Edge-Enabled WBANs for Efficient QoS Provisioning Healthcare Monitoring: A Two-Stage Potential Game-Based Computation Offloading Strategy

XIAOMING YUAN [ID][1]**, (Member, IEEE), HANSEN TIAN**[1]**, HAIYANG WANG**[1]**,
HAORU SU**[2]**, JIEMIN LIU**[1]**, AND AMIR TAHERKORDI** [ID][3]

[1]Qinhuangdao Branch Campus, Northeastern University, Shenyang 110819, China
[2]Faulty of Information, Beijing University of Technology, Beijing 100124, China
[3]Department of Informatics, University of Oslo, 0316 Oslo, Norway

Corresponding author: Haoru Su (suhaoru@bjut.edu.cn)

ABSTRACT Wireless Body Area Network (WBAN) as one of the primary Internet of Things (IoT) provides real time and continuous healthcare monitoring and has been widely deployed to improve the quality of peoples' life. In edge-enabled WBANs, intensive computing tasks could be offloaded to Mobile Edge Computing (MEC) servers, guaranteeing that the massive amount of health data with different user priorities could be processed in lower delay and energy consumption. Efficient computation offloading schemes are more critical to satisfy the massive data access and personalized service requirements for multiple Quality of Service (QoS) parameters constraint WBANs. In this paper, we propose a Two-Stage Potential Game based Computation Offloading Strategy (TPOS) to optimize resource allocation while taking into consideration the task priorities and user priorities of WBANs. Firstly, we construct a system utility maximization problem about the QoS of tasks. The reward, cost and penalty functions are given to model the computation offloading. Then, we propose a two-stage optimization method to solve the problem of mutual restriction strategies existing in the strategy space of the potential game model, reducing the computation complexity and improving the feasibility of the algorithm. Finally, performance evaluations on average processing delay, energy consumption and network utility are conducted to show the significance of the proposed TPOS algorithm.

INDEX TERMS WBAN, edge computing, computation offloading, potential game, healthcare monitoring.

I. INTRODUCTION

Wireless Body Area Network (WBAN) is a kind of wireless communication networks centered on the human body, which can collect physiological, behavioral and other health-related data in real time through multiple medical sensor nodes arranged on the surface, inside or near the human body [1]. WBANs enable users to access healthcare services anytime and anywhere. The application of WBAN satisfies the urgent need for long-term, real-time and high-quality healthcare monitoring service requirements for the elderly, chronic patients and even ordinary people. WBANs alleviate the

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

health care problems precipitated by the shortage of medical resources and an unbalanced medical situation, promoting the development of e-healthcare [2].

With the wide spread and application of WBANs, massive health related data access and user personalized service requirements pose intra- and inter-WBAN interference problems. Moreover, sensor nodes are usually wearable or implanted devices with limited battery power. Battery replacing or charging is difficult for these nodes. Data transmission to remote data centers and the cloud consumes more energy. The critical quality of service (QoS) requirements of WBANs could not be guaranteed in the resource-constrained environment of health monitoring. The mobile/multi-access edge computing (MEC) [3], [4] enabled WBANs to solve a

key related problem, namely that the capability of centralized cloud computing cannot match the explosive growth of massive edge data in the era of Internet of Everything.

Despite this, in the edge-enabled WBANs, the resource allocation for multiple sensor nodes of intra- and inter-WBANs still poses a great challenge. Moreover, when considering how to guarantee the agile connectivity, low delay and energy consumption make resource allocation more difficult. Stymied by the restriction of energy and different delays of various user priorities (UP) data in WBAN, the effective computation offloading strategies are transforming into the multi-objective collaborative optimization problems. Resource allocation and offloading decision belong to two different variables in mathematics. Resource allocation is a continuous variable satisfying the interval, while offloading decision is an 0-1 integral variable that cannot be solved uniformly. Feng *et al.* [5] developed a cooperative computation offloading and resource allocation framework by jointly optimizing offloading decision, power allocation, block size and block interval. Cheng *et al.* [6] proposed a joint resource allocation and task scheduling method. The application of game theory can improve the autonomy and intelligence of individual WBAN tasks. The authors [7] formulated the virtual machine migration problem as a one-to-one contract game model which can achieve higher resource utilization rate and system throughput, as well as reduced service packet loss rate and reduced service delay. However, the inherent different user priorities of WBAN data are not considered.

In this paper, we propose a Two-Stage Potential Game based Computation Offloading Strategy (TPOS) for WBANs, which separates the strategy space in the game decision by stages. We use task priority (TP) and user priority (UP) as important criteria for resource allocation to make the allocation problem transformed into a multi-user game problem. At the first stage, we restrict the game space for intra-WBAN to compete for local computing resources through games among tasks. In the game progress, we increase the utility to improve the competitiveness of high-reward tasks for more computing resources, which can meet the service demand of low time delay and low energy consumption. At the same time, for the tasks with few resources obtained in the game, we put them into the send queue to offload in priority. At the second stage, the game space turn to inter-WBAN. The offload tasks from each WBAN will play on the MEC server, and each task games the resource of the MEC server. Finally, through the two-stage optimization algorithm, we can effectively reduce the influence between the mutually restricting decisions and improve the feasibility of the TPOS algorithm. On the other hand, the game we used in the optimization algorithm is potential game, which has a good finite increment property (FIP) as one of special form of non-cooperative games. The FIP ensures the existence of the pure strategy equilibrium solution without the need to prove the existence of the equilibrium solution. In summary, the main contributions are summarized as follows:

- Firstly, we define a utility function that quantifies the impact of resource allocation and offload decision on the QoS of each task based on its characteristics. At the same time, we use the penalty function to constrain the utility of tasks with different TPs to make full use of the free computing resources, and avoid the hunger phenomenon of low-reward tasks.
- Secondly, we construct the utility maximization problem and model it in game theory. As a result, the autonomy and intelligence of each WBAN are improved and the complexity of algorithm is reduced by solving the original problem in a distributed manner. At the same time, we build the model by the potential game, which is one of the non-cooperative games. The potential game further simplifies the problem without tedious verification of the existence of equilibrium solutions.
- Thirdly, we put forward a two-stage optimization algorithm, which is mainly used to solve the game problem of complex decision space and mutual influence and restriction of policy variables in MEC service scenarios. In the two stages, we conduct the offloading and unloading decisions and allocate different local and server computing resources to tasks of certain WBANs with different UPs. The two-stage optimization improves the feasibility of the algorithm.
- Finally, we evaluate the average processing delay, energy consumption and network utility on different data arrival rates as well as the number of tasks and WBANs. The comparison results validate the effectiveness of the proposed TPOS method.

The remainder of the paper is structured as follows: Section II organizes the related work and introduces the potential game. Section III presents the network model of edge-enabled IoT system. Section IV introduces the analysis and formulation of WBAN computation offloading problem. Section V introduces the specific content of the TPOS algorithm. Section VI discusses and analyzes the simulation results. Section VII concludes the paper.

II. PRELIMINARIES
A. RELATED WORK
Wireless body area networks are human centered, highly reliable short-range wireless communication networks [8]. With flexibility, scalability and low-cost characteristics, WBAN technology has been envisioned as one of the primary technologies for the e-Health Internet of Things (IoT) [9], [10]. WBAN has been widely deployed in densely populated scenarios, like wards and waiting rooms in hospitals [11].

Due to the sensitivity and criticality of the data carried and handled by WBAN, fault tolerance is a critical issue and widely discussed [12]. Ye and Zhuang [13] proposed a distributed and adaptive hybrid medium access control (DAH-MAC) scheme for a single hop IoT-enabled mobile ad hoc network supporting voice and data services. The authors [14], [15] proposed an optimization algorithm to

minimize the energy consumption of the offloading system while taking into account the energy consumption of task calculation and file transfer. Contrasting the more complex multiuser scenario, a distributed iterative algorithm for searching continuous convex approximation was proposed in [16] which minimized energy consumption by jointly optimizing radio and computational resources.

Mobile Edge Computing is a latest technology aimed at offloading mobile devices to nearby resource-rich edge architectures, thus freeing them from computing-intensive workloads. Mobile edge computing enhanced base stations (BSs) and powerful distributed edge devices make it possible for local knowledge extraction from massive IoT data [17], [18]. MEC-based applications can achieve lower latency levels than cloud-based applications [19]. Based on Lyapunov's optimization theory, a joint computation allocation and resource management algorithm [20] was proposed by transforming the original problem into a series of deterministic optimization problems in each time block. In [21] the author divided the research on computation offloading into three key areas: decision-making on computation offloading; allocating computing resources in MEC; and mobility management.

Dynamic heuristic algorithms were proposed to make decisions on computation offloading [20], [22], [23]. In [24] a heuristic collaborative content caching strategy was developed to determine the content items to be cached on each MEC server. The region-based scheme performed better than the average content delivery and other intermediate computing strategies balanced the offloading when comparing the delay of MEC servers. Du *et al.* [25] considered a cognitive vehicular network which used the TV white space(TVWS) band, and formulated a dual-side optimization problem to minimize the cost of vehicular technology sand of the MEC server at the same time. In [26]–[29], the authors adopted adaptive schemes to optimize the allocation of resources. Among the research of mobility management, Liu *et al.* [30] used queuing theory to meet the demand for high mobility and reliability. Besides the above-mentioned articles, some researchers focus on the resource allocation of vehicle edge calculation and networks, Hetnets or cellular networks [31]–[35]. The application scenarios are in high mobility and the service priority is not considered, which is not suitable for WBANs. Without considering the priority of tasks could result in too many high priority data obtain insufficient resources [36]. However, in WBAN application scenario, the service priority is very important to the quality of service of users.

In this paper, we model the resource allocation problem of different tasks in different WBANs as multi-user game problem. They play games as game players to determine the amount of computing resources and communication resources acquired. Through the restriction of penalty function, we make the allocated resources meet the demand of maximizing the service as much as possible, and make the service allocation close to the proportion of its own service value. According to our theoretical assumption, under the premise of ensuring that all tasks can be executed within the rated time delay, the resource allocation strategy will prioritize the resource requirements of high UP users and high TP tasks.

B. DEFINITION OF POTENTIAL GAME
Game theory is a theory of conflict and cooperation among intelligent rational players. It can analyze the interaction between multiple independent entities that need to compete or work together to achieve their goals. The game is divided into cooperative game and non-cooperative game according to whether binding agreement can be reached. In [37], the authors show some applications of game theory in solving wireless network problems.

Non-cooperative game mainly studies how to make decisions to maximize their own benefits in situations where interests interact, which is discussed and applied in more occasions. Nash equilibrium is an important concept in non-cooperative games, but not every game has a Nash equilibrium. Therefore, it is necessary to prove the existence of game equilibrium. If the game model itself converges to the Nash equilibrium, the establishment of the game model will be greatly simplified. As a special form of non-cooperative game, potential game is the type of game, which converges to the Nash equilibrium. Potential game has FIP, aka, the increasing path of the game is of finite length or the game subject can reach the Nash equilibrium after a finite number of iterations.

In this paper, we use a potential game to solve resource allocation and offload decision problems. On the one hand, we can make use of the decentralized decision-making process of network nodes in WBAN. Since each player may have different needs and ultimately will not pursue the same interests. A decentralized scenario is required where each participant chooses the best possible strategy to achieve their goals. By leveraging the intelligence of each node, the distributed solution will reduce the complexity of the problem. On the other hand, in view of the problem P is a constrained hybrid nonlinear programming problem, which is usually NP hard. With potential games, the original centralized problem is simplified to make it feasible.

Definition 1 (Potential Game): For the game $\Gamma = \langle M, \{Y_m\}_{m \in \mathcal{M}}, \{P_m\}_{m \in \mathcal{M}} \rangle$, where M is the number of players and the set of players is expressed as $\mathcal{M} = \{1, 2, \ldots, M\}$, Y_m and P_m is the policy space and profit function of the player m. If there is a function $F : Y \rightarrow R$ with $\forall m \in \mathcal{M}$, $\forall y_{-m} \in Y_{-m}$ and $\forall x, z \in Y_m$ can satisfy the following function

$$P_m(x, y_{-m}) - P_m(z, y_{-m}) = F(x, y_{-m}) - F(z, y_{-m}),$$

Γ could be called a perfect potential game. In the definition, y_m represents the strategy of the mth participant while y_{-m} represents the strategies of other participants.

Definition 2 (Nash Equilibrium): In optimization theory, pareto optimality is often used to describe the ideal state of resource allocation. In consideration of the number of

subjects and the amount of resources which can be allocated, pareto optimality is the change from one allocation state to another. The change is impossible to make someone better off without making anyone worse off. A Nash equilibrium is a stable state of play in which all players have no incentive to change their choices.Pareto optimality is considered from a holistic perspective, which means that the sum of the total benefits of all participants reaches the maximum, that is, the whole optimal or social optimal. Nash equilibrium does not necessarily satisfy pareto optimality. If the game results reach pareto optimality, we can think the result of individual rationality is the choice of collective rationality.

In the potential game model, the Nash equilibrium point corresponds to the maximum value of the potential function. If the objective function of the optimization problem is modeled as the potential function of the potential game, the Nash equilibrium point corresponds to the optimal solution of the optimization problem, which is also the pareto optimal solution.

To sum up, we derive the properties of potential game as follows:

- Attribute 1: if the potential function of game $\Gamma = \langle M, \{Y_m\}_{m \in \mathcal{M}}, \{P_m\}_{m \in \mathcal{M}} \rangle$ is $F : Y \to R$, the Nash equilibrium of $\Gamma = \langle M, \{Y_m\}_{m \in \mathcal{M}}, \{P_m\}_{m \in \mathcal{M}} \rangle$ is consistent with that of game:

$$\Gamma = \langle M, \{Y_m\}_{m \in \mathcal{M}}, \{P_m = F\}_{m \in \mathcal{M}} \rangle.$$

- Attribute 2: every finite ordinal potential game has an equilibrium solution of pure strategy.
- Attribute 3: each finite ordinal potential game has the property of FIP.

These three attributes of potential game ensure that the game model established based on potential game must have a Nash equilibrium. So we only need to prove that the model is a potential game, and we do not need to prove the convergence of the model.

III. NETWORK MODEL

We assume a scenario of edge-enabled IoT system with multiple WBANs and MEC servers in the network to provide health monitoring, as shown in Fig.1. The MEC service with much computation resources is deployed at the mobile communication base station and other communication facilities, which can provide real-time computing services for the covered WBAN users. On the other hand, we assume that $\mathcal{N} = [1, 2, \ldots, N]$ shows the set of active WBANs, which can offload several tasks to the MEC server based on the current WBAN system conditions like the communication interference, system load and residual power and so on.

For WBAN $n \in \mathcal{N}$, there are several sensor nodes for real-time monitoring of user health information and a hub with computation workload X_n (in CPU cycle per bit) to receive and process the task from the sensor nodes efficiently. These nodes deliver real-time tasks to hub uniformly through IEEE 802.15.6 protocol in the
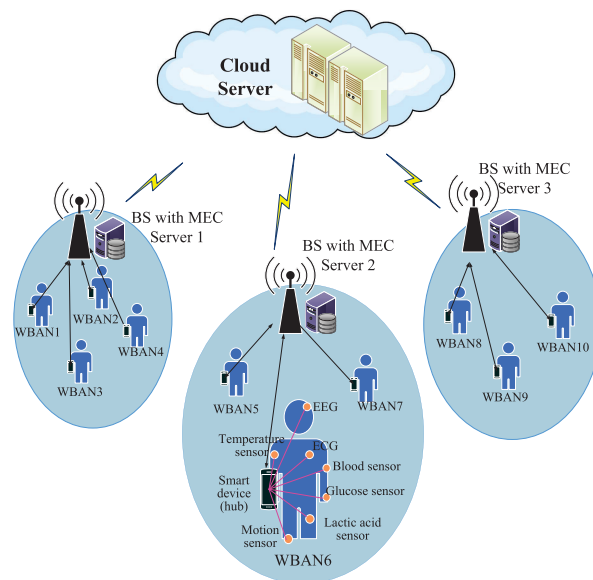

FIGURE 1. Network model.

WBAN network. The set of tasks in WBAN n is denoted as $\mathcal{I}_n = [1, 2, \ldots, I_n]$ and the task $i \in \mathcal{I}_n$ is described by notation $\mathcal{T}_{n,i} = \mathcal{T}(D_{n,i}, K_{n,i})$. Where $D_{n,i}$ represents the amount of data (in bits) of task i. $K_{n,i}$ is the priority of the task which subjects to IEEE 802.15.6 defined task priority $(TP_k | k = 0, 1, 2, \ldots, 7)$, as shown in Table 1. The higher the priority, the lower the delay tolerance of the service, and the more stringent the requirements for QoS in healthcare monitoring.

TABLE 1. TP mapping of WBAN.

Task priorities	Traffic designation
7	Emergency or medical implanted event report
6	High priority medical data or network control
5	Medical data or network control
4	Voice data or management
3	Video data
2	Excellent effort data
1	Best effort data
0	Background data

Given the bandwidth W_n of WBAN n, WBAN can release local computing resources, reduce processing delay and energy consumption and improve tasks processing efficiency through utilizing the server's rich computing resources. Based on current conditions, WBAN can decide a task to be executed locally or offloaded to the MEC server to offloading the current computationally intensive tasks with reasonable offloading strategy. We assume the central processing units (CPUs) of MEC server are idle at the current.

The maximum computing resources of WBAN and MEC are represented by F_n^W and F^M, and F^M is much larger than F_n^W.

IV. PROBLEM FORMULATION

For mathematical modeling analysis of the process of tasks, we assume that it is the process of buying and selling transactions. A series of operations of the tasks are represented by corresponding value functions, as follows

$$U_{n,i} = R_{n,i} - C_{n,i}, \tag{1}$$

where $U_{n,i}$ represents the utility of task i in WBAN n, which maximizes the utility as much possible, while WBAN users try best to maximize the total utility of tasks. At the same time, $R_{n,i}$ represents the reward function of the task, aka, the inherent value that the user will obtain when finish the task. $C_{n,i}$ represents the cost function of the task in the process.

A. REWARD FUNCTION

We assume that WBAN will receive a certain amount of reward which is related to the data size and priority of the task after finishing it. For task $\mathcal{T}_{n,i}$, the reward function could be expressed as

$$R_{n,i} = \left(1 + K_{n,i}\right)^2 \cdot \log_2\left(1 + D_{n,i}\right), \tag{2}$$

where tasks reward is proportional to tasks data and priority, and the impact of priority is greater.

B. COST FUNCTION

At the same time, we believe that WBAN need to pay a certain amount of cost to deal with the tasks, which is related to the processing delay T and energy consumption E. The cost of task i in WBAN n is denoted by $C_{n,i}$. Let λ^t and λ^e respectively denote the delay factor and the energy consumption factor to maximize the network life cycle of WBAN while meeting the low latency and high reliability requirement of the service. In summary, the cost function can be expressed as

$$C_{n,i} = \lambda_{n,i}^t T_{n,i} + \lambda_{n,i}^e E_{n,i}, \tag{3}$$

where $\lambda_{n,i}^t$ is the delay factor of task i, while $\lambda_{n,i}^e$ is its energy consumption factor. $T_{n,i}$ and $E_{n,i}$ indicate respectively the processing delay and energy consumption of task i.

In (3), different tasks can have different weighting parameters depending on the imagined application or even the current system state. For example, if the node battery is low, the energy consumption factor should be increased to save more energy. For delay-sensitive tasks, delay factor is added to reduce delays.

From the system model, WBAN can choose to unload the tasks according to the current system situation, thus releasing the local computing and communication resources. So the tasks can be divided into local execution and offloaded to the MEC server.

1) LOCAL EXECUTION

For the local execution task $\mathcal{T}_{n,i}$ in the WBAN user n. We assume that $f_{n,i}^{loc}$ is the computing resources of WBAN to complete the task, then the local execution time of the task is given by

$$T_{n,i}^{loc} = \frac{D_{n,i} \cdot X_n}{f_{n,i}^{loc}}. \tag{4}$$

Meanwhile, the energy consumption of each CPU cycle is $h\left(f_{n,i}^{loc}\right)^2$ [38], where h is the constant associated with the CPU. The local energy consumption can be denoted by

$$E_{n,i}^{loc} = h_0\left(f_{n,i}^{loc}\right)^2 \cdot D_{n,i} \cdot X_n, \tag{5}$$

where h_0 is the CPU constants, which is the effective capacitance coefficient that depends on the chip architecture in WBAN n.

2) OFFLOADING PROCESSING

For tasks that need to be uninstalled, the total time to finish can be expressed as

$$T_{n,i}^{ser} = T_{n,i}^{tr} + T_{n,i}^c + T_{n,i}^r. \tag{6}$$

From the formula, $T_{n,i}^r$ is the time when the completed task returns to WBAN, which is mainly determined by the MEC server. And the processing result is usually much smaller than the original data size with less impact on the final experimental results. Therefore, in order to simplify the calculation, we ignore the return time $T_{n,i}^r$ in the processing of the offloading task execution time. We assume that a task is taken $T_{n,i}^{tr}$ to transmit from WBAN to MEC, and it is given by

$$T_{n,i}^{tr} = \frac{D_{n,i}}{\mathcal{R}_n}. \tag{7}$$

At the meantime, we assume that a task will transmit data at a fixed subcarrier power P_n^{tr} in WBAN n, and the communication bandwidth of each WBAN is W_n. Furthermore, we consider a flat fading environment that channel gain g_n remains constant during the offloading process. Considering the wireless interference of cellular network, the channel gain is $g_n = d^{-\partial}$, where d is the distance between the WBAN and MEC server and ∂ is the path loss factor. The system noise is in accordance with the zero expectation Gaussian distribution, and the variance is expressed as N_0. The aggregated data rate $\mathcal{R}_n$ is expressed as

$$\mathcal{R}_n = W_n \log_2\left(1 + \frac{P_n^{tr} \cdot g_n}{N_0}\right), \tag{8}$$

$T_{n,i}^c$ is the computation time in the MEC server and can be given by

$$T_{n,i}^c = \frac{D_{n,i} \cdot X_n}{f_{n,i}^{ser}} \tag{9}$$

where $f_{n,i}^{ser}$ is the computing resources of MEC to complete the task.

On the other hand, the WBAN energy consumption of offloading a task can be expressed as

$$E_{n,i}^{ser} = P_n^{tr} \cdot T_{n,i}^{tr} + h_1 \left(f_{n,i}^{ser}\right)^2 \cdot D_{n,i} \cdot X^M, \tag{10}$$

where X^M is the MEC compute workload and h_1 is the CPU constant in MEC server. In this paper, we set $h_1 = 1 \times 10^{-26}$ to maintain the energy consumption per bit at the same order of magnitude [39].

According to the equation (10), when a task is offloaded to the MEC server for processing, the energy consumption includes two aspects. One is the energy consumption during the wireless transmission of the task, and the other is the loss of the task in the MEC server.

We introduce $u = \{u_{n,i} \in \{0, 1\}, \forall i \in \mathcal{I}_n\}$ to indicate the offloading strategy of the task i in WBAN n. It indicates that the task is executed locally when $u_{n,i} = 0$, while $u_{n,i} = 1$ represents that the task is offloaded. We simplify the cost formula by strategy value $u_{n,i}$ with the local cost $C_{n,i}^{loc}$ and the offloading cost $C_{n,i}^{ser}$. The cost function is finally obtained by

$$C_{n,i} = \left(1 - u_{n,i}\right) \cdot C_{n,i}^{loc} + u_{n,i} \cdot C_{n,i}^{ser} \tag{11}$$

Finally, we can conclude that the utility function of each WBAN is

$$U_n = \sum_{i \in \mathcal{I}_n}^{I_n} \left(R_{n,i} - \left(1 - u_{n,i}\right) \cdot C_{n,i}^{loc} - u_{n,i} \cdot C_{n,i}^{ser}\right). \tag{12}$$

C. PENALTY FUNCTION

For one of the WBAN users, the result of resource allocation is usually not satisfactory only by solving the utility function of the whole tasks. Either the allocation of low-reward tasks resources fails to meet the minimum demand, resulting in the phenomenon of hunger; or it cannot meet the service of quality requirements of low delay and low energy consumption in high reward tasks. Therefore, on the basis of the utility function mentioned above, penalty function is introduced to solve the problem. On the one hand, the overall performance of resource allocation can be improved; on the other hand, partial constraints are transformed into unconstrained nonlinear programming problems, which simplifies the complexity of the algorithm.

First of all, we give the function of balanced allocation of resources, the purpose of which is to make the result of resource allocation approximate to the benefit ratio of tasks.

$$G_{n,i} = \frac{R_{n,i}}{\sum\limits_{j}^{I_n} R_{n,j} - \sum\limits_{k,k \neq i}^{I_n} R_{n,k}} - \frac{f_{n,i}}{\sum\limits_{j}^{I_n} f_{n,j} - \sum\limits_{k,k \neq i}^{I_n} f_{n,k}} \tag{13}$$

where $\sum\limits_{j}^{I_n} R_{n,j}$ is the total reward for all current tasks, and $\sum\limits_{k,k \neq i}^{I_n} R_{n,k}$ is the reward from the tasks that have allocated

resources. Similarly, $\sum\limits_{j}^{I_n} f_{n,j}$ and $\sum\limits_{k,k \neq i}^{I_n} f_{n,k}$ represent the total resources of the system and the allocated.

Secondly, with the resource equilibrium distribution function, we can get the penalty function as follows:

$$Pen_{n,i} = -\alpha \left(\frac{G_{n,i} - |G_{n,i}|}{2}\right)^2 \cdot R_{n,i} \tag{14}$$

where α is the penalty factor. As the penalty factor increases, the more severe the utility penalty is for the task, the less computing resources are allocated and tend to be proportional to the reward.

From the penalty function (14), when the resource allocation ratio is greater than the current user benefit ratio, the penalty function result will be negative, and the utility of the tasks will decline, resulting in the reduction of its resource allocation. On the contrary, if the current system has a small number of business and abundant resources, the penalty function result will be equal to 0, so that the task can maximize its utility. Finally, the result is to prevent the high priority business from grabbing too much resources and balance the distribution of resources.

In summary, the tasks process for each WBAN user can be formulated as follows:

$$\mathcal{P} : \max_{u, f^{loc}, f^{ser}} F_n = \sum_{i \in \mathcal{I}_n}^{I_n} \left(U_{n,i} + Pen_{n,i}\right)$$

$$s.t. \ C1 : u_{n,i} \in \{0, 1\}, \forall i \in \mathcal{I}_n$$

$$C2 : \sum_{i \in \mathcal{I}_n}^{I_n} \left(1 - u_{n,i}\right) f_{n,i}^{loc} \leq F_n^W$$

$$C3 : \sum_{i \in \mathcal{I}_n}^{I_n} u_{n,i} f_{n,i}^{ser} \leq F^M$$

$$C4 : T_{n,i} = \left(1 - u_{n,i}\right) T_{n,i}^{loc} + u_{n,i} \cdot T_{n,i}^{ser} \leq \tau_{n,i} \tag{15}$$

where F_n is the total utility of all the tasks in WBAN n. The constraint C1 indicates that the offloading strategy value is a binary variable, and constraint 2, 3 represents the maximum allocation computation resources of WBANs and MEC. C4 shows that each task must be completed in time and the maximum completion time $\tau_{n,i}$.

V. THE PROPOSED TWO-STAGE POTENTIAL GAME BASED COMPUTATION OFFLOADING STRATEGY (TPOS)

In this section, the proposed two-stage computation offloading strategy based on potential game theory is introduced. The strategy can reduce the interaction between the strategies through two different stages and improve the feasibility of the algorithm.

We simplified the original problem $\mathcal{P}$ into a non-cooperative game process based on potential game model. Each task is individually rational in the game, that is, each task tries to maximize its own utility and further increase the

benefit of the overall system. However, the potential game Γ in this paper has two different and mutually restricting strategies in the strategy space. They are resource allocation and offload decisions. When the application of local resources is too small to meet the needs of the task QoS, the offload decision of the task will be transferred to the MEC server to apply for the server resources. As a consequence, the complexity of the algorithm is too large to implement two strategies simultaneously.

The two-stage potential game computation based offloading strategy (TPOS) algorithm is illustrated in Fig. 2, which involves the solutions for the first stage and second stage. The first stage is limited to the intra-WBAN and determine the offload decisions. The second stage moves the game space to MEC server. Tasks of different WBANs start the game to get the computing resources from high to low utility.
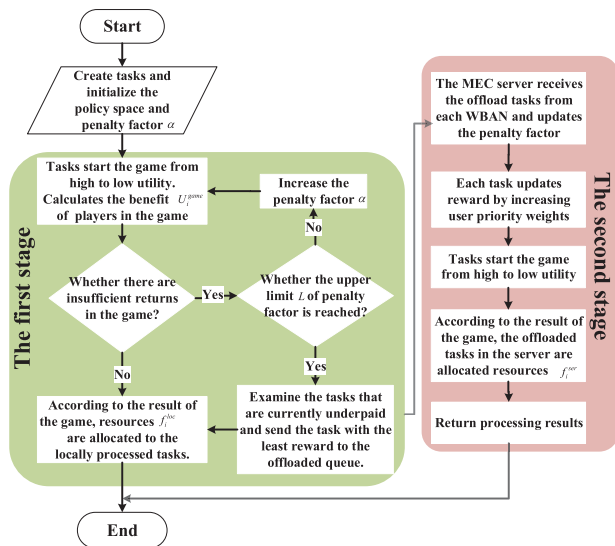

FIGURE 2. The TPOS algorithm.

A. THE FIRST STAGE

In the first stage, we first model the potential game for tasks of each WBAN, which can determine the offload decisions and local computing resources allocation of tasks. The entire game space is limited to the WBAN n where the tasks are generated. Firstly, the WBAN nodes generate the tasks as the players of the potential game, who perceive the system environment and determine the strategy space. The hub will receive these tasks and rank each of them according to its reward value $R_{n,i}$, and players will start the game from high to low.

After the end of the first game, we check the current distribution of resources and determine whether the tasks meet the QoS by determining the profit function $P_{n,i}$ of the players in the game. The function indirectly reflects the current tasks density of WBAN system. Through a given threshold value L, we can judge the allocation of resources

for the tasks. When the profit of the player is greater than the threshold value, we consider the allocation reasonable. On the contrary, when it is less than the threshold value, we consider the allocation of resources to be too few. The profit of the player should be proportional to the utility value of the task, that is, the higher the profit is, the more resources the task applies for, the lower the delay and energy consumption and the better QoS. When there are players whose profits fall below the threshold, we believe that the result of resource allocation is uneven and the strategy will be updated.

In the process of strategy updating, the penalty factor size of each penalty function in task utility will be gradually increased, which will affect the high-utility task that preempts too many resources. The update process will still be sorted according to the utility value, and the unified scheduling of hub will ensure that players play in order. When the penalty factor reaches the upper limit and some utilities of players are still low, we can determine that the current local resources are not enough to allocate all tasks. Among the tasks that do not meet the threshold value, the task with the lowest utility will be offloaded and added to the sending queue. The rest of the tasks restart the game until the current local task is satisfied and all are above the threshold. The first stage of the game is over.

B. THE SECOND STAGE

Unlike the first stage, the game space in the second is located in the MEC server. The MEC server receives the offloading tasks from the transmit queue of each WBAN, and the server resources allocated by each task through the potential game as in the first stage. We consider each task to be a player and to be ranked according to their respective reward. Then the MEC server starts the game.

In addition, considering the differences of different users, we introduced the user priority $\mathcal{K}_n$ at the second stage. It is related with the applications or services of different WBANs. $\mathcal{K}_n$ is defined in (16).

$$\mathcal{K}_n = \lceil \frac{\sum_{i \in \mathcal{I}_n} u_{n,i}}{I_n} + \frac{\max \{u_{n,i} \cdot K_{n,i}\}}{2} \rceil, \quad \forall i \in \mathcal{I}_n, \ \forall n \in \mathcal{N} \tag{16}$$

where $u_{n,i}$ is the offloading decision and $K_{n,i}$ is the priority of task i in WBAN n. The values of $\mathcal{K}_n$ are integer values between 1 and 4. The proportion of offloaded tasks and the highest priority in the unloaded tasks can reflect the tasks density of the WBAN. We use the user priority in the reward weighting of offloaded tasks. The reward function is updated as follows

$$R'_{n,i} = (1 + \mathcal{K}_n) R_{n,i} \tag{17}$$

After the updated task reward function, the tasks will game server resources in the MEC. The specific game process is similar to that of the previous stage, but the difference is that the game in this stage does not need to consider the threshold of the utility of players. Therefore, it is no longer necessary

to gradually increase the penalty factor to make computing resources more fully utilized. But without the iterative process, the optimal size of the penalty factor still needs to be resolved. If the offloaded tasks in the MEC server have little difference, the MEC needs to balance the resources of each task, and the penalty factors increases. On the contrary, when there is a great difference in task rewards, penalty factors need to be reduced. As a result, we adjust the size of the penalty factor according to the current task dynamics unloaded to the MEC.

C. GAME MODEL
Potential game modeling of MEC system is carried out from three elements of the game, which are players, profit function and strategy space. Specific instructions are as follows:

1) PLAYERS
In order to fully reflect the autonomy of each node in WBAN, we map the generated tasks to the players. The players in each game can make their own decisions, maximize their own profit in the process of mutual game, and promote the optimal benefit of the system.

2) PROFIT FUNCTION
From the economic point of view, this paper coordinates the resource distribution of each task, so the profit function of players corresponds to their respective utility functions with each penalty function as follows:

$$P_{n,i} = U_{n,i} + Pen_{n,i}, \quad \forall i \in \mathcal{I}_n, \tag{18}$$

where $P_{n,i}$ is the profit of each player. At the same time, we assume the profit $P_{n,i}$ equals the utility of all tasks in the WBAN, that is $P_{n,i} = F_{n,i}$.

3) POLICY SPACE
The resource allocation and offloading decision to maximize the utility of each task act as the strategic space of each player. Therefore, the potential game is the combination of the current offload state and the local resources or MEC server resources applied for by the task.

$$\begin{cases} Y_{n,i} = \left\{ u_{n,i}, f_{n,i}^{loc}, f_{n,i}^{ser} \right\}, & \forall i \in \mathcal{I}_n \\ u_{n,i} \in \{0, 1\}, & \forall i \in \mathcal{I}_n \\ 0 \le f_{n,i}^{loc} \le F_n^W, & \forall i \in \mathcal{I}_n \\ 0 \le f_{n,i}^{ser} \le F^M, & \forall i \in \mathcal{I}_n. \end{cases} \tag{19}$$

where F_n^W and F_n^M are the total computing resources of WBAN and MEC respectively. $f_{n,i}^{loc}$ and $f_{n,i}^{ser}$ are the allocated WBAN and MEC resources of task i in WBAN n. It should be noticed that, according to the attributes 2 and 3 of potential game, the strategy space needs to be discretized through linear segmentation to make it a finite strategy space and ensure that the game is a finite game.

4) MODEL ANALYSIS
In order to maximize the utility of each task in the game and optimize the benefit of the system in the same time, we consider the construction potential function $J_n(y)$ as the sum of the utility of each player in (20). $J_n(y)$ can maximize the utility of each task and the benefit of the system, which corresponds to the potential function in the potential game.

$$J_n(y) = \sum_{i \in \mathcal{I}_n} P_{n,i} = \sum_{i \in \mathcal{I}_n} F_{n,i}. \tag{20}$$

According to the definition of potential game in *Definition* 1, when a certain player chooses different strategies, the change of the player's own benefit ΔJ_n is the same as the change of the potential function ΔF_n in potential game. Then, we prove the perfect potential game proposed in (21) and (**??**).

$$J_n(y) = J_n(y_{n,i}, y_{n,-i}) = F_{n,i} + \sum_{j \in \mathcal{I}_n, j \neq i} F_{n,j} \tag{21}$$

where y is the set of strategies for all current players. Given $y_{n,i}$ is the strategy of a random player, then $y_{n,-i}$ is the strategy set of other players. $\sum_{j \in \mathcal{I}_n, j \neq i} F_{n,j}$ is the utility of other players except i. Assume that $y'_{n,i}$ is the changed strategy of player i and $F'_{n,i}$ is the changed utility after changing the strategy.

$$\begin{aligned} \Delta J_n &= J_n(y_{n,i}, y_{n,-i}) - J_n(y'_{n,i}, y'_{n,-i}) \\ &= \left(F_{n,i} + \sum_{j \in \mathcal{I}_n, j \neq i} F_{n,j} \right) - \left(F'_{n,i} + \sum_{j \in \mathcal{I}_n, j \neq i} F'_{n,j} \right) \\ &= F_{n,i} - F'_{n,i} \\ &= \Delta F_n \end{aligned} \tag{22}$$

We get the $\Delta J_n = \Delta F_n$, the game model established based on the economic model meets the definition of a perfect potential game and has all the attributes of a potential game. After the strategy space is discretized, the potential game is a finite potential game, which has the property of limited improvement. According to attribute 2, the potential game must have a Nash equilibrium solution of pure strategy.

VI. PERFORMANCE EVALUATION
In this section, we build a simulation environment using Python to evaluate the performance of our proposed algorithm. We set up a MEC server and multiple WBANs in the simulation scenario. Each WBAN consists of a hub and multiple sensor nodes, and each sensor node will only generate tasks with single task priority. In addition, each WBAN has a user priority attribute. We use a queuing model to simulate the process of sending tasks from WBAN to the MEC server. We ignore the transmission delay of data in the channel and only consider the sending delay and queuing delay of the task. Some parameters of the simulation environment are detailed in Table 2.

We conduct the performance evaluation on the average processing delay and average processing energy. The simulation

TABLE 2. System parameters.

Parameters	Values
Task priority	[0,7]
User priority	[1,4]
Processing frequency of single WBAN F^W	0.8-1.0 GHz
Processing frequency of single MEC server F^M	4.0-5.0 GHz
The computation workload X_n	1000 cycles/bit
Subcarrier bandwidth W_n	180 KHz
Maximum transmit power P_n^{tr}	1.0 W
Background noise N_0	-100dBm
Task size $D_{n,i}$	1000-10000 bits
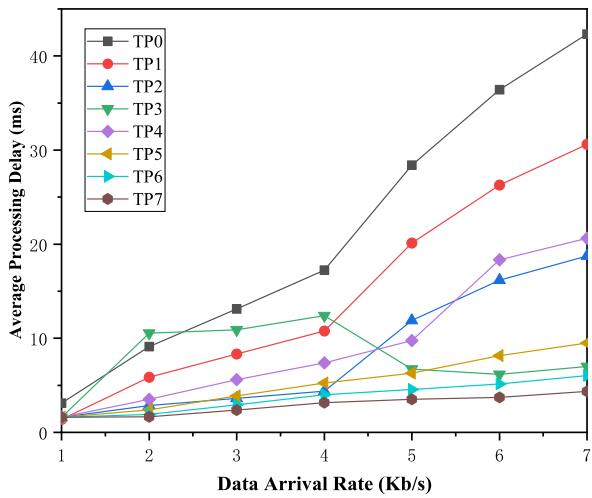
FIGURE 3. The delay of different task priority with different data arrival rates.

FIGURE 4. The energy consumption of different task priority with different data arrival rates.

FIGURE 5. The utility of different task priority with different data arrival rates.

FIGURE 6. The delay of different user priority with different number of WBANs.

results are performed on different arrival rates and the number of WBANs and tasks with different user priorities and task priorities. At the same time, we compared the proposed TPOS with the other two task processing modes. The simulation results are shown in Fig. 3-Fig. 10.

The relationship of the average processing delay of tasks with different task priorities and different task arrival rates is shown in Fig. 3. The data arrival rate here refers to the sum of the data size of tasks generated in a single WBAN during a unit time. In this scenario, there are 4 WBANs within the service range of the MEC server. The WBANs have the same user priority. The results show the average delay of tasks TP0-TP7 increase with the data arrival rates. High TP tasks, such as TP7, TP6, are in low delay than other low TP tasks. That is because the first allocation principle of TPOS is ensuring high-priority tasks can get enough resources to meet the requirements of users. When the data arrival rate is small, low priority tasks are executed locally and can obtain sufficient computing resources to meet user needs. As data arrival rate continues to increases, the computing
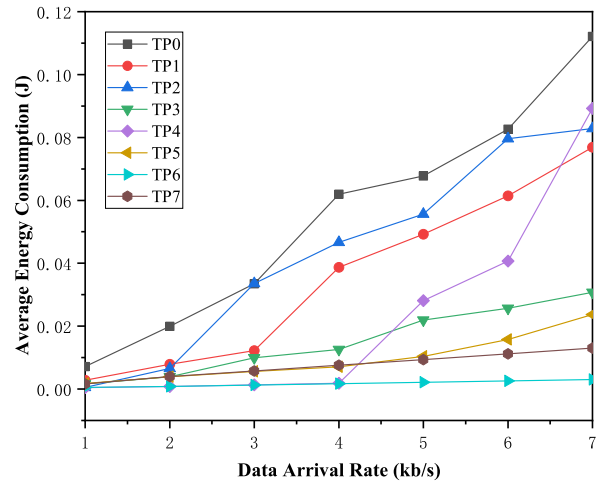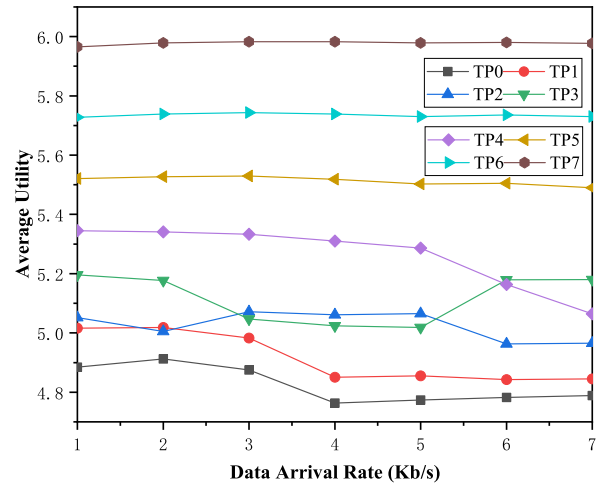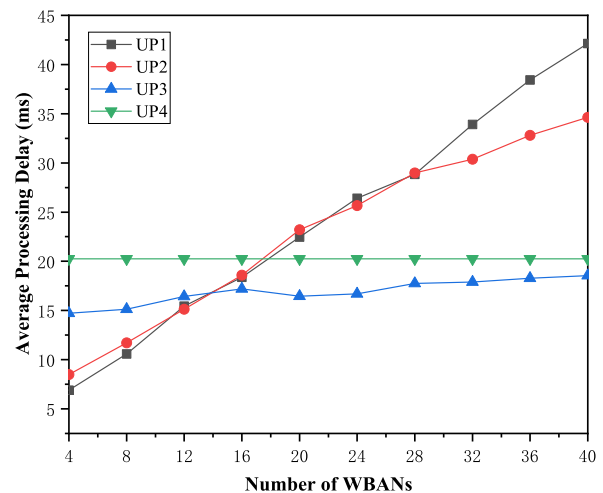
resources of a single WBAN are insufficient to process tasks locally at high speed. In other words, the number of tasks

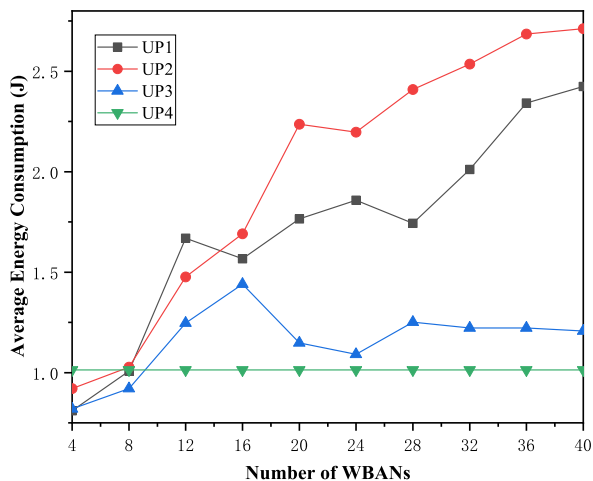**FIGURE 7.** The energy consumption of different user priority with different number of WBANs.
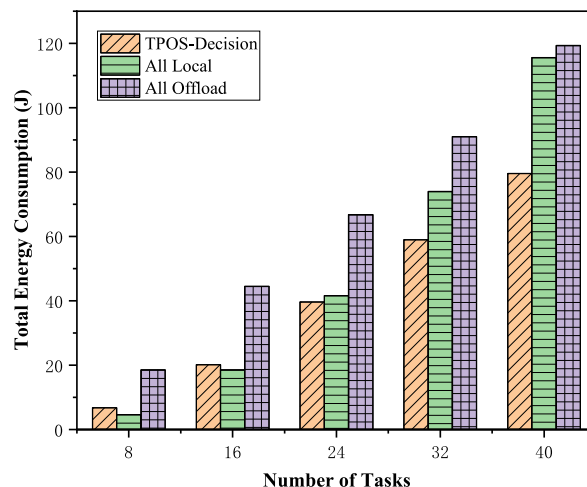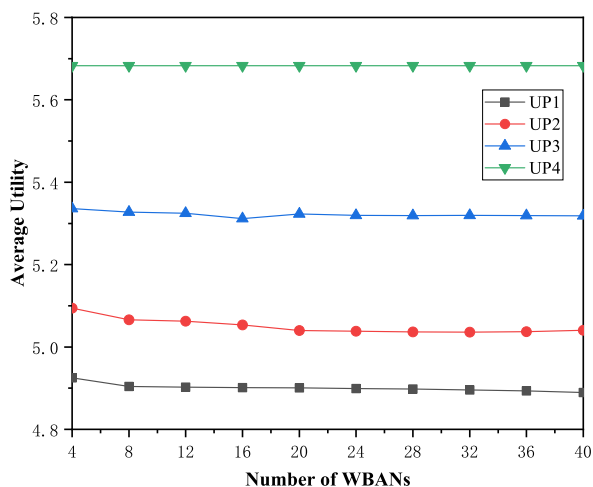

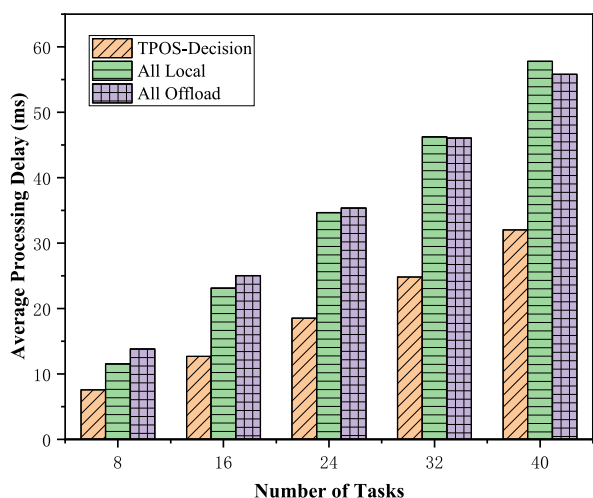**FIGURE 8.** The utility of different user priority with different number of WBANs.

FIGURE 9. The average delay of different processing modes with different number of tasks.

participating in the resource game locally increases, resulting in a gradual decrease and leveling off of the resources

FIGURE 10. The average energy consumption of different processing modes with different number of tasks.

available for each task. The delay of low TP tasks increase faster than high TP tasks. We can observe that the task processing delay curves of 3 and TP 4 all have a significant downward trend when the data arrival rates increase. This is because these priority tasks are offloaded to MEC server to process and occupy the highest order in the queuing model so that get a lower queuing delay. Therefore, these tasks can achieve a phased reduction in processing delay when they are just offloaded compared to execute locally. Overall, TPOS guarantee both high and low TP tasks with lower latency.

We can obtain the average energy consumption of different tasks with different data arrival rates from Fig. 4. With the increasing data arrival rates, the energy consumption of all the task increases. High TP tasks always have priority to obtain enough computing resources. They are processed locally in high probability in the TPOS algorithm. When there are too much data in the network, the local resources are insufficient for all tasks. The low TP tasks are offloaded to the MEC server with rich computing resources for further processing. This will increase the transmission power consumption of low TP tasks.

Figure 5 describes the average utility of tasks with the changing arrival data. The average utility values of tasks decline slowly with the increasing data arrival rates. This is due to the continuous increase of the processing cost of the processing delay and energy consumption. From the definition of utility function shown in (1), we can know the utility of the high TP tasks always better than the low TP tasks. Since the cost of high priority task in WBAN is smaller than low priority task in WBAN.

The impacts of the number of different WBANs with different UPs on the performance are shown in Fig. 6 and Fig. 8. In this scenario, there are four UP kinds of WBANs and each UP only generates 8 tasks with two TPs. We fix the data arrival rates to 5Kb/s and the number of WBANs varies from 4 to 40. From (16) we can know the WBANs with UP3 and UP4 generates TP4-TP7 tasks while UP1 and
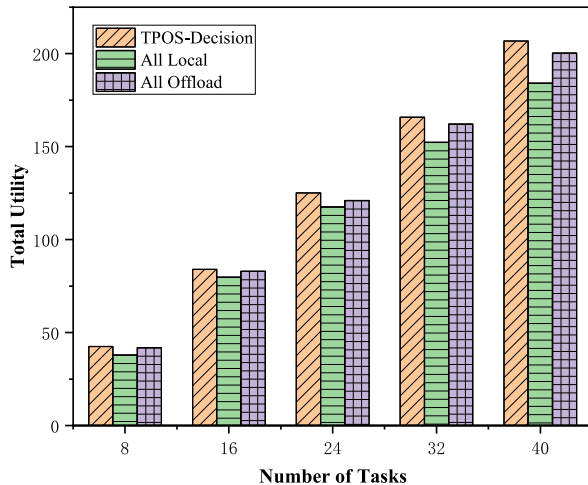
FIGURE 11. The total utility of different processing modes with different number of tasks.

UP2 WBANs generate TP0-TP4 tasks. Different UPs would make different offloading decision according to state of system. As mentioned before, the higher the UP is, the more inclined to process tasks locally and the lower the UP is, the more inclined to offload tasks to MEC server. Because of the limited computational resource of WBANs, the tasks offloaded will obtain more resource than those processed locally and the process delay is less. That is why UP1 and UP2 WBANs have low processing delay than UP3 and UP4 WBANs at first. With the increasing of number of WBANs, there are more and more tasks to play resource games, the penalty function factor also increases, which limits the amount of resources the task obtains, resulting in greater processing delay and energy consumption. As shown in Fig. 6 and Fig. 7, the average processing delay and energy consumption of UP1 and UP2 increases rapidly. On the contrary, UP3 and UP4 process most tasks locally which results in relatively small fluctuations on processing delay and energy consumption.

In a densely WBANs employed scenario, the TPOS computation offloading algorithm can keep the power consumption of the device at a healthy level to meet the user's needs for battery life. The average utility of processing tasks for different UPs is in Fig. 8. All WBANs with different UPs show the stable system utility which validates the effectiveness of the proposed TPOS algorithm. The high UP WBAN has great advantages than low UP WBAN. The utility of UP4 is almost 16% more than UP1.

The relationships of the performance of the TPOS algorithm and the number of tasks are shown in Fig. 9-Fig. 11. The data arrival rate of per task is 3Kb/s. At the same time, a MEC server with four WBANs in the same UP are set in the scenario. Each WBAN generates 8 TP tasks. We make a comparison between TPOS and two other processing modes, all local mode and all offload mode. The number of tasks changes from 8 to 40 in a single WBAN. Since the TPOS considers both the characteristics of tasks and WBANs, we can

clearly observe that the offload decision made by the TPOS algorithm has great advantages compared with the other two processing modes. Here we calculate the average processing delay of each task, total energy consumption and utility of a WBAN. The values of these performance are in positive growth with the increasing number of tasks. The TPOS could save at least 20% energy when the tasks more than 32 of one WBAN. Moreover, the utility of the TPOS is the highest among the three computing resource allocation schemes no matter how many tasks in a WBAN. Even the whole network has more than 300 tasks, TPOS can guarantee the tasks be processed in low delay, low energy consumption and high utility. These advantages make the TPOS more applicable for health monitoring.

VII. CONCLUSIONS

In this paper, we have proposed a Two-Stage Potential Game based Computation Offloading Strategy (TPOS) for WBANs with considering the task priorities and user priorities. We divided the game space into two stages to solve the multi-user game problem. At the first stage, tasks with different priorities in WBANs obtained their offloading decisions according to their utility function and penalty function. At the second stage, MEC server allocates computing resources to offloaded tasks through the potential game. Evaluation results showed the TPOS algorithm can meet the need of low delay and low energy consumption even in heavy tasks and dense employed WBANs scenario. In this paper, the movements of different WBANs have not been considered. As the direction of our future work, we will devote attention to the problems of the selection of MEC server and the computing resource allocation for mobile WBANs.

REFERENCES

[1] R. Chavez-Santiago, K. Sayrafian-Pour, A. Khaleghi, K. Takizawa, J. Wang, I. Balasingham, and H.-B. Li, "Propagation models for IEEE 802.15.6 standardization of implant communication in body area networks," *IEEE Commun. Mag.*, vol. 51, no. 8, pp. 80–87, Aug. 2013.

[2] S. Movassaghi, M. Abolhasan, J. Lipman, D. Smith, and A. Jamalipour, "Wireless body area networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1658–1686, 3rd Quart., 2014.

[3] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[4] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, "Making knowledge tradable in edge-AI enabled IoT: A consortium blockchain-based efficient and incentive approach," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6367–6378, Dec. 2019.

[5] J. Feng, F. R. Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, "Cooperative computation offloading and resource allocation for blockchain-enabled mobile edge computing: A deep reinforcement learning approach," *IEEE Internet Things J.*, early access, Dec. 24, 2019, doi: 10.1109/JIOT.2019.2961707.

[6] X. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/Aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.

[7] S. Kim, "One-on-one contract game-based dynamic virtual machine migration scheme for mobile edge computing," *Trans. Emerg. Telecommun. Technol.*, vol. 29, no. 1, p. e3204, Jan. 2018, doi: 10.1002/ett.3204.

[8] M. Salayma, A. Al-Dubai, I. Romdhani, and Y. Nasser, "Wireless body area network (WBAN): A survey on reliability, fault tolerance, and technologies coexistence," *ACM Comput. Surveys*, vol. 50, no. 1, pp. 1–38, Apr. 2017.

[9] K. Zhang, K. Yang, X. Liang, Z. Su, X. Shen, and H. H. Luo, "Security and privacy for mobile healthcare networks: From a quality of protection perspective," *IEEE Wireless Commun.*, vol. 22, no. 4, pp. 104–112, Aug. 2015.

[10] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

[11] X. Yuan, C. Li, Q. Ye, K. Zhang, N. Cheng, N. Zhang, and X. Shen, "Performance analysis of IEEE 802.15.6-based coexisting mobile WBANs with prioritized traffic and dynamic interference," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5637–5652, Aug. 2018.

[12] A. Samanta and S. Misra, "Energy-efficient and distributed network management cost minimization in opportunistic wireless body area networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 2, pp. 376–389, Feb. 2018.

[13] Q. Ye and W. Zhuang, "Distributed and adaptive medium access control for Internet-of-Things-Enabled mobile networks," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 446–460, Apr. 2017.

[14] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.

[15] M. Kamoun, W. Labidi, and M. Sarkiss, "Joint resource allocation and offloading strategies in cloud enabled cellular networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 5529–5534.

[16] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[17] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," 2019, *arXiv:1908.06849*. [Online]. Available: http://arxiv.org/abs/1908.06849

[18] T. Yang, H. Feng, C. Yang, Y. Wang, J. Dong, and M. Xia, "Multivessel computation offloading in maritime mobile edge computing network," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4063–4073, Jun. 2019.

[19] T. Chanyour, M. El Ghmary, Y. Hmimz, and M. O. C. Malki, "Energy-efficient and delay-aware multitask offloading for mobile edge computing networks," *Trans. Emerg. Telecommun. Technol.*, 2019, p. e3673, doi: 10.1002/ett.3673.

[20] S. Mao, S. Leng, K. Yang, Q. Zhao, and M. Liu, "Energy efficiency and delay tradeoff in multi-user wireless powered mobile-edge computing systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.

[21] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[22] H. Cao, S. Wu, G. S. Aujla, Q. Wang, L. Yang, and H. Zhu, "Dynamic embedding and quality of service-driven adjustment for cloud networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1406–1416, Feb. 2020.

[23] Y. Kim, H.-W. Lee, and S. Chong, "Mobile computation offloading for application throughput fairness and energy efficiency," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 3–19, Jan. 2019.

[24] N. Wang, G. Shen, S. K. Bose, and W. Shao, "Zone-based cooperative content caching and delivery for radio access network with mobile edge computing," *IEEE Access*, vol. 7, pp. 4031–4044, 2019.

[25] J. Du, F. R. Yu, X. Chu, J. Feng, and G. Lu, "Computation offloading and resource allocation in vehicular networks based on dual-side cost minimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1079–1092, Feb. 2019.

[26] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 695–708, Jan. 2019.

[27] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1689–1703, Mar. 2020.

[28] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for energy-efficient mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4188–4200, Jun. 2019.

[29] S. Guo, J. Liu, Y. Yang, B. Xiao, and Z. Li, "Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 319–333, Feb. 2019.

[30] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 283–294, Feb. 2018.

[31] C. Chen, L. Liu, T. Qiu, D. O. Wu, and Z. Ren, "Delay-aware grid-based geographic routing in urban VANETs: A backbone approach," *IEEE/ACM Trans. Netw.*, vol. 27, no. 6, pp. 2324–2337, Dec. 2019.

[32] N. Cheng, F. Lyu, J. Chen, W. Xu, H. Zhou, S. Zhang, and X. Shen, "Big data driven vehicular networks," *IEEE Netw.*, vol. 32, no. 6, pp. 160–167, Nov. 2018.

[33] F. Lyu, N. Cheng, H. Zhu, H. Zhou, W. Xu, M. Li, and X. Shen, "Towards rear-end collision avoidance: Adaptive beaconing for connected vehicles," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 22, 2020, doi: 10.1109/TITS.2020.2966586.

[34] Y. Wei, F. R. Yu, M. Song, and Z. Han, "User scheduling and resource allocation in HetNets with hybrid energy supply: An actor-critic reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 680–692, Jan. 2018.

[35] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[36] T. Chen and G. B. Giannakis, "Bandit convex optimization for scalable and dynamic IoT management," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 1276–1286, Feb. 2019.

[37] J. Moura and D. Hutchison, "Game theory for multi-access edge computing: Survey, use cases, and future trends," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 260–288, 1st Quart., 2019.

[38] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[39] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, "Energy-efficient joint offloading and wireless resource allocation strategy in multi-MEC server systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

XIAOMING YUAN (Member, IEEE) received the B.E. degree in electronics and information engineering from Henan Polytechnic University, Jiaozuo, China, in 2012, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, China, in 2018. From September 2016 to August 2017, she was a Visiting Scholar with the Broadband Communications Research (BBCR) Group, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. Since 2018, she has been an Assistant Professor at the Qinhuangdao Branch Campus, Northeastern University, Shenyang, China. Her research interests include cloud/edge computing, medium access control, and performance analysis for wireless body area networks and the Internet of Things.

HANSEN TIAN is currently pursuing the bachelor's degree in computer science and technology with the Qinhuangdao Branch Campus, Northeastern University, Shenyang, China. His research interests include cloud/edge computing and performance optimization for wireless body area networks.

HAIYANG WANG is currently pursuing the bachelor's degree in Internet of Things Engineering with the Qinhuangdao Branch Campus, Northeastern University, Shenyang, China. His research interests include cloud/edge computing and performance optimization for wireless body area networks and the Internet of Things.

JIEMIN LIU received the B.E. degree in computer application from the Northeastern Heavy Machinery College, Qiqihar, China, in 1988, the M.E. degree in computer and information engineering from Yanshan University, Qinhuangdao, China, in 1997, and the Ph.D. degree from Northeastern University, China, in 2008. Since 2010, he has been a Professor at Northeastern University, Qinhuangdao, China, where he has been the Dean of the Department of Computer and Communication Engineering, since April 2012. He is also the Vice Chairman of ACM China Council, Qinhuangdao, and the Chairman of CCF, Qinhuangdao. His current research interests include next generation networks and information management systems.

HAORU SU received the B.S. degrees in computer science and engineering and the M.S. degree in computer applied technology from Northeastern University, China, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Korea University, South Korea. She is currently working at the Faulty of Information, Beijing University of Technology, China. She is also a Visiting Scholar with the Broadband Communications Research Lab, Department of Electrical and Computer Engineering, University of Waterloo, Canada. Her research interests include multiple access control protocol design and performance evaluation for wireless body area networks, beacon scheduling mechanisms for the IEEE 802.15.4/ZigBee wireless networks, and reader collision problem in dense RFID systems.

AMIR TAHERKORDI received the Ph.D. degree from the Department of Informatics, University of Oslo, in 2011. He is currently a Researcher with the Networks and Distributed Systems Group, Department of Informatics, University of Oslo. His research interests include distributed computing and software engineering aspects of emerging technologies, such as the Internet of Things, clouds and fogs/edges, cyber-physical systems, and smart grids. His current research interests also include software architectures, programming abstractions, service distribution, and middleware systems for the Internet of Things, as well as adaptation of middleware solutions for multicloud applications (EU H2020 MELODIC Project). He has experience from several Norwegian and EU research projects and has published several articles in high-ranked conferences and journals. He was selected as a Young Talented Researcher by the Norwegian Research Council, in 2017, to work on a novel IoT service computing model for future Fog-Cloud computing systems (NFR DILUTE Project).

● ● ●