Edge Resource Prediction and Auction for Distributed Spatial Crowdsourcing With Differential Privacy

Yin Xu, Mingjun Xiao, *Member, IEEE*, An Liu, *Member, IEEE*, and Jie Wu, *Fellow, IEEE*

***Abstract*—Traditional spatial crowdsourcing (SC) systems employ a centralized server platform to provide services for requesters. Such a centralized design requires powerful resource capacity and often cannot accomplish the urgent demands due to the unpredictable network latency. In order to ensure the scalability of systems and the quality of services, we study the distributed SC (DSC), where a diversity of location-relative services provided by various service providers (SPs) can deploy on edge clouds (ECs) with low time latency. Since the edge resources are limited, SPs need to compete for edge resources so as to deploy their desired SC services, and the requested resources must be allocated together to meet the demand of the service. We first design a gated recurrent unit with particle filter (GRUPF) network for SPs to predict future resource demands so as to participate in the competitions judiciously. Then, we model the competitive edge resource allocation problem between SPs and ECs as a combinatorial auction process. Due to the NP-hardness of this problem, an approximation algorithm is proposed to tackle it. Moreover, the leakage of private information such as bids may incur severe economic damage, and most existing studies usually rely on a trusted third party to provide rigorous privacy protection. Therefore, we customize a novel differentially private resource auction (DRA) mechanism, and design a bid confusion strategy based on differential privacy. Through theoretical analysis, we prove that the DRA mechanism meets some desired properties, including ϵ-differential privacy, individual rationality, computational efficiency, and γ-truthfulness. Additionally, we corroborate the significant performances of DRA through extensive simulations on synthetic and real-world data sets.**

Manuscript received 19 February 2022; revised 11 May 2022; accepted 6 June 2022. Date of publication 14 June 2022; date of current version 24 August 2022. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62172386, Grant 61872330, Grant 61572457, Grant 61572336, Grant 61379132, Grant 61936015, and Grant U1709217); in part by NSF under Grant CNS1824440, Grant CNS1828363, Grant CNS1757533, Grant CNS1618398, Grant CNS1651947, and Grant CNS1564128; in part by the NSF of Jiangsu Province in China under Grant BK20191194, Grant BK20131174, and Grant BK2009150; in part by the Anhui Initiative in Quantum Information Technologies under Grant AHY150300; in part by the National Key Research and Development Program of China under Grant 2017YFB0802302; and in part by the Natural Science Research Project of Jiangsu Higher Education Institution under Grant 18KJA520010. This article was presented in part at the 25th International Conference on Database Systems for Advanced Applications (DASFAA) [DOI: 10.1007/978-3-030-59416-9_47]. *(Corresponding author: Mingjun Xiao.)*

Yin Xu and Mingjun Xiao are with the School of Computer Science and Technology, Suzhou Institute for Advanced Research, the CAS Key Laboratory of Wireless-Optical Communications, University of Science and Technology of China, Hefei 230026, China (e-mail: xuyin218@mail.ustc.edu.cn; xiaomj@ustc.edu.cn).

An Liu is with the School of Computer Science and Technology, Soochow University, Suzhou 215006, China (e-mail: anliu@suda.edu.cn).

Jie Wu is with the Center for Networked Computing, Temple University, Philadelphia, PA 19122 USA (e-mail: jiewu@temple.edu).

Digital Object Identifier 10.1109/JIOT.2022.3183006

***Index Terms*—Auction mechanism, deep learning, differential privacy, distributed spatial crowdsourcing (DSC), particle filter (PF), resource usage prediction.**

I. Introduction

With the prosperous development of mobile Internet and smart mobile devices, spatial crowdsourcing (SC) has attracted increasing attention in utilizing the crowd power to complete complex tasks. A typical SC system is comprised of a server on the cloud, mobile users, and some task requesters. Through the SC server, requesters can crowdsource their tasks to mobile users (also known as, workers) to be accomplished. So far, there has been plenty of research on designing various SC systems and the corresponding privacy-preserving protocols, incentive mechanisms, task assignment, or user recruitment, etc., [1]–[10]. Nevertheless, most of these existing SC systems only involve a single server platform, which can be categorized as the centralized crowdsourcing paradigm. In other words, all user recruitment or task assignment would need to be conducted via the centralized SC server. On the one hand, it would weigh the burden on the centralized server and require powerful resource capacity. On the other hand, owing to the expensive bandwidth and the unpredictable network latency, centralized platforms often fail to accomplish the stringent demands for such latency-sensitive users. Thus, with the growth of the number and types of SC services, these simple centralized systems have become increasingly unable to meet users' needs. Distributed SC (DSC) systems, which can support diverse and efficient services, are becoming popular.

In this article, we focus on the edge resource allocation problem in a DSC system. Generally, a DSC system can support a diversity of location-related services provided by different service providers (SPs). By the aid of edge computing, the DSC system can dynamically deploy these services on specified decentralized edge clouds (ECs) according to the demands of SPs. The ECs can be formed of a number of small-scale computing and storage servers, which are placed at network edges. Consequently, services are closer to mobile users, which can significantly reduce high bandwidth costs and time latency. Fig. 1 illustrates an example of a DSC system. Three SPs hope to provide requesters with their SC services, each of which corresponds to one or more locations. SP_1 provides the traffic condition monitoring service with collecting data from locations $\{d_1, d_2, d_3\}$, the service s_2 provided by SP_2

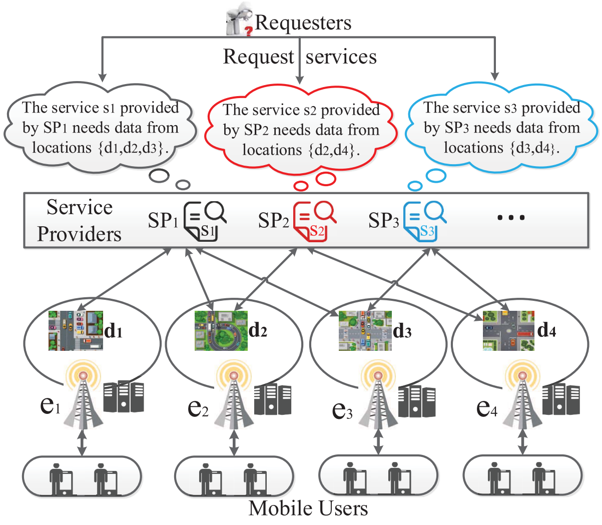

Requesters
Request services
The service s1 provided by SP1 needs data from locations {d1,d2,d3}.
The service s2 provided by SP2 needs data from locations {d2,d4}.
The service s3 provided by SP3 needs data from locations {d3,d4}.
Service Providers
SP1
S1
SP2
S2
SP3
S3
...
d1
d2
d3
d4
e1
e2
e3
e4
Mobile Users

Fig. 1. DSC system model.

needs data from locations $\{d_2, d_4\}$ for air quality monitoring, and SP_3 collects data from locations $\{d_3, d_4\}$ for free parking monitoring. To satisfy these demands, the DSC system can deploy three services to nearby ECs, which cover the related locations. Afterward, the corresponding SPs can recruit mobile users to perform their SC tasks and return the collected data to requesters.

Different from traditional SC systems, which mainly rely on a centralized cloud server, the DSC system needs to deploy services on distributed ECs where the computation and communication resources are generally restricted. Hence, SPs have to compete for the limited edge resources so as to deploy their desired SC services [11]. As we know, designing an auction mechanism is one of the most efficient and fair manners to deal with resource competition issues, and thus, it is also adopted in this article. There are three challenges in the auction mechanism design.

First, there is a need to design a fast and adaptive method for SPs to efficiently predict the usage of resources, such as CPU and memory. Requesting too many resources might result in waste and unnecessary pay, while requesting too few resources might not satisfy service level agreements (SLAs). However, it is difficult to determine accurate resource demands while guaranteeing SLAs and improving resource utilization. Second, the resources that each SP applies for might be distributed among multiple ECs and must be allocated together to support the corresponding SC service. For instance, the service s_1 provided by SP_1 needs to collect the traffic congestion information from three locations $\{d_1, d_2, d_3\}$ during a certain time period, shown in Fig. 1. Because only collecting partial data under the time constraint is meaningless, s_1 is either successfully deployed on an EC bundle $\{e_1, e_2, e_3\}$ simultaneously or fails. Therefore, it actually involves a distributed combinatorial resource allocation issue. Third, bids play a vital role in auctions and may imply some private information (e.g., the valuation of SC services, SPs' interests, and so on). If such sensitive information is divulged, potential adversaries might harness the information to manipulate the auction, leading to unfair edge resource competition. Hence, we must prevent the bids of SPs from being disclosed [12], [13].

Resource prediction using historical sequential data is a widely-studied and long-existing problem. Many typical approaches are derived from traditional or heuristic neural networks [14]–[17] and regression theories [18], [19], which require resource usage with a clear tendency or obvious regularity so as to fulfill the accurate resource prediction. Besides, the traditional neural networks cannot completely explore the correlation between highly variable resource usage. With the superb capability on sequential processing, a state-of-the-art sequence prediction technique, recurrent neural networks (RNNs), has received growing attention. Based on RNNs, some variants [e.g., gated recurrent unit (GRU) [20]] display a great power to learn long-term memory dependencies and avoid gradient vanishing. However, data generated from the real world usually contain some noise, resulting in inaccurate prediction results. Faced with the complexity of uncertain data, one way is to increase the length of the latent vector in networks, but it will sacrifice time efficiency due to the increase of the number of network parameters. Thus, we employ a GRU with particle filter (GRUPF) network for SPs to predict their resource usage. Particle filter (PF) is a dynamic time-discrete filter based on the Monte Carlo method to simulate the transference of particles and to update the estimated state with observation from sensors recursively [21]. Inspired by the idea of using particles to approximate the posterior state distribution, we combine PF with GRU to improve predictive accuracy without lengthening the latent vector.

Although a large number of resource allocation mechanisms based on combinatorial auctions have been proposed [22], [23], most of existing works concentrate on achieving critical economic properties and ignore the significance of bid privacy. Thus far, only a few studies take the privacy issues into account, which generally can be classified into two categories. One category is to encrypt bids by leveraging cryptography techniques [13], [24]–[26]. This category of methods can provide theoretically provable security guarantee while suffering from high computational overheads. The other is to perturb bids by harnessing differential privacy [12], [27]–[29]. Nevertheless, most of such solutions usually rely on a trusted third party. To get around these thorny problems, we customize a differentially private resource auction (DRA) in a DSC system. Specifically, each round of resource allocation process is modeled as a combinatorial auction, which includes a secure winning bid selection problem and a secure payment determination problem. Due to the NP-hardness of winning bid selection, we design an approximation algorithm. Faced with the competitive SPs and the semihonest [30] third-party auctioneer, we take full advantage of differential privacy to achieve bid protection.

Overall, our multifold contributions are listed as follows.

1) A novel DSC system is presented, which can support a diversity of location-related services provided by different SPs. To deal with the edge resource allocation problem in the DSC system, we propose a DRA mechanism, where SPs can compete for the edge resources of ECs so as to deploy their desired SC services.
2) A GRUPF network-based resource prediction algorithm is designed for SPs to learn memory dependencies from historical resource usage. Trace-driven experiments

show that the algorithm can work well on highly variable resource usage and predict more accurately compared with popular networks.

3) We construct a secure combinatorial auction to model the competitive edge resource allocation process with the indivisible requested resources, which includes the secure winning bid selection and the secure payment determination. In addition, we prove the NP-hardness of the bid selection problem, and a greedy algorithm is designed to determine winners and the corresponding reasonable payments.
4) In order to safeguard bid privacy from the untrusted third-party and rivalry SPs, we design a bid confusion strategy in a differentially private manner. By means of this strategy, SPs are allowed to upload confused bids to replace true bids.
5) Through theoretical analysis, we prove that the DRA mechanism satisfies some essential properties, including ϵ-differential privacy, individual rationality, computational efficiency, and γ-truthfulness. Moreover, we carry out lots of simulations on real and synthetic data sets to demonstrate the excellent performance of the DRA mechanism.

The organization of this whole article is presented as follows. Section II introduces our problem in the DSC system model. Section III gives the resource prediction algorithm. The detailed design of the DRA mechanism and the theoretical analysis are elaborated in Sections IV and V, respectively. In Section VI, the simulations and results will be presented. We review some related works in Section VII. After discussing the limitations and the potential future research directions in Section VIII, we make a conclusion in Section IX.

II. Model and Problem Formulation

In this section, we put forward a general DSC system model with the distinct workflow among various parties, followed by the problem description and preliminary.

A. System Model

The DSC model is mainly composed of three parties: 1) many services; 2) a collection of ECs; and 3) the auctioneer. Services proposed by different SPs request the edge resources of ECs, denoted by $S = \{s_1, s_2, \ldots, s_n\}$. Let $E = \{e_1, e_2, \ldots, e_m\}$ denote the set of ECs, and these ECs hold certain limited resources to be used for deploying services. Thus, the ECs can be regarded as sellers, and the services are considered as buyers. The semihonest auctioneer determines the auction results, including the set of winning bids (denoted by W) and the corresponding payments (denoted by P).

For each EC e_j, we use $\langle A_j, c_j\rangle$ to represent the state information where A_j means resource capacity and c_j denotes the unit cost of resources. We use $\langle D_i, Q_i, b_i\rangle$ to describe the state information of each service s_i. SPs bid on a bundle rather than an individual EC. We assume that an SP who possesses s_i is l_i-minded, which indicates that the SP can submit at most l_i bundles $D_i = \{D_{i,1}, D_{i,2}, \ldots, D_{i,l_i}\}$ along with a set of requested resources $Q_i = \{Q_{i,1}, Q_{i,2}, \ldots, Q_{i,l_i}\}$ and a set of the unit bids $b_i = \{b_{i,1}, b_{i,2}, \ldots, b_{i,l_i}\}$, where $D_{i,k} \subseteq E$ $(\forall k \in [1, l_i])$ is a bundle and $Q_{i,k}$ is the set of the corresponding requested resource usage $\{q_{i,j} | \forall e_j \in D_{i,k}\}$. Let B denote the set of all true bids.

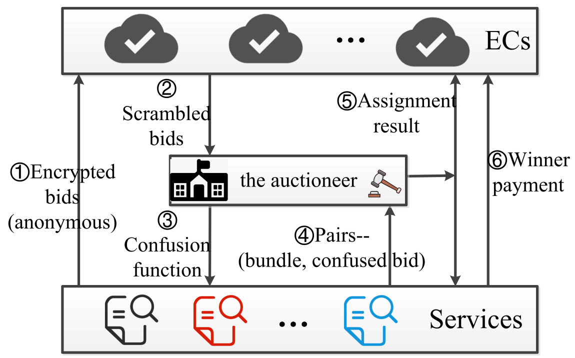


Fig. 2. Workflow of whole system.

Fig. 2 illustrates the interactions among the SPs, ECs, and the auctioneer in each round of resource allocation. Now, we present the workflow of the whole system.

Step 1: On the basis of the preference for various bundles, the SP who provides the service s_i $(\forall i \in [1, n])$ predicts its demanded resource usage $\{q_{i,j} | \forall e_j \in D_{i,k}\}$ through historical usage (to be presented in Section III), and decides the unit bid for each preferred bundle, forming a bundle-bid pair $(D_{i,k}, b_{i,k})$. By using the public key k_p publicized by the auctioneer, SPs encrypt bids via the asymmetric encryption technology, so that the bid information can be protected during the transmission process. Then, all SPs upload encrypted bids to diverse ECs through anonymous communication technology [31]. Note that each SP can upload different encrypted bids to varied ECs but any one encrypted bid only can be uploaded to at most one EC.

Step 2: Each EC executes the shuffle operation, and then transfers all encrypted bids in disorder to the auctioneer. Because of the anonymous transmission, the auctioneer cannot know which bundle a bid belongs to.

Step 3: The auctioneer makes use of its private key k_s to decrypt all encrypted bids. Next, it invokes statistical analysis to obtain the distribution of true bids. Based on this, a confusion function can be generated by taking advantage of the exponential mechanism (to be presented in Section IV-C). Afterward, the auctioneer will broadcast the designed confusion function to all SPs.

Step 4: According to the received bid confusion function, the SP who provides the service s_i $(\forall i \in [1, n])$ computes a confused bid $\widetilde{b}_{i,k}$ to replace the corresponding true bid $b_{i,k}$, and then communicates the bundle-confused bid pairs $(D_{i,k}, \widetilde{b}_{i,k})$ to the auctioneer.

Step 5: When the auctioneer receives all SPs' pairs, the auction process can be triggered. That is, the auctioneer will execute the combinatorial auction algorithm to find out the winning bids (to be elaborated in Algorithm 2). Then, the auctioneer will announce the auction results and notify the corresponding SPs and ECs.

Step 6: Finally, the DSC system deploys the winning services on requested ECs according to the auction results.

The winning SPs can recruit workers to accomplish their tasks and provide their SC services for requesters. Meanwhile, the winning SPs need to pay for the ECs' resources, in which the payment is determined by the auctioneer.

Remark: The auctioneer can define a time interval used for triggering the auction. That is, when the current auction has been accomplished, the auctioneer can start another edge resource auction after the preset time interval. In addition, all requested resources (i.e., $\{Q_1, Q_2, \ldots, Q_n\}$) must have the same type in one resource allocation round, but the resource types of different rounds can be allowed to have diversity. Therefore, each service can have diverse resource requirements (e.g., CPU and memory).

B. Problem Formulation

Our design objective is to maximize the social welfare (SW) while guaranteeing prediction accuracy and bid privacy. The SW is defined as follows.

Definition 1: The *SW* is the total valuations of the winning services minus the total cost.

Next, the DRA problem can be formulated as follows.

Definition 2 (DRA Problem):

$$\text{Maximize}\;\; SW = \sum_{b_{i,k} \in W} \sum_{q_{i,j} \in Q_{i,k}} \left(v_{i,k} - c_j\right) * q_{i,j} \qquad (1)$$

$$\text{Subject to} \sum_{i:(b_{i,k} \in W \cap q_{i,j} \in Q_{i,k})} q_{i,j} \leq A_j \quad \forall e_j \in E \qquad (2)$$

$$\sum_{k=1}^{l_i} \mathbb{1}_{\{b_{i,k} \in W\}} \leq 1 \quad \forall s_i \in S \qquad (3)$$

where $v_{i,k}$ denotes the true unit valuation that the SP evaluates if it deploy its own service s_i on bundle $D_{i,k}$. More specifically, the SP makes preestimation of the reward (paid by requesters) and the recruitment cost (used for recruiting workers). The reward can be estimated according to the historical service records. The recruitment cost arises from the worker recruitment process, which is beyond the scope of this article and can be calculated by adopting many existing state-of-the-art policies [32]–[34]. Then, the true valuation (i.e., $v_{i,k} * q_{i,j}$) is approximately equal to the difference between the reward and the recruitment cost.

Here, (2) indicates the capacity constraint, that is, the total demanded resource usage of services cannot go beyond each EC's capacity. Equation (3) claims that a service can be only deployed on a bundle at most.

Apart from achieving the bid privacy protection, we are reluctant to discard several essential properties, e.g., individual rationality, truthfulness, and computational efficiency. They can be defined in great detail as follows.

Definition 3 (Individual Rationality): For each winning bid $b_{i,k}$, if the corresponding SP has a nonnegative utility, i.e.,

$$u_i = v_{i,k} - p_{i,k} \geq 0 \qquad (4)$$

then the mechanism meets individual rationality.

Definition 4 (γ-Truthfulness): An auction mechanism is γ-truthful in expectation *iff* $E[u_i(b_i', \mathbf{b}_{-i})] \geq E[u_i(b_i, \mathbf{b}_{-i})] - \gamma$ holds for any bid $b_i \neq b_i'$ and any bid profile of other services $\mathbf{b}_{-i}$, where γ is a small positive constant.

Definition 5 (Computational Efficiency [13]): If an auction mechanism can generate results and terminate in a polynomial time, the mechanism is computationally efficient.

C. Preliminary

For a better understanding, we present the introduction on PF and the GRUPF network briefly, which are used in the design of resource prediction.

Definition 6 (PF [21]): The PF is a Bayesian filter, which estimates the belief b_t (i.e., a posterior distribution of the state h_t) given the history of actions $a_{1:t}$ and observations $o_{1:t}$. PF approximates the belief b_t with a set of weighted particles $\{(h_t^\tau, w_t^\tau)\}_{\tau=1}^N$, where $\{h_t^\tau\}_{\tau=1}^N$ are N latent states learned by policy-oriented training, and $\{w_t^\tau\}_{\tau=1}^N$ stands for the corresponding weights. Importantly, the mean state can be estimated as the mean particles, i.e., $\text{mean}(h_t) = \sum_{\tau=1}^N w_t^\tau h_t^\tau$. Moreover, the PF algorithms update the particles in a Bayesian manner. The particle updates include transition update, measurement update, and resampling.

1) *Transition Update:* We sample the next state h_t^τ from a generative transition model, i.e., $h_t^\tau \sim p(h_t | h_{t-1}^\tau, a_t)$.
2) *Measurement Update:* We update the particle weights with the observation model $p(z_t | h_t^\tau)$.

$$w_t^\tau = \eta p(o_t | h_t^\tau) w_{t-1}^\tau$$

$$\eta = 1 / \sum_{i=1}^N p\left(o_t | h_t^\tau\right) w_{t-1}^\tau$$

where η is a normalization factor.

3) *Resampling:* To diminish particle degeneracy, the new particles are resampled based on the weight of importance, with a mean weight of $1/N$.

Definition 7 (GRUPF Network [35]): The GRUPF network is derived from RNNs for improving belief approximation without increasing the length of the latent vector. The GRUPF network not only contains the belief representation and approximate Bayesian inference adopted in PF but also possesses the data-driven approximation capabilities of GRU. Specifically, a GRUPF network approximates the belief as a set of weighted particles $\{(h_t^\tau, w_t^\tau)\}_{\tau=1}^N$, and updates them with the stochastic particle filtering algorithm instead of with a deterministic nonlinear function in the GRU. What is more, the state transition model and the observation model are approximated directly as the learned functions.

On the other hand, differential privacy is a lightweight tool for data privacy protection, which can provide a strong theoretical privacy guarantee for statistics publishing. The exponential mechanism has been widely applied in the privacy-preserving mechanism designs to protect bids. Also, we give some related definitions about differential privacy.

Definition 8 (Differential Privacy (DP) [36]–[38]): A randomized mechanism $\mathcal{M}$ satisfies ϵ-differential privacy if for any two input sets D_1 and D_2 differing on at most one element, and for any set of outcomes $O \subseteq \text{Range}(\mathcal{M})$, there exists $Pr[\mathcal{M}(D_1) \in O] \leq \exp(\epsilon) \times \Pr[\mathcal{M}(D_2) \in O]$. Here, $\epsilon > 0$ is a parameter called as the privacy budget that controls the

TABLE I
DESCRIPTION OF MAJOR NOTATIONS

Variable	Description
E, e_j	the set of all ECs and the j-th EC.
S, s_i	the set of all services and the i-th service.
$\langle A_j, c_j \rangle$	the resource capacity and unit cost of resources.
$\langle D_i, Q_i, b_i \rangle$	the set of preferred bundles, the set of requested resources, and the set of the claimed unit bids.
$D_{i,k}, Q_{i,k}$	a bundle and the set of the corresponding requested resource usage.
$q_{i,j}$	requested resource usage for the EC e_j.
$v_{i,k}, b_{i,k}$	the true valuation and the claimed bid for the bundle $D_{i,k}$.
$\widetilde{b}_{i,k}$	the confused bid in terms of $b_{i,k}$.
P, W, B	all payments, winning bids, the set of all bids.
(h_t^τ, w_t^τ)	the state and weight of the particle.
b_t	a posterior distribution of the state h_t.
x_t, x_t^{norm}	historical resource usage and normalized usage.
$y_t, y_t^{real}, \hat{y}_t$	predicted and real usage after inputting x_t, intermediate predicted result.
$W_{(\cdot)}, I_{(\cdot)}$	coefficient weight and bias term.
η, ζ_t^τ	a normalization factor and a learned noise term.
$\kappa_{1:t}^\tau, \psi$	a historical chain for the particle τ and a weight parameter in the training process.
θ	a trade-off parameter in soft resampling.
ξ, α	the learning rate and the learning rate decay.

strength of privacy protection—the smaller the ϵ, the higher privacy protection level and the lower the data availability.

Definition 9 (Exponential Mechanism): Given an input set A, an outcome space O, a score function f, and a privacy budget ϵ, if a mechanism $\mathcal{M}$ has $\mathcal{M}(A, o) = \{o : |Pr[o \in O] \propto \exp([\epsilon f(A, o)]/[2\Delta f])\}$, $\mathcal{M}$ satisfies ϵ-differential privacy. Here, Δf is the sensitivity of the score function $f(A, o)$.

For ease of reference, we list the frequently used notations throughout this article in Table I.

III. Design of Resource Prediction

In this section, we leverage the GRUPF network for SPs to predict the edge resource usage, such as CPU and memory. The objective is to obtain the requested resource usage as accurately as possible so as to alleviate the resource-wasting, unnecessary payments, and inability to satisfy SLAs.

A. Workflow of Prediction

Given a historical resource usage $X = (x_1, x_2, \ldots, x_t)$, we aim to predict the future resource usage at the time $t + 1$. We let $y_t^{\text{real}} = x_{t+1}$ denote the real resource usage, and y_t denotes the predicted resource usage after inputting x_t to the network. For ease of exposition, we use x_t or y_t instead of $q_{i,j}$ to indicate the requested resource usage, which can avoid confusion caused by the redundant subscripts. The prediction process can be explicitly described as the following steps.

1) Data Preprocessing: The value range of resource usage may be heterogeneous at different time slots. In order to speed up the convergence rate of learning-based algorithms, the origin input should be normalized before the model training process. In this article, we make use of the min–max normalization methods for data preprocessing as follows:

$$x_t^{\text{norm}} = \frac{x_t - \min(X)}{\max(X) - \min(X)}. \tag{5}$$

2) Network Model Training: After data preprocessing, the normalized usage data $X^{\text{norm}} = (x_1^{\text{norm}}, x_2^{\text{norm}}, \ldots, x_t^{\text{norm}})$ is forwarded to the network model training.

First, it is difficult to set the state equation and observation equation of the system in advance when adopting PF. Therefore, we harness neural network prediction instead of the state transition equation of the system. Specifically, we directly approximate the transition model $p(h_t | h_{t-1}^\tau, a_t)$ as a learned function $f_{\text{tr}}(h_{t-1}^\tau, a_t)$ with h_{t-1}^τ and a_t as inputs, instead of formulating it as a generative distribution. Similarly, the observation model $p(o_t | h_t^\tau)$ is replaced by a learned function $f_{ob}(h_t^\tau)$. Thus, the update equations can be represented as follows:

$$h_t^\tau \sim f_{tr}\left(h_{t-1}^\tau, a_t\right) \tag{6}$$

$$w_t^\tau = \eta f_{ob}\left(h_t^\tau\right) w_{t-1}^\tau, \eta = 1 / \sum_{\tau=1}^{N} f_{ob}\left(h_t^\tau\right) w_{t-1}^\tau. \tag{7}$$

Second, multiple neural networks with different parameters are established for sampling particles. Each particle can be regarded as the predicted result of a neural network. Thus, the posterior state distribution can be approximated by a set of N weighted particles $\{(h_t^\tau, w_t^\tau)\}_{\tau=1}^N$, for latent state h_t^τ and weight w_t^τ. Each latent state h_t^τ represents a hypothesis in the belief, and the set of particles provides an approximate representation for the belief.

Finally, we use the learned functions from the GRUPF network to update latent states and predict the resource usage. Specifically, we suppose that the predicted usage is $Y^{\text{train}} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_t)$. Then, the GRUPF network is trained by comparing the errors between the actual usage x_{t+1}^{norm} and the predicted usage $\hat{y}_t$, where x_{t+1}^{norm} denotes the actual resource usage at the time $t + 1$.

3) Prediction: The fitted GRUPF model can be used for 1-step-ahead or multistep-ahead resource usage prediction. Based on the mean particle, the output $\hat{y}_t$ can be calculated by $\hat{y}_t = f_{\text{predict}}(\text{mean}(h_t))$, where $\text{mean}(h_t) = \sum_{\tau=1}^N w_t^\tau h_t^\tau$ and f_{predict} is a prediction function, which transforms the latent state to the output. At last, by converting normalized data to original data, we can obtain the final predicted usage y_t. In order to realize the usage prediction for different future periods, we measure the usage of each trace at each time slot and add it into historical usage, which will be used as the input of the GRUPF network. Thus, after setting the prediction length Len, we can obtain the predicted sequence $Y = (y_t, y_{t+1}, \ldots, y_{t+\text{Len}})$. In this article, we employ the mean absolute error (MAE) to compute the accuracy of resource usage prediction

$$\text{MAE} = \frac{1}{\text{Len}} \sum_{\text{len}=0}^{\text{Len}} \left| y_{t+\text{len}}^{\text{real}} - y_{t+\text{len}} \right|. \tag{8}$$

B. Network Architecture

A standard GRU network architecture for resource prediction needs to maintain a deterministic latent state h_t used for capturing the correlation of the input history, and update h_t

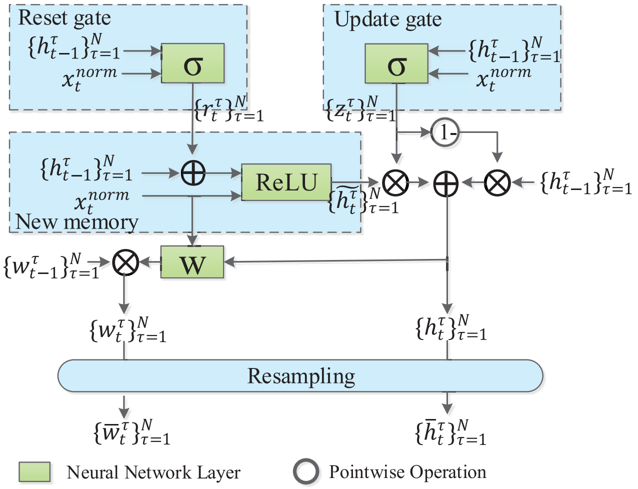

Reset gate
$\{h_{t-1}^{\tau}\}_{\tau=1}^{N}$
x_t^{norm}
σ
$\{r_t^{\tau}\}_{\tau=1}^{N}$
Update gate
σ
$\{h_{t-1}^{\tau}\}_{\tau=1}^{N}$
x_t^{norm}
$\{z_t^{\tau}\}_{\tau=1}^{N}$
1-
$\{h_{t-1}^{\tau}\}_{\tau=1}^{N}$
x_t^{norm}
ReLU
$\{\tilde{h}_t^{\tau}\}_{\tau=1}^{N}$
New memory
$\{h_{t-1}^{\tau}\}_{\tau=1}^{N}$
$\{w_{t-1}^{\tau}\}_{\tau=1}^{N}$
W
$\{w_t^{\tau}\}_{\tau=1}^{N}$
$\{h_t^{\tau}\}_{\tau=1}^{N}$
Resampling
$\{\bar{w}_t^{\tau}\}_{\tau=1}^{N}$
$\{\bar{h}_t^{\tau}\}_{\tau=1}^{N}$
Neural Network Layer
Pointwise Operation

Fig. 3. GRUPF network architecture.

sequentially with new inputs. In order to tackle highly variable noisy resource usage and better mine the sufficient statistic of the input history, the GRUPF network borrows the idea of PF, forming multiple hypotheses over h_t to approximate the posterior state distribution.

As illustrated in Fig. 3, the GRUPF network architecture operates on a variable-length sequence X^{norm} with the three units (i.e., reset gate, update gate, and memory update), so as to maintain a series of latent states $\{h_t^\tau\}_{\tau=1}^N$ and particle weights $\{w_t^\tau\}_{\tau=1}^N$. Then, in order to avoid particle degeneracy, we conduct resampling to construct a new particle set $\{(\bar{h}_t^\tau, \bar{w}_t^\tau)\}_{\tau=1}^N$ for prediction. Similar to a standard GRU network, GRUPF maintains the latent state h_t, and designs two gates (i.e., the update gate z_t and the reset gate r_t) to update it. The difference is that the memory state is comprised of a set of weighted particles $\{(h_t^\tau, w_t^\tau)\}_{\tau=1}^N$, and stochasticity is added to the update process. We first present the mathematical formulation of GRUPF units before diving into the intuition behind this design

$$\text{Reset gate: } r_t^\tau = \sigma\left(W_r\left[h_{t-1}^\tau, x_t^{\text{norm}}\right] + I_r\right) \quad (9)$$

$$\text{Update gate: } z_t^\tau = \sigma\left(W_z\left[h_{t-1}^\tau, x_t^{\text{norm}}\right] + I_z\right) \quad (10)$$

Memory update:

$$\tilde{h}_t^\tau = \text{ReLU}\left(W_h\left[h_{t-1}^\tau \circ r_t^\tau, x_t^{\text{norm}}\right] + I_h + \zeta_t^\tau\right) \quad (11)$$

$$\zeta_t^\tau \sim N\left(0, \varrho_t^\tau\right), \quad \varrho_t^\tau = W_\varrho\left[h_{t-1}^\tau, x_t^{\text{norm}}\right] + I_\varrho \quad (12)$$

$$h_t^\tau = \left(1 - z_t^\tau\right) \circ h_{t-1}^\tau + z_t^\tau \circ \tilde{h}_t^\tau \quad (13)$$

Particle weight update:

$$w_t^\tau = \eta\left(W_p\left[x_t^{\text{norm}}, h_t^\tau\right] + I_p\right) w_{t-1}^\tau \quad (14)$$

where x_t^{norm} is the current input, $W_{(\cdot)}$ is coefficient weight, $I_{(\cdot)}$ is bias term, η is a normalization factor, ζ_t^τ is a learned noise term, and $\circ$ is the elementwise product. We give some intuitive interpretations about the above equations.

1) *Reset Gate:* The gate is used to decide how much past information is forgotten. According to (9), r_t^τ will belong to (0, 1) after using the sigmoid function $\sigma(x) = [1/(1+e^{-x})]$. If r_t^τ is close to 0, the latent state is compelled to ignore the previous latent state and reset with the current input only. Simply speaking, the reset gate can allow the latent state to drop any information that is found to be irrelevant to the computation of the new memory in the future.
2) *Update Gate:* It is devoted to deciding what information is thrown away and what new information is added. According to (13) and $z_t^\tau \in (0, 1)$, we have: if $z_t^\tau \approx 0$, the current input will be ignored and h_{t-1}^τ will be almost entirely copied out to h_t^τ; if $z_t^\tau \approx 1$, the previous latent state will be thrown away and $\tilde{h}_t^\tau$ will be forwarded to h_t^τ. In short, the update gate can determine whether or not the input and the previous latent state are worth retaining.
3) *Memory Update:* A new memory $\tilde{h}_t^\tau$ is updated by partially forgetting the existing memory and adding a new memory according to (11). Especially, we inject the noise ζ_t^τ into $\tilde{h}_t^\tau$ based on (12). We assume that the noise is subject to the Gauss distribution and use the reparameterization trick [39] to imitate the transition in a differentiable way. The advantages of stochastic memory update using ζ_t^τ are adding particle diversity to relieve particle depletion and capturing the randomness of the latent dynamics. Moreover, we adopt the ReLU activation to alleviate the overfitting problem and avoid gradient vanishing. Based on z_t^τ, h_{t-1}^τ, and $\tilde{h}_t^\tau$, the latent state can be updated by (13).
4) *Particle Weight Update:* The particle weight w_t^τ is updated according to (6) and (7). Recall that we employ a learned function $f_{ob}(h_t^\tau)$ to replace the observation model, so that we can circumvent the design of the observation equation. Therefore, (14) is used for updating the particle weight w_t^τ, in which W_p and I_p need to be learned.

Based on the presented GRUPF network architecture, the prediction algorithm for resource usage can be proposed in Algorithm 1. After initialization, we call the normalization method to attain the normalized usage X^{norm}. The learning rate decay α is used for controlling the learning rate ξ. Reducing ξ appropriately can improve the efficiency of learning at different stages and is conducive to training the GRUPF network. Then, we begin to train the model using (9)–(14) in the GRUPF network. In order to avoid the particle set degeneration problem (i.e., except for a few particles, the weights of other particles are small enough to be negligible.), resampling particles is necessary. We adopt the soft-resampling strategy [40], which provides approximate gradients for the nondifferentiable resampling step. Specifically, we sample particles $\{\bar{w}_t^\tau\}_{\tau=1}^N$ from a softened distribution $p(\tau) = \theta w_t^\tau + (1-\theta)(1/N)$, where θ denotes a tradeoff parameter and we set $\theta = 0.5$ in our experiments. Using important sampling $\bar{w}_t^\tau = ([w_t^\tau]/[\theta w_t^\tau + (1-\theta)(1/N)])$, we can get new particles $\{(\bar{h}_t^\tau, \bar{w}_t^\tau)\}_{\tau=1}^N$. For each training sequence, the prediction loss designed in this article is Loss $= L_{\text{MSE}} + \psi L_{\text{ELBO}}$, $L_{\text{MSE}} = (1/|O|)\Sigma_{t\in O}(y_t^{\text{real}} - y_t)^2$, $L_{\text{ELBO}} = -\sum_{t\in\mathcal{O}} \log(1/N) \sum_{\tau=1}^N p(y_t|\kappa_{1:t}^\tau, x_{1:t}^{\text{norm}})$ [35], where O is the set of time indices with outputs, $\kappa_{1:t}^\tau$ is a historical chain for the particle τ, and ψ is a weight parameter. Especially, we leverage the backpropagation through time (BPTT) as the training algorithm for GRUPF.

Algorithm 1 GRUPF-Based Prediction Algorithm for Resource Usage

```
Require: Historical usage $X = (x_1, x_2, \ldots, x_t)$.
Ensure: Predicted usage $y_t$.
Initialize: epoch number $EP$, the learning rate $\xi$, the learning rate decay $\alpha$;
Data preprocessing: normalization using (5) and obtain the new input $X^{\text{norm}} = (x_1^{\text{norm}}, x_2^{\text{norm}}, \ldots, x_t^{\text{norm}})$;
Model training:
Segment the epoch $EP$ and get breakpoints set $BP$;
for each training epoch $ep = 1, 2, \ldots, EP$ do
    if $ep \in BP$ then
        $\xi = \xi * \alpha$;
    end if
    for each batch do
        Update model using (9)~(14);
        Resampling: $\bar{w}_t^\tau = \frac{w_t^\tau}{\theta w_t^\tau + (1-\theta)(1/N)}$;
        Calculate $Loss$ and use BPTT for training;
    end for
end for
Prediction: $\hat{y}_t = f_{\text{predict}}(\sum_{\tau=1}^{N} w_t^\tau h_t^\tau)$;
Anti-normalization to get final predicted usage $y_t$.
```

IV. Design of the DRA Mechanism

After each SP determines its requested resource usage by conducting Algorithm 1, we propose a DRA mechanism to cope with edge resource allocation. It begins with the analysis of the DRA problem hardness and then figures out the basic idea of DRA, followed by the detailed design of bid confusion and the auction process.

A. Problem Hardness Analysis

First, we analyze the complexity of the DRA problem.

Theorem 1: The DRA problem is NP-hard.

Proof: We consider a special case of the DRA problem without the privacy concern, where there is only one EC totally and each SP submits a bid for the EC at most. Without loss of generality, we choose e_1 as the EC. Next, the DRA problem is reduced to determine a subset $B' \subseteq B$ so as to maximize $\sum_{b_{i,1} \in B'} (v_{i,1} - c_1) * q_{i,1}$, while meeting $\sum_{i: b_{i,1} \in B'} q_{i,1} \leq A_1$. This special problem is equivalent to the 0-1 knapsack problem: under the condition that the constraint $\sum_{i=1}^{n} w_i * x_i \leq C, x_i \in \{0, 1\}$ is satisfied, the goal is to maximize the total value (i.e., $\sum_{i=1}^{n} v_i * x_i$). As we all know, the 0-1 knapsack problem is a typical NP-hard problem, so that the special DRA problem is also NP-hard. Consequently, the general DRA problem while considering privacy protection is at least NP-hard. ■

B. Basic Idea

We customize the DRA mechanism in a DSC system, in which the exponential mechanism is embedded into the combinatorial auction skillfully. The objective of our mechanism design is to maximize the SW while achieving bid-privacy preservation, truthfulness, individual rationality, and computational efficiency simultaneously.

In order to conceal the sensitive information of bids, we design a global bid confusion function through which the bids of SPs will be perturbed using the exponential mechanism. Unlike the general bid protection designs that need to rely on a trusted third party, our design can provide a stronger privacy assurance to protect bid privacy for SPs. More specifically, we take full advantage of the asymmetric encryption and the anonymous communication technologies to acquire encrypted bids, which can keep the ECs from knowing true bids and will generate the bid confusion function. Moreover, we employ local differential privacy to shield bids from the untrusted third party.

Given the bundle-confused bid pairs, we model the competitive edge resource allocation problem as a secure combinatorial auction, which is composed of the secure winning bid selection and the secure payment determination. Without knowing true bids, we take the expectation of bids according to the bid confusion function as the input of the auction algorithm. Owing to the NP-hardness of the DRA problem, we propose a greedy algorithm to determine winning bids with the objective of optimizing the SW, and design the pricing strategy for winners without sacrificing some critical economic properties.

C. Bid Confusion

In order to protect the bid privacy, we make full use of the exponential mechanism to design a bid confusion function, which can map a true bid b to a confused bid $\tilde{b}$. Based on Definition 9, we can define the confusion function as

$$P\left(\tilde{b}|b\right) \propto \exp\left(\frac{\epsilon f\left(b, \tilde{b}\right)}{2\Delta f}\right). \tag{15}$$

Here, $P(\tilde{b}|b)$ is the probability of mapping the true bid b to the confused bid $\tilde{b}$. $f(b, \tilde{b})$ denotes the score function measuring the closeness of the confused bid $\tilde{b}$ to the true bid b. The higher the score is, the closer the two are.

In our article, a monotonically nonincreasing function can be put to use so as to meet the properties of the score function. The score function is designed as

$$f\left(b, \tilde{b}\right) = -\ln\left(\left|b - \tilde{b}\right| + 1\right). \tag{16}$$

Based on (16), the score function is satisfactory because the smaller the difference between a true bid b and a confused bid $\tilde{b}$, the higher the probability $P(\tilde{b}|b)$. Meanwhile, the sensitivity of the score function is $\ln(|\hat{b} - \check{b}| + 1) = \ln(\Delta b + 1)$, where $\hat{b}$ and $\check{b}$ denote the maximum value and minimum value in B, respectively, and we define $\Delta b = \hat{b} - \check{b}$.

By substituting the score function into (15), the confusion function can be expressed as follows:

$$\begin{aligned} P\left(\tilde{b}|b\right) &\propto \exp\left(-\frac{\epsilon * \ln\left(\left|b - \tilde{b}\right| + 1\right)}{2\ln(\Delta b + 1)}\right) \\ &= \frac{\exp\left(-\frac{\epsilon * \ln\left(\left|b - \tilde{b}\right| + 1\right)}{2\ln(\Delta b + 1)}\right)}{\sum_{b' \in B} \exp\left(-\frac{\epsilon * \ln(|b - b'| + 1)}{2\ln(\Delta b + 1)}\right)}. \end{aligned} \tag{17}$$

Algorithm 2 Secure Winning Bid Selection

```
Require: $S, E, B$
Ensure: $W$
1:  Initialize $G = \emptyset, W = \emptyset$;
2:  //Compute Grade:
3:  for $s_i \in S$ do
4:      Initialize $G_i = \emptyset$;
5:      for $D_{i,k} \in D_i$ do
6:          Initialize $g_{i,k} = 0, count = 0$;
7:          for $e_j \in D_{i,k}$ do
8:              $g_{i,k} = g_{i,k} + c_j, count = count + 1$;
9:          end for
10:         $g_{i,k} = E[b_{i,k}] - \frac{g_{i,k}}{count}$;
11:         $G_i = G_i + \{g_{i,k}\}$;
12:     end for
13:     $G = G + \{G_i\}$;
14: end for
15: //Greedy Selection:
16: while $S \neq \emptyset$ and $E \neq \emptyset$ and $B \neq \emptyset$ do
17:     Record the index with the maximum grade $g_{i,k}$ as $(i^*, k^*)$;
18:     Initialize $flag = 1$;
19:     for $q_{i^*,j} \in Q_{i^*,k^*}$ do
20:         if $q_{i^*,j} > A_j$ then
21:             $flag = 0$ and break;
22:         end if
23:     end for
24:     if $flag = 1$ then
25:         for $q_{i^*,j} \in Q_{i^*,k^*}$ do
26:             $A_j = A_j - q_{i^*,j}$;
27:         end for
28:         $W = W + \{E[b_{i^*,k^*}]\}$;
29:         $S = S - \{s_{i^*}\}$;
30:         $G = G - \{G_{i^*}\}$;
31:     else
32:         $G_i = G_i - \{g_{i^*,k^*}\}$ and Update $G$;
33:         if $G_i = \emptyset$ then
34:             $S = S - \{s_{i^*}\}$;
35:         end if
36:     end if
37: end while
```

Finally, all SPs receive the designed confusion function broadcasted by the auctioneer. Then, each SP utilizes its true bid $b_{i,k}$ and the confusion function to compute the probability $P(\widetilde{b}_{i,k}|b_{i,k})$. According to different perceptions about privacy and the urgent need level, each SP selects a confused bid $\widetilde{b}_{i,k}$ from the probability distribution judiciously.

D. Auction Mechanism Design

1) Secure Winning Bid Selection: With the ultimate goal of maximizing the SW, the biggest obstacle of selecting winning bids is that the auctioneer only knows the confused bids without the ability to infer the true bids. In response to the difficulty, we harness the expected bids to approximate the true bids. Given the confused bid $\widetilde{b}_{i,k}$ and the confusion function, the expected bid $E[b_{i,k}]$ can be calculated as follows:

$$E[b_{i,k}] = \frac{\sum_{b_{i,k}\in B} P(\widetilde{b}_{i,k}|b_{i,k})P(b_{i,k})P_e(b_{i,k}) * b_{i,k}}{\sum_{b_{i,k}\in B} P(\widetilde{b}_{i,k}|b_{i,k})P(b_{i,k})P_e(b_{i,k})} \tag{18}$$

where $P(b_{i,k}) = \text{num}(b_{i,k})/|B|$ denotes the probability of $b_{i,k}$ in the set B. The function $\text{num}(b_{i,k})$ counts the frequency of $b_{i,k}$ in the set B and $|\cdot|$ denotes the cardinality of the set. We use $P_e(b_{i,k})$ to represent the probability that a true bid $b_{i,k}$ exists in the bundle-bid pairs for bundle $D_{i,k}$, which can be estimated by the following formula:

$$P_e(b_{i,k}) = \frac{\sum_{\widetilde{b}_{i,k}\in B_k} P(\widetilde{b}_{i,k}|b_{i,k})}{\sum_{b_{i,k}\in B}\sum_{\widetilde{b}_{i,k}\in B_k} P(\widetilde{b}_{i,k}|b_{i,k})} \tag{19}$$

where B_k denotes the set of confused bids of the SPs who desire to purchase resources from the bundle $D_{i,k}$.

On the basis of the above designs, the secure winning bid selection process can be completed by the following steps.

Step 1 (Compute Grade): We first give a grade vector for each service ($\forall s_i \in S$). The grading rule is based on (20). $g_{i,k}$ means the grade for the bundle $D_{i,k}$

$$G_i = \left\{ g_{i,k} = E\big[b_{i,k}\big] - \frac{\sum_{e_j\in D_{i,k}} c_j}{|D_{i,k}|} \middle| \forall k \in [1, l_i] \right\}. \tag{20}$$

Step 2 (Greedy Algorithm): The DRA problem is proven to be NP-hard in Theorem 1. Hence, we design a greedy algorithm to determine the winning services in polynomial time. The basic idea is to compute grades and then pick out an expected bid with the highest grade.

Algorithm 2 exhibits the detailed process of secure winning bid selection, which mainly contains two components. At the beginning, we initialize the set G (used to record all grades) and the set W (used to record final winning bids) as empty sets (line 1). In the first part (lines 2–14), we compute the selection criteria. Specifically, for each bundle $D_{i,k}$ of service s_i, we use a variable $g_{i,k}$ to sum up $\{c_j|e_j \in D_{i,k}\}$ with a counter (lines 5–9). Then, $g_{i,k}$ is transformed into the grade for the bundle $D_{i,k}$ according to (20). Finally, we update the sets G_i and G. In the second part (lines 15–37), we perform the greedy winning bid selection strategy. More concretely, we first find the bundle with the highest grade from G and save its index as (i^*, k^*) (line 17). Then, we set an indicator *flag*, indicating whether the remaining resource capacity can satisfy the requested resource usage (lines 18-23). If flag $= 0$, there must exist an EC, which cannot provide the demanded resource usage for the selected bundle. Thus, the bundle should be removed (line 32). Note that $G_i = \emptyset$ suggests that service s_i does not have any eligible request and should be removed (lines 33–35). If $flag = 1$, a winning bid is found and the specified ECs will allocate the requested resource usage Q_{i^*,k^*} to the corresponding service s_{i^*} (lines 25–27). Also, $E[b_{i^*,k^*}]$ will be added into W. Since a service can be only deployed on a bundle at most (i.e., (3)), we remove the service s_{i^*} and the grade set G_{i^*} from the set S and the set G, respectively (lines 28–30).

2) Secure Payment Determination: Even though the auctioneer does not know the true bundle-bid pairs, it is certainly a fact that each true bid must belong to the interval between $\check{b}$

and $\hat{b}$. To ensure the individual rationality, we design the payment of each SP who holds the corresponding winning service as follows:

$$p_{i,k} = \min\left\{\left|E\left[b_{i,k}\right] - \left(\hat{b} - \check{b}\right)\right|, \check{b}\right\}. \tag{21}$$

V. Theoretical Analysis

We analyze that the DRA mechanism can achieve the desired properties of differential privacy, individual rationality, γ-truthfulness, and computational efficiency.

Theorem 2: The DRA mechanism can meet ϵ-differential privacy, ϵ is the privacy budget.

Proof: We assume that there exists two different bids b_1 and b_2 (i.e., $b_1 \neq b_2$), both of them are transformed into the confused bid $\tilde{b}$ according to Section IV-C. $P(\tilde{b}|b_1)$ means that the probability of mapping the bid b_1 to the confused bid $\tilde{b}$. Similarly, the probability of mapping the bid b_2 to the confused bid $\tilde{b}$ is $P(\tilde{b}|b_2)$. Now, we need to derive an exponential upper bound for $P(\tilde{b}|b_1)/P(\tilde{b}|b_2)$. The specific derivation process is as follows:

$$\begin{aligned}\frac{P\left(\tilde{b}|b_1\right)}{P\left(\tilde{b}|b_2\right)} &= \frac{\frac{\exp\left(-\frac{\epsilon*\ln\left(\left|b_1-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln(|b_1-b'|+1)}{2\ln(\Delta b+1)}\right)}}{\frac{\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln(|b_2-b'|+1)}{2\ln(\Delta b+1)}\right)}}\\ &= \frac{\exp\left(-\frac{\epsilon*\ln\left(\left|b_1-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}\\ &\quad *\frac{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-b'\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln(|b_1-b'|+1)}{2\ln(\Delta b+1)}\right)}.\end{aligned} \tag{22}$$

According to (22), $P(\tilde{b}|b_1)/P(\tilde{b}|b_2)$ can be computed by multiplying two parts. We denote the first half of (22) (i.e., the left part) as $\mathcal{L}$, and further have

$$\begin{aligned}\mathcal{L} &= \frac{\exp\left(-\frac{\epsilon*\ln\left(\left|b_1-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)}\\ &= \exp\left(\epsilon*\frac{\ln\left(\left|b_2-\tilde{b}\right|+1\right)-\ln\left(\left|b_1-\tilde{b}\right|+1\right)}{2\ln(\Delta b+1)}\right)\\ &= \exp\left(\epsilon*\frac{\ln\frac{\left|b_2-\tilde{b}\right|+1}{\left|b_1-\tilde{b}\right|+1}}{2\ln(\Delta b+1)}\right)\\ &\leq \exp\left(\epsilon*\frac{\ln(\Delta b+1)}{2\ln(\Delta b+1)}\right) = \exp\left(\frac{\epsilon}{2}\right).\end{aligned} \tag{23}$$

Next, we denote the last half of (22) (i.e., the right part) as $\mathcal{R}$. In order to derive the bound of $P(\tilde{b}|b_1)/P(\tilde{b}|b_2)$, we continue to compute the upper bound of *right* as follows.

Owing to $[1/(\Delta b+1)] \leq [(|b_2-\tilde{b}|+1)/(|b_1-\tilde{b}|+1)] \leq \Delta b+1$, we can obtain $\ln[1/(\Delta b+1)] \leq \ln[(|b_2-\tilde{b}|+1)/(|b_1-\tilde{b}|+1)] \leq \ln(\Delta b+1)$. Therefore, we get

$$\begin{aligned}\mathcal{R} &= \frac{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-b'\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln(|b_1-b'|+1)}{2\ln(\Delta b+1)}\right)}\\ &\leq \frac{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-b'\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(\frac{\epsilon*[-\ln(|b_2-b'|+1)-\ln(\Delta b+1)]}{2\ln(\Delta b+1)}\right)}\\ &= \frac{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln\left(\left|b_2-b'\right|+1\right)}{2\ln(\Delta b+1)}\right)}{\sum_{b'\in B}\exp\left(-\frac{\epsilon*\ln(|b_2-b'|+1)}{2\ln(\Delta b+1)}\right)*\exp\left(-\frac{\epsilon}{2}\right)}\\ &= \exp\left(\frac{\epsilon}{2}\right).\end{aligned} \tag{24}$$

Finally, according to the induction of $\mathcal{L}$ and $\mathcal{R}$, we derive the upper bound of $P(\tilde{b}|b_1)/P(\tilde{b}|b_2)$, i.e.,

$$\frac{P\left(\tilde{b}|b_1\right)}{P\left(\tilde{b}|b_2\right)} = \mathcal{L}*\mathcal{R} \leq \exp\left(\frac{\epsilon}{2}\right)*\exp\left(\frac{\epsilon}{2}\right) = \exp(\epsilon). \tag{25}$$

Based on Definition 8, the proof of the theorem is completed, i.e., the DRA mechanism satisfies ϵ-DP. ■

Theorem 3: The DRA mechanism satisfies the property of individual rationality.

Proof: We need to prove that each winning SP can acquire a nonnegative utility in the DRA mechanism. Without loss of generality, we consider an arbitrary confused bid $\widetilde{b}_{i,k}$. According to (18), we compute the corresponding expected bid $E[b_{i,k}]$. Afterward, the expected bid would undergo two situations: 1) $E[b_{i,k}] \notin W$ and 2) $E[b_{i,k}] \in W$. If $E[b_{i,k}] \notin W$, the payment $p_{i,k}$ equals to 0. Otherwise, the SP who provides the service s_i should pay $p_{i,k} = min\{|E[b_{i,k}] - (\hat{b} - \check{b})|, \check{b}\}$. Obviously, there is $p_{i,k} \leq \check{b}$. Hence, given that $v_{i,k} \in [\check{b}, \hat{b}]$, the utility of the SP is $v_{i,k} - p_{i,k} \geq 0$. Based on Definition 3, the theorem holds. ■

Theorem 4: The DRA mechanism satisfies $2\epsilon\Delta b$-truthful.

Proof: We use b_1 and b_2 to denote two different true bids for the same bundle $D_{i,k}$ of service s_i. According to Theorem 2, we have $P(\tilde{b}|b_1) \leq \exp(\epsilon)P(\tilde{b}|b_2)$. Thus, the utility expectation of the SP who provides the service s_i is

$$\begin{aligned}E[u_i(b_1)] &= \sum_{\tilde{b}\in B}\left[u_i\left(\tilde{b}\right)P\left(\tilde{b}|b_1\right)\right]\\ &\leq \sum_{\tilde{b}\in B}\left[u_i\left(\tilde{b}\right)\exp(\epsilon)P\left(\tilde{b}|b_2\right)\right] = \exp(\epsilon)E[u_i(b_2)].\end{aligned}$$

Since $u_i = v_{i,k} - p_{i,k} \leq \hat{b} - (E[b_{i,k}] - (\hat{b} - \check{b})) = (\hat{b} - \check{b}) + (\hat{b} - E[b_{i,k}]) \leq 2\Delta b$, we further get

$$\begin{aligned}E[u_i(b_2)] &\geq \exp(-\epsilon)*E[u_i(b_1)] \geq (1-\epsilon)*E[u_i(b_1)]\\ &\geq E[u_i(b_1)] - \epsilon E[u_i(b_1)]\\ &\geq E[u_i(b_1)] - 2\epsilon\Delta b.\end{aligned} \tag{26}$$

Based on Definition 4, the proof has been completed. ■

Theorem 5: The mechanism is computationally efficient.

Proof: The computation overhead of the DRA mechanism is mainly dominated by the bid confusion and the secure winning bid selection. During the process of the bid confusion, each SP can bid for at most $l_{\max} = \max\{l_i | i \in [1, n]\}$ bundles for each service, and there are at most n services in a certain period of time, so the computational overhead is $O(nl_{\max})$. When we compute the grade vectors in the secure winning bid selection, lines 7–9 are enclosed in a loop that iterates at most m times and lines 5–12 are enclosed in a loop that repeats at most $l_{\max}$ times. Thus, lines 3 and 14 are enclosed in a loop that iterates n times having the worst case complexity of $O(nml_{\max})$. For the process of the greedy selection, line 17 needs at most $nl_{\max}$ times to find the maximum grade. Lines 19–23 (or lines 25–27) are enclosed in a loop that iterates at times. Lines 16–37 are enclosed in a loop that repeats at most nl_{max} times and have the complexity of $O(nl_{max}(nl_{max} + m + m))$. The rest have constant time complexities. Consequently, the overall computational overhead of Algorithm 2 is According to Definition 5, the theorem holds. ■

VI. Evaluations

By conducting a series of simulations, we corroborate the performance of the GRUPF-based prediction algorithm for resource usage and the DRA mechanism for edge resource allocation. It starts with the introduction of the compared algorithms and basic simulation settings, followed by the detailed evaluation results. Additionally, we conduct the simulations on a computer with Inter Core i5-10400 CPU @2.9 GHz and 16-GB RAM under a Windows platform, and all experiments are implemented in MATLAB language.

A. Algorithms in Comparison

For the prediction evaluation, we compare the GRUPF network model with the corresponding GRU model and RNN model. Then, since the proposed DRA mechanism is the first solution for the combinatorial auction and the bid-privacy protection in DSC systems against the untrusted third party, we compare our DRA mechanism with some state-of-the-art bid protection algorithms with a trustworthy third party [27], [28]. Nevertheless, we cannot directly compare them since their problems and models in these studies are different from ours. Thus, we extract the basic idea from these algorithms for our model and carefully design three secure resource allocation algorithms for comparison: 1) LIN-M [27]; 2) LOG-M [27]; and 3) DPS [28].

B. Simulation Setup

We implement the GRUPF model for resource usage prediction based on PyTorch. A real data set called Alibaba cluster traces [41] is used in our experiments, which contains the runtime resource usage of 4000 machines in eight days. We select a host machine and feed its CPU usage data into our resource prediction model in batches. Specifically, we randomly split the Alibaba cluster data set into the testing set and the training set. We make use of the training set to train the GRUPF model, i.e., determining the parameters of the neural network. Then, the testing set is used for evaluating the performance of the trained GRUPF model. Besides, we set the initial learning rate, the number of training epochs, and the batch size as 0.01, 50, and 32, respectively. Also, the number of particles is set as 30 and the latent state size is 64. Given a fixed training data set, we hope to get better learning performance by increasing the number of particles, but the results fail to meet our expectations because of the greater computational complexity. Owing to the randomness of the neural network model, we set seeds from 100 to 2000 so as to get the averaged results.

TABLE II
Simulation Settings

Parameter name	Values
the privacy budget	0.1, 0.3, **0.5**, 0.7, 0.9, 1.1, 1.3, 1.5
the number of edge clouds	**20**, 30, 40, 50, 60
the number of services	50, **100**, 150, 200, 250
the unit recruitment cost	0.1, **0.2**, 0.25, 0.3, 0.35, 0.4
range of the unit cost	[1, 5]
range of resource capacity	[10, 20]
range of requested resource usage	[1, 5]

In order to evaluate the auction mechanism, we artificially generate some ECs. Each EC possesses a restricted resource capacity and a unit cost. The edge resource capacity and the unit cost are uniformly distributed over [10, 20] and [1, 5], respectively. For simplicity, we assume that bundles are determined, but the generation of bundles in each round of simulations is random. Next, various SPs hope to deploy some SC services on ECs and can bid for determined bundles. The requested resource usage is generated randomly from [1, 5]. Each bid is approximately equal to the true unit valuation due to the property of truthfulness. Thus, we calculate the difference between the reward and the recruitment cost, and divide it by the demanded resource usage, so as to generate the value of the bid. Specifically, the reward is generated randomly from [20, 30]. The recruitment cost is the product of the unit recruitment cost and the number of recruited workers. We vary the unit recruitment cost from 0.1 to 0.4 with a step of 0.05, and attain the number of recruited workers by adopting the greedy selection solution similar to that in [32]. Then, the number of services ranges from 50 to 250 and the number of ECs is selected from 20 to 60. The values of the privacy budget used in the DP technology belong to [0.1, 1.1] and its default value is set as $\epsilon = 0.5$. Note that under the same setting, all experimental results are averaged on 1000 random repetitions. Moreover, Table II lists some simulation parameters, in which default values are in bold fonts.

To evaluate the performance of our DRA mechanism and compared algorithms, we apply the following metrics.

1) *SW:* It is defined in Section II.
2) *Total Payment:* The payments paid by the winning SPs to the DSC system.
3) *Privacy Leakage:* We measure the privacy leakage of DRA by applying the Kullback–Leibler divergence [27]. We use b and $\tilde{b}$ to denote the true bid and the confused bid, respectively. $P(b = \tilde{b})$ indicates the probability

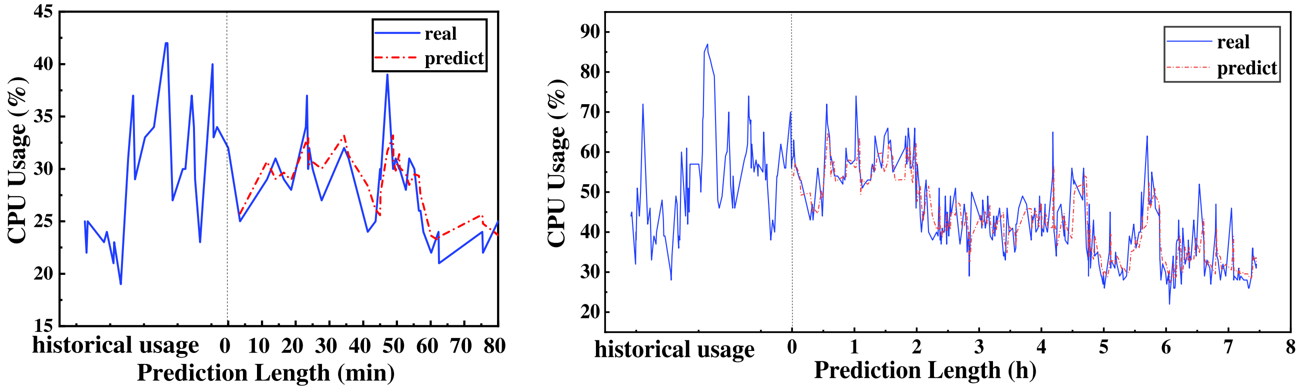


Fig. 4. Evaluation of prediction: minute-level prediction and hour-level prediction.

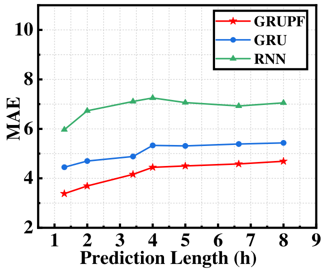


Fig. 5. Prediction accuracy (MAE).

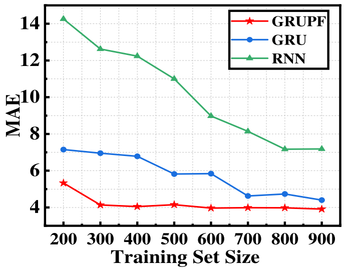


Fig. 6. Evaluation of convergence efficiency.

when b equals $\tilde{b}$. Then, the privacy leakage can be defined as $\text{PL} = (1/[\sum_{b \in B} P_e(b) \ln(1/[P(b = \tilde{b})])])$.

C. Simulation Results

1) Evaluation of Prediction: We first evaluate the performance of the GRUPF network for resource usage prediction, i.e., measuring the CPU usage under different levels of prediction length. Specifically, we depict the performance of GRUPF at minute-level prediction and hour-level prediction, as illustrated in Fig. 4. From the perspective of CPU usage, we can see that the GRUPF network can achieve a highly accurate resource usage prediction. It demonstrates that the GRUPF network can still acquire excellent prediction results even though there exists highly random usage (i.e., exhibiting an extremely random feature) in Alibaba cloud data centers. Moreover, increasing the prediction length will lead to the growth of the prediction errors, but it can also control within a certain range.

Next, we evaluate the performance of the GRUPF network and other recent RNN-based approaches for resource usage prediction, such as the RNN and gated recurrent unit. We mainly compare the prediction accuracy among these methods by measuring MAE. In Fig. 5, we change the hour-level prediction length and then record the MAE of different RNN-based approaches. Generally, increasing the prediction length will result in the growth of MSE for all these methods. It is noteworthy that the GRUPF network obviously exceeds RNN and GRU in terms of prediction accuracy and has a dramatic performance improvement. The reason is that we can make full use of data through particles. Each particle of GRUPF independently aggregates information based on the input history, and then the final output is derived from the averaged particle.

Finally, we evaluate the convergence efficiency among the GRUPF model, GRU model, and RNN model. We provide a fixed number of 100 testing sets, and then change the number of training sets from 200 to 900. Fig. 6 reports the changing trends of MAE under different methods of resource usage prediction. When we enlarge the training set size, the prediction errors of three curves are reduced, and the final averaged MAE of GRUPF is lower than GRU and RNN, which is consistent with the results in Fig. 5. More importantly, we can note that the speed of convergence of GRUPF is faster than GRU, which means the GRUPF has better convergence efficiency.

2) Evaluation of Social Welfare: We compare the SW values of different algorithms (i.e., DRA, LIN-M, LOG-M, and DPS), and the results are shown in Fig. 7. When the number of ECs is 20, Fig. 7(a) illustrates the effect of the number of services on the SW. It can be observed that the SW values of all implemented mechanisms increase along with the growth of the number of services. This is because, with more candidate services, the auctioneer may select more suitable services to allocate resources. Moreover, we can notice that the SW of DPS is higher than that of DRA, LIN-M, and LOG-M. The reason is that DPS can be regarded as an optimal circumstance when selecting winners. That is, there exists a trustworthy third party in the DPS algorithm, which can know all true bids so as to pick out the services with the highest grade. It is worth noting that the SW of DRA is higher than LIN-M and LOG-M according to Fig. 7(a). This reason is that though DRA

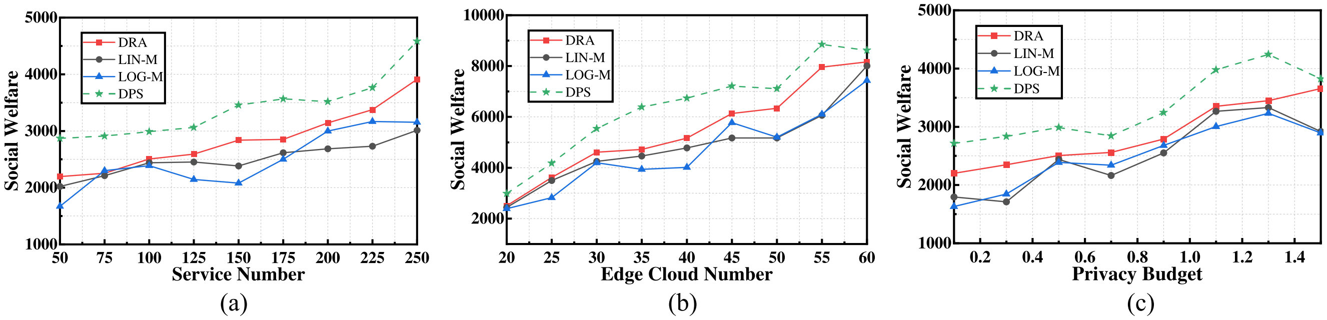


Fig. 7. Evaluation of SW. (a) Effect of the number of services. (b) Effect of the number of ECs. (c) Effect of privacy budget.

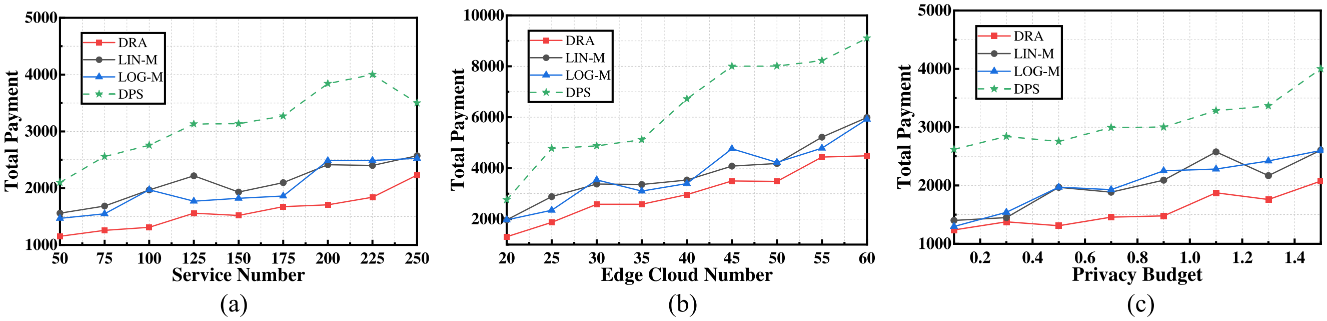


Fig. 8. Evaluation of total payment. (a) Effect of the number of services. (b) Effect of the number of ECs. (c) Effect of privacy budget.

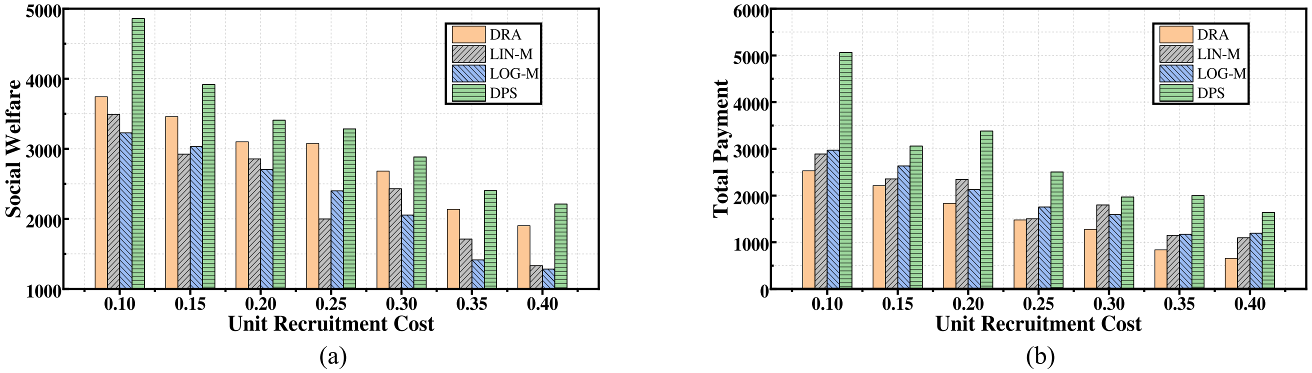


Fig. 9. Effect of unit recruitment cost. (a) SW versus unit recruitment cost. (b) Total payment versus unit recruitment cost.

perturbs the true bids, we opt for the winning SPs based on their expected bids rather than choose them randomly with a certain probability in the LIN-M mechanism and the LOG-M mechanism.

When we change the number of ECs from 20 to 60 with a step of 5 under the circumstance that SPs provide 100 services, we evaluate the SW values of all implemented mechanisms. Fig. 7(b) illustrates the SW values of four mechanisms will raise if the number of ECs grows. This is because each SP would hold more choices on different ECs when deploying their desired services, so that more services will be selected as winners by the auctioneer. Meanwhile, our proposed mechanism DRA can achieve a higher SW compared with LIN-M and LOG-M.

As shown in Fig. 7(c), we vary the values of the privacy budget from 0.1 to 1.5 with a step of 0.05 to evaluate the SW values. The SW values of all implemented algorithms show an increasing trend. It happens because a larger privacy budget indicates a lower privacy protection level. Therefore, a large privacy budget will narrow the gap between the true bids and the expected bids, so as to acquire the high SW.

3) Evaluation of Total Payment: We evaluate the total payments of four mechanisms by changing the number of services, the number of ECs, and the privacy budget. The results are depicted in Fig. 8(a)–(c), respectively. We can see that the changing trend of the total payment is the same as the SW, i.e., the total payment grows slightly when we increase the number of services, the number of ECs, and the privacy budget. The reason is similar to that discussed in Fig. 7. More importantly, the total payment of LIN-M/LOG-M/DPS is higher than that of DRA. This is because the payment determined by the DRA mechanism is less than $\check{b}$ so as to ensure individual rationality based on (21).

4) Effect of Unit Recruitment Cost: In Fig. 9(a) and (b), we can observe the impact of the unit recruitment cost on the SW and the total payment, respectively. When we increase

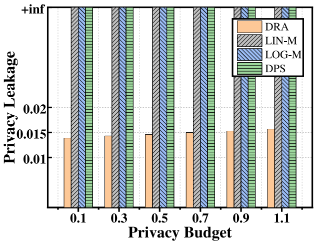


Fig. 10. Evaluation of privacy leakage.

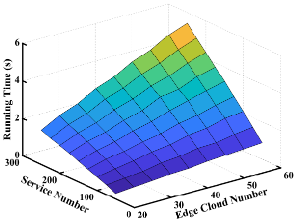


Fig. 11. Evaluation of computational efficiency.

the unit recruitment cost from 0.1 to 0.4, the results show that both the SW and the total payment will have a decrease. This is because, with a higher unit recruitment cost, the SPs might invest more expenses in the worker recruitment process. In such a case, SPs could tend to cut down the payments of resources (i.e., bids) appropriately. Therefore, the total payment also shows a downward trend. Moreover, we can also see that the SW of DRA is higher than LIN-M and LOG-M, and the total payment of DRA is the lowest among all implemented algorithms. The results are consistent with the above simulations (i.e., Figs. 7 and 8).

5) Evaluation of Privacy Leakage: Different privacy budgets might lead to diverse degrees of privacy leakage, and thus, we evaluate the effect of privacy budgets on privacy leakage. Fig. 10 shows that the privacy leakage risk of DRA becomes higher along with the growth of the privacy budget. When we select a small privacy budget, the probability of perturbing any bid to others will be higher according to (17). Thus, the leakage of the bid privacy is small. More importantly, the privacy leakage values of compared mechanisms are positive infinity. The reason is that these mechanisms work with a trusted third party and all bidders will upload true bids to it.

6) Evaluation of Computational Efficiency: As illustrated in Fig. 11, we run our DRA mechanism to verify the computational efficiency. When the number of services and the number of ECs increase, the running time of DRA will grow slowly. Especially, the running time of DRA is less than 6 s when the number of services is 250 and the number of ECs is 60. This time is much smaller than the auction cycle, which indicates that the DRA mechanism can work well in many practical applications. Our theoretical analysis is in accordance with the results of the experiment.

VII. Related Work

Both the resource prediction problem and the resource allocation problem have attracted much attention all the time, while many researchers have worked on addressing two problems separately. Since we focus on dealing with two problems simultaneously and protecting bid privacy, we discuss some related studies in the following three aspects. Specifically, we first review the classical approaches and RNN-based methods for resource usage prediction, then we review some auction-based resource allocation mechanisms and privacy-preserving mechanisms.

A. Methods for Resource Prediction

The prediction of time series is generally conducted by applying the family of autoregressive (AR) models, moving average (MA) models, and Integrated models (I). These traditional models have a strong assumption that the time series may be modeled linearly with some given statistical distribution. Guo *et al.* [18] constructed high-dimensional composite features based on basic features and predicted driver revenue by developing a linear regression model. Although the linear model has interpretability, it is not suitable for long-term prediction. A regressive ensemble method has been proposed in [19] for CPU usage prediction, in which the final prediction results would take the accuracies of eight regression models into account. However, the limitation of this approach lies in consuming the long training time for various models. In the past few years, the emergence of RNN provides a new pathway for usage prediction. For instance, Jiao *et al.* [15] proposed a deep RNN architecture to predict the remaining useful life of the rollers. Duggan *et al.* [42] implemented the RNN to predict host CPU utilization, which can retain information and make predictions with greater accuracy when resolving short-term dependencies. However, the problem of gradient varnishing cannot be addressed by the traditional RNN. To cope with this problem, long short-term memory (LSTM) [43] and GRU were proposed as an improvement of RNN. For example, Bi *et al.* [44] integrated the bidirectional LSTM model and the grid LSTM model to capture the dependence characteristics and different dimension information, which achieves the high-accurate prediction of workloads and resources. Chen *et al.* [45] combined the GRU block and the top-sparse autoencoder (TSA) to solve the long-term memory dependencies, which can attain accurate workload prediction results. Nevertheless, these works need a longer latent vector to achieve accuracy. In [35], PF-RNNs exploit the general idea of sequence prediction. Nevertheless, it does not refer to the resource usage prediction.

B. Auction-Based Resource Allocation Mechanisms

Since auctions can allocate resources in an efficient manner and cater to our goal of maximizing the SW, we focus on investigating auction-based resource allocation mechanisms. So far, auctions have been widely adopted in many state-of-the-art systems for resource allocation. For example, Shi *et al.* [22] introduced an online combinatorial auction framework for resource provisioning dynamically, which achieves truthfulness, SW maximization, and computational efficiency. Kumar *et al.* [23] designed a truthful combinatorial double auction mechanism and payment schemes to trade the cloud resources, which is truthful and weakly budget-balanced. Gao *et al.* [46] devoted to allocating virtual machine resources in network edges by using the auction theory, which achieves an approximately optimal solution and a series of excellent properties. Jiao *et al.* [47] focused on the efficient computing resource allocation between the fog/cloud SPs and miners, and presented an auction-based market model to pursue the optimal SW. Nie *et al.* [6], [48] proposed an incentive mechanism based on the Stackelberg game, which aims to achieve an equilibrium state among all participants, rather than to maximize the SW. Different from the above works, we take the resource competition and security requirements into consideration simultaneously.

C. Privacy-Preserving Mechanisms

In order to achieve the privacy preservation of sensitive bid information, a wide variety of efforts has been devoted to this aspect based on classic techniques (e.g., cryptography techniques and DP). For example, Xiao *et al.* [13] protected the privacy of workers' quotations from being revealed to others by adopting homomorphic encryption. Even though these mechanisms [24]–[26] can keep sensitive information secret completely, they would bring in a large quantity of computation and communication costs. To circumvent the drawback, DP has been proposed to protect bid privacy since it is a lightweight privacy-preserving technique. DP was first introduced by Dwork *et al.* [36]. The first differentially private auction mechanism was introduced in [49], where the mechanism design and the exponential mechanism are incorporated to realize the DP property under different objectives. Hu and Zhang [28] designed a differentially private reverse auction mechanism under a budget constraint for crowdsourcing-based spectrum sensing, and it can protect crowdsourcing workers' bids based on DP and ensure the accuracy of the radio environment map. Han *et al.* [50] developed a dynamic pricing mechanism by using DP and multiarmed bandits for mobile crowdsourcing, which can keep users' costs (i.e., bids) private. The research [27] devised two frameworks (i.e., BidGuard and BidGuard-M) with two score functions based on the exponential mechanism, which can protect bid privacy and approximately minimize the social cost. However, it has a strong assumption that there exists a trusted platform and only protects bids from being leaked to other bidders. That is, if the platform is vulnerable or semihonest, the private information of bids might be disclosed with a high probability.

To address these open challenges, we propose a DRA mechanism in a DSC system. First, we can use the GRUPF network, which combines the GRU and PF into an RNN structure to achieve the accurate prediction of future resource usage. Then, the proposed DRA mechanism can ensure the series of properties in the auction.

VIII. Limitation and Discussion

In this section, we discuss other issues that are not addressed in this work due to space and time constraints, and then point out the potential future research directions.

Unknown Worker Quality and Incomplete Historical Resource Usage: In this article, we need a worker recruitment policy to estimate the recruitment costs, in which a critical issue is how to identify the quality of each worker. Most existing researches assume that the quality information of workers is known in advance, which is not practical. On the other hand, the resource prediction process applies a sequence of historical edge resource usage as input. However, some SPs may hold incomplete or even unknown information about the historical resource usage. Reinforcement learning techniques may be required to learn these unknown or incomplete information. In the future, we will attempt to integrate the unknown worker recruitment mechanism and incomplete resource prediction mechanism into DRA to extend the functionality of the DSC system.

Dynamic Arrival of Services and Workers: Usually, an EC covers a specific geographical area so that workers within its overage can connect to it via wireless access. We assume that all connections between ECs and workers remain unchanged and stable. Meanwhile, all services proposed by different SPs in our DSC system are already fixed and known. Nevertheless, in a multiservice-oriented DSC system, new online services might be publicized anytime. Besides, the smooth mobility of workers should be supported and new workers may participate in the system freely. Therefore, a newly emerging challenge is how to deal with the dynamics of new services and new workers. Based on this challenge, some potential directions may be worth investigating in our future work, e.g., the accurate prediction of the arrival time for new services and workers.

Privacy Protection of Bundle-Bid Pairs: In order to safeguard the privacy of bids from being leaked, all SPs depend on the anonymous communication technology to upload encrypted bids to different ECs, and then report the bundle-confused bid pairs to the auctioneer. Apart from the bid values, the bundle-confused bid pairs of an SP may still involve some sensitive information. For example, the set of bundles may imply the SP's preferences and demands for some ECs. Thus, it would be better to encrypt each SP's bundles via homomorphic encryption locally. On the other hand, the DSC system needs an auctioneer to perform the combinatorial auction algorithm, so as to find out the winning bids and determine the corresponding payments. To further enhance the security and trustworthiness of the whole DSC system, we may take full advantage of the blockchain technology and smart contracts to replace the auctioneer. In future work, we will consider more potential attacks and adopt more sophisticated mechanisms, which may lead to an entirely new research direction.

IX. Conclusion

In this article, we have studied the edge resource allocation problem in a DSC system where SPs need to compete for the finite EC resources so as to deploy their desired SC services. For the purpose of making rational use of resources, we have designed an efficient GRUPF network for SPs to predict demanded resource usage used in the subsequent competitions. Then, the competitive problem of edge resources among SPs can be formalized as a secure combinatorial auction, and we have proposed a DRA mechanism to address this problem. To safeguard the bid privacy from the untrusted third-party and rivalry SPs, we further design a bid confusion strategy based on the exponential mechanism, which can avoid the leakage of SPs' sensitive information by letting SPs upload their confused bids. Moreover, we have proved that the DRA mechanism has some desired properties, including ϵ-differential privacy, individual rationality, computational efficiency, and γ-truthfulness. Extensive simulations have been performed to confirm the excellent performance of DRA.

References

[1] Y. Xu, M. Xiao, X. Zou, and A. Liu, "Differentially private resource auction in distributed spatial crowdsourcing," in *Proc. DASFAA*, (Lecture Notes in Computer Science), vol. 12113, Springer, 2020, pp. 728–745. [Online]. Available: https://doi.org/10.1007/978-3-030-59416-9_47

[2] Y. Tong, Z. Zhou, Y. Zeng, L. Chen, and C. Shahabi, "Spatial crowdsourcing: A survey," *VLDB J.*, vol. 29, no. 1, pp. 217–250, 2020.

[3] Y. Zhao, J. Guo, X. Chen, J. Hao, X. Zhou, and K. Zheng, "Coalition-based task assignment in spatial Crowdsourcing," in *Proc. IEEE ICDE*, 2021, pp. 241–252.

[4] M. Li, J. Wu, W. Wang, and J. Zhang, "Toward privacy-preserving task assignment for fully distributed spatial crowdsourcing," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 13991–14002, Sep. 2021.

[5] J. Zhang, F. Yang, Z. Ma, Z. Wang, X. Liu, and J. Ma, "A decentralized location privacy-preserving spatial crowdsourcing for Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 4, pp. 2299–2313, Apr. 2021.

[6] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and H. V. Poor, "A multi-leader–multi-follower game-based analysis for incentive mechanisms in socially-aware mobile crowdsensing," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1457–1471, Mar. 2021.

[7] Y. Liu *et al.*, "Boosting privately: Federated extreme gradient boosting for mobile crowdsensing," in *Proc. IEEE ICDCS*, 2020, pp. 1–11.

[8] D. Yuan, Q. Li, G. Li, Q. Wang, and K. Ren, "PriRadar: A privacy-preserving framework for spatial crowdsourcing," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 299–314, 2020.

[9] Y. Zhao, K. Zheng, Y. Li, H. Su, J. Liu, and X. Zhou, "Destination-aware task assignment in spatial Crowdsourcing: A worker decomposition approach," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 12, pp. 2336–2350, Dec. 2020.

[10] P. Wang, Z. Chen, M. Zhou, Z. Zhang, A. Abusorrah, and A. C. Ammari, "Cost-effective and latency-minimized data placement strategy for spatial crowdsourcing in multi-cloud environment," *IEEE Trans. Cloud Comput.*, early access, Oct. 14, 2021, doi: 10.1109/TCC.2021.3119862.

[11] H. Yuan and M. Zhou, "Profit-Maximized collaborative computation offloading and resource allocation in distributed cloud and edge computing systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 3, pp. 1277–1287, Jul. 2021.

[12] D. Li, Q. Yang, W. Yu, D. An, Y. Zhang, and W. Zhao, "Towards differential privacy-based online double auction for smart grid," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 971–986, 2020.

[13] M. Xiao *et al.*, "SRA: Secure reverse auction for task assignment in spatial Crowdsourcing," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 4, pp. 782–796, Apr. 2020.

[14] J. Pan, C. Li, Y. Tang, W. Li, and X. Li, "Energy consumption prediction of a CNC machining process with incomplete data," *IEEE/CAA J. Automatica Sinica*, vol. 8, no. 5, pp. 987–1000, May 2021.

[15] R. Jiao, K. Peng, and J. Dong, "Remaining useful life prediction for a roller in a hot strip mill based on deep recurrent neural networks," *IEEE/CAA J. Automatica Sinica*, vol. 8, no. 7, pp. 1345–1354, Jul. 2021.

[16] J. Bi, H. Yuan, M. Zhou, and Q. Liu, "Time-dependent cloud workload forecasting via multi-task learning," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2401–2406, Jul. 2019.

[17] J. Kumar and A. K. Singh, "Workload prediction in cloud using artificial neural network and adaptive differential evolution," *Future Gener. Comput. Syst.*, vol. 81, pp. 41–52, Apr. 2018.

[18] S. Guo *et al.*, "ROD-Revenue: Seeking strategies analysis and revenue prediction in ride-on-demand service using multi-source urban data," *IEEE Trans. Mobile Comput.*, vol. 19, no. 9, pp. 2202–2220, Sep. 2020.

[19] G. Kaur, A. Bala, and I. Chana, "An intelligent regressive ensemble approach for predicting resource usage in cloud computing," *J. Parallel Distrib. Comput.*, vol. 123, pp. 1–12, Jan. 2019.

[20] K. Cho *et al.*, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. EMNLP*, 2014, pp. 1724–1734.

[21] R. van der Merwe, A. Doucet, N. de Freitas, and E. A. Wan, "The unscented particle filter," in *Proc. NIPS*, 2000, pp. 584–590.

[22] W. Shi, L. Zhang, C. Wu, Z. Li, and F. C. M. Lau, "An online auction framework for dynamic resource provisioning in cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 2060–2073, Aug. 2016.

[23] D. Kumar, G. Baranwal, Z. Raza, and D. P. Vidyarthi, "A truthful combinatorial double auction-based marketplace mechanism for cloud computing," *J. Syst. Softw.*, vol. 140, pp. 91–108, Jun. 2018.

[24] Q. Wang, J. Huang, Y. Chen, C. Wang, F. Xiao, and X. Luo, "PROST: Privacy-preserving and truthful online double auction for spectrum allocation," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 2, pp. 374–386, Feb. 2019.

[25] M. F. Balli, S. Uludag, A. A. Selcuk, and B. Tavli, "Distributed multi-unit privacy assured bidding (PAB) for smart grid demand response programs," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4119–4127, Sep. 2018.

[26] Y. Chen, X. Tian, Q. Wang, M. Li, M. Du, and Q. Li, "ARMOR: A secure combinatorial auction for heterogeneous spectrum," *IEEE Trans. Mobile Comput.*, vol. 18, no. 10, pp. 2270–2284, Oct. 2019.

[27] J. Lin, D. Yang, M. Li, J. Xu, and G. Xue, "Frameworks for privacy-preserving mobile crowdsensing incentive mechanisms," *IEEE Trans. Mobile Comput.*, vol. 17, no. 8, pp. 1851–1864, Aug. 2018.

[28] Y. Hu and R. Zhang, "Differentially-private incentive mechanism for Crowdsourced radio environment map construction," in *Proc. IEEE INFOCOM*, 2019, pp. 1594–1602.

[29] Q. Tao, Y. Tong, Z. Zhou, Y. Shi, L. Chen, and K. Xu, "Differentially private online task assignment in spatial crowdsourcing: A tree-based approach," in *Proc. IEEE ICDE*, 2020, pp. 517–528.

[30] O. Goldreich, *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[31] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-basedservices," in *Proc. ICPS*, 2005, pp. 88–97.

[32] S. Sarker, M. A. Razzaque, M. M. Hassan, A. Almogren, G. Fortino, and M. Zhou, "Optimal selection of crowdsourcing workers balancing their utilities and platform profit," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8602–8614, Oct. 2019.

[33] M. T. Arafat, M. H. Emon, S. Sarker, M. A. Razzaque, and M. M. Rahman, "Balancing worker utility and recruitment cost in spatial crowdsensing: A Nash game approach," in *Proc. ACM NSysS*, 2021, pp. 50–59.

[34] S. Saha *et al.*, "Quality-of-experience-aware incentive mechanism for workers in mobile device cloud," *IEEE Access*, vol. 9, pp. 95162–95179, 2021.

[35] X. Ma, P. Karkus, D. Hsu, and W. S. Lee, "Particle filter recurrent neural networks," in *Proc. AAAI*, 2020, pp. 5101–5108.

[36] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. Theory Cryptography Conf.*, 2006, pp. 265–284.

[37] R. Chen, Q. Xiao, Y. Zhang, and J. Xu, "Differentially private high-dimensional data publication via sampling-based inference," in *Proc. ACM SIGKDD*, 2015, pp. 129–138.

[38] L. Pournajaf, D. A. Garcia-Ulloa, L. Xiong, and V. S. Sunderam, "Participant privacy in mobile crowd sensing task management: A survey of methods and challenges," *ACM SIGMOD Rec.*, vol. 44, no. 4, pp. 23–34, 2016.

[39] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. ICLR*, 2014, p. 6.

[40] P. B. Choppala, P. D. Teal, and M. R. Frean, "Soft resampling for improved information retention in particle filtering," in *Proc. ICASSP*, 2013, pp. 4036–4040.

[41] J. Guo *et al.*, "Who limits the resource efficiency of my datacenter: An analysis of Alibaba datacenter traces," in *Proc. IEEE IWQoS*, 2019, pp. 1–10.

[42] M. Duggan, K. Mason, J. Duggan, E. Howley, and E. Barrett, "Predicting host CPU utilization in cloud computing using recurrent neural networks," in *Proc. ICITST*, 2017, pp. 67–72.

[43] S. Li, J. Bi, H. Yuan, M. Zhou, and J. Zhang, "Improved LSTM-based prediction method for highly variable workload and resources in clouds," in *Proc. IEEE SMC*, 2020, pp. 1206–1211.

[44] J. Bi, S. Li, H. Yuan, and M. Zhou, "Integrated deep learning method for workload and resource prediction in cloud systems," *Neurocomputing*, vol. 424, pp. 35–48, Feb. 2021.

[45] Z. Chen, J. Hu, G. Min, A. Y. Zomaya, and T. El-Ghazawi, "Towards accurate prediction for high-dimensional and highly-variable cloud workloads with deep learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 4, pp. 923–934, Apr. 2020.

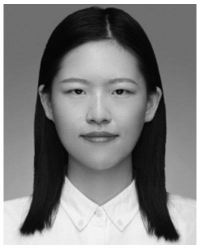
[46] G. Gao, M. Xiao, J. Wu, H. Huang, S. Wang, and G. Chen, "Auction-based VM allocation for deadline-sensitive tasks in distributed edge cloud," *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 1702–1716, Nov./Dec. 2021.

[47] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 9, pp. 1975–1989, Sep. 2019.

[48] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and M. Guizani, "An incentive mechanism design for socially aware Crowdsensing services with incomplete information," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 74–80, Apr. 2019.

[49] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Proc. IEEE FOCS*, 2007, pp. 94–103.

[50] K. Han, H. Liu, S. Tang, M. Xiao, and J. Luo, "Differentially private mechanisms for budget limited mobile Crowdsourcing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 934–946, Apr. 2019.



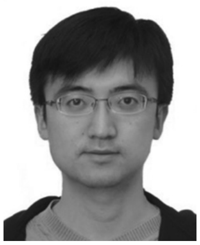
Yin Xu received the B.S. degree from the School of Computer Science and Technology, Anhui University, Hefei, China, in 2019. She is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, University of Science and Technology of China, Hefei.

Her research interests include spatial crowdsourcing, deep learning, auction theory, edge computing, privacy preservation, and resource allocation mechanism.

Mingjun Xiao (Member, IEEE) received the Ph.D. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2004.

He is a Professor with the School of Computer Science and Technology, USTC. He has published more over 90 papers in referred journals and conferences, including IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS (TPDS), IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON SERVICES COMPUTING, INFOCOM, ICDE, and ICNP. His research interests include crowdsourcing, mobile social networks, vehicular ad hoc networks, mobile cloud computing, auction theory, and data security and privacy.

Prof. Xiao served as the TPC member of INFOCOM'21, IJCAI'21, INFOCOM'20, INFOCOM'19, ICDCS'19, DASFAA'19, and INFOCOM'18. He is on the reviewer board of several top journals, such as IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and IEEE TRANSACTIONS ON CLOUD COMPUTING.



An Liu (Member, IEEE) received the Ph.D. degree in computer science from both the City University of Hong Kong, Hong Kong, and the University of Science and Technology of China, Hefei, China, in 2009.

He is a Professor with the Department of Computer Science and Technology, Soochow University, Suzhou, China. He has published more than 80 papers in referred journals and conferences, including IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON SERVICES COMPUTING, *GeoInformatica*, *Knowledge and Information Systems*, ICDE, and WWW. His research interests include spatial databases, crowdsourcing, data security and privacy, and cloud/service computing.

Prof. Liu served as the Workshop Co-Chair of WISE 2017 and DASFAA 2015. He is on the reviewer board of several top journals, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON CLOUD COMPUTING, *ACM Transactions on Internet Technology*, *Journal of Systems and Software*, *Data & Knowledge Engineering*, *Future Generation Computer Systems*, *World Wide Web*, and *Journal of Current Science and Technology*.

Jie Wu (Fellow, IEEE) received the Ph.D. degree in computer engineering from Florida Atlantic University, Boca Raton, FL, USA, in 1989.

He is the Director of the Center for Networked Computing and a Laura H. Carnell Professor with Temple University, Philadelphia, PA, USA, where he also serves as the Director of International Affairs, College of Science and Technology. He served as the Chair of the Department of Computer and Information Sciences from summer 2009 to summer 2016 and an Associate Vice Provost for International Affairs from fall 2015 to summer 2017. Prior to joining Temple University, he was a Program Director with the National Science Foundation and was a Distinguished Professor with Florida Atlantic University, Boca Raton, FL, USA. He regularly publishes in scholarly journals, conference proceedings, and books. His current research interests include mobile computing and wireless networks, routing protocols, network trust and security, distributed algorithms, and cloud computing.

Dr. Wu is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award. He serves on several editorial boards, including IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON SERVICE COMPUTING, *Journal of Parallel and Distributed Computing*, and *Journal of Computer Science and Technology*. He is/was the General Chair/Co-Chair for IEEE IPDPS'08, IEEE DCOSS'09, IEEE ICDCS'13, ACM MobiHoc'14, ICPP'16, IEEE CNS'16, WiOpt'21, and ICDCN'22, as well as the Program Chair/Co-Chair for IEEE MASS'04, IEEE INFOCOM'11, CCF CNCC'13, and ICCCN'20. He was an IEEE Computer Society Distinguished Visitor, an ACM Distinguished Speaker, and the Chair for the IEEE Technical Committee on Distributed Processing. He is a Fellow of AAAS.