IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 17, 2022, accepted June 12, 2022, date of publication June 15, 2022, date of current version June 20, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3183237

Bargaining Game Based Offloading Service Algorithm for Edge-Assisted Distributed Computing Model

SUNGWOOK KIM
Department of Computer Science, Sogang University, Mapo-gu, Seoul 04107, South Korea
e-mail: swkim01@sogang.ac.kr

This work was supported in part by the Ministry of Science and ICT (MSIT), South Korea, under the Information Technology Research Center (ITRC) Support Program through the Institute for Information & Communications Technology Planning & Evaluation (IITP) under Grant IITP-2021-2018-0-01799; and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2021R1F1A1045472.

ABSTRACT Computation offloading is a new paradigm to provide cloud computing capabilities at the edge of pervasive radio access networks in close proximity to mobile users. In this paper, we propose a new computation offloading model for the 5G networks and beyond. Based on the edge computing platform, intensive computing tasks can be partially offloaded from local devices to edge clouds to supplement the computation capability of resource-limited devices. This approach leverages the edge server's idle computing power to assist individual devices in model training. To implement control decision algorithms for the distributed computing process, we adopt the concepts of different bargaining solutions for the dynamic offloading services. According to the cooperative game theory, the proposed method can maximize the full synergy that gives mutual advantages for devices and edge clouds while improving the system efficiency. Therefore, we can take various benefits to reach a fair-efficient consensus under the edge-assisted distributed computing system environment. Finally, experimental results demonstrate the effectiveness of our bargaining based computation offloading scheme by comparing with the existing state-of-the-art distributed computing protocols; we can accelerate training process thanks to our efficient bargaining approach.

INDEX TERMS Distributed computing, edge offloading computing, cooperative game theory, loss aversion bargaining solutions, utilitarian bargaining solution.

I. INTRODUCTION

Recently, Internet of Things (IoT) devices are widely used in industrial control, network physical system, medical devices, environmental monitoring and other fields. Smart IoT devices will be deployed everywhere in future networks to meet the growing demand for services such as smart cities, smart homes and smart medical systems. Connected IoT devices can potentially provide insights that lead beyond-5G systems to cost reductions, efficiency gains, and new business opportunities. With massive usage of smart IoT devices, machine learning and artificial intelligence techniques target not only communication and networking tasks but also augmented environmental perception services. This combination is expected to provide advanced services while paving the way to the Internet of Conscious Things [1]–[3].

The rapid advancement of IoT devices and social networking applications results in an exponential growth of the generated data. However, these devices require low latency and power consumption to perform application tasks. Therefore, computation intensive and delay-sensitive applications are hardly executed on resource-constrained mobile IoT devices. To tackle this challenge, distributed computing has been proposed as a promising solution to alleviate the computing burdens of IoT devices and reduce service delay. It can leverage the computing capabilities of IoT devices via computation offloading at anytime and anywhere. In the distributed computing infrastructure, local data storing and processing with global coordination may be possible by the emerging technology of mobile edge computing (MEC). It extends the capability of cloud computing, and edge clouds

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

are equipped with storage and computation in the edge of the network. Multiple edge clouds work together to perform large-scale distributed tasks that involve both local processing and remote coordination [2]–[4].

With edge computing, IoT devices can offload their computation tasks to edge clouds. As a natural consequence, they can save more energy and still maintain the quality of the services they should provide. However, it has shown that there is a trade-off between efficiency and privacy. For the computation offloading service, user-sensitive data are required to offload to the edge cloud, it may impose great privacy leakage risk. Therefore, all of these factors should be rationally considered in the distributed MEC paradigm [1], [4], [5].

In this study, our major goal is to design a new distributed edge computing scheme. The major challenge of our proposed scheme is to coordinate the different individual agents while ensuring good global properties. However, it is a complex and difficult work under dynamically changing MEC system environments. Therefore, we need a new intelligent control paradigm and novel solution concept. In our scheme, autonomous, distributed, and intelligent IoT devices coordinately make rational and strategic decisions to deploy deep learning algorithms. This scenario may fall into cooperative game theory. Cooperative game theory offers an effective model of cooperation between rational game players. The critical issue of cooperative games is how to distribute surplus outcome among all players. Various solutions in cooperative games embody different criteria. Currently, they are widely used in economics, computer science, wireless telecommunications, political science, and so on [6].

A. TECHNICAL CONCEPTS

Game theory studies mathematical models of negotiation, conflict and cooperation between strategic actors. It has been applied in various areas of study to understand why a game player makes a particular decision and how the decisions made by one player affect others. As a type of game theory, a bargaining game is a cooperative game model in which groups of players coordinate their actions and pool their winnings. Conceptually, bargaining is the opposite of idealized 'perfect competition' among players. In bargaining games, strategic actors wish to enter into contracts to generate a surplus, which must be divided among the players. Therefore, a solution to the cooperative bargaining game enables players to determine their payoffs fairly and optimally to make joint agreements. First, a solution for the bargaining game was introduced in a fundamental paper by J. Nash in 1950. Based on axiomatic theory, Nash derived an idealized representation of the bargaining problem and developed a methodology to resolve this problem. Since then, various bargaining solutions have been proposed based on slightly different assumptions about the properties desired for the final agreement point. In 1975, E. Kalai and M. Smorodinsky introduced another solution, known as the Kalai and Smorodinsky bargaining solution. To date, these two bargaining solutions have been regarded as standard solutions to bargaining problems. Additionally, utilitarian bargaining solution has been introduced. In the point view of efficiency, this solution maximizes only the sum of effectiveness without fairness concerns [15].

Recently, much research in bargaining problems has dealt with the concept of risk aversion. From a realistic point of view, game players tend to attach greater importance to losses than to the corresponding gains. Therefore, they want to minimize loss more than maximize gain, and the player's risk aversive manner tends to play an important role in how that player behaves. Nowadays, much attention has been paid to the influence of risk on the outcomes assigned by specific bargaining solutions [16], [17]. In 2002, J. Shalev dealt with the connection between risk aversion and the Nash solution, and proposed a new solution, called the *loss aversive Nash bargaining solution* (*LNBS*). In the *LNBS*, the reference points are endogenized to incorporate loss aversion into the bargaining model; increasing loss aversion for a player leads to worse outcomes for that player in bargaining games. To find a unique solution, the psychological element plays an important role in *LNBS* [16]. In 2011, B. Driesen et al proposed another loss aversion bargaining idea, called the *loss aversive Kalai-Smorodinsky bargaining solution* (*LKSBS*). The original Kalai and Smorodinsky solution is an intersection point between the bargaining set and the line connecting that outcome to the disagreement and utopia points. However, the *LKSBS* is a Pareto optimal outcome while satisfying a given proportion which depends on the players' loss aversion coefficients [17].

B. MAIN CONTRIBUTIONS

According to the main concept of *LNBS* and *LKSBS*, we develop a novel edge cloud computing scheme for IoT devices. In the distributed MEC paradigm, each individual device adaptively decides the amount of offload computing. By considering the current condition of its own device, this decision is made based the utilitarianism. To effectively coordinate multiple devices, each edge cloud dynamically allocates its computation and communication resources. By considering the heterogeneity of devices, these limited resources should be shared fair-efficiently. To implement our scheme, the ideas of *LNBS* and *LKSBS* are adopted, and they work together toward an appropriately-balanced system performance. Our joint bargaining approach provides the most proper combination of different solutions while ensuring good global properties. In detail, the major contributions of this study are as follows:

- This study considers the computation offloading problem in the distributed MEC platform. During the interactive cooperative game process, the control decisions for computation offloading and resource allocations are made in an effective online fashion based on the ideas of different bargaining solutions.

- Each individual IoT device partially offloads its computation task to the corresponding edge cloud. Based on main characteristics of utilitarian bargaining solution, the offloading amount is dynamically decided; this decision reflects the current condition of IoT device.
- Each edge cloud allocates its computation and communication resources for its corresponding devices. The *LNBS* is used to distribute the computation resource, and the *LKSBS* is adopted to allocate the communication resource.
- Under a joint-bargaining procedure, we explore the interaction of utilitarian solution, *LNBS* and *LKSBS* while leveraging the synergistic features. The main novelty of our approach lies in the reciprocal combination of different bargaining solutions.
- We evaluate the performance of our proposed scheme via extensive experiments in a simulated environment. Our experimental results reveal that the proposed joint-bargaining approach can achieve a higher system performance compared with the existing MEC based offloading protocols.

C. ORGANIZATION

The rest of this paper is organized as follows. Section II presents related work of the edge computing platform. Section III describes a general control framework in the MEC infrastructure. And, we explain the basic ideas of different bargaining solutions such as *LNBS* and *LKSBS*. To implement our proposed scheme, we formulate our joint-bargaining game model, and the main steps of our proposed algorithm are given based on the interactive bargaining procedure. In Section IV, we show performance evaluations and numeric results in comparison with the existing offloading methods. Finally, Section V concludes our work, and draws some future investigations.

II. RELATED WORK

The unprecedented amount of data necessitates the use of distributed computational framework to provide solutions for various MEC applications. Using distributed optimization techniques, researchers have recently proposed a few algorithms for federated learning based edge computing process. They address the challenges of resource management involving collaborations of a number of IoT devices. The paper [19] presents a reinforcement learning based mobile offloading scheme for edge computing against jamming attacks and interference, which uses safe reinforcement learning to avoid choosing the risky offloading policy that fails to meet the computational latency requirements of the tasks. It has provided the computational complexity of the proposed scheme and its performance bound containing the computational latency of the tasks, the energy consumption and the utility of the mobile device based on game theory [19].

In [1], the *Edge Computing supported Federated Learning* (*ECFL*) scheme uses multiple agents to indicate the decisions of the IoT devices. With the aim of making decisions feasible and further reducing the transmission costs between the smart devices and edge clouds, learning algorithm is used to train agents in a distributed fashion. Through the joint allocation of communication and computing resources, the *ECFL* scheme supports the edge computing and learning algorithm in the IoT environment. Based on the deployed multiple edge clouds, this approach is to implement the learning algorithm based decision mechanism in the dynamic IoT system; it is the main contribution of the *ECFL* scheme. Finally, the effectiveness of the *ECFL* scheme is verified by the experimental results in terms of accuracy and efficiency [1].

S. Wang *et al.* propose the *Resource Constrained Federated Learning* (*RCFL*) scheme that determines the best tradeoff between local update and global parameter aggregation under a limited resource constraint [4]. The main goal of this scheme is to address the problem of how to efficiently utilize the limited computation and communication resources at the edge cloud. Based on the typical edge cloud computing architecture, the raw data is collected and stored at multiple edge clouds, and a learning model is trained in a distributed fashion. From a theoretical perspective, the *RCFL* scheme analyzes the convergence bound of gradient-descent based learning approach. According to the theoretical convergence bound, this method learns the data distribution, system dynamics, and model characteristics for the optimal learning performance. It dynamically adapts the frequency of global aggregation in real time to minimize the learning loss. Finally, they estimate the performance through extensive experiments, and confirm the near-optimal outcome for different system configurations [4].

The paper [5] provides the *Cloud based Personalized Federated Learning* (*CPFL*) scheme to exploit a massive amount of user-generated data samples on IoT devices. This scheme advocates a personalized learning model in the edge cloud architecture for intelligent IoT applications. To cope with the heterogeneity issues in IoT environments, the *CPFL* scheme investigates an emerging personalized learning method which is able to mitigate the negative effects caused by heterogeneities in the complex IoT environments. To tackle the high communication and computation cost issues in device heterogeneity, each IoT device can offload its computationally intensive application task to the edge cloud. This approach fulfills the requirement for fast-processing capacity and low latency. In addition, the *CPFL* scheme also enables that IoT devices and edge clouds jointly train a global model in a coordinative paradigm. Under a limited resource constraint, the performance of this approach is verified by a simulation analysis [5].

Although some researches have exploited extensively the edge computing paradigm, an efficient cooperation of edge clouds and IoT devices has not been fully investigated. Different from the existing *ECFL*, *RCFL* and *CPFL* protocols [1], [4], [5], clouds and IoT devices in our proposed scheme make rational decisions in a cooperative manner, and

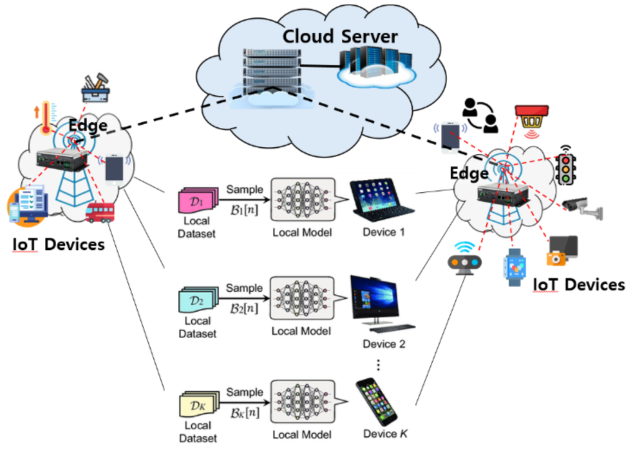

FIGURE 1. Edge-assisted computing platform for offloading services.

effectively share the limited system resource while ensuing mutual advantages; it has more potential benefits for the computation offloading process.

III. THE PROPOSED COMPUTATION OFFLOADING ALGORITHM

In this section, a short description of the MEC system infrastructure is presented with a review of *LNBS* and *LKSBS*. Based on the interactive cooperative game approach, we elaborate the main challenges of computation offloading process while discussing relevant control issues in MEC-assisted IoT environments. Finally, the main step procedures of our proposed algorithm are delineated to help readers understand better.

A. EDGE CLOUD SYSTEM INFRASTRUCTURE FOR FEDERATED LEARNING

In this study, a system model in IoT environment with edge clouds is taken. In this model, IoT devices have the computation capability. However, portable IoT devices have restricted computation capacities along with little onboard batteries. To overcome the device's computing power limitation, computation tasks can be partly offloaded to the nearby edge cloud. In the MEC infrastructure, edge cloud is a small-scale cloud datacenter that extends the capability of cloud computing by bringing it to the edge of the network. The main purpose of edge cloud is to support interactively computation-intensive tasks by providing offloading services. It is a new architectural element that enlarges today's cloud computing services in the 5G IoT environment. Fig.1 illustrates the edge-assisted computing infrastructure for offloading services.

Based on the edge cloud system model, $\mathcal{E} = \{E_1, \ldots, E_n\}$ is the set of edge clouds, and $\mathfrak{D} = \{D_1, \ldots, D_m\}$ is the set of IoT devices. One edge cloud out of $\mathcal{E}$ can be chosen by nearby devices for offloading computation tasks. The $E_{1 \le k \le n} \in \mathcal{E}$ has its computation power Γ_{E_k} with wireless spectrum bandwidth $\mathcal{W}_{E_k}$. One IoT device $D_{1 \le i \le m} \in \mathfrak{D}$ has its own battery cell and computation power. The computation power of D_i is Υ_{D_i}, and the battery cell stores electrical energy; the full-energy and remaining-energy of D_i are represented by $\mathfrak{E}_{D_i}$ and $e^c_{D_i}$, respectively. $\mathcal{T}_{D_i} = \left[\mathfrak{N}_{D_i}, \kappa_{D_i}, \mathcal{M}^{\mathcal{T}}_{D_i}, \mathcal{L}_{D_i}\right]$ is the learning task of D_i where $\mathfrak{N}_{D_i}$ is the data bit size of $\mathcal{T}_{D_i}$, and κ_{D_i} represents the number of CPU cycles required to process one-bit data input. $\mathcal{M}^{\mathcal{T}}_{D_i}$ is the maximum delay-latency to complete the $\mathcal{T}_{D_i}$, and $\mathcal{L}_{D_i}$ is the ratio of data offloading for the $\mathfrak{N}_{D_i}$. During the computation offloading process, time horizon is discretized into time epochs indexed by t_c with equivalent duration in seconds [1].

From the viewpoint of devices, how to leverage the trade-offs between local and offloading computing is a critical issue. Therefore, the D_i decides the value of $\mathcal{L}_{D_i}$, which can be viewed as a control parameter between 0 and 1. To effectively complete the learning process, the $\mathcal{L}_{D_i}$ value should be dynamically adjusted while coordinating its offloading and local computation amounts. In the point view of edge clouds, they are responsible for offloading services. Therefore, they adaptively provide their communication ($\mathcal{W}$) and computation (Γ) resources to successfully complete the requested offloading devices. Based on the MEC infrastructure, cloud edges and devices make control decisions intelligently in pursuit of their individual objectives.

In this study, the interaction process of cloud edges and devices is formulated as a joint bargaining game ($\mathbb{G}$) in a coordination manner; $\mathbb{G}$ is subdivided into $\mathbb{G}^{\mathcal{L}}_D$, $\mathbb{G}^{\Gamma}_E$, and $\mathbb{G}^{\mathcal{W}}_E$ to solve different bargaining problems. The edge computing system operates in a slotted time structure and task offloading services are implemented at each time period. Formally, we define game entities, i.e., $\mathbb{G} = \{\mathbb{G}^{\mathcal{L}}_D, \mathbb{G}^{\Gamma}_E, \mathbb{G}^{\mathcal{W}}_E\} = \{\{E_k \in \mathcal{E}, D_i \in \mathfrak{D}\}, \{\mathbb{G}^{\mathcal{L}}_D \,|\, \mathcal{T}^O_D, \mathcal{T}^L_D, \mathcal{L}_D, U^L_D(\cdot), U^O_D(\cdot)\}, \{\mathbb{G}^{\Gamma}_E \,|\, \Gamma_E, \Gamma^D_E, U^{\Gamma}_D(\cdot)\}, \{\mathbb{G}^{\mathcal{W}}_E \,|\, \mathcal{W}_E, \mathcal{W}^D_E, U^{\mathcal{W}}_D(\cdot)\}, T\}$ of gameplay.

- $\mathbb{G}^{\mathcal{L}}_D$, $\mathbb{G}^{\Gamma}_E$ and $\mathbb{G}^{\mathcal{W}}_E$ are bargaining games for the $\mathcal{L}$ value decision by D, the Γ and $\mathcal{W}$ resource distributions by E, respectively. They are mutually and reciprocally interdependent in an interactive manner.
- $\mathcal{E}, \mathfrak{D}$ are the sets of cloud edges and devices; they are game entities for the $\mathbb{G}^{\mathcal{L}}_D$, $\mathbb{G}^{\Gamma}_E$ and $\mathbb{G}^{\mathcal{W}}_E$ bargaining games.
- In the $\mathbb{G}^{\mathcal{L}}_D$, $\mathcal{T}^O_D$ and $\mathcal{T}^L_D$ represent the D's computation offloading part and local computing part, respectively. They are game players, and $\mathcal{L}_D$ and $(1 - \mathcal{L}_D)$ are their strategies. $U^L_D(\cdot)$ and $U^O_D(\cdot)$ are utility functions of $\mathcal{T}^O_D$ and $\mathcal{T}^L_D$.
- In the $\mathbb{G}^{\Gamma}_E$, Γ_E is the computation power of E, and devicescontacting to the E are game players. The Γ^D_E represents the assigned computing power for the D; it is the D's strategy and $U^{\Gamma}_D(\cdot)$ is the D's utility function for the Γ_E resource allocation problem.
- In the $\mathbb{G}^{\mathcal{W}}_E$, $\mathcal{W}_E$ is the wireless bandwidth of E, and devices contacting to the E are game players. The $\mathcal{W}^D_E$ represents the assigned bandwidth amount for the D; it is the D's strategy and $U^{\mathcal{W}}_D(\cdot)$ is the D's utility function for the $\mathcal{W}_E$ resource allocation problem.

- The limited Γ_E^D and $\mathcal{W}_E$ resources are rationally distributed according to the *LNBS* and *LKSBS*, respectively.
- $T = \{t_1, \ldots, t_c, t_{c+1}, \ldots\}$ denotes time, which is represented by a sequence of time steps.

B. THE BASIC CONCEPTS AND FUNDAMENTAL IDEAS OF LNBS AND LKSBS

To characterize the basic concepts of bargaining solutions, we first define mathematical expressions. The set of bargaining game players is denoted by $N = \{1, \ldots, n\}$ with $n \geq 2$. For $x, y \in \mathbb{R}^N$ we write $x \geq y$ if $x_i \geq y_i$ for all $i \in N$, and $x > y$ if $x_i > y_i$ for all $i \in N$. The utility function of player $i \in N$ ($u_i : S \to \mathbb{R}$) can be represented as the degree of satisfaction received by player i; S is the set of feasible outcomes, which are translated into real numbers. Usually, bargaining games $(S, d) \in \mathcal{B}^N$ consist of S and disagreement, d. If the bargaining players fail to reach some other outcome $x = \{x_1, \ldots, x_n\} \in S$, d results; S contains a vector $x > d$ for all $x \in \mathcal{B}^N$. An n-player bargaining problem is a set $S \subset \mathbb{R}^N$ with $S \subset d + \mathbb{R}_+^N$ where $\mathbb{R}_+^N = \{x \in \mathbb{R}^N | x \geq 0\}$. Players seek agreement on an outcome $x = \{x_1, \ldots, x_n\}$ in S, yielding an utility $x_{1 \leq i \leq n}$ to player $i \in N$. The utopia point of (S, d) is the highest possible utility payoff for all players; it is simply defined by the vector $\mathcal{I}(S, d) = (\mathcal{I}_1(S, d), \ldots, \mathcal{I}_n(S, d))$ where $\mathcal{I}_i(S, d) = \mathbf{max}\{x_i \,|\, x \in (S, d)\}$ [16], [17].

In the *LNBS* and *LKSBS*, the outcomes of players' utility functions are evaluated with respect to a reference point and loss aversion tendency. To capture the loss aversion aspect of a player's preference, its utility function has a constant level of loss aversion, called the player's loss aversion coefficient (λ). A higher coefficient value indicates a higher level of loss aversion. Finally, the player i's utility function with reference point Θ_i from an outcome x_i, i.e., $U_i(x_i, \Theta_i)$, is formally defined as follows [16];

$$U_i(x_i, \Theta_i) = \begin{cases} u_i(x_i), & \text{if } u_i(x_i) \geq \Theta_i \\ u_i(x_i) - (\lambda_i \times (\Theta_i - u_i(x_i))), & \text{otherwise} \end{cases}$$
$$\text{s.t., } \Theta_i \in \mathbb{R}, \quad x_i \in \mathcal{B}^N \text{ and } \lambda_i \in \mathbb{R}_+ \tag{1}$$

where λ_i is the non-negative loss aversion coefficient of player i, and Θ_i is the expected utility to depict the player i's reference level. When the player i obtains a payoff x_i, which is below his reference outcome Θ_i, the player i experiences a disutility that is equal to his loss $(\Theta_i - u_i(x_i))$, multiplied by the λ_i. If the player i's payoff is above than the Θ_i, it is remains unchanged [16], [17].

To obtain the *LNBS*, we can define a unique solution function $\varphi^{LNBS}(\cdot) : \mathcal{B}^N \to \mathbb{R}^N$ based on the $U_{1 \leq i \leq n}(x_i, \Theta_i)$. It assigns to each bargaining problem $(S, d) \in \mathcal{B}^N$ a single point $\varphi^{LNBS}(\cdot) \in S$ with $\lambda_{1 \leq i \leq n}$ and $\Theta_{1 \leq i \leq n}$ [11];

$$\varphi^{LNBS}\left(\mathcal{B}^N, d, N, \Lambda, \mathcal{R}\right) = \mathop{\mathbf{max}}_{\substack{x \in \mathcal{B}^N, \\ \lambda_i \in \Lambda, \\ \theta_i \in \mathcal{R}}} \prod_{i \in N} \left(U_i(x_i, \Theta_i) - U_i(d_i, \Theta_i)\right)$$
$$\text{s.t., } \Lambda = (\lambda_1, \ldots, \lambda_n) \in \mathbb{R}_+^N \quad \text{and} \quad \mathcal{R} = (\Theta_1, \ldots, \Theta_n) \in \mathbb{R}^N \tag{2}$$

The *LKSBS* is a map $\varphi^{LKSBS}(\cdot) : \mathcal{B}^N \to \mathbb{R}^N$ based on the $U_{1 \leq i \leq n}(x_i)$, which also implies that a single point $\varphi^{LKSBS}(\cdot)$ is obtained in the bargaining problem $(S, d) \in \mathcal{B}^N$ for all $\Lambda \in \mathbb{R}_+^{\mathrm{N}}$ and $\mathcal{R} \in \mathbb{R}^N$. Finally, $\varphi^{LKSBS}(\cdot)$ is defined as follows [17];

$$\varphi^{LKSBS}(\mathcal{I}(S, d), N, \Lambda) = \left\{ x = \{x_1 \ldots x_n\} \in P(S) \,|\, \frac{(\mathcal{I}_n(S, d) - U_n(x_n))}{(U_n(x_n) - d_n)} = \left(\mathfrak{T}_k \times \frac{(\mathcal{I}_k(S, d) - U_k(x_k))}{(U_k(x_k) - d_k)}\right) \right\}$$
$$\text{s.t., } \begin{cases} x \geq d, \quad \tilde{N} = N \backslash \{n\}, \; k \in \tilde{N} \text{ and } \Gamma \in \mathbb{R}_{++}^{\tilde{N}} \\ \mathfrak{T}_k \in \mathfrak{T} = \left\{ \dfrac{1 + \lambda_n}{1 + \lambda_1}, \dfrac{1 + \lambda_n}{1 + \lambda_2}, \ldots \dfrac{1 + \lambda_n}{1 + \lambda_{n-1}} \right\} \\ P(S) = \left\{ x \in S \,|\, \text{forall } y \in \mathbb{R}^N, \text{ if } y \geq x \text{ and } x \neq y, \text{ then } y \notin S \right\} \end{cases} \tag{3}$$

The axioms involved in characterizing bargaining solutions are defined as follows. The *LNBS* is characterized by a collection of desirable axioms ***PO***, ***S***, ***I***, ***IIA***, and ***RI***, and the *LKSBS* satisfies the axioms ***PO***, ***SI***, ***IM***, ***SIR*** and ***PCI*** [16], [17].

- *Pareto Optimality* (***PO***)**:** The solution is not weakly dominated by any point in $\mathcal{B}^N$ except itself.
- *Symmetry* (***S***)**:** If $\mathcal{B}^N$, d, Λ and $\mathcal{R}$ are symmetrical in the plane, the solution assigns the same outcome to each player.
- *Invariance* (***I***)**:** The solution is invariant with respect to the positive linear transformations of $\mathcal{B}^N$, d and $\mathcal{R}$.
- *Independence of Irrelevant Alternatives* (***IIA***)**:** If the solution of $\left(\mathcal{B}^N, d, N, \Lambda, \mathcal{R}\right)$ is x^*, $X' \subseteq \mathcal{B}^N$ and $x^* \in X'$, then the solution of $\left(X', d, N, \Lambda, \mathcal{R}\right)$ is also x^*.
- *Representation Invariance* (***RI***)**:** If two elements of X give the same set of utility pairs, then the evaluations of the solution points of the two problems give the same utilities to the players.
- *Scale Invariance* (***SI***)**:** $\mathfrak{P} : \mathbb{R}^N \to \mathbb{R}^N$ is a linear transformation $\mathfrak{P}(x) := \alpha + (\beta \cdot x)$, where $\alpha \in \mathbb{R}^N$, $\beta \in \mathbb{R}_{++}^N$, and $\mathfrak{P}(S) := \alpha + (\beta \cdot S)$ for $S \subseteq \mathbb{R}^N$. If $\mathfrak{P}(\varphi(S, d)) = \varphi(\mathfrak{P}(S), \mathfrak{P}(d))$, then $\varphi : \mathcal{B}^N \to \mathbb{R}^N$ satisfies ***SI***.
- *Individual Monotonicity* (***IM***)**:** If $\varphi_i(S, d) \leq \varphi_i(T, d)$ for all $(S, d), (T, d) \in \mathcal{B}^N$ and $i \in N$ with $S \subseteq T$ and $u_j(S) = u_j(T)$ for all $j \in N \backslash \{i\}$, then $\varphi : \mathcal{B}^N \to \mathbb{R}^N$ satisfies ***IM***.
- *Strong Individual Rationality* (***SIR***)**:** If $\varphi(S, d) > d$ for all $(S, d) \in \mathcal{B}^N$, then $\varphi : \mathcal{B}^N \to \mathbb{R}^N$ satisfies ***SIR***.
- *Proportional Concession Invariance* (***PCI***)**:** Define a bargaining problem $(\hat{S}, d)$ with $\hat{S} := \left\{x \in \mathrm{S} \,|\, x \leq \hat{u}\right\}$, where $\hat{u} = (\alpha \times \varphi(S, d)) + ((1 - \alpha) \times u(S, d))$ for

some $\alpha \in [0, 1]$. If $\varphi\left(\hat{S}, d\right) = \varphi(S, d)$, $\varphi{:}\mathcal{B}^N \rightarrow \mathbb{R}^N$ satisfies ***PCI***.

C. THE COOPERATIVE CONTROL SCHEME FOR FEDERATED LEARNING PROCESS

In the IoT devices, energy consumption rate and task's delay-constraint are major control factors to decide the $\mathcal{L}_D$ value. In addition, personal data in $\mathfrak{N}_D$ usually contain sensitive information with individual privacy. Therefore, a novel computation offloading mechanism is necessary to complete computing tasks while considering all current conditions; it is quite beneficial to maximize the learning system performance. For the task $\mathcal{T}_{D_i}$ in D_i, total computation work is $\left(\mathfrak{N}_{D_i} \times \kappa_{D_i}\right)$ CPU cycles, and it is partially offloaded to the edge cloud according to the $0 \leq \mathcal{L}_{D_i} \leq 1$; $\mathcal{T}^O_{D_i} = \left(\mathfrak{N}_{D_i} \times \kappa_{D_i} \times \mathcal{L}_{D_i}\right)$ indicates the computation offloading part, and $\mathcal{T}^L_{D_i} = \left(\mathfrak{N}_{D_i} \times \kappa_{D_i} \times \left(1 - \mathcal{L}_{D_i}\right)\right)$ is the local computing part. By considering the current situation of D_i and corresponding edge cloud, the $\mathcal{L}_{D_i}$ decision problem is formulated as a bargaining game model $\left(\mathbb{G}^{\mathcal{L}}_{D_i}\right)$ [12]–[14].

In the game $\mathbb{G}^{\mathcal{L}}_{D_i}$, $\mathcal{T}^O_{D_i}$ and $\mathcal{T}^L_{D_i}$ are assumed as game players, and $\mathcal{L}_{D_i}$, $\left(1 - \mathcal{L}_{D_i}\right)$ are their strategies. To reduce computation complexity, strategies are specified in terms of basic control unit (Δ). $\mathcal{T}^O_{D_i}$ and $\mathcal{T}^L_{D_i}$ players have their own utility functions $U^O_{D_i}(\cdot)$ and $U^L_{D_i}(\cdot)$; each function maps the player's satisfaction to a real number, which represents the resulting outcome in the game $\mathbb{G}^{\mathcal{L}}_{D_i}$. When the E_k is D_i's corresponding edge, $U^O_{D_i}(\cdot)$ and $U^L_{D_i}(\cdot)$ are defined as follows;

$$
\begin{cases}
U^L_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right) \\
\quad = \left(\dfrac{e^c_{D_i}}{\mathfrak{E}_{D_i}} \times log\left(\alpha + \dfrac{\mathcal{T}^L_{D_i}}{\mathcal{X}_{D_i}}\right) \times \mathcal{G}^c_{D_i}\left(\Upsilon^c_{D_i}, \mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right)\right) \\
U^O_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right) \\
\quad = \left(\dfrac{\left(\Gamma_{E_k} - \Phi_{E_k}\right)}{\Gamma_{E_k}} \times log\left(\psi + \dfrac{\mathcal{T}^O_{D_i}}{\mathcal{X}_{D_i}}\right) \times \mathcal{F}^p_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right)\right) \\
\mathcal{F}^p_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right) = \gamma - exp\left(\dfrac{\mathcal{T}^O_{D_i}}{\mathcal{X}_{D_i}}\right) \\
\quad \text{and } \mathcal{G}^c_{D_i}\left(\Upsilon^c_{D_i}, \mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right) = \mu - exp\left(1 - \dfrac{\Upsilon^c_{D_i}}{\Upsilon_{D_i}}\right)
\end{cases}
$$

$$
\text{s.t., } \begin{cases}
\mathcal{T}^L_{D_i} = \left(\mathfrak{N}_{D_i} \times \kappa_{D_i} \times \left(1 - \mathcal{L}_{D_i}\right)\right) \\
\mathcal{T}^O_{D_i} = \left(\mathfrak{N}_{D_i} \times \kappa_{D_i} \times \mathcal{L}_{D_i}\right) \text{ and } \mathcal{X}_{D_i} = \left(\mathfrak{N}_{D_i} \times \kappa_{D_i}\right)
\end{cases} \tag{4}
$$

where α, ψ, γ, μ are control factors for $U^O_{D_i}(\cdot)$ and $U^L_{D_i}(\cdot)$, and Φ_{E_k} is the currently used E_k's computing power. $\mathcal{G}^c_{D_i}(\cdot)$ is a function to represent the D_i's computation overhead, and $\mathcal{F}^p_{D_i}(\cdot)$ is a function to indicate the D_i's privacy sensitivity. $\mathbb{H}_{E_k}$ is the set of devices contacting to the E_k, and $\Upsilon^c_{D_i}$, $e^c_{D_i}$ are the currently available Υ_{D_i} and $\mathfrak{E}_{D_i}$ in the D_i. In the game $\mathbb{G}^{\mathcal{L}}_{D_i}$, game players, i.e., $\mathcal{T}^O_{D_i}$ and $\mathcal{T}^L_{D_i}$, work together for the D_i, itself. Therefore, bargaining solution for the $\mathbb{G}^{\mathcal{L}}_{D_i}$ may not consider the fairness issue among game players; only the sum of players' payoffs should be maximized in the viewpoint of efficiency. In this case, the utilitarian bargaining solution for the D_i $\left(UBS_{D_i}\right)$ is most appropriate [18]. This solution can be formulated as the following:

$$
UBS_{D_i} = \arg \max_{0 \leq \mathcal{L}_{D_i} \leq 1} \sum \left(U^L_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right) + U^O_{D_i}\left(\mathcal{T}_{D_i}, \mathcal{L}_{D_i}\right)\right) \tag{5}
$$

According to (5), the D_i decides the $\mathcal{L}_{D_i}$ value to maximize the sum of $U^O_{D_i}(\cdot)$ and $U^L_{D_i}(\cdot)$. And then, the D_i's corresponding edge $E_k \in \mathcal{E}$ distributes its computation $\left(\Gamma_{E_k}\right)$ and communication $\left(\mathcal{W}_{E_k}\right)$ resources to individual devices in the $\mathbb{H}_{E_k}$. In this study, the Γ_{E_k} and $\mathcal{W}_{E_k}$ resource distribution problems are formulated as bargaining games, i.e., $\mathbb{G}^{\Gamma}_{E_k}$ and $\mathbb{G}^{\mathcal{W}}_{E_k}$. In these games, devices in the set $\mathbb{H}_{E_k}$ are game player and the allocated Γ_{E_k} and $\mathcal{W}_{E_k}$ resources for the D_i are strategies in the $\mathbb{G}^{\Gamma}_{E_k}$ and $\mathbb{G}^{\mathcal{W}}_{E_k}$. In the $\mathbb{G}^{\Gamma}_{E_k}$, the utility function for the D_i, i.e., $U^{\Gamma}_{D_i}(\cdot)$, is defined based on some factors; E_k's assigned computation power for the D_i, i.e., $\Gamma^{D_i}_{E_k}$, the currently used E_k's computing power, i.e., Φ_{E_k}, and D_i's loss aversion propensity, i.e., λ_{Γ,D_i}. Under the heavy computing overhead of E_k, the loss aversion tendency of the D_i is alleviated. However, if it was vice versa, i.e., the Φ_{E_k} is light, it means that the E_k has enough available computing power to support offloading services. In this case, the tendency toward loss aversion increases. Therefore, based on the current E_k conditions, we dynamically modify the loss aversion propensity λ_{Γ,D_i}. According to the $\Gamma^{D_i}_{E_k}$, Φ_{E_k} and Θ_{Γ,D_i} values, the D_i's utility function, i.e., $U^{\Gamma}_{D_i}(\cdot)$, can be derived as follows.

$$
U^{\Gamma}_{D_i}\left(\Gamma_{E_k}, \Gamma^{D_i}_{E_k}, \mathcal{T}^O_{D_i}, \Phi_{E_k}, \Theta_{\Gamma,D_i}\right)
$$

$$
= \begin{cases}
u^{\Gamma}_{D_i}\left(\Gamma^{D_i}_{E_k}\right) = \left(\dfrac{T_{D_i}}{\eta + exp\left(-\dfrac{\Gamma^{D_i}_{E_k}}{\Gamma_{E_k}}\right)}\right), \\
\quad \text{if } u^{\Gamma}_{D_i}\left(\Gamma^{D_i}_{E_k}\right) \geq \Theta_{\Gamma,D_i} \\
u^{\Gamma}_{D_i}\left(\Gamma^{D_i}_{E_k}\right) - \left(\lambda_{\Gamma,D_i} \times \left(\Theta_{\Gamma,D_i} - u^{\Gamma}_{D_i}\left(\Gamma^{D_i}_{E_k}\right)\right)\right), \\
\quad \text{otherwise}
\end{cases}
$$

$$
\text{s.t., } \begin{cases}
T_{D_i} = \left(\zeta + \dfrac{\left(\Omega_O \times \mathcal{T}^O_{D_i}\right) + \left(\Omega_L \times \mathcal{T}^L_{D_i}\right)}{\mathcal{M}^{\mathcal{T}}_{D_i}}\right) \\
\quad \text{and } D_i \in \mathbb{H}_{E_k} \\
\Theta_{\Gamma,D_i} = \left(u^{\Gamma}_{D_i}\left(\mathcal{T}^O_{D_i}\right) \times \rho_{D_i}\right) \\
\quad \text{and } \lambda_{\Gamma,D_i} = \left(\beta \times \dfrac{\left(\Gamma_{E_k} - \Phi_{E_k}\right)}{\Gamma_{E_k}}\right)
\end{cases} \tag{6}
$$

where η, ζ are control parameters for $U^{\Gamma}_{D}(\cdot)$, and Ω_O, Ω_L are time delay parameters for offloading and local computing processes, respectively. ρ_{D_i}, β are the system adjustment factors for the loss aversion tendency. Especially, during the operation of offloading computation services, unexpected growth of offloading requests may develop in a specific edge cloud. Under this system overload condition, the ***S***, ***I*** and ***RI*** axioms are desirable to ensure a relative fairness. Therefore, the idea of *LNBS* is applied for the game $\mathbb{G}_{E_k}^{\Gamma}$. It is given by:

$$\varphi^{LNBS}\left(D_i \in \mathbb{H}_{E_k} \mid U_{D_i}^{\Gamma}(\cdot), \left(\Lambda_{E_k}, \Theta_{\Gamma,D_i}\right)\right) = \max_{\Gamma_{E_k}^{D_i}} \prod_{D_i \in \mathbb{H}_{E_k}} \left(U_{D_i}^{\Gamma}(\cdot) - U_{D_i}^{d}(\cdot)\right)$$
$$\text{s.t., } \Lambda_{E_k} = \left(\ldots \lambda_{\Gamma,D_i} \ldots\right) \in \mathbb{R}_{+}^{\left|\mathbb{H}_{E_k}\right|}, \left(\ldots \Theta_{\Gamma,D_i} \ldots\right) \in \mathbb{R}^{\left|\mathbb{H}_{E_k}\right|} \text{ and } \sum_{D_i \in \mathbb{H}_{E_k}} \Gamma_{E_k}^{D_i} \leq \Gamma_{E_k} \tag{7}$$

where $U_{D_i}^{d}(\cdot)$ is the disagreement point of D_i. According to (7), the $\Gamma_{E_k}^{D_i}$ value is adaptively decided in the $\mathbb{G}_{E_k}^{\Gamma}$ game. As the same manner as the $U_{D_i}^{\Gamma}(\cdot)$, the D_i's utility function in the spectrum allocation process, i.e., $U_{D_i}^{\mathcal{W}}(\cdot)$, is decided based on the $\mathcal{W}_{E_k}^{D_i}$ and $\Theta_{\mathcal{W},D_i}$; it can be derived as follows.

$$U_{D_i}^{\mathcal{W}}\left(\mathcal{W}_{E_k}, \mathcal{W}_{E_k}^{D_i}, \mathfrak{N}_{D_i}, \mathcal{L}_{D_i}, \Theta_{\mathcal{W},D_i}, \varpi_{E_k}\right) = \begin{cases} u_{D_i}^{\mathcal{W}}\left(\mathcal{W}_{E_k}^{D_i}\right) = \left(T_{D_i} \times log\left(\sigma + \dfrac{\mathcal{W}_{E_k}^{D_i}}{\mathcal{W}_{E_k}}\right)\right), & \text{if } u_{D_i}^{\mathcal{W}} \geq \Theta_{\mathcal{W},D_i} \\ u_{D_i}^{\mathcal{W}}\left(\mathcal{W}_{E_k}^{D_i}\right) - \left(\lambda_{\mathcal{W},D_i} \times \left(\Theta_{\mathcal{W},D_i} - u_{D_i}^{\mathcal{W}}\left(\mathcal{W}_{E_k}^{D_i}\right)\right)\right), & \text{otherwise} \end{cases}$$
$$\text{s.t., } \Theta_{\mathcal{W},D_i} = \left(u_{D_i}^{\mathcal{W}}\left(\mathfrak{N}_{D_i} \times \mathcal{L}_{D_i}\right) \times \rho_{D_i}\right) \text{ and } \lambda_{\mathcal{W},D_i} = \left(\beta \times \frac{\left(\mathcal{W}_{E_k} - \varpi_{E_k}\right)}{\mathcal{W}_{E_k}}\right) \tag{8}$$

where σ is a control parameter for $U_{D_i}^{\mathcal{W}}(\cdot)$ and $\left(\mathfrak{N}_{D_i} \times \mathcal{L}_{D_i}\right)$ is the requested bandwidth of D_i, and ϖ_{E_k} is the currently used E_k's bandwidth amount. For the bandwidth sharing problem, the ***IM***, ***SIR*** and ***PCI*** axioms are preferred to give higher preferences to mission-critical tasks. Therefore, the concept of *LKSBS* is applied to distribute the $\mathcal{W}_{E_k}$ resource to devices in the set $\mathbb{H}_{E_k}$. It is given for the $\mathbb{G}_{E_k}^{\mathcal{W}}$ game, and the $\mathcal{W}_{E_k}^{D_i}$ value is adaptively decided according to (9).

$$\varphi^{LKSBS}\left(D_i, D_j \in \mathbb{H}_{E_k} \mid U_{D_i}^{\mathcal{W}}(\cdot), \left(\Lambda_{E_k}, \Theta_{\mathcal{W},D_i}\right)\right) = \left\{\left\{\ldots \mathcal{W}_{E_k}^{D_i} \ldots\right\} \mid \frac{\left(\mathfrak{I}_{D_j}(\cdot) - U_{D_j}^{\mathcal{W}}(\cdot)\right)}{\left(U_{D_j}^{\mathcal{W}}(\cdot) - U_{D_j}^{d}(\cdot)\right)} = \left(\mathfrak{T}_{D_i} \times \frac{\left(\mathfrak{I}_{D_i}(\cdot) - U_{D_i}^{\mathcal{W}}(\cdot)\right)}{\left(U_{D_i}^{\mathcal{W}}(\cdot) - U_{D_i}^{d}(\cdot)\right)}\right)\right\}$$
$$\text{s.t., } \Lambda_{E_k} = \left(\ldots \lambda_{\mathcal{W},D_i} \ldots\right) \in \mathbb{R}_{+}^{\left|\mathbb{H}_{E_k}\right|}, \quad \mathfrak{T}_{D_i} = \frac{1 + \lambda_{\mathcal{W},D_j}}{1 + \lambda_{\mathcal{W},D_i}} \text{ and } \sum_{D_i \in \mathbb{H}_{E_k}} \mathcal{W}_{E_k}^{D_i} \leq \mathcal{W}_{E_k} \tag{9}$$

D. MAIN STEPS OF OUR PROPOSED FEDERATED LEARNING ALGORITHM

In this study, we have developed a new computation offloading control scheme for a MEC system infrastructure. By adopting the main concepts of *UBS*, *LNBS* and *LKSBS*, multiple IoT devices can share fair-efficiently the limited system resources. To design our proposed scheme, we formulate a novel joint bargaining game model in a distributed online manner. To make control decisions, game players bargain with each other to get mutual advantages, and work together through the dynamics of edge cloud platform. Based on the interactive bargaining approach, different bargaining solutions are interdependent to strike the appropriate performance balance for the MEC process.

Usually, control algorithms have exponential time complexity in order to solve classical optimal problems. These methods are impractical to be implemented for realistic system operations. In this study, we do not focus on trying to get an optimal solution based on the traditional optimal approach. But instead, the decision mechanism in our joint bargaining game model is implemented with polynomial complexity. In the point view of practical operations, it is suitable approach for the real world edge-assisted computing system. The main steps of the proposed scheme can be described as follows:

Step 1: For our simulation model, the values of system parameters and control factors can be discovered in Table 1, and the simulation scenario is given in Section IV.

Step 2: In each time step of bargaining game process, individual devices generate their computing tasks while contacting their neighboring edge cloud for offload services.

Step 3: In each device, the game $\mathbb{G}_{D}^{\mathcal{L}}$ is operated to decide its $\mathcal{L}$ value. According to (4), utility functions, i.e., $U_{D}^{L}(\cdot)$ and $U_{D}^{O}(\cdot)$, are defined for local and offload computing services, and the $\mathcal{L}$ value decision problem is solved based on the equation (5).

Step 4: In each edge cloud (E), computation (Γ_E) and communication ($\mathcal{W}_E$) resources are distributed its corresponding devices in the set $\mathbb{H}_E$ through the $\mathbb{G}_{E}^{\Gamma}$ and $\mathbb{G}_{E}^{\mathcal{W}}$ game models.

Step 5: In the $\mathbb{G}_{E}^{\Gamma}$, the Γ resource in the E is allocated for the E's corresponding devices; each individual device's utility function, i.e., $U_{D}^{\Gamma}(\cdot)$, is defined by using (6). Based on this information, the Γ resource is shared by the idea of *LNBS*, and the Γ_{E}^{D} value is decided according to (1),(2) and (7).

Step 6: In the $\mathbb{G}_{E}^{\mathcal{W}}$, the $\mathcal{W}$ resource in the E is allocated for the E's contacting devices; each individual device's utility function, i.e., $U_{D}^{\mathcal{W}}(\cdot)$, is defined by using (8). Based on this information, the $\mathcal{W}$ resource is shared by the idea of *LKSBS* and the $\mathcal{W}_{E}^{D}$ value is decided according to (1),(3) and (9).

TABLE 1. System parameters used in the simulation experiments.

Parameter	Value	Description
n	10	the number of edge clouds in the MEC system
m	100	the number of devices in the MEC system
Γ_E	1 THz	the computing power of edge cloud
$\mathcal{W}_E$	1.2 Tbps	the communication resource of edge cloud
Υ_D	1 GHz	the computation power of D
$\mathfrak{E}_D$	100 joules	the initial energy of D
κ_D	1 cycle /1 bit	the number of CPU cycles to process one-bit data
Δ	0.1	the basic control unit to adjust $\mathcal{L}$ value
α, ψ	1, 1	control factors for D's utility function
γ	3	control factor for $\mathcal{F}_D^{\mathcal{P}}(\cdot)$ function
μ	2.7	control factor for $\mathcal{G}_D^{c}(\cdot)$ function
η, ζ	1, 1	control parameters for $U_D^{\Gamma}(\cdot)$ function
β	0.2	system adjustment factor for loss aversion tendency
σ	1	a control parameter for $U_D^{\mathcal{W}}(\cdot)$ function
Ω_O	0.7 µs/cycle	a time delay parameters for offloading service
Ω_L	1.5 µs/cycle	a time delay parameters for local service

Type	$\mathcal{M}_D^{\mathcal{T}}$	Factor (ρ) for Θ	Data bit size of $\mathcal{T}_D$	Duration (average/s)
I	12 sec.	$\rho_D = 0.75$	$\mathfrak{N}_D = 10$ Mb	120 sec.
II	9 sec.	$\rho_D = 0.7$	$\mathfrak{N}_D = 7.5$ Mb	90 sec.
III	7.5 sec.	$\rho_D = 0.85$	$\mathfrak{N}_D = 5$ Mb	180 sec.
IV	5 sec.	$\rho_D = 0.8$	$\mathfrak{N}_D = 2.5$ Mb	60 sec.
V	15 SEC.	$\rho_D = 0.9$	$\mathfrak{N}_D = 12$ Mb	240 sec.
VI	10 SEC.	$\rho_D = 0.75$	$\mathfrak{N}_D = 9$ Mb	150 sec.

Step 7: In a distributed online fashion, individual edge clouds and devices make their control decisions adaptively. They work together in a coordinated manner to strike the appropriate performance balance between efficiency and fairness principles.

Step 8: Constantly, the game entities are self-monitoring the current MEC platform situations, and proceed to Step 2 for the next bargaining process.

IV. PERFORMANCE EVALUATION

In this section, we first describe the experiment settings and then evaluate system performance to show how our proposed protocol works effectively. The performance improvement of our presented scheme is compared with other existing *ECFL*, *RCFL* and *CPFL* schemes [1], [4], [5]; these existing schemes are recently published state-of-the-art MEC protocols. The assumptions of our simulation environments are as follows:

- The simulated MEC system platform consists of 10 edge clouds and 100 IoT devices.
- Multiple edge clouds and devices are regularly positioned. Therefore, one edge cloud is associated with 10 devices.
- The computing power (Γ_E) and communication resource ($\mathcal{W}_E$) of each edge cloud are 1 Tera Hz and 1.2 Tera bps, respectively.
- Each device generates its tasks ($\mathcal{T}_D$), which are six different kinds of service types based on the connection duration, spectrum requirement, loss aversion tendency and maximum delay-latency.
- The process for task generations in individual devices is Poisson with rate λ (services/s), and the range of offered task load was varied from 0 to 3.0.
- In this study, the value of $\mathcal{L}_D$ is defined as integer multiples of control unit (Δ); we set $\Delta = 0.1$ in this study.
- The Υ_D and $\mathfrak{E}_D$ are initially set to one Giga Hz and 100 joules for each device. We assume that 1 Pico joule is consumed to process one bit process.
- To reduce computation complexity, the Γ_E and $\mathcal{W}_E$ allocations are specified in terms of basic units (BUs), where one BU for Γ_E is one GHz and one BU for $\mathcal{W}_E$ is one Gbps.
- The utility of disagreement point, i.e., $U_{D_i}^{d}(\cdot)$, is zero in our system.
- System performance measures obtained on the basis of 100 simulation runs are plotted as a function of the offered task request load.
- Performance measures obtained are normalized device's payoff, bandwidth utilization, and task failure probability in the MEC system.
- For simplicity, we assume the absence of physical obstacles in the wireless communications.

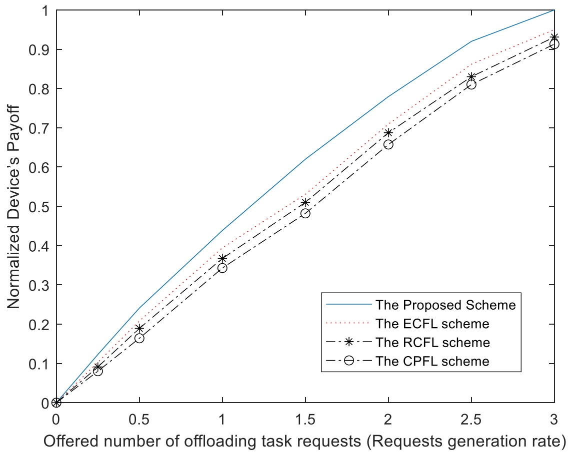

FIGURE 2. Normalized payoff of devices.

In Fig.2, we plot the normalized payoff of device as a function of task generation rate. In the point view of device users, this is the most important performance criterion. The more offered task requests in each device, the more service applications are executed. The simulation results reveal that our joint cooperative game approach effectively controls the system resource under light to heavy task load distributions. It can lead to higher device's payoff in the MEC infrastructure. Especially, the *UBS*, *LNBS* and *LKSBS* in our proposed scheme work together in an interactive manner, and they attempt to enhance the impact of our combined cooperative game. It is a primary advantage of our joint bargaining model in contrast with the *ECFL*, *RCFL* and *CPFL* schemes.

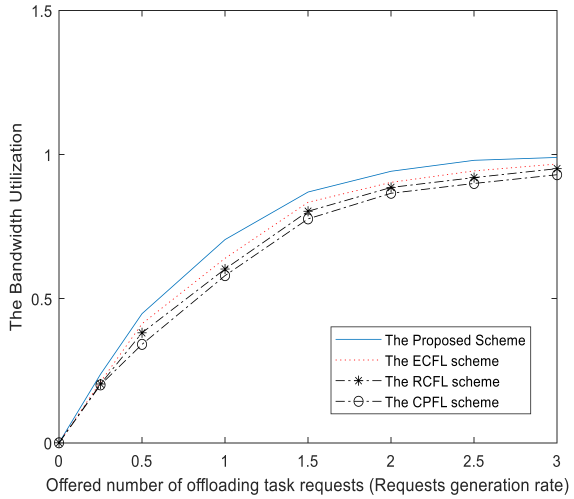

FIGURE 3. The bandwidth utilization.

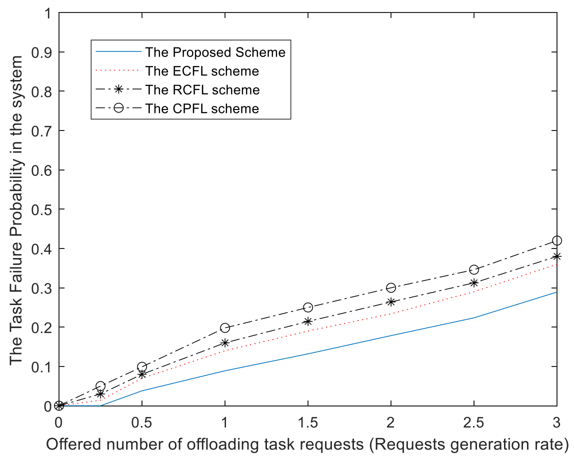

FIGURE 4. Task failure probability in the system.

Fig.3 shows the comparison results about the bandwidth utilization in each edge cloud. Usually, the spectrum usage increases in proportion to task generation rate; it is intuitive correct. Therefore, it is strongly related to the device's payoff, and the performance trend showing in Fig.3 is very similar to the curves in Fig.2. Thus, similar conclusions to the ones of Fig.3 are reached. Under different task load intensities, our communication resource distribution algorithm can ensure an efficient spectrum usage with desirable characteristics according to the basic concept of *LKSBS*. Therefore, we can maintain a stable and higher bandwidth utilization than other existing schemes through the bandwidth distribution process in the MEC system platform.

Under diversified system task load changes, the successful task completeness is another prominent issue in the MEC operation. Fig.4 is plotted to assess the task failure probability among different protocols. In this study, learning tasks are generated with their time constrained. Therefore, system entities should fine tune the limited computation and communication resources to increase the rate of task completeness. Major novelty of our proposed scheme is to provide the best compromise in the presence of current system conditions until the best solution has been found during the MEC process. An interesting observation in Fig.4 is that, we can fair-efficiently share the limited system resources among different devices while maintaining a lower task failure probability than the existing state-of-the-art MEC protocols.

From the simulation results shown in Fig.2 to Fig.4, we present the numerical analysis to draw insights for validation. Finally, we can confirm that our joint bargaining approach can attain an appropriate performance balance in the federated system infrastructure while outperforming the existing *ECFL*, *RCFL* and *CPFL* protocols.

V. SUMMARY AND CONCLUSION

FL paradigm stands in contrast to traditional centralized machine learning techniques where all data samples are uploaded to one server. With the idea of computation offloading from smart devices to edge clouds, the MEC emerged to supplement resource-limited IoT devices in the FL. However, computational offload operations involve complex control problems and should be determined in an effective cooperative manner to dynamic edge system environments. In this paper, we design a novel task offloading scheme based on the joint cooperative game model. By using the idea of *UBS*, each device decides its own computation offloading amount, and the concepts of *LNBS* and *LKSBS* are adopted to solve the communication and computation resource distribution problems for each edge cloud. These three solutions are interactively combined and mutually dependent in our proposed scheme. By taking into account the current MEC system condition, different bargaining solutions act cooperatively and collaborate with each other in a real-time online manner. Therefore, task offloading services are effectively operated under the dynamically changing MEC system environment. Extensive simulation is conducted to demonstrate the performance enhancement of our proposed approach compared to the *ECFL*, *RCFL* and *CPFL* protocols, in terms of normalized device's payoff, bandwidth utilization, and task failure probability.

For the future work, our current study can be extended in a number of ways. One future direction is to design a crowdsourcing framework to leverage the MEC process that considers the incentive-based interaction between the crowdsourcing platform and the participating IoT devices. Another potential direction for the future research is to apply MEC paradigm to the demand prediction problem while accurately forecasting the more popular application types in the network. In addition, we will construct a new control algorithm that determines the best tradeoff between local update and global parameter aggregation under the limited system resources.

COMPETING OF INTERESTS

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

AUTHOR' CONTRIBUTION

The author is a sole author of this work and ES (i.e., participated in the design of the study and performed the statistical analysis.

AVAILABILITY OF DATA AND MATERIAL

The data used to support the findings of this study are available by contacting the corresponding author at *swkim01@sogang.ac.kr*.

REFERENCES

[1] J. Ren, H. Wang, T. Hou, S. Zheng, and C. Tang, "Federated learning-based computation offloading optimization in edge computing-supported Internet of Things," *IEEE Access*, vol. 7, pp. 69194–69201, 2019.
[2] S. Savazzi, M. Nicoli, and V. Rampa, "Federated learning with cooperating devices: A consensus approach for massive IoT networks," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4641–4654, May 2020.
[3] S. Savazzi, M. Nicoli, V. Rampa, and S. Kianoush, "Federated learning with mutually cooperating devices: A consensus approach towards serverless model optimization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 3937–3941.
[4] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 1205–1221, Jun. 2019.
[5] Q. Wu, K. He, and X. Chen, "Personalized federated learning for intelligent IoT applications: A cloud-edge based framework," *IEEE Open J. Comput. Soc.*, vol. 1, pp. 35–44, 2020.
[6] H. Yang, W. Wang, and Z. Ding, "The weighted surplus division value for cooperative games," *Symmetry*, vol. 11, no. 9, p. 1169, 2019.
[7] C. Herrero and A. Villar, "The rights egalitarian solution for NTU sharing problems," *Int. J. Game Theory*, vol. 39, nos. 1–2, pp. 137–150, 2010.
[8] J.-M. Giménez-Gómez, A. Osório, and J. Peris, "From bargaining solutions to claims rules: A proportional approach," *Games*, vol. 6, no. 1, pp. 32–38, Mar. 2015.
[9] B. Dietzenbacher and H. Peters, "Characterizing NTU-bankruptcy rules using bargaining axioms," Graduate School Bus. Econ., Maastricht Univ., Maastricht, The Netherlands, CentER Discuss. Paper 005–2018, 2018.
[10] Y. Chun and W. Thomson, "Bargaining problems with claims," *Math. Social Sci.*, vol. 24, no. 1, pp. 19–33, 1992.
[11] B. Dietzenbacher, A. Estévez-Fernández, P. Borm, and R. Hendrickx, "Proportionality, equality, and duality in bankruptcy problems with nontransferable utility," *Ann. Oper. Res.*, vol. 301, nos. 1–2, pp. 65–80, Jun. 2021, doi: 10.1007/s10479-020-03643-3.
[12] H.-S. Lee and J.-W. Lee, "Task offloading in heterogeneous mobile cloud computing: Modeling, analysis, and cloudlet deployment," *IEEE Access*, vol. 6, pp. 14908–14925, 2018.
[13] F. Guo, H. Zhang, H. Ji, X. Li, and V. C. M. Leung, "An efficient computation offloading management scheme in the densely deployed small cell networks with mobile edge computing," *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2651–2664, Dec. 2018.
[14] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
[15] S. Kim, *Game Theory Applications in Network Design*. Hershey, PA, USA: IGI Global, 2014.
[16] J. Shalev, "Loss aversion and bargaining," *Theory Decis.*, vol. 52, no. 3, pp. 201–232, 2002.
[17] B. Driesen, A. Perea, and H. Peters, "The Kalai–Smorodinsky bargaining solution with loss aversion," *Math. Social Sci.*, vol. 61, no. 1, pp. 58–64, Jan. 2011.
[18] R. G. Garroppo, S. Giordano, and D. Iacono, "Radio-aware scheduler for WiMAX systems based on time-utility function and game theory," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2009, pp. 1–6.
[19] L. Xiao, X. Lu, T. Xu, X. Wan, W. Ji, and Y. Zhang, "Reinforcement learning-based mobile offloading for edge computing against jamming and interference," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6114–6126, Jul. 2020.