Edge QoE: Intelligent Big Data Caching via Deep Reinforcement Learning

Xiaoming He, Kun Wang, Haodong Lu, Wenyao Xu, and Song Guo

Abstract

In mobile edge networks (MENs), big data caching services are expected to provide mobile users with better quality of experience (QoE) than normal scenarios. However, the increasing types of sensors and devices are producing an explosion of big data. Extracting valuable contents for caching is becoming a vital issue for the satisfaction of QoE. Therefore, it is urgent to propose some rational strategies to improve QoE, which is the major challenge for content-centric caching. This article introduces a novel big data architecture consisting of data management units for content extraction and caching decision, improving quality of service and ensuring QoE. Then a caching strategy is proposed to improve QoE, including three parts: (1) the caching location decision, which means the method of deploying caching nodes to make them closer to users; (2) caching capacity assessment, which aims to seek suitable contents to match the capacity of caching nodes; and (3) caching priority choice, which leads to contents being cached according to their priority to meet user demands. With this architecture and strategy, we particularly use a caching algorithm based on deep reinforcement learning to achieve lower cost for intelligent caching. Experimental results indicate that our schemes achieve higher QoE than existing algorithms.

Introduction

Mobile edge networks (MENs) renovate traditional networks at low cost and aim to offer interactive, real-time, and intelligent services to mobile users [1]. With the increasing quality of abundant big data caching services over MENs, a huge amount of workload is caused by the increasing demands of valuable contents and storage. Therefore, the emerging big data architecture and intelligent technology with deep reinforcement learning (DRL) [2] carry out mobile edge computing optimization and caching decision while reducing unnecessary cost. Specifically, data management units, that is, base stations (BSs) [3] equipped with computation and storage capacity, can cope with the requirements of contents. As a consequence, mobile users can get some services from BSs, especially big data caching service, which can not only enhance the quality of service (QoS) from the perspective of MENs, but also provide good quality of experience (QoE) [4] to mobile users.

The concept of QoE is a well-known measurement of the overall perception of the QoS mechanism. The International Telecommunication Union (ITU) has expanded the definition of QoE, which is subjectively the definition perceived by users [4] of *the whole accepted application/service*. For example, Mok *et al.* [5] proposed a QoE prediction model with low complexity and high accuracy for the real-time measurement of the video streaming traffic in future 5G networks.

At the moment, caching service in MENs has attracted researcher's interests. Nevertheless, caching query results with QoE are not easy to obtained, especially in this scenario, which might contain some constraints in caching conditions. The disadvantages of existing work are discussed in the following. A solution to the caching query issue is straightforward use of caching methods including the prediction of popularity distributions, user preferences, and the impact of erroneous information to cache contents [6]. Although the results are accurate, they have relatively poor performance as they fail to take the decision of caching location and caching capacity into consideration. Another straightforward solution is to use the *cache-hit-wonder* method [7] based on context-specific content popularity online. Specifically, *one-hit-wonder* focused on the trade-off between caching insertion rates [8]. Li *et al.* [9] derived the QoE-based caching placement issue in mobile edge for dynamic video streaming. The proposed methods can also solve the caching issue. However, they cannot achieve good effects with QoE. From what we understand, no special attention has been paid to the actual caching. Therefore, we still need to pay more effort to investigating caching strategies in MENs to achieve better QoE.

Nowadays, caching services with QoE have been widely applied in MENs, which help mobile users to acquire improved experience, enhance the quality of contents, and alleviate the traffic pressure of the network. The issue of improving QoE for MENs is the key point in this article. Therefore, we propose a caching strategy, which consists of three parts: caching location decision of a caching-node, caching capacity assessment, and caching sequence choice of valuable contents due to popularity. In short, the reasonable location decision of a caching-node can shorten the distance to mobile users, which reduces the cost of transmission. The rational capacity assessment can match contents with the proper cach-

Xiaoming He is with Hohai University; Haodung Lu is with Nanjing University of Posts and Telecommunications; Kun Wang (corresponding author) is with the University of California, Los Angeles; Wenyao Xu is with the University at Buffalo, State University of New York; Song Guo is with Hong Kong Polytechnic University.

Digital Object Identifier: 10.1109/MNET.011.1900393

0890-8044/20/$25.00 © 2020 IEEE

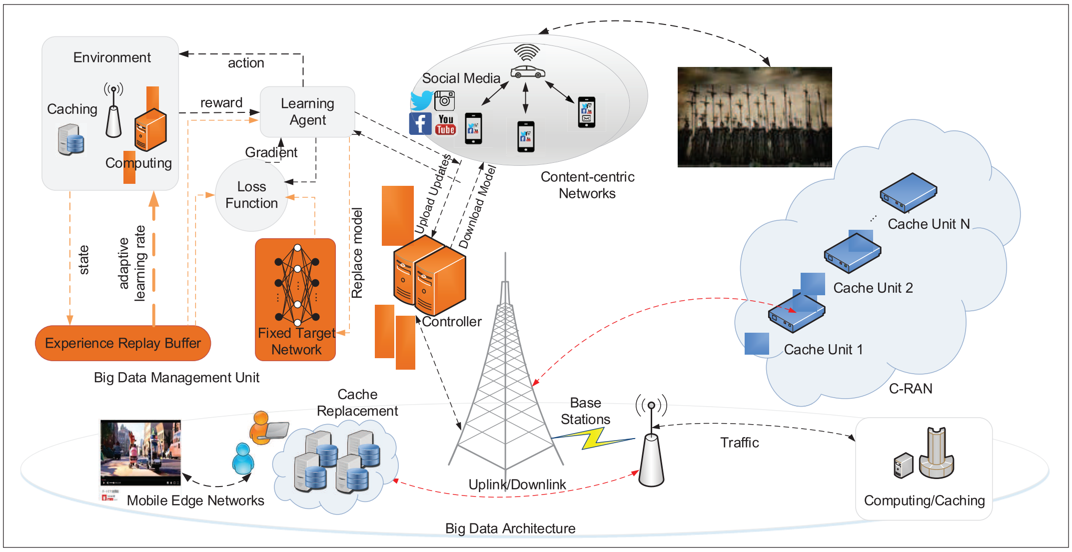

FIGURE 1. Content-centric caching with big data architecture in mobile edge networks (MENs).

ing-node according to their sizes. The caching sequence of contents is the last step in the strategy, and it can adjust the order of contents to mobile users subscribing to their requirements.

However, caching in MENs raises intelligent demands in caching strategies because of a vast volume and variety of valuable contents. Caching algorithms based on DRL [2] in the proposed architecture have some satisfying advantages during the caching decision compared to traditional caching methods. Furthermore, the algorithms show excellent performance in reducing cost. Because of this property, much work has implemented caching algorithms based on DRL in MENs. For example, Wang *et al.* [6] used a deep Q-learning (DQL) method to jointly optimize the communication, caching, and computing, improving the quality of MENs. Generally, DRL technique can provide a powerful support for caching decision in MENs. Based on the proposed caching strategy, MENs can cache valuable contents for big data caching service with better QoE.

Motivation: Given that the varieties of demands continuously grow, the caching strategy is supposed to deal with a series of issues with the satisfaction of QoE. Specifically, caching location decision can be reasonable, caching capacity assessment should be rational, and caching priority choice might be well founded. Designing an intelligent caching algorithm based on DRL for caching decision is an emergency. We use QL to determine the Q-value. Then the Q-value is estimated by a deep neural network (DNN) approximator. Finally, an adaptive learning rate is proposed to accelerate stability in DQL.

The major contributions in this article are as follows.

- To make the big data caching decision in MENs, we propose a novel big data architecture to extract valuable contents and store them with the satisfaction of QoE.
- To emphasize the satisfaction of QoE, we study a caching strategy that contains caching location decision, caching capacity assessment, and caching priority choice. Several schemes based on DRL are implemented as benchmarks to compare their performance separately. The extensive simulation results reveal that the proposed caching approach can significantly improve the value of QoE.
- To study potential future directions, some challenging open issues are pointed out with the proposed content-centric caching strategy.

Big Data Caching in Content-Centric Mobile Edge Networks

Figure 1 shows a novel big data architecture to intelligently manage high-dimensional data. Since user behavior that refers to user demands for big data caching services can be highly acknowledged, the architecture gathers valuable information and caches selective contents according to user demands in MENs. The proposed architecture parallelizes the computation and execution of content extraction and caching placement. As a result, user demands are greatly satisfied, yielding low cost with QoE. The following sections examine the architecture.

Big Data Analysis for Caching in Mobile Edge Networks

A novel big data architecture is proposed to analyze data in MENs. The architecture is used to extract valuable contents for caching. The requirements of this architecture for big data analysis are as follows.

High-dimensional data preprocessing: To perform caching decision, data processing units inside the big data architecture should be competent in reading and combining data from data sources, as well as transferring intelligent insights quickly and reliably. According to this reason, the raw data can be exported into a Hadoop Distributed File System (HDFS) via data-integrated methods after mirroring the data streaming interface. After that, data has to be cleaned, parsed, and formatted. First, raw data including some inappropriate, malfunctioning, and inconsistent packets and character encodings can be cleaned. The second step is to extract the required headers from the raw data because of the data analysis and modeling requirements. Finally, the parsed data can be encoded in HDFS for appropriate storage.

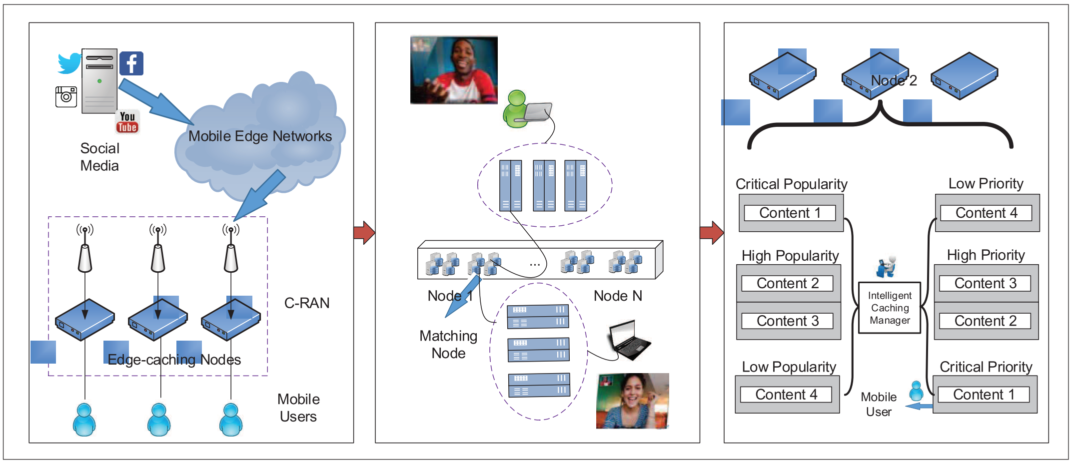

FIGURE 2. The proposed caching strategy for mobile edge networks.

Data intelligence analysis: Formatted data analysis techniques in control and data planes can be applied to offload computation. The aim of data analysis is to find relationships between control and data packets. That is, users located in the control plane request contents from data packets via intelligent operations. Specifically, before the model is downloaded from the controller, the user trains it based on local training data and uploads the updated weights of the big data management unit. The results of analysis can be stored, reused, and re-formatted for further analysis using appropriate tools. Moreover, tables and graphs can also represent the data in a visual format for ease of understanding.

To sum up, big data architecture can be adaptive to enhance the QoS for caching in MENs.

Content-Centric Caching in Mobile Edge Networks with QoE

With the application/service of big data architecture in content-centric MENs, it needs further optimization from the perspective of mobile users. Fortunately, reasonable caching can greatly improve the satisfaction of QoE. At present, an ocean of contents that have great popularity are accessed very frequently by mobile users. On the other hand, the types of popular contents in a caching-node differ because of the demands of mobile users. Therefore, a variety of caching services should be optimized by adjusting the location of a caching-node and popular contents in Fig. 2.

Services in MENs can bring great benefits while reducing the cost of QoE, which are specified in the next section. Different kinds of popular contents in a caching-node account for a large proportion. Typically, content-centric caching can bring key services, including Voice over Internet Protocol (VoIP).

Challenge and Motivation

Although the content-centric caching in MENs brings several benefits with the satisfaction of QoE, we are still faced with some challenges. The demands of mobile users increase rapidly, and huge amounts of contents need to be cached intelligently. Fortunately, the improvements of content-centric caching strategies can effectively deal with these issues and improve the satisfaction of QoE.

Specifically, the caching strategy is supposed to deal with a series of issues. First, the decision of caching location can be reasonable. Second, the strategy guarantees the rationality of adequate contents under the storage capacity of a caching-node. Third, the strategy also determines the cache due to the priority of contents located in a caching-node. Designing an intelligent caching algorithm based on DRL for caching decision is necessary. As the requirements are stochastic, we use QL to determine the Q-value. Constrained by a maximum QoE value, estimating the Q-value can be conducted in a DNN approximator. Adaptive learning rate is proposed to ensure the Q-value accuracy and accelerate stability in DQL.

Caching Strategy for Content-Centric Mobile Edge Networks

Content-centric caching strategy is given in detail for MENs including caching location decision, caching capacity assessment, and caching priority choice, each of which is specified as follows.

Caching Location Decision

The decision of where to put a caching-node should be made in MENs as a strategy for content-centric caching, which is of great importance for QoE. The idea is that locating caching-node at the radio access network (RAN) [3] of BSs is a wise decision, as shown in Fig. 2. Different from the previous cloud data center model, the user demands are greatly increasing, and a great experience can be achieved if a caching-node can be deployed in the RAN. The RAN helps store popular contents in BSs because it empowers local communication and efficiently alleviates the end-to-end cost. As a consequence, it brings many benefits; for example, the distance of transmission has been shortened compared to traditional caching-node deployments. Thereby, the cost in the RAN is smaller when data is served from the caching-node.

The proposed caching strategy not only brings the advantages mentioned in the previous part, but can acknowledge the next requests of users according to the current requests. Each request has potential information about the next request made by the user. In the proposed strategy, potential information is used to cache contents requested by the same user. Thus, the purpose mainly serves the content of a specific user at the first time. The advantage is that it can increase cache-hit ratio.

Caching Capacity Assessment

After locating caching-nodes in the RAN, we can assess which contents deserve to be cached according to the following statements.

First, caching capacity [10] in the RAN is smaller due to financial issues. Therefore, it is important to calculate the size of contents to decide whether to cache it. Although caching can improve the performance in MENs, it is impossible to cache everything. Second, the development of hardware technology is slow for caching, while the requirements of contents for mobile users are increasing dramatically. Therefore, the assessment of what to cache should be made carefully.

The key point of our proposed strategy is ensuring the capacity of a caching-node to match the contents. For instance, if a huge number of users choose to stream video, video segments with high bit rate cannot be cached at a caching-node. Even though they can be partly cached, users do not tend to meet their demands with poor QoE. Meanwhile, more lower-bit-rate segments are cached so that more specified video requests can benefit from them. Therefore, contents with adequate sizes ought to be cached in the deployed caching-node.

Caching Priority Choice

The caching sequence of contents is equally important for the contents of a certain size that is not bigger than the capacity of caching-node.

We can then cache contents according to the popularity only after contents within a certain scope is selected. The popularity of contents determines the priority of them, aiming to determine caching sequence. The data management unit in Fig. 1 acts as a decision-maker for caching, i.e., video segment. This decision-maker considers the context information on content popularity characteristics [11], determining which contents are inclined to be cached in sequence for future demands. For example, according to the network provider, the user demands for video is growing rapidly. Even if the popular contents are deployed in MENs, it also cannot completely satisfy the demands of mobile users. Therefore, we need to select hot videos and put them into caching-nodes, reducing the cost of caching and improving the QoE level.

In conclusion, the proposed caching strategy in MENs can reduce the cost of caching and enhance the satisfaction of QoE.

Case Study: Caching in Content-Centric Networks

In this section, a content-centric network (CCN) [12, 13] is proposed to apply the caching strategy in MENs. In such a scenario as that shown in Fig. 1, big data architecture extracts contents and stores them to a caching-node. Then mobile users can receive contents from the caching-node according to their preferences. Consequently, the proposed caching strategy takes QoE into account and can also minimize the caching cost.

Even though they can be partly cached, users do not tend to meet their demands with poor QoE. Meanwhile, more lower-bit-rate segments are cached so that more specified video requests can benefit from them. Therefore, contents with adequate sizes ought to be cached in the deployed caching-node.

The Measurement of QoE Level

QoE is seen as a vital measurement in our case study. Considering the QoS and user preferences, the QoE level is evaluated in a CCN. Note that varying QoE performance of different caching algorithms under our proposed caching strategy is studied in this article instead of focusing on building new QoE models [14]. Therefore, the measurement model of QoE is adopted to return a QoE value if a low cost is given. Then we can give a QoE value of the maximum caching capacity, popularity, and bandwidth. As a result, this model uses both services and algorithm parameters for evaluating the QoE under consideration, providing an appropriate performance metric.

Caching Algorithm

In a CCN, the state of the system is high-dimensional and dynamic since a data management unit manages the caching and computing resources. Therefore, DQL [9] is selected to achieve the maximum rewards via optimizing actions.

As shown in Fig. 1, the agent and environment can mutually interact in DQL. At one decision period, the agent perceives state of caching or computing from the current environment. Indicated by the given policy, the agent selects action to work and offloads the computation task from users to a computing unit. For example, the agent decides a caching-node to cache the content requested by users. The environment turns into a new state. The learning agent gains the corresponding reward. Q-value is optimized by the reward via a temporal difference method [9] in QL. Then the DNN with the experience replay buffer [9] and fixed target network [9] can be trained to estimate Q-value referring to the QoE value through minimizing the loss function. Finally, QL updates its adaptive learning rate [9] to ensure the training stability. The critical procedure of such an algorithm is explained as follows.

Step 1: The stochastic policy with parameters is initialized by the agent. Then the definition of value function approximation is proposed to parameterize. The target network is initialized using the weights.

Step 2: The agent produces an action in light of current state and policy. The next state in the environment and the reward are observed in QL.

Step 3: QL is employed to the experience replay buffer for storing the tuple I. A minibatch of I is sampled from the replay buffer at random.

Step 4: For any sample, the DNN estimates the Q-value. Then the DNN with the averaged value updates the parameter over the minibatch, minimizing the loss function. Finally, QL updates its adaptive learning rate using it to ensure the training stability.

Experimental Results

To validate the proposed QoE model and caching algorithm, performance evaluation is given quantitatively.

QoE models named QoE-1 metric, QoE-2 metrics, QoE-3 metrics, and QoE-4 metrics are proposed to evaluate their performance by Matlab. Then schemes named QL, DQL, DQL-L, and double DQN (DDQN) [2, 9] are implemented by Matlab and evaluated using Tensorflow. Finally, the performance of our proposed QoE models and schemes can be compared with each other. We take the general simulator since it is used to input realistic traces which are federal communications commission databases [15].

Figure 3 compares the efficiency of four QoE models with different metrics. We can see that the QoE-1 metric model achieves the lowest efficiency. The QoE-2 metrics model has a lower efficiency than the QoE-4 metrics model. When we adopt the QoE-3 metrics model, the rewards are the highest. It appears that the efficiency of a QoE model depends, in a certain way, on three metrics.

The performance of the four schemes is compared in Fig. 4. The measurements mainly include stability, and the indicator is the change of stability with the increase of iterations. The results can be seen, and the stability of DDQN is poor. DQL-L shows its unique performance due to updating the adaptive learning rate. Results in all cases approve that the QL and DQL schemes provide better stability than the DDQN scheme. The conclusion of this experiment partly accounts for the choice of the DQL-L scheme.

The rewards in terms of QoE values of the proposed schemes are compared and computed with a specific video streaming using the aforementioned QoE model [14]. Different schemes associated with the quality of video are chosen by mobile users. Then we execute the schemes to collect the QoE measurements.

The efficiency of schemes for each test case linked to the cost is evaluated by experiment. Note that the cost is the most important indicator in our proposed schemes. The cost in the caching process includes computing, caching, and communication. Although these schemes can contribute greatly to saving cost, there are some gaps. The results can be seen in Fig. 5. DDQN is not efficient since the average cost is higher than the others, while DQL-L has the best performance when the caching capacity is large. In terms of average cost, DQL-L shows excellent performance since the stability is the highest and brings the best result.

As shown in Figure 6, the average QoE values of QL, DQL, DQL-L, and DDQN are 45, 76, 82, and 59, the maximum QoE values of QL, DQL, DQL-L, and DDQN are 50, 85, 90, and 65, and then their minimum QoE values are 39, 67, 73, and 57, respectively. It can be seen that the DQL-L scheme yields the highest QoE value since the caching cost is the lowest. On the contrary, because caching cost is higher than other mechanisms, the QL scheme generates the lowest QoE value.

In conclusion, we can always perceive from Figs 5 and 6 that the QoE performance of these schemes is negatively correlated to the efficiency of cost.

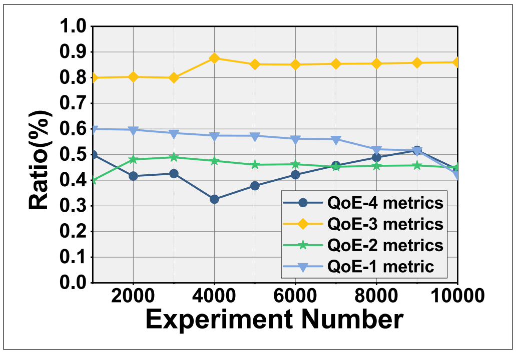

FIGURE 3. The QoE model performance comparison under different metrics.

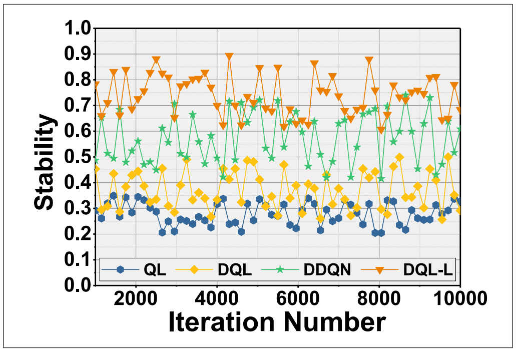

FIGURE 4. The stability comparison under different schemes.

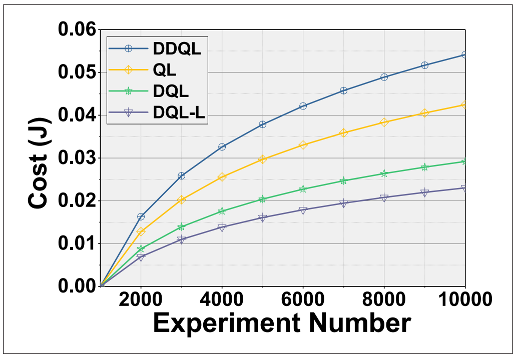

FIGURE 5. The cost comparison under different schemes.

Open Issues

We have introduced a big data architecture and proposed a novel caching strategy for MENs. The following open issues are well worth studying in the future:

- **On-Device Caching:** If on-device caching and communication between the bottom devices get popular, it is worth searching for opportunistic coordination among caching-nodes. The files will be transferred from influential users to requesting users via device-to-device (D2D) communications.
- **Privacy and Security:** If the invaders obtain the privilege of a caching manager, the whole network is likely to be hazardous. Therefore, how to keep the caching manager work from the attack is a hot direction.
- **Green MENs:** Since edge facilities consume over 70 percent of the cable operator energy, substantial innovation is requested to further enhance the energy efficiency in MENs.

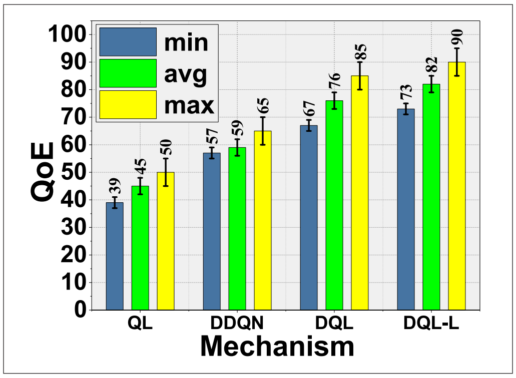

FIGURE 6. The QoE-value comparison under different schemes.

- **Heterogeneity:** In future edge environments, the heterogeneity in applications of networking, communication, and caching becomes a critical issue. The challenge is that handling the heterogeneity should be investigated under a unified network architecture.

Conclusion

In this article, we introduce a big data architecture and present a novel caching strategy achieving better QoE value for mobile users. As a case study of this strategy, we design an intelligent caching algorithm with deep reinforcement learning in MENs to acquire the lower cost of caching with the higher QoE value. Finally, the experimental results indicate the efficiency of our proposed QoE model and scheme compared to the existing caching models and strategies.

References

[1] X. He, K. Wang, and W. Xu, "QoE-Driven Content-Centric Caching with Deep Reinforcement Learning in Edge-Enabled IoT," *IEEE Computational Intelligence Mag.*, vol. 4, no. 4, Nov. 2019, pp. 12–20.

[2] Y. Wang *et al.*, "Traffic and Computation Co-Offloading with Reinforcement Learning in Fog Computing for Industrial Applications," *IEEE Trans. Industrial Informatics*, vol. 15, no. 2, Feb. 2019, pp. 976–86.

[3] K. Guo *et al.*, "On the Interplay Between Communication and Computation in Green C-RAN with Limited Fronthaul and Computation Capacity," *IEEE Trans. Commun.* DOI: 10.1109/TCOMM.2018.2811479.

[4] X. He *et al.*, "Green Resource Allocation Based on Deep Reinforcement Learning in Content-Centric IoT," *IEEE Trans. Emerging Topics in Computing*. DOI:10.1109/TETC.2018.2805718, Feb. 2018.

[5] R. Mok, R. Chang, and W. Li, "Detecting Low-Quality Workers in QoE Crowdtesting: A Worker Behavior-Based Approach," *IEEE Trans. Multimedia*, vol. 19, no. 3, Oct. 2017, pp. 530–43.

[6] K. Wang *et al.*, "Crowdsourcing-Based Content-Centric Network: A Social Perspective," *IEEE Network*, vol. 31, no. 5, Sept./Oct. 2017, pp. 28–34.

[7] X. He *et al.*, "QoE-Driven Big Data Architecture for Smart City," *IEEE Commun. Mag.*, vol. 56, no. 2, Feb. 2018, pp. 88–93.

[8] N. Carlsson and D. Eager, "Ephemeral Content Popularity at the Edge and Implications for On-Demand Caching," *IEEE Trans. Parallel and Distributed Systems*, vol. 28, no. 6, June 2017, pp. 1621–34.

[9] S. Li *et al.*, "Popularity-Driven Content Caching," *Proc. IEEE INFOCOM 2016*, San Francisco, CA, Apr. 2016.

[10] C. Xu *et al.*, "Making Big Data Open in Edges: A Resource-Efficient Blockchain-Based Approach," *IEEE Trans. Parallel and Distributed Systems*, vol. 30, no. 4, Apr. 2019, pp. 870–82.

[11] K. Wang *et al.*, "Cluster Frameworks for Efficient Scheduling and Resource Allocation In Data Center Networks: A Survey," *IEEE Commun. Surveys & Tutorials*, vol. 20, no. 4, Nov. 2018, pp. 3560–80.

[12] X. Zhou *et al.*, "Reinforcement Learning-Based Adaptive Resource Management of Differentiated Services in Geo-Distributed Data Centers," *IEEE/ACM IWQOS 2017*, Vilanova, Spain, June 2017.

[13] R. Li and H. Asaeda, "A Blockchain-Based Data Life Cycle Protection Framework for Information-Centric Networks," *IEEE Commun. Mag.*, vol. 57, no. 6, June 2019, pp. 20–25.

[14] C. Xu *et al.*, "Renewable Energy-Aware Big Data Analytics in Geo-Distributed Data Centers With Reinforcement Learning," *IEEE Trans. Network Science and Engineering*. DOI: 10.1109/TNSE.2018.2813333, Mar. 2018.

[15] M. J. Marcus, "Unlicensed Cognitive Sharing of TV Spectrum: The Controversy at the Federal Communications Commission," *IEEE Commun. Mag.*, vol. 43, no. 5, May 2005, pp. 24–25.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (2018YFB1004700), the National Science Foundation of China (61872195, 61832005), and the U.S. National Science Foundation (1718375).

Biographies

Xiaoming He (isxmhe@gmail.com) received his M.E. degree from Nanjing University of Posts and Telecommunications (NJUPT), China. He is a Ph.D. student in the College of Computer and Information, Hohai University,China. His current research interests include deep reinforcement learning, edge computing, and edge caching.

Kun Wang [SM'17] (wangk@ucla.edu) received two Ph.D. degrees from NJUPT in 2009 and the University of Aizu in 2018. He was a postdoctoral fellow at the University of California Los Angeles (UCLA) from 2013 to 2015, and a research dellow at the University of Aizu in 2016. He is currently a senior research professor at UCLA and a professor at NJUPT. His research interests include big data, wireless communications and networking, energy Internet, and information security technologies.

Haodong Lu (ihaodonglu@gmail.com) is an M.Sc. student in the College of Internet of Things, NJUPT. His current research interests include distributed processing, parallel computing, and deep reinforcement learning.

Wenyao Xu (wenyaoxu@buffalo.edu) received his Ph.D. degree from UCLA in 2013. He got his M.S. degree in 2008 and B.S. degree in 2006 from Zhejiang University (both with honors), China. He is an associate professor in the Computer Science and Engineering Department at the State University of New York (SUNY) at Buffalo, where he founded and directs the Embedded Sensing and Computing Group. He has published over 150 technical papers, co-authored 2 books, and is a named inventor on many international and U.S. patents.

Song Guo (song.guo@polyu.edu.hk) [SM'11, F'19] received his Ph.D. degree in computer science from the University of Ottawa. He is currently a full professor in the Department of Computing, Hong Kong Polytechnic University. Prior to joining PolyU, he was a full professor with the University of Aizu, Japan. His research interests are mainly in the areas of cloud and green computing, big data, wireless networks, and cyber-physical systems.