producer: XU Ying

Circulation Executive: WANG Pingping

Liaison Executive: LU Dan

Assistant: WANG Kun

Editorial Correspondence:

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

Tel: +86-551-65533356

Email: magazine@zte.com.cn

Website: <https://tech-en.zte.com.cn>

Annual Subscription: RMB 80

Printed by:

Hefei Tiancai Color Printing Company

Publication Date: June 25, 2020

China Standard Serial Number: ISSN 1673-5188
CN 34-1294/TN