IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON CONVERGENCE OF SENSOR NETWORKS, CLOUD COMPUTING, AND BIG DATA IN INDUSTRIAL INTERNET OF THING

Received December 5, 2017, accepted March 8, 2018, date of publication March 13, 2018, date of current version April 25, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2815578

An Effective Sensor Cloud Control Scheme Based on a Two-Stage Game Approach

SUNGWOOK KIM
Department of Computer Science, Sogang University, Seoul 121-742, South Korea
e-mail: swkim01@sogang.ac.kr

This work was supported in part by the Ministry of Science and ICT, South Korea, through the Information Technology Research Center Support Program supervised by the Institute for Information & communications Technology Promotion (IITP) under Grant IITP-2017-2014-0-00636 and in part by Basic Science Research Program through the National Research Foundation of Korea (NRF), Ministry of Education, under Grant NRF-2015R1D1A1A01060835.

ABSTRACT Motivated by complementing the ubiquitous sensor networks and cloud computing technique, a lot of attentions have been drawn to Sensor-Cloud (SC). The idea of SC thrives on the principle of virtualization of physical sensor nodes and has introduced the intermediate processing between physical sensor nodes and end users. This paper proposes an efficient interactive SC control scheme to provide on-demand sensing services for multiple applications. By adopting the game theory, we develop a new two-stage game model, which consists of a judicious mixture of selection and incentive algorithms. In our game model, a user can choose the most adaptable data center to execute its task, and each data center can give appropriate incentives to the participating sensors. The main merit possessed by our two-stage game approach is to shed light on the practical SC control problem while providing excellent adaptability and flexibility to satisfy the different application requirements. To the best of our knowledge, this is the first work to include a novel incentive algorithm in the SC system. Simulation results demonstrate that our approach can outperform existing schemes by about 5%∼15% in terms of the normalized user profit, service delay, and SC system throughput. Finally, we discuss future directions for designing SC control frameworks including other issues.

INDEX TERMS Sensor Cloud, game theory, incentive mechanism, two-stage game approach, Internet of Everything.

I. INTRODUCTION

Due to the rapid advances in sensor electronics, digital communications, and miniaturization manufacturing technologies, small-sized smart sensing devices have become possible. The emergence of smart sensor devices has enhanced the standard of living of mankind with the touch of advanced 5G technology. In numerous real-life applications such as target tracking, battlefield monitoring, tele-monitoring, ubiquitous monitoring and several other applications, sensor devices are used to collect data in a self-governing manner. In the future, smart devices are connected to the global Internet based on the concept of Internet of Everything (IoE). Recent advancement in communication systems as well as all-connected sensing devices has the potential to significantly improve the human life standard [1], [2].

In parallel with the development of smart sensing technology, Cloud Computing (CC) has been implemented rapidly and becomes very common. Usually, *cloud* is used in science to describe a large agglomeration of objects that visually appear from a distance. In the information and communications technology field, it is a kind of Internet-based computing paradigm. This paradigm represents a distributing computing model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources. Therefore, the CC technology can provide flexible and scalable services by integrating the smart sensing devices [3]–[6].

Nowadays, Sensor-Cloud (SC) is receiving a growing attention from both the academic and industrial communities. SC is motivated by incorporating the data sensing ability of wireless sensor networks into the powerful data storage and data processing capabilities of CC. Usually, traditional sensor networks are single-user centric, in which a user organization owns and deploys its personalized sensor network. Typically, this network does not share the accessed data to other user organizations. Without security centric applications, it is obvious that sensed data sharing is more profitable. One of main objectives of the SC is to enable a single sensor network to be able to produce sensing services for multiple applications at the same time. Therefore, the SC has been conceived as a potential solution for multi-organization sensor deployment and data access mechanisms [2], [7].

According to MicroStrains, a SC is formally defined as follows [2]: *SC is a unique sensor data storage, visualization and remote management platform that leverages powerful cloud computing technologies to provide excellent data scalability, rapid visualization, and user programmable analysis*. Therefore, virtualization is the basic concept of SC. It allows the abstraction of physical smart sensing devices into logical units, enabling the efficient usage by multiple independent users. In this approach, multiple applications can be able to co-exist on the same virtualized sensor network while allowing the elimination of tight coupling between services and physical sensor network deployments [8].

Although SC technology has been receiving attention, there are still many challenges to solve. In particular, the appropriate sensor management, service quality control and system efficiency maximization are critical issues. Even though current studies have investigated to develop SC control algorithms, there is currently no study designing an efficient integrated scheme to addressing all above issues in the SC system. In addition, a fine-grained interactive control procedure is necessary to provide high Quality of Service (QoS) to SC users [9].

To design a novel interactive SC control scheme, we need a new control paradigm. Nowadays, interactive conrol procedures to satisfy conflicting objectives are often characterized using game theory. Currently, game theory is extensively used for the modelling and analysis of competition and cooperation situations between the rational agents. Being the control theory of multiple goal-driven agents, game theory can provide effective solutions for dealing with SC situations and questions. Motivated by this factor, we have adopted a game theoretic approach to develop a practical SC control scheme. In this way, we are able to ease the heavy computational burden of theoretically optimal centralized solutions [8], [10].

In this study, the basic concept of our scheme is to design a new two-stage game model while harnessing the synergies between competitive and cooperative interactions among SC agents. Upper level game addresses the data center selection algorithm for each user, and our lower level game handles the incentive algorithm for participating sensors. These algorithms are combined in the proposed SC control scheme that is able to adapt dynamically changing SC network environments. As game players, users, data centers and sensors chose their best strategies to maximize the profit in an entirely distributed fashion. This decision process is operated according to the step-by-step timed manner. Finally, our game model achieves greater and reciprocal advantages for all players while ensuring adaptability and flexibility to obtain the finest solution under the current SC system conditions.

Based on the two-stage game approach, our proposed scheme mainly considers three control decision issues; i) how to make SC participants achieve the best outcome just by truthfully acting, ii) how to select a sensible strategy to maximize the payoff, and iii) how to design a computation mechanism with polynomial time complexity. To effectively address these issues, we focus on three design principles such as *truthfulness*, *individual rationality*, and *computational feasibility*.

A. RELATED WORK

There has been considerable research into the implementation of SC systems. Misra *et al.* [2] focused on the theoretical characterization of virtualization, and presented a mathematical formulation of SC. The process of mapping an application to its physical resources and the procedure for virtualization of the resources were also discussed. In addition, they suggested a paradigm shift of technology from traditional sensor networks to SC architecture while providing a detailed analysis based on the performance metrics such as fault tolerance, lifetime of a sensor node, and energy consumption. Finally, future work was provided about the design issues and standardization of communication protocols [2].

Chatterjee *et al.* [11] proposed a dynamic and optimal pricing scheme for the SC infrastructure. The proposed scheme was consisted of two components; one part addressed the problem of pricing the physical sensor nodes subject to variable demand and utility of users, and their part mainly focused on the pricing incurred due to the virtualization of resources. This scheme took into account the end-users' satisfaction and their net utility as one of the factors to establish the optimality in the pricing. Therefore, the main objective was to maximize the expected individual profit made by the several registered sensor owners along with the profit [11].

The paper [12] discussed the potential applications and recent work about SC and observed control issues regarding green SC. And then, a new multi-method data delivery scheme was introduced to solve two control issues - 1) a large number of repeated data transmissions from the cloud to SC users exclusively increased the demand regarding the energy and resources as well as the bandwidth of SC, 2) substantial data delivery from the cloud to multiple SC users increased the requirement with respect to the energy and resources as well as the bandwidth of SC. This scheme strategically incorporated four kinds of delivery: i) delivery from cloud to SC users; ii) delivery from sensor network to SC users; iii) delivery from SC users to SC users; and iv) delivery from cloudlet to SC users. Finally, evaluation results showed that the proposed scheme could achieve lower delivery cost or less delivery time for SC users [12].

In [13], a new SC architecture, called *Mils-Cloud*, was proposed for the integration of military tri-services in a battlefield scenario. It was a hierarchical architecture of SC with users having different levels of priority. Usually, sensor networks are widely used in the military domain, and their integration with CC enhanced their utilization.

Therefore, a SC-based system is best suited for the specific military objectives. Finally, considering the dynamic nature of military operations and associated decision making processes, adaptive learning mechanisms were incorporated into the proposed system. The results showed that the performance was improved for priority users without compromising the availability for normal users [13].

The study in [14] proposed a solution to the optimal gateway selection problem in the SC system. In particular, the gateway selection problem was modelled as a delay optimization problem in the SC architecture. Authors considered request priorities for gateway selection, that is, all the high priority requests were serviced first, and showed how the user requests can be mapped to the optimal gateway through the SC environment. Finally, they observed that their proposed scheme worked well for delay optimization. The future extension of this work was included how the requests could be serviced by the gateways more effectively in order to have a reliable, efficient and real-time monitoring system [14].

T. Dinh et al proposed the *Efficient Interactive Sensor Cloud* (*EISC*) scheme to provide on-demand periodic sensing services to multiple applications which may have different latency requirements [1]. In the *EISC* scheme, a request aggregator was designed on the SC to aggregate applications' latency requests so that workloads required for physical sensors could be minimized to save energy. In particular, the aggregator enabled a sensor to run only a single task and scheduled while serving multiple applications with different requirements. The *EISC* scheme automatically adjusted the scheduling of sensors to meet the latency requirements of all applications. To control packet latency of sensing flows, a QoS controller located on the SC was designed to control end-to-end packet latency of sensing flows. Finally, analysis and experimental results showed that the *EISC* scheme effectively controled the latency of sensing flows with low signaling overhead and high energy efficiency [1].

E. Rachkidi et al proposed the *Resources Optimization and Efficient Distribution* (*ROED*) scheme to share real and virtual sensors between different applications based on their requirements [15]. To minimize the burden on physical and logical sensors, the *ROED* scheme attempted to minimize the number of instantiated virtual sensors in the SC, as well the traffic to and from the physical and logical sensors while achieving the QoS objectives of applications. Therefore, physical, logical, and virtual sensors were shared across applications if it possible. This scheme formulated the optimization problem to achieve the best distribution of virtual sensors and machines in the SC infrastructure. The simulation results showed that the *ROED* scheme could minimize the deployment cost, as well as the data transit delay within the SC infrastructure [15].

Some earlier studies [1], [2], [11]–[15] have attracted considerable attention while introducing unique challenges in handling the SC control problems. In this paper, we demonstrate that our proposed scheme significantly outperforms these existing *EISC* [1] and *ROED* [15] schemes. These schemes also considered the SC platform with the concept of sensor virtualization and QoS provisioning. Therefore, like as our approach, they had developed control mechanisms based on the inter-relationship among different system agents.

B. CONTRIBUTION

Our work focuses on a dynamic controlling of SC and finding an effective solution, which is a well-balanced performance among conflicting requirements. To analyze the interactions among users, data centers and sensors, we design a new two-stage game model. At the upper level game process, users and data centers are game players, and users select the most adaptable data center to complete their tasks. While selecting the data center, the QoS of the application is also taken into account. At the lower level game process, each data center and set of geographically scattered physical sensors are game players, and their interactions are modelled as an incentive algorithm. Using our two-stage game approach, a fair-balanced solution can be obtained under diversified SC network situations. In summary, the contributions of this paper are as follows:

- **Two-stage game model:** we introduce a new game model while capturing dynamic SC interactions depending widely different and diversified system situations. This approach is generic and applicable to implement a real-world SC control scheme.
- **Data center selection algorithm:** as an upper level game, we design a data center selection algorithm to complete requested tasks. Considering the QoS requirement, users select the most adaptable data center to maximize their payoffs.
- **Incentive algorithm for sensors:** as a lower level game, we implement an incentive mechanism based on the three characteristics such as *truthfulness*, *individual rationality*, and *computational feasibility*.
- **The synergy of combined algorithms:** we explore the sequential interaction of data center selection and incentive distribution algorithms, and jointly design an integrated scheme to approximate an optimal SC performance. The synergy effect lies in its responsiveness to the reciprocal combination of different control algorithms.
- **Performance analysis:** Numerical study shows that our two-stage game approach can improve the normalized user profit, service delay and SC system throughput by 5% to 15% under different service request rates, comparing to the existing *EISC* [1] and *ROED* [15] schemes.

C. ORGANIZATION

The remainder of this article is organized as follows. Section II presents the addressing problem, the formation of our game model, and the work done so far on the proposed scheme. Particularly, selection algorithm and incentive algorithm are explained in detail. In addition, this section highlights the main steps of the proposed scheme. In Section III, we provide our simulation scenario and discuss the experimental results while comparing with some existing methods. Finally, Section IV concludes the work. In this section, we also discuss the remaining open challenges in this research area along with possible solutions.

II. THE PROPOSED INTEGRATED SC CONTROL SCHEME

A. TWO-STAGE GAME MODEL FOR THE SC SYSTEM

In this study, we consider that the SC infrastructure is a new dimension of cloud based sensor-management platform, which is functioning as an interface between the physical and cyber space. There are multiple user applications requesting for various types of sensor data from different regions. Therefore, each user individually submits his service request including seeking data, i.e., temperature, humidity, GPS, wind speed, etc., and the target region to its corresponding Data Center (DC). To effectively response applications' requests, the SC system renders *Sensors-as-a-Service* (*Se-aaS*) to the end-users by sharing the pool of sensors and cloud resources [15]. The scenario we assuming in this work consists of a set of user applications, multiple DCs and Physical Sensors (PSs) for data collections. PSs are located in different geographical locations, and they are grouped logically as a temporal set to meet the request of target application [11], [15].

The SC system virtualizes PSs and maps them into Virtual Sensors (VSs) which are instances that abstract the functionalities of physical and logical data sources. Usually, VSs may be formed across multiple overlapping regions and serve as proxies between the data sources and *Se-aaS* of each user's applications. In the SC infrastructure, several Data Centers (DCs) are geographically distributed as service providers. Residing inside each DC, Virtual Machines (VMs) are generated to manage multiple VSs depending on their allocated resources. Originating from different regions, VSs' servings of particular user applications are stored, and processed within the dedicated VM, which is responsible to form instantaneously a *Se-aaS* [11], [15]. In this study, our major goal is to design and propose a control scheme for the dynamic SC system that would serve particular user applications with requesting data types from the respective VSs. The main entities of our scheme are defined as follows;

Physical Sensors (PSs): to form a physical wireless sensor network, each PS has its own attributes, i.e., ID and type where $\mathbb{S}$ is the set of PSs, i.e., $SN_i \in \mathbb{S} = \{SN_1 \ldots SN_n\}$ and $\mathfrak{I}$ is the set of sensor types, i.e., $\tau_j \in \mathfrak{I} = \{\tau_1 \ldots \tau_m\}$; each SN has its own τ.

Virtual Sensors (VSs): VS is a software emulation of PS, and inherits all attributes from the corresponding PS. $\mathcal{V}$ is the set of VSs where $\mathcal{V}_i \in \mathcal{V}$ represents the i^{th} VS. A mapping function to map a physical sensor SN_i to a virtual sensor $\mathcal{V}_i$ is defined as $\mathcal{F}_{p\to v}(SN_i) = \{\mathcal{V}_i, VM_A\}$ where VM_A is a running virtual machine including the $\mathcal{V}_i$.

Virtual Machines (VMs): VM consists of VSs which are software images of PSs and responsible to provide *Se-aaSs* to user applications. The VM receives the sensing data from the underlying VSs and provides the final information transparently to user applications. Let $\mathfrak{W}$ be the set of VMs where $VM_{A_v} \in \mathfrak{W} = \{VM_{A_1} \ldots VM_{A_u}\}$.

Data Centers (DCs): the SC infrastructure is composed of several geographically distributed DCs. In each DC, a number of created VMs are running to handle user applications. Let $\mathfrak{DC}$ be the set of DCs where $\mathfrak{A}_s \in \mathfrak{DC} = \{\mathfrak{A}_1 \ldots \mathfrak{A}_h\}$.

Application: $\mathbb{A} = \{A_1 \ldots A_v\}$ is the set of user applications, and we characterize a user application A with service type ($\tau \in \mathfrak{I}$), region of interest and QoS requirement. Therefore, A contacts the same type (τ) sensors.

The SC has been a compelling paradigm; a user who requests his application task to its corresponding DC, needs to pay the price for the acting DC, and this DC rewards the participating PSs with incentive. The DC generates a VM to acquire SC services while procuring the information from the VSs to complete that task. Each VS is mapped to a corresponding PS, which is a selfish agent and needs to bear cost in providing its service. To select the most adaptable DC for the task accomplishment, it is natural to consider the user's payment, and how to design an incentive algorithm for the VS's service provisioning. The general architecture of a SC system platform is shown in Figure 1.

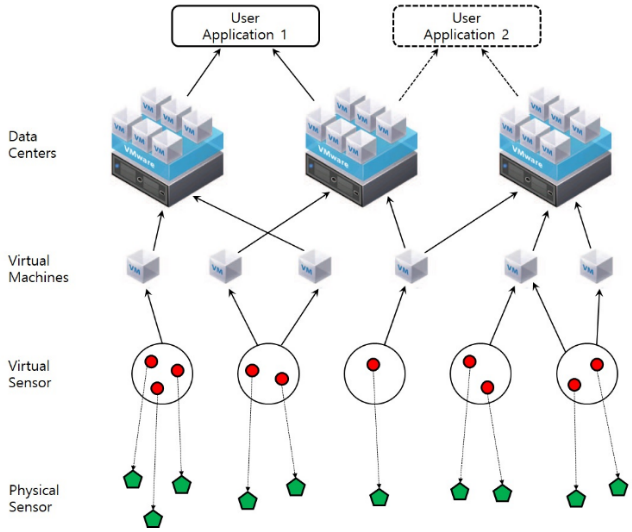


FIGURE 1. The general architecture of a SC system platform.

In this study, we develop a new two-stage game model ($\mathbb{G}$) assuming dynamic SC situations. Formally, we define the $\mathbb{G} = \{\mathbb{G}^{upper}, \mathbb{G}^{lower}_{VM_A \in \mathfrak{W}}\}$ where $\mathbb{G}^{upper}$ is the upper level game to formulate interactions among users and DCs to decide the corresponding DC, and $\mathbb{G}^{lower}_{VM_A \in \mathfrak{W}}$ is the lower level game to formulate interactions among the DC and its corresponding VSs to reward VSs' incentives. During the $\mathbb{G}$ game operations, $\mathbb{G}^{upper}$ and multiple $\mathbb{G}^{lower}_{VM_A \in \mathfrak{W}}$ are executed sequentially and repeatedly. These two-step iterations work together in a coordinated manner, and each application task is completed in an entirely distributed fashion. Table 1 lists the notations used in this paper.

TABLE 1. Parameters used in the proposed algorithm.

Notations	Explanation
SC	sensor cloud
CC	cloud computing
DC	data center for the sensor cloud services
PS	physical sensors in the sensor cloud system
VS	virtual sensor
VM	virtual machine
$\mathbb{S}$	the set of PSs where $\mathbb{S} = \{SN_1 \dots SN_n\}$
SN_i	the i^{th} PS where $SN_i \in \mathbb{S}$
$\mathfrak{I}$	the set of sensor types where $\mathfrak{I} = \{\tau_1 \dots \tau_m\}$
$\boldsymbol{\mathcal{V}}$	the set of VSs
$\mathcal{V}_i$	the i^{th} VS where $\mathcal{V}_i \in \boldsymbol{\mathcal{V}}$
$\mathfrak{W}$	the set of VMs where $\mathfrak{W} = \{VM_{A_1} \dots VM_{A_u}\}$
VM_{A_v}	virtual machine VM_{A_v} where $VM_{A_v} \in \mathfrak{W}$
$\boldsymbol{\mathcal{DC}}$	the set of DCs where $\boldsymbol{\mathcal{DC}} = \{\mathfrak{A}_1 \dots \mathfrak{A}_h\}$.
$\mathfrak{A}_s$	data center where $\mathfrak{A}_s \in \boldsymbol{\mathcal{DC}}$
$\mathbb{A}$	the set of user applications where $\mathbb{A} = \{A_1 \dots A_v\}$
$\mathbb{G}$	two-stage game model where $\mathbb{G} = \{\mathbb{G}^{upper}, \mathbb{G}^{lower}_{VM_A \in \mathfrak{W}}\}$
$\mathbb{G}^{upper}$	the upper level game among users and DCs
$\mathbb{G}^{lower}_{VM_A \in \mathfrak{W}}$	the lower level game among the DC and its corresponding VSs
$\boldsymbol{S}_{A_v}$	the A_v's strategy set where A_v
$\mathcal{P}_k^{A_v}$	the k^{th} price level of $A_v \in \mathbb{A}$
$\boldsymbol{S}^{\mathfrak{A}_s}_{VM_{A_v}}$	the VM_{A_v}'s strategy set where $\mathfrak{A}_s \in \mathcal{DC}$ and $VM_{A_v} \in \mathfrak{W}$
$\mathfrak{P}^{\mathfrak{A}_s}_{VM_{A_v}}$	the VM_{A_v}'s QoS level where $\mathfrak{P}^{\mathfrak{A}_s}_{VM_{A_v}} \in \boldsymbol{S}^{\mathfrak{A}_s}_{VM_{A_v}}$
$\mathcal{H}_t$	the t^{th} time step in the game process
∂_{A_v}	the QoS threshold for the A_v (the predefined minimum delay requirement)
$\mathfrak{y}^{\mathfrak{A}_s}$	the unit cost for the CC to process the A_v
$\mathfrak{p}^{\mathfrak{A}_s}$	the unit cost for the user-DC communication to process the A_v
$\mathfrak{C}^{A_v}_{\tau_m}$	the needed capacity for the CC to process the A_v
$\mathfrak{T}^{A_v}_{\tau_m}$	the needed capacity for the user-DC communication to process the A_v
$\mathcal{R}^{VM_{A_v}}_{\mathcal{V}_i}$	the payment to the $\mathcal{V}_i$ in VM_{A_v}
$\boldsymbol{\mathcal{V}}_{VM_{A_v}}$	the set of VSs for the VM_{A_v}
$\mathfrak{N}^{SN_j}_{\mathcal{V}_j}$	the data transmission delay from the SN_j to $\mathcal{V}_j$
ξ_{A_v}	the A_v's coefficient of service cost distributions between CC and communication
$\mathfrak{C}^{\mathfrak{A}_s}$	the total CC capacity of $\mathfrak{A}_s$
$\mathfrak{T}^{\mathfrak{A}_s}$	The total user-DC communication bandwidth of $\mathfrak{A}_s$
$\Omega_{\mathcal{H}_t}$	the cost for using the $\mathfrak{C}^{\mathfrak{A}_s}$
$\mu_{\mathcal{H}_t}$	the cost for using the $\mathfrak{T}^{\mathfrak{A}_s}$
$\Upsilon^{VM_{A_v}}_{\mathcal{V}}$	the incentive for $\mathcal{V}$ in the VM_{A_v}
$\mathbb{S}_{VM_{A_v}}$	the set of VSs which form the VM_{A_v} where $\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}$
$\mathcal{V}_j^{VM_{A_v}}$	the $\mathcal{V}_i$ in VM_{A_v}
$\boldsymbol{\mathcal{S}}_{\mathcal{V}_j^{VM_{A_v}}}$	the pronounced cost range of $\mathcal{V}_j^{VM_{A_v}}$ to process the requested task A_v
$\mathcal{U}_{\mathcal{V}_j^{VM_{A_v}}}$	the $\mathcal{V}_j^{VM_{A_v}}$'s received payoff
$\delta_{\mathcal{V}_j^{VM_{A_v}}}$	the $\mathcal{V}_j^{VM_{A_v}}$'s true sensing cost
$\overline{\delta}_{\mathcal{V}_j^{VM_{A_v}}}$	the cost bid of $\mathcal{V}_j^{VM_{A_v}}$

B. UPPER LEVEL GAME MODEL FOR THE DC SELECTION ALGORITHM

In the $\mathbb{G}^{upper}$ game process, users and DCs negotiate with each other to decide the service price for each application task and the most adaptable DC is finally selected. From the viewpoint of users, they want to complete their tasks as low price as possible while meeting the requested QoS. To implement the DC selection algorithm, the $\mathbb{G}^{upper}$ can be defined as $\mathbb{G}^{upper} = \left\{\{\mathbb{A}, \mathfrak{DC}\}, \{\boldsymbol{S}_{A_v \in \mathbb{A}}, \boldsymbol{S}^{\mathfrak{A}_s \in \mathfrak{DC}}_{VM_{A_v} \in \mathfrak{W}}\}, T\right\}$ at each time period $\mathcal{H}$ of gameplay. In this model, user applications are used interchangeably with users. Formally, we define the $\mathbb{G}^{upper}$ as follows;

- $\{\mathbb{A}, \mathfrak{DC}\}$ represents a set of $\mathbb{G}^{first}$'s game players.
- $\boldsymbol{S}_{A_v \in \mathbb{A}} = \left\{\mathcal{P}_k^{A_v} \mid \mathcal{P}_k^{A_v} \in \left[\mathcal{P}_{min}^{A_v} \dots \mathcal{P}_k^{A_v} \dots \mathcal{P}_{max}^{A_v}\right]\right\}$ is the A_v's strategy set where $\mathcal{P}_k^{A_v}$ is the A_v's announced price level to complete the requested task. For simplicity, $\boldsymbol{S}_{A_v}$ is discretely quantified.
- $\boldsymbol{S}^{\mathfrak{A}_s \in \mathfrak{DC}}_{VM_{A_v} \in \mathfrak{W}}$ is the VM_{A_v}'s strategy set and VM_{A_v} is in the $\mathfrak{A}_s$. $\mathfrak{P}^{\mathfrak{A}_s}_{VM_{A_v}} \in \boldsymbol{S}^{\mathfrak{A}_s}_{VM_{A_v}}$ represents the VM_{A_v}'s QoS level. In the $\mathbb{G}^{first}$ game, the QoS level is estimated as the A_v' service delay; the range is varied from 0 to ∞.
- $T = \{\mathcal{H}_1, \dots, \mathcal{H}_t, \mathcal{H}_{t+1}, \dots\}$ denotes time, which is represented by a sequence of time steps with imperfect information for the $\mathbb{G}^{upper}$ game process.

To develop the DC selection algorithm, the $\mathbb{G}^{upper}$ is designed as a practical dynamic game model that is a suitable approach to study the interactions among applications and DCs. We consider that A_v with its type τ_m offers the price $\mathcal{P}_k^{A_v}$. According to the application type, QoS threshold (∂_A) is defined beforehand. Based on the $\mathcal{P}_k^{A_v}$, the $\mathfrak{A}_s \in \mathfrak{DC}$ generates the VM_{A_v} and calculate the minimum cost to complete the A_v.

$$\Gamma^{\mathfrak{A}_s}(A_v) = \mathcal{J}^{\mathfrak{A}_s}(A_v) + \mathcal{H}^{\mathfrak{A}_s}(A_v) + \sum_{\mathcal{V}_i \in \boldsymbol{\mathcal{V}}_{VM_{A_v}}} \mathcal{R}^{VM_{A_v}}_{\mathcal{V}_i}$$
$$\text{s.t., } \mathcal{J}^{\mathfrak{A}_s}(A_v) = \mathfrak{y}^{\mathfrak{A}_s} \times \mathfrak{C}^{A_v}_{\tau_m} \text{ and } \mathcal{H}^{\mathfrak{A}_s}(A_v) = \mathfrak{p}^{\mathfrak{A}_s} \times \mathfrak{T}^{A_v}_{\tau_m} \tag{1}$$

where $\mathfrak{y}^{\mathfrak{A}_s}$, $\mathfrak{p}^{\mathfrak{A}_s}$ $\left(or\ \mathfrak{C}^{A_v}_{\tau_m}, \mathfrak{T}^{A_v}_{\tau_m}\right)$ are the unit costs (*or* needed capacities) for the CC and the user-DC communication to process the A_v, respectively. $\mathcal{R}^{VM_{A_v}}_{\mathcal{V}_i}$ is the payment to the $\mathcal{V}_i$ in VM_{A_v} and $\boldsymbol{\mathcal{V}}_{VM_{A_v}}$ is the set of VSs for the VM_{A_v} where $\mathcal{V}_i \in \boldsymbol{\mathcal{V}}_{VM_{A_v}}$. If $\Gamma^{\mathfrak{A}_s}(A_v)$ is less than the $\mathcal{P}_k^{A_v}$, i.e., $\Gamma^{\mathfrak{A}_s}(A_v) < \mathcal{P}_k^{A_v}$, the $\mathfrak{A}_s$ expects to be profitable in the A_v processing, and estimates the service QoS level $\mathfrak{P}^{\mathfrak{A}_s}_{VM_{\mathfrak{A}_v}}$ to complete the A_v by considering the current situation. To estimate the $\mathfrak{P}^{\mathfrak{A}_s}_{VM_{A_v}}$, the $\mathfrak{A}_s$ takes into account three main factors; information collection delay, CC delay and service transmission delay.

$$\mathfrak{P}^{\mathfrak{A}_s}_{VM_{A_v}} = \max\left\{\mathfrak{N}_{\mathcal{V}_j} \mid \mathcal{V}_j \text{ in } VM_{A_v}\right\} + \Theta\left(A_v, \vartheta^{\mathfrak{A}_s}_{VM_{A_v}}\right) + \psi\left(A_v, \vartheta^{\mathfrak{A}_s}_{VM_{A_v}}\right)$$
$$s.t.,\ \vartheta^{\mathfrak{A}_s}_{VM_{A_v}} = \mathcal{P}_k^{A_v} - \sum_{\mathcal{V}_i \in \boldsymbol{\mathcal{V}}_{VM_{A_v}}} \mathcal{R}^{VM_{A_v}}_{\mathcal{V}_i} \tag{2}$$

where $\mathfrak{N}_{\mathcal{V}_j}^{SN_j}$ is the data transmission delay from the SN_j to $\mathcal{V}_j$. $\Theta\left(A_v, \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}\right)$ and $\psi\left(A_v, \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}\right)$ are defined as follows;

$$\begin{cases} \Theta\left(A_v, \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}\right) \\ \quad = \mathcal{D}^{CC}\left(\min\left\{\left(\left\lceil \frac{\xi_{A_v} \times \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}}{\Omega_{\mathcal{H}_t}} \right\rceil \times \mathfrak{C}^{\mathfrak{A}_s}\right), \mathfrak{C}_{\mathcal{H}_t}^{\mathfrak{A}_s}\right\}, A_v\right) \\ \psi\left(A_v, \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}\right) \\ \quad = \mathcal{D}^{Comm}\left(\min\left\{\left(\left\lceil \frac{(1-\xi_{A_v}) \times \vartheta_{VM_{A_v}}^{\mathfrak{A}_s}}{\mu_{\mathcal{H}_t}} \right\rceil \times \mathfrak{T}^{\mathfrak{A}_s}\right), \mathfrak{T}_{\mathcal{H}_t}^{\mathfrak{A}_s}\right\}, A_v\right) \end{cases} \tag{3}$$

where ξ_{A_v} is a coefficient to assign service cost distributions between the CC and the user-DC communication. $\Omega_{\mathcal{H}_t}$ and $\mu_{\mathcal{H}_t}$ are the costs for using the total CC capacity $\left(\mathfrak{C}^{\mathfrak{A}_s}\right)$ and the total user-DC communication bandwidth $\left(\mathfrak{T}^{\mathfrak{A}_s}\right)$. $\mathfrak{C}_{\mathcal{H}_t}^{\mathfrak{A}_h}$ and $\mathfrak{T}_{\mathcal{H}_t}^{\mathfrak{A}_h}$ are the available $\mathfrak{C}^{\mathfrak{A}_s}$ and $\mathfrak{T}^{\mathfrak{A}_s}$ capacities, at the time $\mathcal{H}_t$. $\mathcal{D}^{CC}(\cdot)$ and $\mathcal{D}^{Comm}(\cdot)$ are the delay estimation functions of CC and user-DC communications to process the A_v, respectively. They Each individual DC estimate its QoS level using the equation (2). Finally, the A_v selects the best DC to complete the task A_v as follows;

$$\begin{aligned} &\min\left\{\mathfrak{P}_{VM_{A_v}}^{\mathfrak{A}_s} \mid \mathfrak{A}_s \in \mathfrak{DC}\right\} \\ &\text{s.t., } \mathfrak{P}_{VM_{A_v}}^{\mathfrak{A}_s} \le \partial_{A_v} \text{ and } \Gamma^{\mathfrak{A}_s}(A_v) < \mathcal{P}_k^{A_v} \end{aligned} \tag{4}$$

where ∂_{A_v} is the predefined minimum delay requirement. If all DCs cannot meet the ∂_{A_v} or the $\Gamma^{\mathfrak{A}}(A_v)$, the A_v offers a new $\mathcal{P}^{A_v}$ for the next $\mathbb{G}^{upper}$ game round. At this time, the new $\mathcal{P}^{A_v}$ is larger than the $\mathcal{P}_k^{A_v}$. Therefore, A_v and $\mathfrak{A}_s$ re-negotiate with each other based on the new proposing $\mathcal{P}^{A_v}$. To prevent infinite negotiation rounds, negotiation process must have completed within a negotiation constraint. Our $\mathbb{G}^{upper}$ game process must have completed within a round constraint such as the cardinality of $\boldsymbol{S}_{A_v}$. Therefore, less than the number of $\left|\boldsymbol{S}_{A_v}\right|$ negotiation rounds, the A_v and a specific $\mathfrak{A}$ finally reach an agreement or not.

C. LOWER LEVEL GAME MODEL FOR THE SC INCENTIVE ALGORITHM

According to the result of $\mathbb{G}^{upper}$ game, the $\mathbb{G}_{VM}^{lower}$ game is triggered. In the selected DC, one VM is generating for the assigned task, and this VM distributes incentives to encourage VSs to participate SC services. For example, to complete the application task A_v, the selected DC generate the VM_{A_v}, which consists of VSs, i.e., $\mathcal{V}_i \in \boldsymbol{\mathcal{V}}_{VM_{A_v}}$. Sensors are geographically distributed in a local area, and the VM allocates the sensing work to each $\mathcal{V}$ and pays the incentive $\Upsilon_{\mathcal{V}}^{VM_{A_v}}$. In the $\mathbb{G}_{VM}^{lower}$ game, we employ a truthful incentive algorithm, which is able to ensure the properties including individual rationality and truthfulness of participation.

To design the SC incentive algorithm, our lower level game is defined as

$$\mathbb{G}_{VM_{A_v} \in \mathfrak{W}}^{lower} = \left\{\mathbb{S}_{VM_{A_v}}, \mathcal{P}_k^{A_v}, \mathcal{S}_{\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}}, \mathcal{U}_{\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}}, T\right\}$$

at each time period $\mathcal{H} \in T$ of gameplay. The $\mathbb{G}_{VM_{A_v}}^{lower}$ game is operated based on the selected strategy $\mathcal{P}_k^{A_v}$ made in the $\mathbb{G}^{upper}$ game process. In the SC environment, there can be multiple VMs. Therefore, multiple $\mathbb{G}_{VM}^{lower}$ games work in parallel and independently in a distributed fashion. Formally, we define the $\mathbb{G}_{VM_{A_v}}^{lower}$ as follows;

- $\mathbb{S}_{VM_{A_v}}$ is the set of VSs which form the VM_{A_v}, i.e., $\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}$. They are game players of $\mathbb{G}_{VM_{A_v}}^{lower}$.
- $\mathcal{P}_k^{A_v}$ is the decided price to complete the A_v where $\mathcal{P}_k^{A_v} \in \boldsymbol{S}_{A_v \in \mathbb{A}}$.
- $\mathcal{S}_{\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}}$ is the $\mathcal{V}_j^{VM_{A_v}}$'s strategy set. In other word, $\mathcal{S}_{\mathcal{V}_j^{VM_{A_v}}}$ is the pronounced cost range of $\mathcal{V}_j^{VM_{A_v}}$ to process the requested task A_v.
- $\mathcal{U}_{\mathcal{V}_j^{VM_{A_v}} \in \mathbb{S}_{VM_{A_v}}}$ is the $\mathcal{V}_j^{VM_{A_v}}$'s received payoff from the $\mathbb{G}_{VM_{A_v}}^{lower}$ game process.
- The T is a time period. The $\mathbb{G}_{VM_{A_v}}^{lower}$ is repeated $\mathcal{H}_t \in T < \infty$ time periods with imperfect information.

In our lower game process, assume that the VM_{A_v} includes multiple VSs, i.e., $\left\{\mathcal{V}_1^{VM_{A_v}} \ldots \mathcal{V}_m^{VM_{A_v}}\right\}$ and each $\mathcal{V}^{VM_{A_v}}$ has a heterogeneous cost in provisioning the requested service. For example, when the sensing task is assigned to the $\mathcal{V}_{1 \le j \le m}^{VM_{A_v}}$, the $V_j^{VM_{A_v}}$ incurs a cost $\delta_{V_j^{VM_{A_v}}}$ to contribute its effort in the sensing process. To stimulate each $\mathcal{V}^{VM_{A_v}}$'s truthful participation, our incentive algorithm is formulated as follows. Each sensor is allowed to report its private cost information to perform the assigned sensing task. In order to increase its payoff, each sensor may untruthfully report its cost information to the corresponding VM [16]. Let $\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}$ be the cost bid of $\mathcal{V}_j^{VM_{A_v}}$ and $\delta_{\mathcal{V}_j^{VM_{A_v}}}$ be its true cost. After receiving the bids, the VM_{A_v} needs to decide how much reward, denoted as $\mathcal{R}_{\mathcal{V}_j}^{VM_{A_v}}$, should be paid to $\mathcal{V}_j^{VM_{A_v}}$. The payoff of $\mathcal{V}_j^{VM_{A_v}}$ is accordingly

$$\begin{aligned} &\mathcal{U}_{\mathcal{V}_j^{VM_{A_v}}}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}}\right) = \mathcal{R}_{\mathcal{V}_j}^{VM_{A_v}} - \delta_{\mathcal{V}_j^{VM_{A_v}}} \\ &\text{s.t., } \bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}} = \left(\bar{\delta}_{\mathcal{V}_1^{VM_{A_v}}} \ldots \bar{\delta}_{\mathcal{V}_{j-1}^{VM_{A_v}}}, \bar{\delta}_{\mathcal{V}_{j+1}^{VM_{A_v}}} \ldots \bar{\delta}_{\mathcal{V}_m^{VM_{A_v}}}\right) \end{aligned} \tag{5}$$

Based on the real contribution of $\mathcal{V}_j^{VM_{A_v}}$, we need to design an incentive algorithm satisfying the following desirable properties: i) *truthfulness*, ii) *individual rationality*, and iii) *computational feasibility* [16].

i) ***truthfulness***: truthful reporting is a dominant strategy for the $\mathcal{V}_j^{VM_{A_v}}$, i.e., $\mathcal{U}_{\mathcal{V}_j^{VM_{A_v}}}\left(\delta_{\mathcal{V}_j^{VM_{A_v}}}, \bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}}\right) \ge \mathcal{U}_{\mathcal{V}_j^{VM_{A_v}}}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}}\right)$, for any $\delta_{\mathcal{V}_j^{VM_{A_v}}} \ne \bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}$ and any $\bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}}$.

TABLE 2. System parameters used in the simulation experiments.

Task Type	Cloud Computing	Communication Requirement	Sensing (ψ) cycles	QoS (∂)	ξ	Service Duration
τ_1	200 MHz/s	640 Mbps	200 cycles/s	1.5	0.5	3,000 sec (50 min)
τ_2	240 MHz/s	768 Mbps	225 cycles/s	1.2	0.48	2,700 sec (45 min)
τ_3	270 MHz/s	896 Mbps	250 cycles/s	1.3	0.52	2,880 sec (48 min)
τ_4	330 MHz/s	1.28 Gbps	275 cycles/s	1.5	0.55	2,760 sec (46 min)
Parameter	**Value**	**Description**				
n	1000	the number of physical sensors				
m	4	the number of sensor/application types				
h	10	the number of DCs				
$\mathfrak{y}$	$0.8 \leq \mathfrak{y} \leq 1.4$	the unit cost per Hz for the CC service				
$\mathfrak{p}$	$1 \leq \mathfrak{y} \leq 1.5$	the unit cost per bps for the communication service				
$\bar{\delta}$	$0.7 \leq \mathfrak{y} \leq 1.6$	the unit cost bid of virtual sensor				
δ	$\bar{\delta} - 0.1$	the rue unit cost of virtual sensor				
$\mathfrak{C}$	50 GHz	the cloud computation capacity				
$\mathfrak{T}$	100 Gbps	the DC's front-haul link capacity				
Υ	1	the incentive for each sensor cycle				

ii) ***individual rationality***: the payoff of $\mathcal{V}_j^{VM_{A_v}}$ is non-negative, i.e., $\mathcal{U}_{\mathcal{V}_j^{VM_{A_v}}}\left(\delta_{\mathcal{V}_j^{VM_{A_v}}}, \bar{\boldsymbol{\delta}}_{-\mathcal{V}_j^{VM_{A_v}}}\right) \geq 0$.

iii) ***computational feasibility***: the reward $\mathcal{R}(\cdot)$ should be computed in polynomial time.

To design a truthful incentive algorithm, we first present a payment mechanism determining how much to pay an incentive reward for the $\mathcal{V}_j^{VM_{A_v}}$. The main factors of our incentive algorithm are the cost per payment rate $\Upsilon_{\mathcal{V}_j^{VM_{A_v}}}$ by the VM_{A_v}, and the contribution function $\mathbb{F}\left(\delta_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right)$ from the $\mathcal{V}_j^{VM_{A_v}}$. $\mathbb{F}(\cdot)$ is inspired by the recent work devoted to the contribution maximization problem with limited budget while determining the payment as little as possible to meet the target contribution [16]–[18].

$$
\begin{aligned}
\mathcal{R}_{\mathcal{V}_j}^{VM_{A_v}} &= \mathcal{R}_{\mathcal{V}_j}^{VM_{A_v}}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right) \\
&= \left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}} \times \mathbb{F}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right)\right) \\
&\quad + \int_{\bar{\delta}_{\mathcal{V}_j}}^{\infty} \mathbb{F}\left(\delta_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right) d\delta_{\mathcal{V}_j} \\
\text{s.t.,} & \begin{cases}
\mathbb{F}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right) = 0, \quad \text{if } \left(e - \dfrac{\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}}{\Upsilon_{\mathcal{V}_j^{VM_{A_v}}}}\right) < 1 \\
\mathbb{F}\left(\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}, \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right) = \ln\left(e - \dfrac{\bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}}}{\Upsilon_{\mathcal{V}_j^{VM_{A_v}}}}\right), \\
\text{otherwise}
\end{cases}
\end{aligned} \tag{6}
$$

If $\left(e - \bar{\delta}_{\mathcal{V}_j^{VM_{A_v}}} / \Upsilon_{\mathcal{V}_j^{VM_{A_v}}}\right) < 1$, the $\mathcal{V}_j^{VM_{A_v}}$ does not contribute any effort. According to the Myerson's characterization, $\mathcal{R}(\cdot)$ function maps the cost for a sensing task into a payment for the service provisioning [16], [17].

D. MAIN STEPS OF THE PROPOSED SC CONTROL ALGORITHMS

In this study, the prime focus is to combine the DC selection algorithm and the incentive computation algorithm comprehensively to get full synergy of dynamic SC system operations. However, designing a proper combination of these two algorithms is a particularly challenging problem. Inspired by this, our aim is to offer an integrated SC control scheme based on a novel two-stage game model, which is implemented as a distributed and dynamic repeated game. Our game approach plays a crucial role to provide a suitable tradeoff between the different goals of users, DCs and PSs. Through a step-by-step interactive game process, individual game players can capture the current SC system condition and determine their best strategies to maximize their payoffs; it is a promising approach to implement the real world decision making process. The main steps of the proposed SC control scheme are described as follows.

Step 1: At the initial time, application features and system parameters are determined by the simulation scenario and Table 2.

Step 2: At the upper-level game process, a user independently requests A with τ while offering $\mathcal{P}^A$. Each $\mathfrak{A}$ generates a VM and calculate the $\Gamma^{\mathfrak{A}}(A)$ to complete the task A based on the equation (1).

Step 3: If $\Gamma^{\mathfrak{A}}(A) < \mathcal{P}^A$, the $\mathfrak{A}$ estimates the $\mathfrak{P}_{VM_A}^{\mathfrak{A}}$. By considering three delay factors, it is obtained according to (2).

Step 4: The user selects the best $\mathfrak{A}$ to complete the task A. By using the (4), the $\mathfrak{A}$ with the minimum $\mathfrak{P}_{VM_A}^{\mathfrak{A}}$ is chosen while $\mathfrak{P}_{VM_A}^{\mathfrak{A}} \leq \partial_A$ and $\Gamma^{\mathfrak{A}}(A) < \mathcal{P}^A$.

Step 5: The lower-level game is triggered according to the result of the upper-level game. In the selected $\mathfrak{A}$, the VM announces the sensing task to the $\mathcal{V}^{VM_A}$ and procures the services from the participating sensors.

Step 6: To pay a truthful incentive, we calculate a reward $\mathcal{R}(\cdot)$ for each $\mathcal{V}^{VM_A}$ based on the equation (6).

According to the Myerson's characterization, it can satisfy the desirable incentive properties.

Step 7: Based on the step-by-step interactive game process, users, DCs and sensors attempt to maximize their payoffs by selecting their best strategies, and cause a cascade of interactions in an online distributed fashion.

Step 8: Under the dynamic SC system environment, individual users are constantly requesting applications for the next two-stage game process; go to **Step 2**.

III. PERFORMANMCE EVALUATION

In this section, we evaluate the performance of our proposed protocol, and compare it with that of the existing *EISC* [1] and *ROED* [15] schemes. Based on the simulation results, we confirm the superiority of the proposed approach. To ensure a fair comparison, the following simulation assumptions and system scenario are used.

- 1000 physical sensors are used in which the sensors are distributed randomly in a region of size 10,000 m × 10,000 m.
- Each sensor and application type is randomly decided among $\{\tau_1, \tau_2, \tau_3, \tau_4\}$.
- The simulated system consists of 100 users, and the 10 DCs. They are geographically distributed.
- Each DC's front-haul link capacity ($\mathfrak{T}$) is 100 Gbps, and cloud computation capacity ($\mathfrak{C}$) is 50 GHz.
- The service price strategies in $\boldsymbol{S}_{A_v}$ are defined as $\mathcal{P}^{A_v}_{1=min} = 1$, $\mathcal{P}^{A_v}_{2} = 1.2$, $\mathcal{P}^{A_v}_{3} = 1.4$, $\mathcal{P}^{A_v}_{4} = 1.6$, $\mathcal{P}^{A_v}_{4} = 1.8$ and $\mathcal{P}^{A_v}_{6=max} = 2$ for each service unit, i.e., Hz/s, bps and cycles/s.
- According to each user's characteristics, application tasks are generated based on the Poisson process, which is with rate λ (services/s), and the range is varied from 0 to 3.
- There are 4 different application requests, which are specified according to the connection duration, CC and bandwidth requirements, and sensing cycles per second. They are generated with equal probability.
- All applications need the full region coverage information.
- System performance measures obtained on the basis of 100 simulation runs are plotted as functions of the application service request generation rate.
- For simplicity, we assume the absence of physical obstacles in the experiments.

To demonstrate the validity of our proposed method, we measured the normalized user profit, service delay and SC system throughput. Table 2 shows the system parameters used in the simulation. Major system control parameters of the simulation, presented in Table 2, facilitate the development and implementation of our simulator.

Figure 2 gives the performance comparison of each scheme in terms of the normalized user profit. In this simulation study, the user profit is measured as the ratio of paid price to each application task. From this figure, we can see that all schemes exhibit a similar trend. However, the proposed scheme selects the most adaptable DC through the interactive negotiation process and completes the requested task to meet its preferences. Therefore, users and SC system can reach jointly a mutually acceptable agreement within the constraint round of negotiation. It leads to higher user profit, and provides an ideal solution characterized by SC system environments. Therefore, the proposed scheme outperforms the existing methods from low to high service request load distributions.

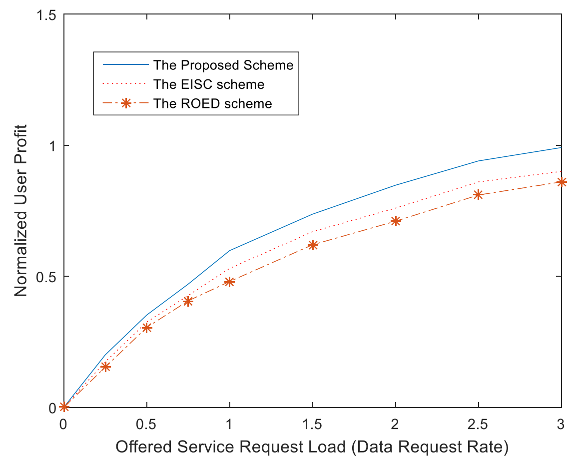


FIGURE 2. **Normalized user profit.**

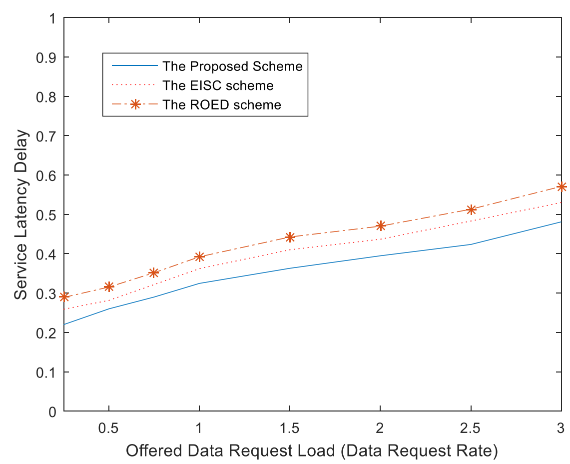


FIGURE 3. **Normalized service latency delay.**

Figure 3 presents the normalized service delay for each scheme. From the viewpoint of QoS, service delay is an important performance criterion. It is intuitively correct to see that the service delay slowly increases under the heavy service request situations. In the simulation result, we have supposed that our decision criteria to select the DC plays an important role to reduce the service latency. Among various QoS constraints, we especially focus on the delay issue and attempt to minimize the service delay. For this reason, the proposed scheme can attain the lower service latency delay to other schemes. In addition, it is also interesting to see that we consistently maintain the operational excellence from low to high service load intensities.

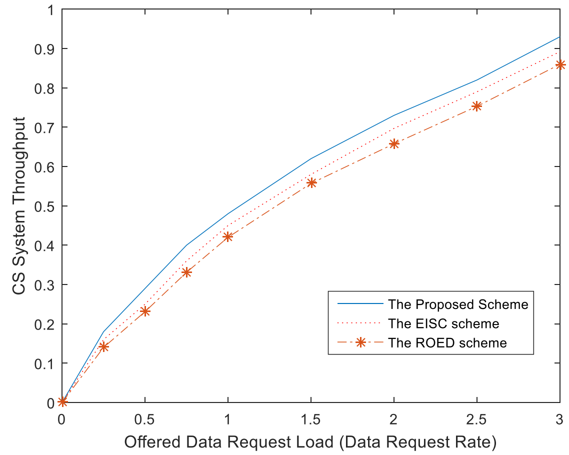


FIGURE 4. CS system throughput.

The curves in Figure 4 indicate the normalized throughput of SC system. From the viewpoint of system operator, throughput maximization is a main concern. The curves in Fig. 4 indicate that our scheme improves the system throughput performance more significantly than the existing schemes. According to our interactive two-stage game model, SC system agents individually recognize the current SC system environment, and effectively decide control decisions to improve their payoffs. In particular, the VM in selected DC distributes incentives to encourage VSs to participate SC services. Therefore, many VSs actively perform the SC services, and successfully complete application tasks; it leads a higher system throughput.

In summary, simulation results shown in Figures 2 to 4 demonstrate that the proposed scheme, which uses a novel two-stage game model, can monitor the current SC system conditions and leverages the DC selection and the incentive algorithm to get the full synergy of SC system operation. Through ongoing game theoretic operations and functionality, users, DC and sensors make intelligent control decisions in a self-adapting manner to maximize their payoffs. Therefore, game players can avoid the inevitable burden of uncertainty while achieving the improved system performance. From the viewpoint of practical operations, it is a suitable approach for real world SC system management. In conclusion, simulation results show that our scheme attains an attractive SC system performance, something that the *EISC* [1] and *ROED* [15] schemes cannot offer.

IV. SUMMARY AND CONCLUSIONS

Recently, sensor networks and cloud computing integration has been receiving a great interest among researchers. To make networked connections more relevant and valuable, SC technology enables sharing the pool of sensors and cloud resources to complete various applications. In this article, we have proposed an integrated SC control scheme by combining the DC selection and incentive algorithms. In particular, we formulate the SC control problem as a new two-stage game process that models the relations among users, DCs and sensors to successfully complete application tasks. Using the step-by-step interactive game process, we explore effective answers to the fundamental questions of how to complete tasks with the best price and how to design an incentive algorithm while ensuring desirable properties. By capturing the tradeoff between cooperative and non-cooperative propensities, all SC agents in our scheme interact sequentially with each other, and select their best strategies to maximize their expected benefits. Compared to the state-of-the-art protocol, simulation results are presented to show the superiority of our integrated scheme. Further, our two-stage game approach will be an interesting topic for future work. New research issues will focus on algorithm designs for optimization of SC resource utilization. In addition, dynamic VM migration methods can be proposed to ensure localized load sharing and balancing among the DCs. By current research, security issue still lacks exploration. Therefore, another interesting direction is to address the network security issues in the SC system from the operator's perspective.

COMPETING OF INTERESTS

The author, Sungwook Kim, declares that there are no competing interests regarding the publication of this paper.

REFERENCES

[1] T. Dinh and Y. Kim, "An efficient sensor-cloud interactive model for on-demand latency requirement guarantee," in *Proc. IEEE ICC*, May 2017, pp. 1–6.

[2] S. Misra, S. Chatterjee, and M. S. Obaidat, "On theoretical modeling of sensor cloud: A paradigm shift from wireless sensor network," *IEEE Syst. J.*, vol. 11, no. 2, pp. 1084–1093, Jun. 2017.

[3] Y. Pan and N. Hu, "Research on dependability of cloud computing systems," in *Proc. IEEE ICRMS*, Aug. 2014, pp. 435–439.

[4] Y. Liu, Y. Sun, J. Ryoo, S. Rizvi, and A. V. Vasilakos, "A survey of security and privacy challenges in cloud computing: Solutions and future directions," *J. Comput. Sci. Eng.*, vol. 9, no. 3, pp. 119–133, 2015.

[5] K. Lee and I. Shin, "User mobility model based computation offloading decision for mobile cloud," *J. Comput. Sci. Eng.*, vol. 9, no. 3, pp. 155–162, 2015.

[6] W. Zhu and C. Lee, "A new approach to Web data mining based on cloud computing," *J. Comput. Sci. Eng.*, vol. 8, no. 4, pp. 181–186, 2014.

[7] C. Zhu, V. C. M. Leung, K. Wang, L. T. Yang, and Y. Zhang, "Multi-method data delivery for green sensor-cloud," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 176–182, May 2017.

[8] L. Guijarro, V. Pla, J.-R. Vidal, and M. Naldi, "Game theoretical analysis of service provision for the Internet of Things based on sensor virtualization," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 3, pp. 691–706, Mar. 2017.

[9] K. Kim, S. Uno, and M. Kim, "Adaptive QoS mechanism for wireless mobile network," *J. Comput. Sci. Eng.*, vol. 4, no. 2, pp. 153–172, 2010.

[10] S. Kim, *Game Theory Applications in Network Design*. Hershey, PA, USA: IGI Global, 2014.

[11] S. Chatterjee, R. Ladia, and S. Misra, "Dynamic optimal pricing for heterogeneous service-oriented architecture of sensor-cloud infrastructure," *IEEE Trans. Serv. Comput.*, vol. 10, no. 2, pp. 203–216, Mar./Apr. 2017.

[12] C. Zhu, V. C. M. Leung, K. Wang, L. T. Yang, and Y. Zhang, "Multi-method data delivery for green sensor-cloud," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 176–182, May 2017.

[13] S. Misra, A. Singh, S. Chatterjee, and M. S. Obaidat, "Mils-cloud: A sensor-cloud-based architecture for the integration of military Tri-services operations and decision making," *IEEE Syst. J.*, vol. 10, no. 2, pp. 628–636, Jun. 2016.

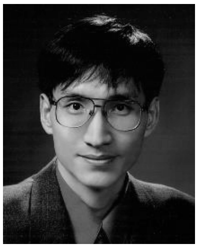
[14] S. Misra, S. Bera, A. Mondal, R. Tirkey, H.-C. Chao, and S. Chattopadhyay, "Optimal gateway selection in sensor-cloud framework for health monitoring," *IET Wireless Sensor Syst.*, vol. 4, no. 2, pp. 61–68, Jun. 2014.

[15] E. El Rachkidi, N. Agoulmine, N. Chendeb, and D. Belaid, "Resources optimization and efficient distribution of shared virtual sensors in sensor-cloud," in *Proc. IEEE ICC*, May 2017, pp. 1–6.

[16] W. Wu *et al.*, "Incentive mechanism design to meet task criteria in crowdsourcing: How to determine your budget," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 2, pp. 502–516, Feb. 2017.

[17] N. Anari, G. Goel, and A. Nikzad, "Mechanism design for crowdsourcing: An optimal 1-1/e competitive budget-feasible mechanism for large markets," in *Proc. IEEE FOCS*, Oct. 2014, pp. 266–275.

[18] R. Myerson, "Optimal auction design," *Math. Oper. Res.*, vol. 6, no. 1, pp. 58–73, 1981.



SUNGWOOK KIM received the B.S. and M.S. degrees in computer science from Sogang University, Seoul, South Korea, in 1993 and 1995, respectively, and the Ph.D. degree in computer science from Syracuse University, Syracuse, NY, USA, in 2003, under the supervision of Prof. P. K. Varshney. He has held faculty positions with the Department of Computer Science, Chung-Ang University, Seoul. In 2006, he returned to Sogang University, where he is currently a Professor with the Department of Computer Science and Engineering. He is the Research Director of the Network Research Laboratory. His research interests include resource management, online algorithms, adaptive quality-of-service control, and game theory for network design.