EAI Endorsed Transactions Preprint Research Article

Deep Model Training and Deployment in Heterogeneous IoT Networks

Bowen Lu[1,*], Shiwei Lai[2], Yajuan Tang[2], Tao Cui[2], Chengyuan Fan[3], Jianghong Ou[4], and Dahua Fan[4]

[1]Shantou University, Shantou, China.
[2]Guangzhou University, Guangzhou, China.
[3]Software Engineering Institute of Guangzhou, Guangzhou, China
[4]AI Sensing Technology, Foshan, China.

Abstract

As a typical form of machines learning, deep learning has attracted much attention from researchers. It can independently construct (train) basic rules according to the sample data in the learning process. Especially in the field of machine vision, neural networks are usually trained by supervised learning, that is, by example data and predefined results of example data. In this paper, we firstly overview the current research progress on the deep model training and deployment on the scalable Internet of Things (IoT) networks, by taking into account both the latency and energy consumption. We then summarize the existing challenges on the model training and model deployment on the scalable IoT devices. We further give some feasible solutions to solve the challenges on the model training and model deployment on the scalable IoT devices. The study in this paper can serve as an important reference for the development of deep model training and model deployment for scalable IoT networks.

Received on 03 December 2022; accepted on 29 December 2022; published on 11 January 2023

Keywords: Deep learning, deep model training, deep model deployment.

doi:10.4108/eetmca.v7i3.2899

1. Introduction

AI usually refers to the architecture constructed by machines (usually computer programs) by imitating or copying human behavior [1–3]. The term “AI" covers many sub domains, such as expert systems, pattern analysis systems, or robots. AI based systems will use different methods to simulate or model human behavior and decision-making structure, including statistical algorithms, heuristic programs, artificial neural networks (ANN) or other machine learning derivative technologies [4–6].

Machine learning is a sub field of AI, which can be classified into “supervised learning" and “unsupervised learning" [7–9]. In supervised learning, the sample data of learning contains both the input data and the corresponding expected results (such as classification) [10–12], while in unsupervised learning, the system should determine the possible results of the input data by itself [13–15]. As a typical form of machines learning, deep learning has attracted much attention from researchers [16–18]. As an artificial neural network, it can independently construct (train) basic rules according to the sample data in the learning process. Especially in the field of machine vision, neural networks are usually trained by supervised learning, that is, by example data and predefined results of example data. Deep learning uses some form of artificial neural network (ANN) technology, so it must be trained with sample data first [19–21]. The trained ANN can be used to perform related tasks. The process of using trained ANN is called “inference". In reasoning, ANN will evaluate the data provided according to the learned rules. For example, it is possible to evaluate whether an object in an input image has a defect or not.

*Corresponding author. Email: 19bwlu@stu.edu.cn

2. Analysis of the current state of research

The acquisition and deployment of intelligent models is the core of implementing B5G edge intelligence [22–24]. However, the training of intelligent models relies on superb computing devices, while large-scale intelligent models are difficult to deploy in IoT devices where computing and storage resources are extremely scarce. In this regard, researchers have conducted extensive and in-depth research work to propose a series of efficient and feasible solutions from various metrics such as training and inference latency and energy consumption.

We should study the parallel training of intelligent models. To address the difficulty of storing massive data in a single node in a cloud computing center, researchers investigated a model training strategy based on data parallelism, proposed a partitioning scheme for the data set, and subsequently allocated training samples according to the memory capacity of each computing node within the central cloud, and theoretically demonstrated the convergence of parallel training based on this scheme [25–27]. To address the difficulty of training large-scale intelligent models in a single node, researchers proposed a model random partitioning strategy for the structural characteristics of neural networks, which randomly partitioned the model into multiple copies and stored them in different computing nodes, and optimized the transfer process of gradient parameters within computing nodes based on the global topology to improve the training efficiency of intelligent models. In addition, the researchers adopt an asynchronous hierarchical training method for the problem of device dropout during parallel training of the model, and also combine the temporal update of the global with the generation of gradient parameters, which greatly accelerates the training process and improves the robustness of the intelligent learning system.

We should study the communication mechanism of distributed training. For the distributed parameter transmission process, in order to minimize the bandwidth consumption of the transmission process, the researchers investigate the depth gradient compression strategy to sparse the gradients and then send some of the gradient elements at each iteration as a way to reduce the communication overhead of computing node interactions. In addition, the researchers investigate the impact of wireless networks on distributed training, using over-the-air computational transmission compression quantization parameters for multi-access channels to minimize transmission errors by regulating power to achieve low-energy and low-latency distributed training. In addition, the gradient merging transmission of adjacent layers of the deep network can be used instead of the traditional hierarchical transmission to improve the bandwidth utilization, and the iterative delay of the training process can be significantly reduced by optimizing the resource scheduling of the merged transmission.

We should further study efficient model deployment and inference for mobile devices. To overcome the shortage of computing power in mobile devices, the previous work reduces the time complexity of operations by building new convolutional operators and optimizes the feature extraction of intelligent models to significantly reduce the end-to-end inference latency while ensuring a certain accuracy. The researchers further explored the use of pruning techniques to remove redundant information from intelligent models and establish a trade-off between latency, energy consumption and model size to achieve flexible and efficient deep model inference in mobile devices. In addition, to reduce the inference latency, a cloud-based fusion model deployment scheme is proposed to utilize the computing power of mobile devices and the central cloud for accelerated inference, and the scheme also reduces the inference latency and energy consumption by scheduling the computing volume based on the real-time channels of wireless networks. In this aspect of research, the researchers propose a training and deployment mechanism based on multiple exit points for the heterogeneity of computing power presented by mobile devices in the B5G edge intelligence network, and realize real-time scheduling of resources through an efficient greedy strategy, which significantly reduces the overall latency of the intelligent system.

3. Challenges on model training and deployment across data centers

From the analysis of the above research status, it can be seen that the existing research has conducted in-depth research on the training and deployment of intelligent models based on the central cloud, and has conducted in-depth analysis from multiple perspectives, such as training and deployment latency, energy consumption, communication and computation efficiency, and data security, etc. The performance of model training and deployment has been significantly improved by combining the optimal scheduling of communication and computation resources. These research works provide important references for the training and deployment of intelligent models in B5G edge intelligence networks. However, B5G edge intelligence networks can also be applied to high-speed mobile scenarios, where the cross-data center characteristics have an important impact on the training and deployment of intelligent models. It is a difficult challenge to design a new intelligent model training and deployment scheme for B5G edge

intelligence networks by deeply exploring the cross-data center characteristics under high mobility and combining over-the-air computing and federal learning technologies.

4. Feasible solutions to model training and deployment across data centers

First, we study the efficient aggregation and processing of intelligent models to achieve fast real-time response and decision making at the control layer and improve the efficiency of distributed model training across data centers. Consider an over-the-air federation learning system consisting of a parameter server and $L \geq 0$ edge data centers. Under the coordination of the parameter server, the edge data centers aggregate and collaborate to train shared machine learning models through wireless updates. Let the parameter vector $\boldsymbol{w}$ denote this federated learning model, where q denotes the model size; and let $\mathcal{D}_l$ denote the local dataset of edge data center l, where the dth sample and its label are denoted by $\boldsymbol{x}_d$ and $\boldsymbol{y}_d$, respectively. Then, the local loss function of the model vector $\boldsymbol{w}$ on $\mathcal{D}_l$ is

$$F_l(\boldsymbol{w}) = \frac{1}{|\mathcal{D}_l|} \sum_{(\boldsymbol{x}_i, y_j) \in \mathcal{D}_l} f\left(\boldsymbol{w}, \boldsymbol{x}_d, y_d\right) + \rho R(\boldsymbol{w}), \tag{1}$$

where $f\left(\boldsymbol{w}, \boldsymbol{x}_d, y_d\right)$ denotes the sample-by-sample loss function that quantifies the prediction error of model $\boldsymbol{w}$ in sample $\boldsymbol{x}_d$ for its labels y_d, and $R(\boldsymbol{w})$ is a strongly convex regularization function with hyperparameters $\rho \geq 0$ as scaling factors. For the convenience of the representation, $f_i(\boldsymbol{w})$ is replaced by $f\left(\boldsymbol{w}, \boldsymbol{x}_d, y_d\right)$. Thus, the global loss function for all distributed data sets is $F(\boldsymbol{w}) = \frac{1}{L} \sum_{l \in L} D_l F_l(\boldsymbol{w})$ where $\mathcal{D} = \cup_{l \in \mathcal{L}} \mathcal{D}_l$, and for simplicity of notation, it is assumed that the size of the local data set in all edge data centers is the same, i.e., $D_l = |\mathcal{D}_l| = \tilde{D}$. The goal of the model training process is to minimize the global loss function:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} F(\boldsymbol{w}). \tag{2}$$

In addition to uploading all local data directly to the parameter server for centralized training,the learning process can be implemented iteratively in a distributed manner based on the gradient averaging method, i.e., as shown in Fig. 1. In each communication process τ, the machine learning model is represented by $\boldsymbol{w}^{(\tau)}$ and each edge data center can use its local dataset $\mathcal{D}_l$ to compute the local gradient $\boldsymbol{g}_l^{(\tau)} = \frac{1}{|\mathcal{D}_l|} \sum_{(\boldsymbol{x}_d, \boldsymbol{y}_d) \in \mathcal{D}_l} \nabla f_d\left(\boldsymbol{w}^{(\tau)}\right) + \rho \nabla R(\boldsymbol{w})$, where ∇ is the gradient operation and it is assumed that the whole local dataset is used to estimate the local gradient. Next, the edge data center sends all local gradients simultaneously to the parameter server and averages them to obtain the global gradient $\overline{\boldsymbol{g}^{(\tau)}} = \frac{1}{L} \sum_{l \in \mathcal{L}} \boldsymbol{g}_l{}^{(\tau)}$. Then, the parameter server broadcasts the global gradient estimate to the edge data center, and the edge device can update the local model based on this estimate: $\boldsymbol{w}^{(\tau+1)} = \boldsymbol{w}^{(\tau)} - \eta \cdot \overline{\boldsymbol{g}^{(\tau)}}$, where η is the learning rate. The above learning process is repeated until the convergence criterion is satisfied or the maximum number of iterations is reached.

An efficient and feasible scheme is to make full use of the superposition characteristics of waveforms in air computing and an efficient model/gradient aggregation technology based on air computing should be studied. Let $\hat{h}_l^{(\tau)}$ denote the complex channel coefficient from the edge data center l to the parameter server in the communication process τ, then let $h_l^{(\tau)} = |\hat{h}_l^{(\tau)}|$. When uploading the gradient, all edge devices transmit on the same time-frequency block, so the received aggregated signal is

$$\boldsymbol{y}^{(\tau)} = \sum_{l \in \mathcal{L}} h_1^{(tau)} \sqrt{p_1^{(\tau)}} \boldsymbol{g}_1^{(\tau)} + \boldsymbol{z}^{(\tau)}, \tag{3}$$

where $p_l^{(\tau)}$ is the transmission power, $\boldsymbol{z}^{(\tau)}$ is the additive Gaussian white noise, subject to $\boldsymbol{z}^{(\tau)} \sim \mathcal{C}N(0, N_0 \boldsymbol{I}_0)$, in which N_0 is the noise power density and $\boldsymbol{I}_0$ is the identity matrix. Therefore, the global gradient estimation of the parameter server is $\hat{\mathbf{g}}^{(\tau)} = \frac{\mathbf{y}^{(\tau)}}{L}$.

The edge data center can adaptively adjust its transmission power to enhance learning performance. In addition, each edge is limited by the maximum transmit power $\tilde{P}_l$, i.e., $p_l^{(\tau)} \leq \tilde{P}_l, \forall l \in \mathcal{L}, \forall l$, and the average power constraint $\tilde{P}_l$, i.e., $\frac{1}{L} \sum_{l \in \mathcal{L}} p_l^i(\tau) \leq \tilde{P}_l, \forall l \in \mathcal{L}$. In general, the above constraints need to satisfy $\tilde{P}_l \leq \tilde{P}_l, \forall l \in \mathcal{L}$.

Secondly, the training accuracy and convergence rate are established as the performance metrics of federated learning, and an accurate and reliable mathematical model is established according to the basic computing theory, communication theory and federated learning framework. Let τ_0 be the required total number of communications, and use $F^{(\tau+1)}$ to simplify $F\left(\boldsymbol{w}^{(\tau+1)}\right)$, and let $F^\star = F\left(\boldsymbol{w}^\star\right)$. After τ_0 communications, the optimal gap of the loss function, that is, $F^{(\tau_0+1)} - F^\star$, can obtain an upper bound related to the transmission power $\left\{p_1^{(\tau)}\right\}$, the learning rate η and τ_0: $F^{(\tau_0+1)} - F^\star \leq \mathfrak{G}\left(\left\{p_1^{(\tau)}\right\}, \eta, \tau_0\right)$. Since $F^\star$ is a constant, the problem of minimizing $F\left(\mathbf{w}^{(\tau_0)}\right)$ can be approximated as minimizing the upper bound $\mathfrak{G}\left(\left\{p_l^{(\tau)}\right\}, \eta, \tau_0\right)$.

At the same time, according to different application scenarios, the optimization parameters such as transmission power at the transmitting end and learning

rate at the receiving end are designed. According to different performance metrics, an optimization model of cross data center federated learning based on over the air computing is constructed. According to the above model, the obtained optimization problem is modeled as:

$$\begin{aligned} \min_{\left\{p_l^{(\tau)}\right\},\eta,N} \quad & \mathcal{G}\left(\left\{p_k^{(\tau)}\right\},\eta,\tau_0\right) \\ \text{s.t.} \quad & \tilde{P}_l \le \bar{P}_l, \forall l \in \mathcal{L} \\ & \frac{1}{\tau_0}\sum_{\tau=1}^{\tau=\tau_0} p_l^{(\tau)} \le \tilde{P}_l, \forall l \in l \\ & \left(\left\{p_l^{(\tau)}\right\},\eta,\tau_0\right) \in \mathbb{S} \end{aligned} \tag{4}$$

The constraint space $\mathbb{S}$ may vary according to different task requirements. However, this problem is a nonconvex optimization problem with large dimension of design parameters and high complexity of solution. Non-convex optimization, online optimization and other methods can be used to reasonably allocate wireless resources (such as time, bandwidth and transmission power) in combination with deep reinforcement learning and other means, so as to improve the convergence speed of the model and realize efficient air federation edge learning while ensuring the training accuracy.

Further, the deployment and deduction of lightweight intelligent models based on model pruning should be studied to minimize the end-to-end delay and energy consumption in the deduction process. Specifically, according to the different application requirements of B5G edge intelligence, the edge end joint inference method is proposed to perform network cutting / pruning on the artificial intelligence model, so as to obtain the lightweight model under different compression rates from the heavyweight complex model. Then, the artificial intelligence model is mixed and deployed in the edge server and the terminal, so that the distributed computing resources can be used to realize rapid model inference. Take the deep neural network as an example, as shown in Fig. 2. In the process of model inference, the neural network can be cut into two layers, and the bottom network is carried out at the terminal to extract the feature information of real-time data, and compress and transmit it to the edge server. The upper layer network performs model inference at the edge server. Finally, the edge server sends the inference result to the terminal device.

It is assumed that there are L terminal devices in the system. Under the coordination of the edge server, they cooperate to complete the inference of the intelligent model $\mathbb{N}$. According to different application requirements, $\mathbb{N}$ is compressed by model compression technologies such as network cutting / pruning, and L lightweight sub networks $\{\mathbb{N}_l\}$ and heavy sub networks $\mathbb{N}_0$ are obtained. The structure is expressed as $\mathcal{N}_l$, the compression rate is $c(\mathcal{N}_l)$, and the computing storage and other resources allocated to it by the node are $b(\mathcal{N}_l)$. The energy consumption of completing the sub network is $e(\mathcal{N}_l)$ and the delay $\tau(\mathcal{N}_l)$ are both related to $b(\mathcal{N}_l)$. Define $a(\mathcal{N}_l)$ as the accuracy index of the sub network, and Ω as the deployment strategy of L lightweight sub networks at different terminals.

Finally, we should optimize the network wide communication computing resources and model deployment strategy. In order to cope with different performance metric constraints, assuming that the maximum computing resource given by the terminal l to the model $\mathbb{N}_k$ is B_l, the energy consumption limit $\gamma_{E,l}$ and the delay limit $\gamma_{th,l}$, the problem can be expressed as a multi-objective optimization problem:

$$\begin{aligned} \min_{(b(\mathcal{N}_l),c(\mathcal{N}_l),\Omega,G)} \quad & \sum_{l\in\mathcal{L}} \varphi_l \left(\alpha_1 e\left(\mathcal{N}_l\right) + \alpha_2 l\left(\mathcal{A}_l\right) + \alpha_3 b\left(\mathcal{A}_l\right)\right) \\ \text{s.t.} \quad & e\left(\mathcal{N}_l\right) \le \gamma_{E,l} \\ & l\left(\mathcal{N}_l\right) \le \gamma_{th,l} \\ & b\left(\mathcal{N}_l\right) \le B_l \\ & \left(b\left(\mathcal{N}_l\right), c\left(\mathcal{N}_l\right), \Omega, G\right) \in \mathbb{S}, \end{aligned} \tag{5}$$

where $\{\varphi_l\}$ is the weight coefficient of the edge server, $\{\alpha_1, \alpha_2, \alpha_3\}$ is the weight coefficient of the performance metric compromise, and $l \in \{1, ..., L\}$. The weighted sum problem is a non-convex optimization problem, which is difficult to solve directly. Non-convex optimization, machine learning and other methods can be used to optimize data compression and transmission and lightweight network deployment strategies. In combination with computing and communication resource allocation in the network, the end-to-end transmission load can be reduced, and the end-to-end delay and energy efficiency deduced by the model can be optimized.

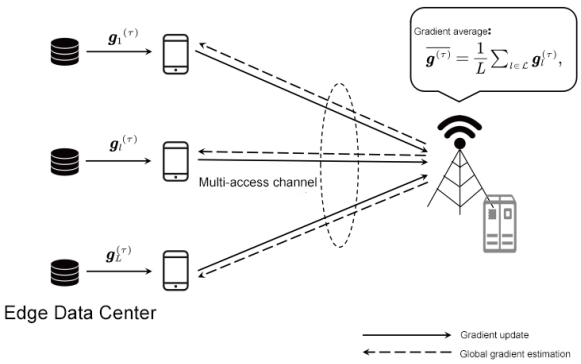

Figure 1. Air Federal Edge Learning Schematic.

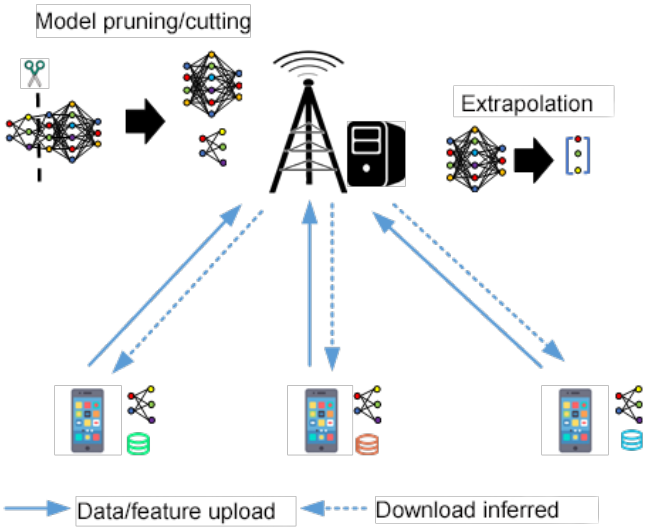

Figure 2. Schematic diagram of edge end joint inference

5. Conclusions

As a typical form of machines learning, deep learning has attracted much attention from researchers. It can independently construct (train) basic rules according to the sample data in the learning process. Especially in the field of machine vision, neural networks are usually trained by supervised learning, that is, by example data and predefined results of example data. In this paper, we firstly overview the current research progress on the deep model training and deployment on the scalable Internet of Things (IoT) networks, by taking into account both the latency and energy consumption. We then summarize the existing challenges on the model training and model deployment on the scalable IoT devices. We further give some feasible solutions to solve the challenges on the model training and model deployment on the scalable IoT devices. The study in this paper can serve as an important reference for the development of deep model training and model deployment for scalable IoT networks.

5.1. Acknowledgements

The work in this paper was supported by the NSFC with grant number 61871235.

5.2. Copyright

References

[1] H. Wang and Z. Huang, "Guest editorial: WWWJ special issue of the 21th international conference on web information systems engineering (WISE 2020)," *World Wide Web*, vol. 25, no. 1, pp. 305–308, 2022.

[2] Y. Guo and S. Lai, "Distributed machine learning for multiuser mobile edge computing systems," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 3, pp. 460–473, 2022.

[3] L. He, K. He, L. Fan, X. Lei, A. Nallanathan, and G. K. Karagiannidis, "Toward optimally efficient search with deep learning for large-scale MIMO systems," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3157–3168, 2022.

[4] H. Wang, J. Cao, and Y. Zhang, *Access Control Management in Cloud Environments*. Springer, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-31729-4

[5] K. He and Y. Deng, "Efficient memory-bounded optimal detection for GSM-MIMO systems," *IEEE Trans. Commun.*, vol. 70, no. 7, pp. 4359–4372, 2022.

[6] S. Tang, "Dilated convolution based CSI feedback compression for massive MIMO systems," *IEEE Trans. Vehic. Tech.*, vol. 71, no. 5, pp. 211–216, 2022.

[7] H. Wang, Y. Wang, T. Taleb, and X. Jiang, "Editorial: Special issue on security and privacy in network computing," *World Wide Web*, vol. 23, no. 2, pp. 951–957, 2020.

[8] X. Lai, "Outdated access point selection for mobile edge computing with cochannel interference," *IEEE Trans. Vehic. Tech.*, vol. 71, no. 7, pp. 7445–7455, 2022.

[9] S. Tang and L. Chen, "Computational intelligence and deep learning for next-generation edge-enabled industrial IoT," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 105–117, 2022.

[10] J. Sun, X. Wang, Y. Fang, X. Tian, M. Zhu, J. Ou, and C. Fan, "Security performance analysis of relay networks based on-shadowed channels with rhis and cees," *Wireless Communications and Mobile Computing*, vol. 2022, 2022.

[11] X. Deng, S. Zeng, L. Chang, Y. Wang, X. Wu, J. Liang, J. Ou, and C. Fan, "An ant colony optimization-based routing algorithm for load balancing in leo satellite networks," *Wireless Communications and Mobile Computing*, vol. 2022, 2022.

[12] C. Wang, W. Yu, F. Zhu, J. Ou, C. Fan, J. Ou, and D. Fan, "Uav-aided multiuser mobile edge computing networks with energy harvesting," *Wireless Communications and Mobile Computing*, vol. 2022, 2022.

[13] J. Chen, Y. Wang, J. Ou, C. Fan, X. Lu, C. Liao, J. Huang, and H. Zhang, "Albrl: Automatic load-balancing architecture based on reinforcement learning in software-defined networking," Wireless Communica-tions and Mobile Computing, vol. 2022, 2022.

[14] C. Ge, Y. Rao, J. Ou, C. Fan, J. Ou, and D. Fan, "Joint offloading design and bandwidth allocation for ris-aided multiuser mec networks," Physical Communication, p. 101752, 2022.

[15] C. Yang, B. Song, Y. Ding, J. Ou, and C. Fan, "Efficient data integrity auditing supporting provable data update for secure cloud storage," *Wireless Communications and Mobile Computing*, vol. 2022, 2022.
[16] J. Lu, "Analytical offloading design for mobile edge computing based smart internet of vehicle," *EURASIP J. Adv. Signal Process.*, vol. 2022, no. 1.
[17] L. Zhang, "DQN based mobile edge computing for smart internet of vehicle," *EURASIP J. Adv. Signal Process.*, vol. 2022, no. 1.
[18] L. Chen, "Physical-layer security on mobile edge computing for emerging cyber physical systems," *Computer Communications*, vol. PP, no. 99, pp. 1–12, 2022.
[19] J. Lu, L. Chen, J. Xia, F. Zhu, M. Tang, C. Fan, and J. Ou, "Analytical offloading design for mobile edge computing-based smart internet of vehicle," *EURASIP journal on advances in signal processing*, vol. 2022, no. 1, pp. 1–19, 2022.
[20] L. Zhang, W. Zhou, J. Xia, C. Gao, F. Zhu, C. Fan, and J. Ou, "Dqn based mobile edge computing for smart internet of vehicle," *EURASIP journal on advances in signal processing*, vol. 2022, no. 1, pp. 1–19, 2022.
[21] J. Liu, Y. Zhang, J. Wang, T. Cui, L. Zhang, C. Li, K. Chen, S. Li, S. Feng, D. Xie *et al.*, "Outage probability analysis for uav-aided mobile edge computing networks," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 9, no. 31, pp. e4–e4, 2022.
[22] R. Zhao and M. Tang, "Profit maximization in cache-aided intelligent computing networks," *Physical Commu-nication*, vol. PP, no. 99, pp. 1–10, 2022.
[23] L. Zhang and C. Gao, "Deep reinforcement learning based IRS-assisted mobile edge computing under physical-layer security," *Physical Communication*, vol. PP, no. 99, pp. 1–10, 2022.
[24] S. Tang and X. Lei, "Collaborative cache-aided relaying networks: Performance evaluation and system optimiza-tion," *IEEE Journal on Selected Areas in Communications*, vol. PP, no. 99, pp. 1–12, 2022.
[25] R. Zhao and M. Tang, "Impact of direct links on intelligent reflect surface-aided MEC networks," *Physical Communication*, vol. PP, no. 99, pp. 1–10, 2022.
[26] J. Lu and M. Tang, "Performance analysis for IRS-assisted MEC networks with unit selection," *Physical Communication*, vol. PP, no. 99, pp. 1–10, 2022.
[27] Y. Wu and C. Gao, "Intelligent task offloading for vehicular edge computing with imperfect CSI: A deep reinforcement approach," *Physical Communication*, vol. PP, no. 99, pp. 1–10, 2022.