计算机科学
COMPUTER SCIENCE

Distributed Weighted Data Aggregation Algorithm in End-to-Edge Communication Networks Based on Multi-armed Bandit

Yifei ZOU, Senmao QI, Cong'an XU, Dongxiao YU

Citation

Yifei ZOU, Senmao QI, Cong'an XU, Dongxiao YU. Distributed Weighted Data Aggregation Algorithm in End-to-Edge Communication Networks Based on Multi-armed Bandit[J]. Computer Science, 2023, 50(2): 13-22.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221100134

Distributed Weighted Data Aggregation Algorithm in End-to-Edge Communication Networks Based on Multi-armed Bandit

Yifei ZOU[1], Senmao QI[1], Cong'an XU[2,3] and Dongxiao YU[1]
1 School of Computer Science and Technology, Shandong University, Qingdao 266237, China
2 Naval Aviation University, Yantai 264000, China
3 Advanced Technology Research Institute, Beijing Institute of Technology, Jinan 250300, China

Abstract As a combination of edge computing and artificial intelligence, edge intelligence has become a promising technique and provided its users with a series of fast, precise, and customized services. In edge intelligence, when learning agents are deployed on the edge side, the data aggregation from the end side to the designated edge devices is an important research topic. Considering the various importance of end devices, this paper studies the weighted data aggregation problem in a single hop end-to-edge communication network. Firstly, to make sure all the end devices with various weights are fairly treated in data aggregation, a distributed end-to-edge cooperative scheme is proposed. Then, to handle the massive contention on the wireless channel caused by end devices, a multi-armed bandit (MAB) algorithm is designed to help the end devices find their most appropriate update rates. Different from the traditional data aggregation works, combining the MAB enables our algorithm a higher efficiency in data aggregation. With a theoretical analysis, we show that the efficiency of our algorithm is asymptotically optimal. Comparative experiments with previous works are also conducted to show the strength of our algorithm.

Keywords Weighted data aggregation, End-to-edge communication, Multi-armed bandit, Edge intelligence
Chinese Library Classification TP393

1 Introduction

As a combination of edge computing and artificial intelligence, edge intelligence (EI) has developed rapidly in recent years and gained increasing attention from industrial and academic research[1-3]. By offloading the computing tasks from the cloud to the edge, EI significantly reduces its communication latency compared to traditional cloud computing[4]. Besides, with the help of artificial intelligence, edge nodes can provide users with flexible and customized services after a series of complex computing based on machine learning models.

To support the computing tasks on the edge side, the data of end devices should be aggregated to the designated edge devices rapidly. Thus, the data aggregationis a fundamental and important research topic in EI[5]. As a branch, the weighted data aggregation additionally considers the weight of data in aggregation. Thus, it is more realistic and flexible. A typical example is that some emergency messages should have higher priorities on intelligent transportation systems[6].

In general, the existing weighted data aggregation approaches can be divided into two categories: centralized and distributed. In the centralized scheme, a centralized node is responsible for scheduling all the data according to the global information of the network[7-10]. Although the centralized method is highly efficient, it requires that the centralized node should have a global view of the weights and number of nodes, and undertake a heavy computing task for the centralized control. Thus, it is not suitable for large-scale edge intelligence scenarios.

In the distributed scenario, without a centralized control, the contention on the wireless channel is a critical problem. Specifically, in a single hop wireless channel with physical interference constraint, two nodes simultaneously transmitting may fail, causing a collision and no one can succeed. To resolve the contention from a distributed view, letting nodes transmit with a probability is a good choice, and two classical channel utilization methods have been proposed: one with static transmission probability[11] and another one with adaptive transmission probability[12].

In the static method, nodes transmit with a fixed transmission probability after the number of transmitting nodes is estimated. In other words, the efficiency of transmission in the static method highly relies on the accurate estimation of the number of nodes, which is hard in large-scale edge intelligence

Submission data: 2022-11-15 Modification data: 2023-01-16
This work was supported by the National Natural Science Foundation of China (NSFC) (62102232, 62122042, 61971269) and Natural Science Foundation of Shandong Province Under (ZR2021QF064).
Corresponding author: Dongxiao YU (dxyu@sdu.edu.cn)

scenarios. Different from the static method, nodes in the adaptive method keep changing their transmission probability according to the feedback from the wireless channel. For example, the famous exponentially back-off scheme[12-13]. With such an adaptive transmission scheme, it is possible for nodes to find some suboptimal transmission probabilities. Whereas, the adaptive scheme will cause some constant fluctuation on channel utilization, making it difficult to achieve high efficiency[13].

Additionally, there also exist a few hybrid data aggregation approaches[14-16], but they are not suitable for edge intelligence scenarios because the centralized nodes in these works still need certain global information. Therefore, we have this question: whether there exists a distributed approach to achieve high efficiency and fairness in the weighted data aggregation without knowing the global information?

In this paper, we answer this question positively by proposing a distributed weighted data aggregation algorithm based on the multi-armed bandit. Specifically, we consider the weighted data aggregation from n end nodes to an edge node in an end-to-edge network with the physical interference constraint of the edge intelligence scenario. The weight of each end node varies from 1 to W according to its importance. The more critical the node is, the larger its weight will be, and its data should have a higher priority in the aggregation process.

When we study the weighted data aggregation problem, efficiency, fairness, and privacy issues should be considered. Specifically, the efficiency requires that the edge device receives the data packets from the end devices as much as possible in each time interval; fairness means the end nodes with larger weight should have a higher priority in data aggregation; from a privacy view, the weight of nodes should be kept as local information.

To address these issues, a distributed data aggregation algorithm is proposed in this paper. Specifically, an end-to-edge cooperative framework is proposed in our algorithm, to ensure the fairness and protect the privacy of nodes. A multi-armed bandit scheme is deployed on the edge device to maximize the channel utilization, which guarantees fast data aggregation in our algorithm. It is worth noting that there are already some studies using the multi-armed bandit theory to solve the data aggregation pro-blem, mainly of which focus on the spectrum selection under multi-channel or node selection in a single channel[17-19]. Whereas, our work focuses on how to use the MAB for fast, fair, and private data aggregation in a single channel wireless network.

To the best of our knowledge, this paper is the first one to solve the distributed weighted data aggregation in a single channel with physical interference constraints by using the multi-armed bandit model. Specifically, we propose a distributed algorithm that uses a MAB scheme to maximize the efficiency of data aggregation, and bases on an end-to-edge cooperative framework to guarantee the fairness and privacy of weighted end nodes. The efficiency of our algorithm is asymptotically optimal in terms of data aggregation. Both theoretical analysis and numerical simulations are presented to show the performance of our algorithm.

The remaining parts of this paper are organized as follows. In Section 2, we summarize the related work on data aggregation. In Section 3, the model of the end-to-edge communication network and the weighted data aggregation problem are formulated. In Section 4, we describe the challenges in algorithm design, redefine the problem from the perspective of a multi-armed bandit, and give a detailed description of our distributed weighted data aggregation algorithm. In Section 5, theoretical analysis is given to show the efficiency and correctness of our algorithm. The experimental results are presented in Section 6. Finally, this paper gets concluded.

2 Related Work

As a fundamental building block, data aggregation has a long research history in wireless sensor network[20-23], and becomes an indispensable process in edge intelligence[24]. According to the working mode of nodes, the previous works on data aggregation algorithms can be divided into three categories: centralized, distributed, and hybrid. In the centralized algorithm, there is a centralized node in the network to control the data aggregation processes among all nodes with the help of global information[7-10,22-23]. For example, S. Randhawa and G. P. Gupta both adopt a hierarchical-cluster-based aggregation method, where nodes are divided into different clusters and the network allocates a cluster head for the implementation of data aggregation[7-8]. For weighted data aggregation, S. Abbasi-Daresari proposed a clustering-based weighted compressive data aggregation (WCDA) approach and applied the WCDA to each cluster to reduce the number of involved nodes during each compressive sampling measurement[25]. M. Razzaq proposed a scheme that calculates a weight function for the cluster head selection process and weights a link cost using traditional Dijkstra algorithm. This scheme can enhance the throughput and reduce energy consumption in the cluster-based network[26].

In the distributed domain, all nodes execute the same data aggregation algorithm independently. A main concern of the distributed data aggregation is how to avoid collisions[27-31] on the channel and destination. In [28], the authors proposed a distributed and randomized algorithm to build a message aggregation tree, and used the geographical location of the nodes to avoid interference. Thus, data aggregation and dissemination in the blockchain system can be realized. However, in this paper, the authors do not consider the weight of the message.

In recent years, extensive research about distributed federated learning has noticed that the weighted data aggregation can speed up the training process of the model and reduce the traffic load of the network[30-31]. However, these works rarely focus on the specific implementation of the weighted data aggregation algorithm under the network with the physical interference constraint.

There are relatively few studies on distributed weighted data aggregation. Because the weight, as private information, is only known by the node itself, ensuring fairness between nodes with different weights is not easy. A. B. Alexandru proposed a solution that involves two layers of encryption based on the learning with errors problem and can protect the privacy of packet by using homomorphic encryption[32]. In delayed sensor networks, R. Adachi proposed scalable communication laws where each node compensates communication delays of the received data, and aggregates these data with the consideration of weight coefficients[33].

The hybrid algorithm is a combination of the above centralized and distributed modes, such as Cooperative data aggregation Algorithms[14], Centralized energy allocation and Distributed energy allocation algorithms[15]. For example, S. Xiao designed a centralized algorithm to find a near-optimal energy allocation strategy and adopted a distributed method for data aggregation[15].

Several research in recent years has also considered the use of multi-armed bandits to solve data aggregation problems[17-19]. However, these studies are more concerned with the allocation of spectrum and the selection of clients in federated learning and do not give a definite solution for the contention resolution in a single channel.

In this paper, we use a distributed algorithm to realize the weighted data aggregation in an end-to-edge communication network. Compared with the previous distributed algorithms, the MAB is adopted in our algorithm so that the nodes with various weights can find the most appropriate transmission probability in data aggregation. Thus, efficiency, fairness, and privacy can be obtained in our algorithm.

3 Model and Problem Definition

In this section, we consider a practical case of weighted data aggregation in an end-to-edge communication network. Specifically, n end nodes are arbitrarily deployed in a circle with the edge node as the center and the radius R, where R is the maximum transmission range of all nodes. End nodes continuously collect information from the environment and aggregate data to the edge node. Let V represent the set of end nodes. We assume that each end node has sufficient data that is divided into equal size packets. At the beginning of each slot, each end node can choose to transmit a data packet to the edge device. Whether the edge device can receive and decode the data packet is formulated in the following communication model. Edge node v also has an integer of $w_v \in [1, W]$ as its weight where W is the upper bound of weight. The larger the weight, the earlier data collected by this node should be aggregated to the edge node. The execution of our algorithm is divided into multiple synchronized slots, and each slot is a minimum time slice for a node to transmit.

3.1 Communication Model

In a single-hopwireless network, the distance between any end node and edge node does not exceed R. The nodes communicate in a half-duplex manner, that is, in each slot, the nodes can choose to send a message or listen to the channel, but cannot do both. Considering the actual wireless network environment, the signal strength will gradually attenuate with the propagation distance, and at the same time, multiple signals will be superimposed at the receiving end. To better describe this phenomenon, we use the SINR (Signal to interference plus noise ratio) model to describe the signal strength in the wireless channel. Specifically, for communication from u to v, the following equation is satisfied:

$$Signal(u,v) = P_u / d\ (u,v)^{\alpha} \tag{1}$$

$$SINR(u,v) = \frac{\mathrm{Signal}(u,v)}{\sum\limits_{w \in S/\{u\}} \mathrm{Signal}(u,v) + N} \tag{2}$$

In the above equation, $Signal(u,v)$ represents the signal strength from u felt by v. P_u is the transmit power of node u, $d(u,v)$ is the Euclidean distance between u and v, and α is the path loss exponent determined by the medium and some other factors in environment. When v is trying to decode a message from u, $\sum\limits_{w \in S/\{u\}} Signal(u,v)$ is the interference, where S is the set of nodes transmitting simultaneously with u, and N is the ambient noise. We consider that the transmission from u to v is successful when $SINR(u,v) \geqslant \beta$, where $\beta > 1$ is the threshold determined by hardware.

3.2 Problem Definition

We consider the weighted data aggregation from n end nodes to 1 edge node in a single-hop wireless network. Each end node v has an integer $k_v \in [1, W]$ to represent its weight. I is a time interval, consisting of sufficient time slots. For each node v, let m_v be the number of its packets received by the edge node in the interval I. Then, our weighted data aggregation is to let end nodes find their specific transmission probabilities, to maximize the efficiency of data aggregation with fairness and privacy constraints. A mathematical formulation is given in the following:

$$\begin{aligned} &\text{objective}: Max \sum_{v \in V} m_v \\ &\text{s. t. } m_v : m_u = \Theta(w_v : w_u) \\ &\qquad b(v,u) = 0\ \forall\ v, u \in V \end{aligned} \tag{3}$$

where $b(v,u) = 0$ if and only if the weight of v is unknown to u.

Otherwise, $b(v,u)=1$.

3.3 Capability and Knowledge of Nodes

We assume that all nodes adopt a uniform power allocation strategy[11], that is, all nodes use a consistent power P to transmit. In order to ensure that any two nodes u and v in a single-hop network can communicate with each other, we assume $\frac{P}{d\ (u,v)^{\alpha}} \geqslant N\beta$. The edge node is equipped with a physical carrier sensing function, if the node chooses to listen to the channel, it can obtain the current signal strength. We assume that the ambient noise N is fixed. Therefore, the node can distinguish three states of the channel according to the signal strength and received information:

(1) Idle channel: signal strength equal to N.

(2) Successful transmission: the node has successfully received the information from the transmitted node.

(3) Busy channel: signal strength is greater than N and no message is received.

In our algorithm, all nodes know the uniform power P, the SINR parameter α, β, the ambient noise N, and its own weight. However, the nodes know nothing about the topology of the network and other nodes.

4 Distributed Weighted Data Aggregation Algorithm based on MAB

In this section, we discuss the main challenges in designing a highly efficient, fair, and private distributed weighted data aggregation algorithm. At the same time, we redefine the problem from the perspective of multi-armed bandits, and give a detailed description of our distributed weighted data aggregation algorithm.

4.1 Challenges and Solutions

Since the weight of a node is only known by itself, how to satisfy the weighted fairness while guaranteeing privacy protection is a difficult problem to solve. In the previous weighted data aggregation algorithms, the realization of weighted fairness relies on the knowledge of the weights of other nodes, but this destroys the principle of privacy protection. The privacy constraint makes all centralized and distributed random algorithms based on the sum of weights invalid[9,11,25,28].

To solve the above problem, we design an end-to-edge cooperative framework. Specifically, the edge node maintains a stem probability p_e for all end nodes. In each slot, the edge node will broadcast the stem probability p_e to all the end devices. The end node v transmits with the probability of $p_v = \min\left(\frac{1}{2}, p_e * w_v\right)$ in the transmission slot, where w_v is the weight of v. Using this method, we can ensure that the transmission probability of a node is proportional to its weight.

Another problem brought about by this simple and effective method is the contention resolution. More specifically, how to choose an appropriate stem probability p_e? Obviously, a larger p_e will cause more nodes to transmit at the same time, resulting in the collision in the wireless channel. On the contrary, a smaller p_e will cause no node transmitting, which is a waste of the channel. The previous work generally uses the binary backoff method to adjust p_e dynamically according to the state of the channel. But the disadvantage of this method is that p_e will constantly fluctuate, so it is difficult to achieve a highly efficient channel utilization[12-13].

To overcome this problem, we first prove that the optimal choice of stem probability is $1/\hat{W}$ (Theorem 1), where $\hat{W}$ is the sum of all node weights. Therefore, our goal is to find an integer i that is as close as possible to $\log \hat{W}$ and use $1/2^i$ as the stem probability. We treat the choice of i as a multi-armed bandit problem where i is closer to $\log \hat{W}$, the reward of choosing i is greater. To get an approximate range of i, we adopt an adaptive method that can find an interval containing the optimal setting with high probability. Besides, we use the idea of dichotomy to design a weighted ε-greedy strategy, in which the probability of choosing different arms is weighted during exploration. Due to the adoption of the multi-armed bandit, after a period of exploration, we can always keep the optimal channel utilization with the probability of $1-\varepsilon$, outperforming the binary backoff method. Besides, compared with the traditional ε-greedy strategy, the weighted method can explore those arms with greater rewards earlier, thus reducing the regret for each decision.

4.2 Problem Redefinition

As stated above, one of the core problems to be solved is how to find an integer i that is close enough to $log\ \hat{W}$. Through the adaptive method, we can find an interval $L=[left, right]$ with $left < \log \hat{W} < right$. Therefore, we can consider the choice of i as a K armed bandit problem, where $K=(right-left)+1$ and each arm corresponds to an integer in L. When the edge node chooses the arm of integer i, end nodes will use stem probability $p_e = 1/2^i$ to transmit its data packet. If its transmission succeeds, the reward obtained in MAB is 1. Otherwise, the reward is 0. We assume that the reward of each arm is K probability distributions $\langle D_1, D_2, \cdots, D_K \rangle$, with the expectation of $\langle \mu_1, \mu_2, \cdots, \mu_K \rangle$ and the variance of $\langle \sigma_1^2, \sigma_2^2, \cdots, \sigma_K^2 \rangle$. Obviously, D_i obeys the Bernoulli distribution with parameter p, where p is the probability of successful transmission when $p_e = 1/2^i$.

Thus, our problem can be redefined as minimizing the total regret of MAB. We assume that the arm selected by round t is $j(t)$ and $r_j(t)$ is the corresponding reward. Let $\mu^* = \max(\mu_1, \mu_2, \cdots, \mu_K)$ is the maximum expected reward, then in a sufficiently long period T, our goal is to minimize the total expected regret R_T, that is:

$$\min R_T = T * \mu^* - \sum_{t=1}^{T} r_j(t) \tag{4}$$

4.3 MAB-based Algorithm

Our distributed weighted data aggregation algorithm with multi-armed bandit (DWDA-MAB) is shown in Algorithm 1 and 2 which both consists of multiple synchronous rounds, and each round consists of three synchronous slots, to listen to the channel or transmit a message.

Specifically, each end node v will listen to channel in the first slot. If v successfully received the stem probability p_e from the edge node, it will conduct a mapping process, i.e. $p_v = \min\{w_v * p_e, 1/2\}$. In the second slot, v randomly selects a data packet from its data and transmits it with probability p_v. In the third slot, node v will judge whether the transmission succeeds according to the feedback from the edge node, and if the transmission is successful, it will remove the packet from its local data.

For edge node u, its core task is to find an integer i that is as close to $\log\hat{W}$ as possible with the help of MAB. Before the MAB process, we need to use Algorithm 3 to determine the approximate range of i and the exploration weights for each arm. Specifically, Algorithm 3 can be divided into two key stages with three slots in each synchronized round. The first stage is called the adaptive stage (the first while loop in Algorithm 3), and its role is to find a search interval L containing $\log\hat{W}$. At the beginning, u maintains an integer $i=1$. In the first slot of each round, u broadcasts the stem probability $p_e = 1/2^i$. In the second slot, u adjust i according to the state of the channel. It will double i when the channel is not idle. The third slot is to give feedback on the successfully received message. When the channel is idle, u ends the first phase and we can get an interval $L=[i/2, i*2]$ containing $\log\hat{W}$ with high probability. The second stage is to determine the exploration weight of each integer in L in MAB. Specifically, it is a process in which u continuously bisects L. We use the bisection method to find an exploration weight array EP. Specifically, u broadcasts stem probability $p_e = 1/2^{mid}$, where mid is the midpoint of the current interval. When the channel is idle, the next interval is $[low, mid]$, otherwise it is $[mid, high]$, where low and $high$ represent the left and right endpoints of the current interval. We determine the search weight of each integer in L according to its frequency of occurrences in the second stage.

Algorithm 1 DWDA-MAB for end node v

Initialization: $Q_v = \emptyset$;

Slot 1:

1. Listen to channel;
2. if receive a message $M_u = \{p_e\}$
3. $\quad p_v = \min\{w_v * p_e, 1/2\}$;
4. end if

Slot 2:

5. Randomly select a packet M_v from its local data;
6. Transmit M_v with probability p_v;

Slot 3:

7. Listen to channel;
8. if receive a ACK_u and $u=v$
9. $\quad$ Remove M_v from Q_v;
10. end if

Algorithm 2 DWDA-MAB for edge node u

Initialization: $p_e = 0$;

1. [L, EW] = Search and Weight ();
2. reward(j) = 0 for integer $i \in L$;

Slot 1:

3. X = 1 or 0 with probability ε and $1-\varepsilon$;
4. if X = 1
5. $\quad p_e = 1/2^i$ with probability of EW(i) where integer $i \in L$;
6. else
7. $p_e = 1/2^i$ where i = argmax(reward) and $i \in L$;
8. end if
9. Broadcast p_e;

Slot 2:

10. Listen to channel;
11. Update reward(j);

Slot 3:

12. if receive a message M_v in slot 2
13. $\quad$ Broadcast an ACK_v;
14. end if

Algorithm 3 Search and Weight

Initialization: i = 1;

Phase 1:

1. while true

Slot 1:

2. $\quad p_e = 1/2^i$;
3. $\quad$ Broadcast $M_u \leftarrow \{p_e\}$;

Slot 2:

4. $\quad$ Listen to channel;
5. $\quad$ if sense an idle channel;
6. $\quad\quad$ break;
7. $\quad$ else
8. $\quad\quad$ i = i * 2;
9. $\quad$ end if

Slot 3:

10. $\quad$ if receive a message M_v in slot 2
11. $\quad\quad$ Broadcast an ACK_v;
12. $\quad$ end if
13. end while
14. L = [i/2, i * 2];
15. low = i/2, high = i * 2;
16. count[low: high] = {0};

Phase 2:

17. while low + 1 < high

Slot 1:

18. $\quad$ count(j)++; ($j \in$ [low: high])
19. $\quad$ mid = ⌈(low + high)\2⌉;

```
    Broadcast M_e←{p_e=1\2^mid};
Slot 2:
    Listen to channel;
    if sense an idle channel;
      high=mid;
    else
      low=mid;
    end if
Slot 3:
    if receive a message M_v in slot 2
      Broadcast an ACK_v;
    end if
end while
EW = count[low:high] / sum(count[low:high]);
Return L, EW
```

The stage of MAB is a process of exploration and exploitation about i. We design a weighted ε-greedy strategy. Specifically, u chooses i with the largest reward with probability $1-\varepsilon$. u will choose to explore the integer in L with probability ε, and i is chosen with probability $EW(i)$. Each round of MAB is also divided into three synchronized slots. In the first slot, u broadcasts the stem probability $p_e=1/2^i$. In the second slot, u listen to channeland get a reward. Then, u will update its estimate of the reward. For example, for integer i, $reward(i)$ should be updated to $acc\_reward(i)/t(i)$ where $acc\_reward(i)$ is the accumulative reward of arm i and $t(i)$ is the number of rounds arm i was explored. The third slot, consistent with the previous statement, is still a feedback process. We summarize all important variables in the paper in Table 1.

Table 1 Important variables

Parameters	Definition
n	Number of end nodes
V	Set of all end nodes
W	Upper bound of all weights
w_v	Weight of end node v
$\hat{W}$	Sum of all weights
K	Number of arms
R_T	Total expected regret
p_e	Stem probability
p_v	Transmission probability of node v
ε	Exploration rate in MAB

At the end of this section, we give the following theorems about the correctness and efficiency of the algorithm, and the proofs of these theorems will be given in the next section.

Theorem 1 Let $Pr(X=1)$ be the probability that only one end node chooses to transmit. Then, the closer p_e is to $1/\hat{W}$, the larger $Pr(X=1)$ is and $Pr(X=1)$ gets maximized when $p_e=1/\hat{W}$.

Theorem 2 We consider a weighted data aggregation with n end nodes and an edge node, where each end node has enough data to be aggregated to the edge node and a weight varying within $[1,W]$. By executing our DWDA-MAB algorithm for a sufficient long time, the following properties are satisfied:

(1) High Efficiency: after executing the algorithm for sufficient long rounds, we can obtain the optimal channel utilization with probability of $1-\varepsilon$.

(2) Weighted Fairness: let $\hat{p}_i$ be the probability that the packet of end node i is received on the edge side. Then, for any two end nodes i,j, we have: $\hat{p}_i:\hat{p}_j=\Theta(w_i:w_j)$.

(3) Privacy Protection: the weight of each node is kept as local and private knowledge only known by itself in the execution of our algorithm.

5 Algorithm Analysis

In this section, we proved Theorem 1 and 2.

5.1 Proof for Theorem 1

Let $f(p_e)=Pr(X=1\mid p_e)$ denote the probability that only one end node chooses to transmit when stem probability is p_e. Then, we have:

$$f(p_e)=\sum_u p_u * \prod_{v\neq u}(1-p_v)$$
$$=\sum_u \frac{p_u}{1-p_u} * \prod_u (1-p_u) \tag{5}$$

Let $g(p_e)=\prod_u(1-p_u)$. Then, the derivative of $f(p_e)$ can be written as:

$$\frac{\mathrm{d}f(p_e)}{\mathrm{d}p_e}=\sum_u\left[\frac{p_u}{(1-p_u)^2}g(p_e)+\frac{\mathrm{d}g(p_e)}{\mathrm{d}p_e}*\frac{p_u}{1-p_u}\right], \text{where}$$
$$\frac{\mathrm{d}g(p_e)}{\mathrm{d}p_e}=g(p_e)*\left(-\sum_u\frac{p_u}{1-p_u}\right) \tag{6}$$

We can find $\left.\frac{\mathrm{d}f(p_e)}{\mathrm{d}\,p_e}\right|_{p_e=1/\hat{W}}=0$. When $p_e<1/\hat{W}$, we have $\frac{\mathrm{d}f(p_e)}{\mathrm{d}p_e}>0$ and when $p_e>1/\hat{W}$, we have $\frac{\mathrm{d}f(p_e)}{\mathrm{d}p_e}<0$. Therefore, when $p_e=1/\hat{W}$, $Pr(X=1)$ is greatest and when p_e is closer to $1/\hat{W}$, the $Pr(X=1)$ is greater. Here we have successfully proved Lemma 1.

5.2 Proof for Theorem 2

Proof for High Efficiency. The efficiency of the algorithm depends on the choice of the stem probability, so we first analyze the help of the Algorithms 3 in choosing the stem probability in lemma 1.

Lemma 1 Algorithm 3 will be executed for $O(\log\log\hat{W})$ rounds with high probability. After Algorithm 3 stops, we can get an interval L of length $\theta(\log\hat{W})$ and $\log\hat{W}\in L$. We can also get an exploration weight array EP, where the larger $EP(i)$, the better channel throughput with $p_e=1/2^i$.

Proof: Let random variance X denote the number of transmitted nodes in a specific round. Thus, when $i<\frac{1}{2}*\log\hat{W}$, we have: $Pr(X=0)=\prod_u\left(1-\frac{w_u}{2^i}\right)\leqslant \mathrm{e}^{-\sqrt{\hat{W}}}\leqslant\frac{1}{n}$. When $i>2*$

$\log\hat{W}$, we have: $E(X)\geqslant\frac{1}{\hat{W}}$. According to Markov's inequality, we know $Pr(X\geqslant 1)\leqslant\frac{1}{\hat{W}}\leqslant\frac{1}{n}$.

The above two conclusions show that when $i<\frac{1}{2}*\log\hat{W}$, the channel will not be idle with high probability. On the contrary, when $i>2*\log\hat{W}$, the channel will be idle with high probability. Therefore, the first while loop of Algorithm 3 will stop when $i\in[\frac{1}{2}*\log\hat{W}, 2*\log\hat{W}]$ with high probability. Considering i will be doubled in each round, the first while loop will stop after $\theta(\log\log\hat{W})$ rounds. Besides, we can prove that $|L|=\left|\left[\frac{i}{2}, i*2\right]\right|=\theta(\log\hat{W})$ and $\log\hat{W}\in L$. At the same time, since the initial bisection interval is L, the second while loop will end after $\theta(\log\log\hat{W})$ rounds. Here we successfully prove the time complexity of Algorithm 3. We next provide a brief analysis of the process of dichotomy.

We denote mid_j as the midpoint of the *j-th* dichotomy. A round of dichotomy is good if and only if $\log\hat{W}$ is in the dichotomy interval of round $j+1$. Similar to the proof above, we have: when $|mid_j-\log\hat{W}|>c$, $Pr(round\ j\ is\ good)\geqslant 1-\frac{1}{2^c}$. This conclusion tells us that when mid_j is further away from $\log\hat{W}$, we have a greater probability of making a good bisection. For example, when $c=\theta(\log\log\hat{W})$, the probability of error does not exceed $1/log\ \hat{W}$. From this, we can get that the closer the integer i is to $\log\hat{W}$, the more likely it will appear in the bisection process. Combining the results in Theorem 1, we can prove that the larger $EP(i)$, the better the transmission efficiency with $p_e=1/2^i$.

Lemma 2 After a period of exploration, we can obtain the optimal channel throughput with probability of $1-\varepsilon$. The regret of our algorithm is $O(T)$ where T is the execution time and lower than the traditional ε-greedy strategy.

Proof: Since the search range of MAB includes $\log\hat{W}$, and the reward corresponding to $\log\hat{W}$ is the largest. Therefore, after a period of exploration we can always find the arm with the largest reward with probability $1-\varepsilon$. Besides, since we use a weighted exploration process, those arms with larger rewards will be explored earlier. Clearly, the regret will be lower than the traditional ε-greedy strategy. The analysis of the upper bound of the regret has been given in the previous study[34], so the detailed analysis will not be shown in this paper.

Proof for Weighted Fairness. Let p_i be the transmission of end node i in a specific round. Obviously, for any two different end nodes i, j, we have: $\hat{p}_i=p_i*\prod_{u\neq i}(1-p_u)$ and $\hat{p}_j=p_j*\prod_{u\neq j}(1-p_u)$. Considering in the transmission slot, $p_i=\min\{w_i*p_e, 1/2\}$ and $p_j=\min\{w_j*p_e, 1/2\}$. Therefore, $\hat{p}_i:\hat{p}_j=w_i*(1-p_j):w_j*(1-p_i)=\Theta(w_i:w_j)$ for $(1-p_i)$ and $(1-p_j)$ are both variables in $\left[\frac{1}{2}, 1\right]$.

Proof for Privacy Protection. Since the weight of the node is only known by itself, it is obvious that the algorithm can protect the privacy of the node.

6 Experimental Results

In this section, we show the performance of our DWDA-MAB algorithm. We measure the efficiency of our algorithm from the perspectives of communication and multi-armed bandits, respectively. Besides, we take the channel throughput and generated regret in the traditional MAB process as metrics. We discuss the effect of the number of end nodes n and the exploration rate ε on the above metrics. Comparative experiments show that our algorithm outperforms previous work both in channel throughput and regret.

6.1 Parameter Setting

In the experiment, n end nodes are arbitrarily distributed in a circle area with the edge node as the center and the radius R. Note that $R=200m$ is the maximum transmission range of nodes. Therefore, such a deployment makes sure a single hop end-to-edge communication network. The number of end nodes n and the exploration rate vary within [2000, 5000] and [0.05, 0.2], respectively. Considering that the upper bound of weights and the distribution of weights will not directly affect the effect of the algorithm, we assume that the weights of all nodes are integers and obey a uniform distribution $U(1, 100)$. We summarize the settings of some important parameters in Table 2. Without loss of generality, for each reported result, we performed the simulation over 100 runs for an averaged result. We summarize the settings of some important parameters in Table 2.

Table 2 Parameters inExperiment

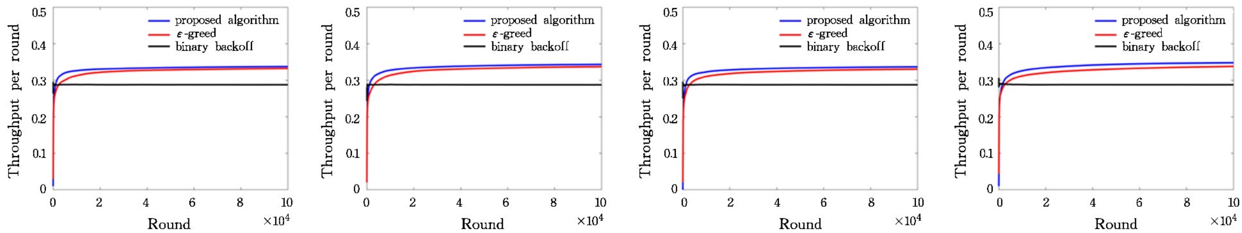
Parameters & Values	
$n=[2,3,4,5]*10^3$	$\varepsilon=[1,2,3,4]*0.05$
$R=200m$	$N=1.0$
$\alpha=3$	$\beta=1.5$
$W=100$	$w_v\sim U(1,100)$ and $w_v\in z$

6.2 Performance Evaluation

Since we have defined the weighted data aggregation problem from the perspective of communication and multi-armed bandits, respectively, we choose throughput and regret as metrics for presentation. The throughput reflects the utilization rate of channel during a sufficientlylargeinterval, and the regret reflects the gap between the actual choice and the theoretical optimal choice.

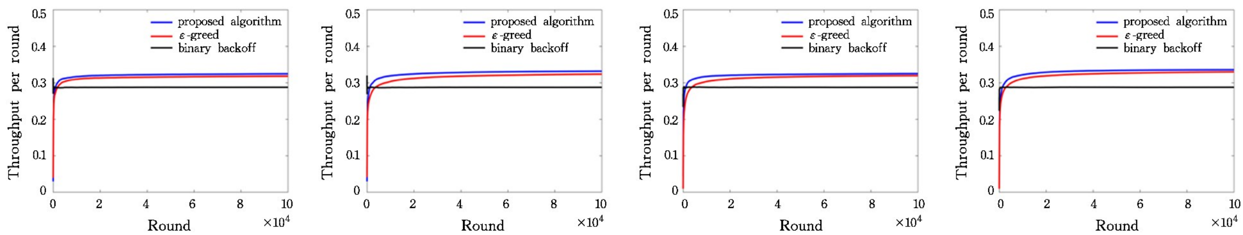
From the Fig. 1(a), we can find that as the number of rounds increases, the throughput of our algorithm keeps increasing and gradually remains around 0.334. By comparing the four subgraphs, we can find that n has little effect on stable value of throughput when the exploration rate ε is fixed but can affect the convergence speed of throughput.

When we keep n unchanged, from Fig. 1 to Fig. 4, we can find that as the exploration rate ε increases, the throughput will converge faster, but the converged throughput will be lower. For example, the stable value of throughput in Fig. 1(a) is 0.334, while it is 0.297 in Fig. 4(a) with $\varepsilon=0.20$. It is because the increased exploration rate ensures that the algorithm can explore all the arms faster, but it also leads to underutilization of the optimal solution, which reduces the throughput.



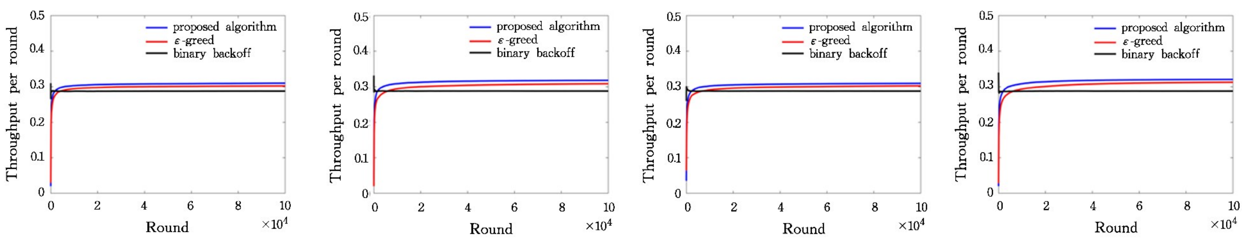
(a) $n=2\,000$ (b) $n=3\,000$ (c) $n=4\,000$ (d) $n=5\,000$

Fig. 1 Throughput when n varies and $\varepsilon=0.05$



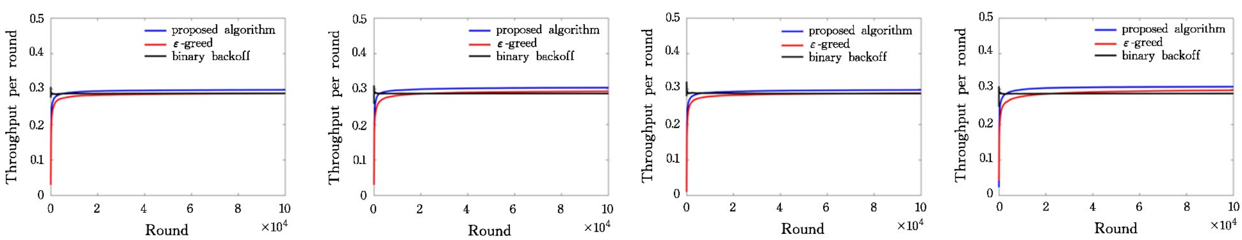
(a) $n=2\,000$ (b) $n=3\,000$ (c) $n=4\,000$ (d) $n=5\,000$

Fig. 2 Throughput when n varies and $\varepsilon=0.10$



(a) $n=2\,000$ (b) $n=3\,000$ (c) $n=4\,000$ (d) $n=5\,000$

Fig. 3 Throughput when n varies and $\varepsilon=0.15$



(a) $n=2\,000$ (b) $n=3\,000$ (c) $n=4\,000$ (d) $n=5\,000$

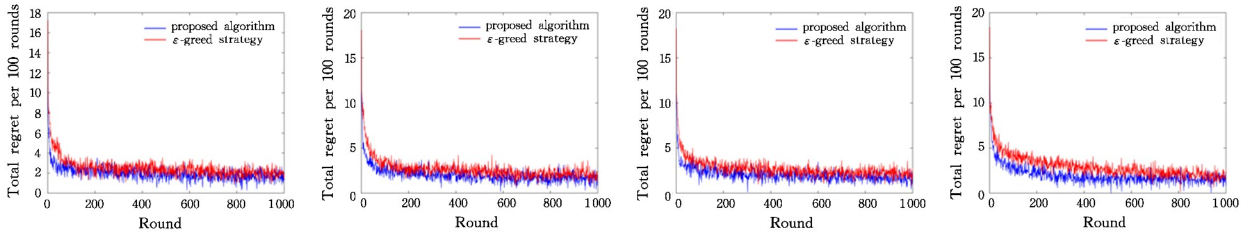
Fig. 4 Throughput when n varies and $\varepsilon=0.20$

In the experiment about throughput, we choose the ε-greedy strategy and the binary backoff strategy[10-11] for comparison. By comparison, we can see that compared to the ε-greedy strategy, the throughput of our algorithm converges faster and the stable value is higher than that of the ε-greedy strategy. This is because our weighted ε-greedy strategy can explore those arms with greater rewards earlier. Compared with binary-backoff, our algorithm can obtain better throughput after a period of exploration.

Our next experiment evaluates the efficiency of the algorithm from the perspective of a MAB machine. The reason for the above conclusion can be better explained from the perspective of the regret.

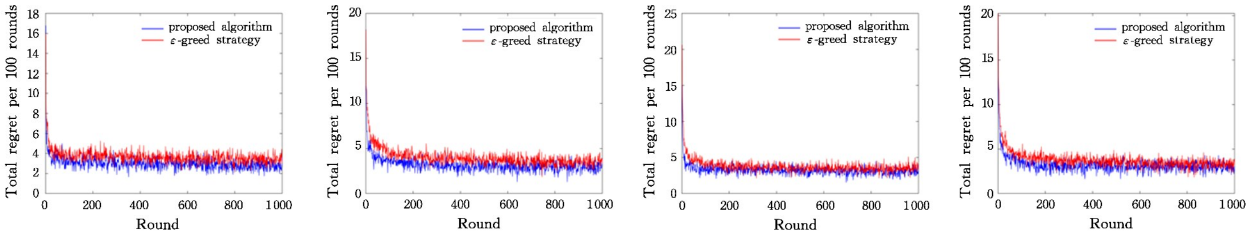
Fig. 5—Fig. 8 show the effects of four different values of

n on the total regret per 100 rounds when the exploration rate $\varepsilon = 0.05, 0.10, 0.15, 0.20$, respectively. The x-axis represents the number of rounds executed in MAB, and the y-axis represents total regret per 100 rounds.



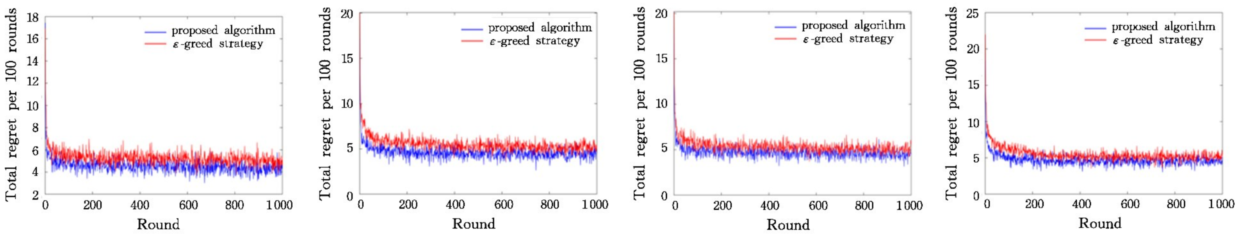
(a) $n = 2\,000$ (b) $n = 3\,000$ (c) $n = 4\,000$ (d) $n = 5\,000$

Fig. 5 Total regret per 100 rounds when n varies and $\varepsilon = 0.05$



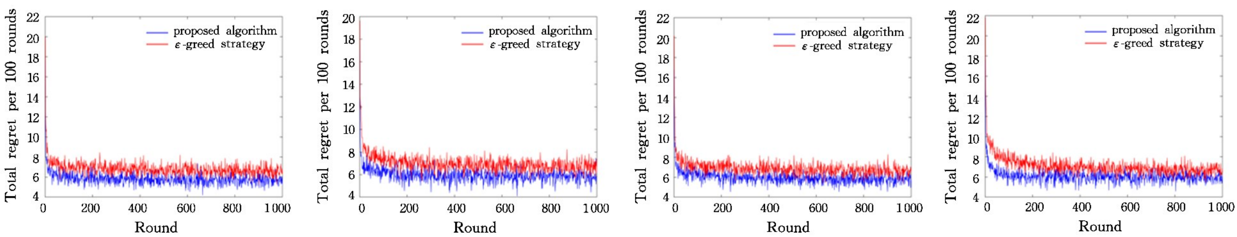
(a) $n = 2\,000$ (b) $n = 3\,000$ (c) $n = 4\,000$ (d) $n = 5\,000$

Fig. 6 Total regret per 100 rounds when n varies and $\varepsilon = 0.10$



(a) $n = 2\,000$ (b) $n = 3\,000$ (c) $n = 4\,000$ (d) $n = 5\,000$

Fig. 7 Total regret per 100 rounds when n varies and $\varepsilon = 0.15$



(a) $n = 2\,000$ (b) $n = 3\,000$ (c) $n = 4\,000$ (d) $n = 5\,000$

Fig. 8 Total regret per 100 rounds when$n$ varies and $\varepsilon = 0.20$

From Fig. 5, we can see that when the exploration rate is fixed, the regret in the initial stage will be larger with the increase of n, which indicates that the exploration time in MAB needs to be longer. When we fix n, we find that as the exploration rate increases, the total regret also increases, because the probability of exploration error increases. At the same time, by comparing with the traditional ε-greedy algorithm, we can obviously see the superiority of our algorithm in regret.

Conclusion This paper studies the weighted data aggregation problem in end-to-edge communication networks. Based on an end-to-edge cooperative framework and a modified MAB scheme, we propose a distributed algorithm that is efficient, fair, and private on solving the weighted data aggregation problem. Compared with the previous works, our work is the first one solving the distributed weighted data aggregation in a single wireless channel with physical interference constraints based on a MAB model. Extending our work to the mobile scenarios will be our work in the future.

Reference

[1] CAI Z, SHI T. Distributed Query Processing in the Edge Assisted IoT Data Monitoring System[J]. IEEE Internet of Things Journal, 2021, 8(16): 12679-12693.

[2] ZHU T, SHI T, LI J, et al. Task Scheduling in Deadline-Aware Mobile Edge Computing Systems[J]. IEEE Internet of Things Journal, 2019, 6(3): 4854-4866.

[3] YU L, SHEN H, CAI Z, et al. Towards Bandwidth Guarantee for Virtual Clusters under Demand Uncertainty in Multi-tenant

Clouds[J]. IEEE Transactions on Parallel and Distributed Systems, 2018, 29(2): 450-465.
[4] DENG S, ZHAO H, FANG W, et al. Edge intelligence: The confluence of edge computing and artificial intelligence[J]. IEEE Internet of Things Journal, 2020, 7(8): 7457-7469.
[5] KAUR M, MUNJAL A. Data aggregation algorithms for wireless sensor network: a review[J]. Ad Hoc Networks, 2020, 100: 102083.
[6] SUMALEE A, HO H W. Smarter and more connected: Future intelligent transportation system[J]. Iatss Research, 2018, 42(2): 67-71.
[7] RANDHAWA S, JAIN S. Dahda: dynamic adaptive hierarchical data aggregation for clustered wireless sensor networks[J]. Wireless Personal Communications, 2017, 97(4): 6369-6399.
[8] GUPTA G P, MISRA M, GARG K. Towards scalable and load-balanced mobile agents-based data aggregation for wireless sensor networks[J]. Computers & Electrical Engineering, 2017, 64: 262-276.
[9] EL FISSAOUI M, BENI-HSSANE A, SAADI M. Mobile agent protocol-based energy aware data aggregation for wireless sensor networks[C]// Procedia computer science. 2017: 25-32.
[10] LEI Y, QIN H, CHEN W, et al. Secure Data Aggregation Scheme Based on Dynamic Multi-cluster Key Management Model[J]. Computer Science, 2011, 38(9): 36-40.
[11] YU D, ZOU Y, XU M, et al. Competitive age of information in dynamic iot networks[J]. IEEE Internet of Things Journal, 2020, 8(20): 15160-15169.
[12] YU D, ZOU Y, ZHANG Y, et al. An exact implementation of the abstract mac layer via carrier sensing in dynamic networks[J]. IEEE/ACM Transactions on Networking, 2021, 29(3): 994-1007.
[13] RICHA A, SCHEIDELER C, SCHMID S, et al. An efficient and fair mac protocol robust to reactive interference[J]. IEEE/ACM Transactions on Networking, 2012, 21(3): 760-771.
[14] XU H, HUANG L, ZHANG Y, et al. Energy-efficient cooperative data aggregation for wireless sensor networks[J]. Journal of Parallel and Distributed Computing, 2010, 70(9): 953-961.
[15] XIAO S, LI B, YUAN X. Maximizing precision for energy-efficient data aggregation in wireless sensor networks with lossy links[J]. Ad Hoc Networks, 2015, 26: 103-113.
[16] NIKNAM S, DHILLON H S, REED J H. Federated learning for wireless communications: Motivation, opportunities, and challenges[J]. IEEE Communications Magazine, 2020, 58(6): 46-51.
[17] LI F, YU D, YANG H, et al. Multi-armed-bandit-based spectrum scheduling algorithms in wireless networks: A survey[J]. IEEE Wireless Communications, 2020, 27(1): 24-30.
[18] CHEN S, WANG X, ZHOU P, et al. Heterogeneous semi-asynchronous federated learning in internet of things: A multi-armed bandit approach[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2022, 6(5): 1113-1124.
[19] XIA W, QUEK T Q, GUO K, et al. Multi-armed bandit-based client scheduling for federated learning[J]. IEEE Transactions on Wireless Communications, 2020, 19(11): 7108-7123.
[20] DHAND G, TYAGI S. Data aggregation techniques in wsn: Survey[J]. Procedia Computer Science, 2016, 92: 378-384.
[21] GHATE V V, VIJAYAKUMAR V. Machine learning for data aggregation in wsn: A survey[J]. International Journal of Pure and Applied Mathematics, 2018, 118(24): 1-12.
[22] DONG H, WEI S, LIU X, et al. A Survey of Routing Methods for Wireless Sensor Networks Oriented to Environment Awareness[J]. Computer Science, 2018, 45(1): 14-22.
[23] FENG H Y, ZHOU L, DING Q L. Data Aggregation Algorithm for Wireless Sensor Networks Based on Data Stream and Network Coding[J]. Computer Science, 2015, 42(5): 136-140.
[24] XU D, LI T, LI Y, et al. Edge intelligence: Architectures, challenges, and applications[J]. arXiv: 2003.12172, 2020.
[25] ABBASI-DARESARI S, ABOUEI J. Toward cluster-based weighted compressive data aggregation in wireless sensor networks[J]. Ad Hoc Networks, 2016, 36: 368-385.
[26] RAZZAQ M, KWON G R, SHIN S. Energy efficient dijkstra-based weighted sum minimization routing protocol for wsn[C]// 2018 Third International Conference on Fog and Mobile Edge Computing (FMEC). IEEE, 2018: 246-251.
[27] ZHANG X, TAO L, YAN F, et al. Shortest-latency opportunistic routing in asynchronous wireless sensor networks with independent duty-cycling[J]. IEEE Transactions on Mobile Computing, 2019, 19(3): 711-723.
[28] YANG L, ZOU Y, XU M, et al. Distributedconsensus for blockchains in internet-of-things networks[J]. Tsinghua Science and Technology, 2022, 27(5): 817-831.
[29] ZOU Y, XU M, YU J, et al. A fast consensus forpermissioned wireless blockchains[J]. IEEE Internet of Things Journal, 2021.
[30] CHEN Y, SUN X, JIN Y. Communication-efficient federated deeplearning with layerwise asynchronous model update and temporallyweighted aggregation[J]. IEEE Transactions on Neural Networks Andlearning Systems, 2019, 31(10): 4229-4238.
[31] MA Z, ZHAO M, CAI X, et al. Fast-convergent federated learning with class-weighted aggregation[J]. Journal of Systems Architecture, 2021, 117: 102125-102131.
[32] ALEXANDRU A B, PAPPAS G J. Private weighted sum aggregationfor distributed control systems[J]. IFAC-Papers OnLine, 2020, 53(2): 11081-11088.
[33] ADACHI R, YAMASHITA Y, KOBAYASHI K. Distributed estimationbased on weighted data aggregation over delayed sensor networks[J]. IFAC Journal of Systems and Control, 2020, 14: 100109-100115.
[34] KULESHOV V, PRECUP D. Algorithms for multi-armed bandit problems[J]. arXiv: 1402.6028, 2014.

Yifei ZOU, born in 1994, Ph.D, assistant professor. His main research interests include wireless networks, ad hoc networks, and distributed computing.

Dongxiao YU, born in 1984, Ph.D, professor. His main research interests include wireless networks, distributed computing, and graph algorithms.

(Responsible editor: Yahui LI)