Received February 1, 2018, accepted March 19, 2018, date of publication March 22, 2018, date of current version April 23, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2818111

Spatial and Temporal Computation Offloading Decision Algorithm in Edge Cloud-Enabled Heterogeneous Networks

HANEUL KO[1,2], JAEWOOK LEE[3], AND SANGHEON PACK[3], (Senior Member, IEEE)
[1]Smart Quantum Communication Research Center, Korea University, Seoul 02841, South Korea
[2]The University of British Columbia, Vancouver, BC V6T 1Z4, Canada
[3]School of Electrical Engineering, Korea University, Seoul 02841, South Korea

Corresponding author: Sangheon Pack (shpack@korea.ac.kr)

This work was supported in part by the National Research Foundation under Grant 2017M3C4A7065980 and Grant 2017R1E1A1A01073742, and in part by the Institute for Information and Communications Technology Promotion through the Korean Government (MSIT) under Grant 2015-0-00575.

ABSTRACT A novel concept of the edge cloud has recently been introduced to reduce transmission costs in mobile cloud computing services. Heterogeneous networks with diverse radio access networks will be pervasive in the future. In this paper, we propose a spatial and temporal computation offloading decision algorithm (ST-CODA) in edge cloud-enabled heterogeneous networks. In ST-CODA, a mobile device decides where and when to process tasks by means of a Markov decision process with the consideration of the processing time and energy consumption of different computation nodes and the transmission cost in heterogeneous networks. Extensive evaluation results are given to demonstrate the effectiveness of the ST-CODA in terms of the transmission cost, the energy efficiency of the mobile device, and the number of tasks that can be processed before their deadline.

INDEX TERMS Cloud computing, edge cloud, mobile edge computing (MEC), computation offloading, Markov decision process (MDP), optimization, heterogeneous networks.

I. INTRODUCTION

The explosive growth of mobile devices enables users to be connected to the Internet anytime and anywhere through wireless connectivity. Meanwhile, various applications that require high computing power and rich resources (e.g., video games, data mining, and gesture recognition) have become more popular [1]–[4]. However, most mobile devices have limited computing power and resources to support these applications [5]. To address this problem, computation offloading, where mobile devices transfer tasks to an external cloud and receive the results from the cloud, has been introduced as a promising solution [6], [7]. By exploiting computation offloading, mobile devices can obtain results with short processing delays and reduce their energy consumption for processing tasks.

To reduce the transmission cost in mobile cloud computing services, a novel concept of the edge cloud has been introduced [8]. In addition, heterogeneous networks with diverse radio access networks will be pervasive in the future [9]. In edge cloud-enabled heterogeneous networks, one of the most important issues is to decide where and when to offload tasks (i.e., spatial and temporal decisions about computation offloading).

For the spatial decision, tasks can be processed in three types of computation nodes, i.e., the central cloud,[1] the edge cloud, and the mobile device. In other words, tasks can be offloaded to an external node (i.e., central or edge cloud) or can be processed at the mobile device. These computation nodes have several advantages and disadvantages. Computation offloading to the central cloud may lead to the increased transmission cost due to long distance between the central cloud and the mobile device [12]. Meanwhile, the edge cloud is closer to the mobile device than the central cloud, and therefore computation offloading to the edge cloud results in lower transmission cost. However, the edge cloud has less computation power than that of the central cloud, which can increase the processing delay. On the other hand,

[1]Representative examples of central clouds are Amazon EC2 [10] and Windows Azure [11].

when tasks are processed at the mobile device, there is no transmission cost. However, the mobile device consumes energy to process tasks. In addition, the mobile device has the smallest computation power among the three computation nodes (i.e., the mobile device has the longest processing delay). The decision of where to offload tasks can be made by considering these salient features.

Various wireless access technologies with different transmission costs are deployed in heterogeneous networks. Intuitively, if a low-cost access network (e.g., a Wi-Fi network) is actively used to offload tasks to clouds, the transmission cost can be reduced. However, since the available networks change depending on the location of the mobile device, the low-cost access network cannot always be exploited. In such situations, the mobile device delays the task processing and then offloads tasks opportunistically through the low-cost access network by using the mobility of the mobile device. Meanwhile, when task processing is delayed excessively, the task cannot be completed within the deadline. Therefore, it is not a trivial issue to decide when to offload tasks.

In this paper, we propose a spatial and temporal computation offloading decision algorithm (ST-CODA) in edge cloud-enabled heterogeneous networks. In ST-CODA, the mobile device decides where and when to offload tasks in consideration of the advantages and disadvantages of the computation nodes and the different transmission costs in heterogeneous networks. To optimize the performance of ST-CODA, a Markov decision process (MDP) problem is formulated, and the optimal policy on the spatial and temporal offloading is obtained by a value iteration algorithm. Extensive evaluation results are given to demonstrate the effectiveness of ST-CODA with the optimal policy compared to that of other schemes in terms of transmission cost, energy efficiency of the mobile device, and the number of tasks processed within the deadline.

The key contributions of this paper are twofold: 1) to the best of our knowledge, this is the first work to simultaneously decide and optimize where and when to offload tasks in edge cloud-enabled heterogeneous networks; and 2) we present and analyze extensive evaluation results under various practical environments, which provides guidelines for designing offloading services. Based on these results, an intelligent offloading service can be implemented in edge cloud-enabled heterogeneous networks.

The remainder of this paper is organized as follows. Related works are summarized in Section II. The detailed operation of ST-CODA is described in Section III, and the MDP model is developed in Section IV. Evaluation results are given in Section V, followed by concluding remarks in Section VI.

II. RELATED WORK

A number of studies on the computation offloading have been conducted to address the resource limitation of the mobile device [14]–[28]. These studies can be categorized into: 1) architecture design [14]–[19]; and 2) algorithm design [20]–[28].

Li *et al.* [14] suggested a three-tier architecture that leverages user location prediction, real-time network performance, and cloud servers' loads to optimize the offloading decision and designed a cloud-enabled Wi-Fi access point (AP) selection scheme to find the most energy efficient AP. Shukla and Munir [15] proposed a computation offloading architecture to process the huge amount of data while guaranteeing the task completion before the deadline. Kovachev *et al.* [16] proposed a mobile augmentation cloud service framework that enables adaptive execution of Android applications from a mobile device to the cloud. Tong *et al.* [17] organized edge cloud servers into a hierarchical architecture. After that, when the loads exceed the capacities of lower tiers of edge cloud servers, they can be aggregated and offloaded by other servers at higher tiers in the edge cloud hierarchy. Puente *et al.* [18] presented a detailed approach for edge cloud deployment, architecture, and protocol, which allows the deployment of edge clouds with no modification in the long-term evolution (LTE) architecture. Lobillo *et al.* [19] introduced a small cell manager which optimizes the overall operation in cloud-enabled small cells.

Zhang *et al.* [20] proposed an optimal offloading algorithm for mobile devices with consideration of the mobile device's local load and the availability of cloud servers. Hoang *et al.* [21] developed an optimization model to address the admission control for mobile cloud computing based on semi-MDP (SMDP). Geng *et al.* [22] developed an energy-efficient computation offloading algorithm for cellular networks by formulating the offloading problem as a shortest path problem and solved the problem by means of the Dijkstra's algorithm. Chen [23] introduced a game-theoretic approach to achieve efficient computation offloading for mobile cloud computing. Wolski *et al.* [24] investigated a decision maker that determines when to move a part of a computations to more capable resources by statistically predicting the execution time. Truong-Huu *et al.* [25] proposed a dynamic and opportunistic offloading algorithm to decide whether to offload or defer the task processing based on MDP. Zhao *et al.* [26] developed an optimization problem whose objective function is to maximize the probability that task execution satisfies the given delay bound. The problem was proved to be concave, and an optimal algorithm was proposed. Tang and Chen [27] studied a social-aware computation offloading game, and designed a distributed computation offloading algorithm to achieve the Nash equilibrium. Al-Shatri *et al.* [28] proposed a distributed computation offloading algorithm to minimize the total network energy in multi-hop networks.

These works improve the performance of computation offloading; however, no previous studies simultaneously consider different types of computation nodes and the heterogeneity of wireless networks.

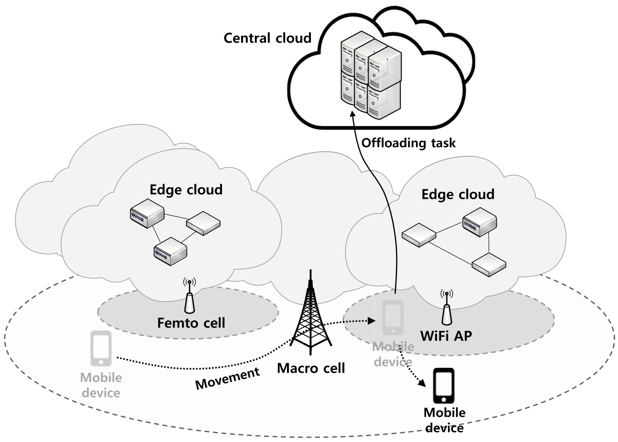


FIGURE 1. System model.

III. SPATIAL AND TEMPORAL COMPUTATION OFFLOADING DECISION ALGORITHM (ST-CODA)

As shown in Figure 1, we consider edge cloud-enabled heterogeneous networks that deploy different wireless access technologies (e.g., Wi-Fi networks and cellular networks). In addition, we consider three types of computation nodes, that is, tasks can be processed by the central cloud, the edge cloud, and the mobile device.

Tasks processed in different computation nodes incur different transmission costs. Since the central cloud is generally located far from the mobile device, computation offloading to the central cloud may lead to higher transmission cost. Meanwhile, the edge cloud is closer to the mobile device than the central cloud. Therefore, computation offloading to the edge cloud can be achieved with lower transmission cost. By contrast, there is no transmission cost when the task is processed in the mobile device. However, the mobile device consumes energy to process the task. Note that energy is a valuable resource in the mobile device due to its limited battery capacity.[2] On the other hand, computation nodes have different computation powers; thus, different processing delays are expected. In summary, the three types of computation nodes have different transmission costs, computation powers, and energy consumptions. By considering these features in ST-CODA, the mobile device can carefully decide where to process tasks.

Meanwhile, in heterogeneous networks with low-cost access networks (e.g., Wi-Fi networks), computation offloading can be delayed and opportunistically conducted through the low-cost access network. However, if the task processing is excessively delayed, the task cannot be completed within its deadline r. Therefore, in ST-CODA, the mobile device carefully decides when to process the task in consideration of the deadline r and the probability that the mobile device will move to a low-cost wireless network. For example, if the deadline is far enough away and the probability that the mobile device handovers to a low-cost wireless network is sufficiently high, the mobile device can delay the task processing. Then, after moving to a low-cost wireless network, the mobile device can offload the tasks to the central or edge cloud through low-cost wireless networks.

To sum up, meticulous spatial and temporal decisions are required to achieve high efficiency computation offloading in edge cloud-enabled heterogeneous networks. These decisions can be handled by four types of actions in ST-CODA: 1) offloading to the central cloud; 2) offloading to the edge cloud; 3) processing the task by itself; and 4) delaying the task processing. The mobile device decides the most beneficial action based on the current state information (e.g., the time remaining until the deadline, the transmission cost, and energy consumption). This selection can be optimized by means of MDP, which will be further elaborated in Section IV.

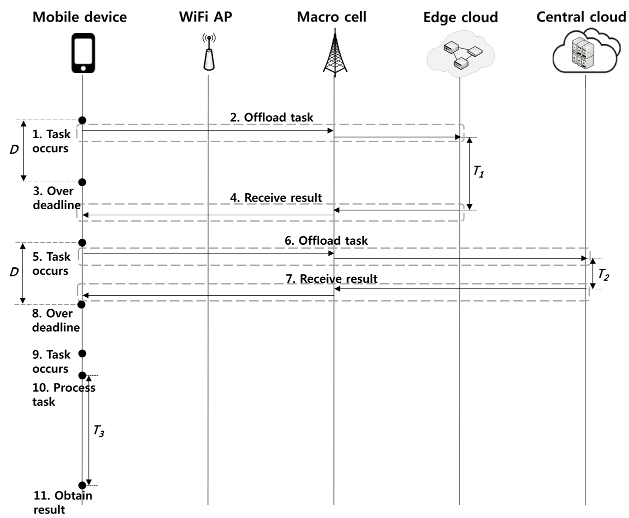


FIGURE 2. Operation example for the spatial decision.

Figure 2 shows an operational example of ST-CODA for the spatial decision. In this example, we assume that the mobile device resides in a cellular network, and the deadline of the task is D. When a task occurs (Step 1), the mobile device offloads the task to the edge cloud through cellular networks. Since the edge cloud has limited computation power, the task can be processed with a long delay denoted by T_1; thus, the result can be received after the deadline (Steps 3-4). On the other hand, when another task is offloaded to the central cloud, the result can be received before the deadline (Steps 5-8). Note that, since the central cloud has higher computation power than the edge cloud, the task can be processed with a short delay, denoted by T_2.[3] In this case, a shorter processing delay can be achieved at the expense of a higher transmission cost. Since the central cloud is generally

[2]In general, other computation nodes (i.e., the central and edge clouds) do not have energy constraints.

[3]The total delay consists of the processing delay and the transmission delay. Typically, since the processing delay is a dominant factor in the total delay, the difference of transmission delays to edge and central clouds is neglected.

located farther from the mobile device than the edge cloud, the transmission cost to the central cloud is higher than that to the edge cloud. On the other hand, when another task occurs (Step 9), the mobile device decides to process the task by itself (Step 10); therefore, there is no transmission cost. However, some energy ϵ of the mobile device is consumed to handle the task. Moreover, since the computation power of the mobile node is lowest among the computation nodes, the task is processed with the longest delay, denoted by T_3.

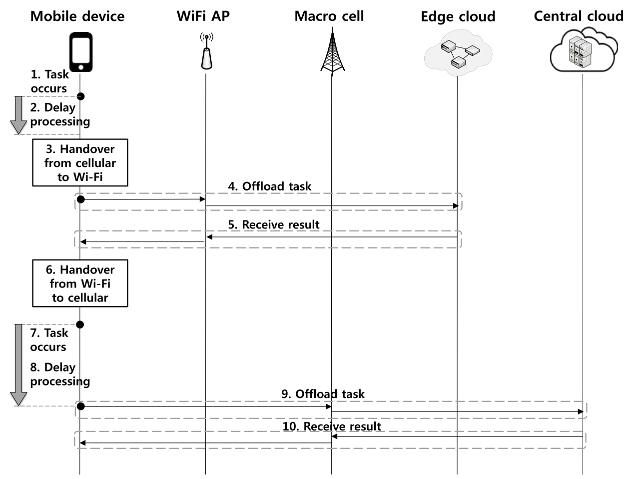


FIGURE 3. Operation example for the temporal decision.

Figure 3 shows an operational example of ST-CODA for the temporal decision. In this example, we assume that the mobile device is initially in a cellular network. When a task occurs (Step 1), the processing can be delayed to offload through low-cost access networks (Step 2) if the mobile device anticipates that it will moves to a Wi-Fi network soon. After handover from the cellular network to the Wi-Fi network (Step 3), the mobile device offloads the task to the edge cloud (Step 4) and receives the result from the edge cloud through the Wi-Fi network (Step 5). Then, the mobile device hands over to the cellular network (Step 6). When another task occurs (Step 7), the processing can be delayed (Step 8). If the mobile device moves to a low-cost access network while processing is delayed, the task can be offloaded at a lower transmission cost (like Steps 1-5). However, since the mobile device does not handover to the low-cost access network, the mobile device offloads the task and receives the result through the cellular network (Steps 9-10). An unnecessary delay occurs in this example, which increases the probability that the task is not processed before the deadline. Therefore, the delay should occur only when the deadline is far enough in the future and the probability that the mobile device will enter a low-cost access network within a short time is sufficiently high.

IV. MDP FORMULATION

In edge cloud-enabled heterogeneous wireless networks, three types of computation nodes with different computing powers and latencies are deployed in heterogeneous networks with different transmission costs. In ST-CODA, by considering these factors, the mobile device decides its action: 1) offloading to the central cloud; 2) offloading to the edge cloud; 3) processing the task by itself; and 4) delaying the task processing. This process improves the performance of computation offloading in edge cloud-enabled heterogeneous networks. To optimize this decision, we formulate an MDP model[4] with five elements: 1) decision epoch; 2) state; 3) action; 4) transition probability; and 5) reward and cost functions [30], [31]. Subsequently, we introduce the optimality equation and a value iteration algorithm to solve the equation. Important notations for the MDP model are summarized in Table 1.

TABLE 1. Summary of notations.

Notation	Description
s_t	State at decision epoch t
a_t	Action chosen at decision epoch t
τ	Duration of each decision epoch
$\mathbf{S}$	State space
$\mathbf{T}$	State for denoting the task phase
$\mathbf{L}$	State for denoting the location of the mobile device
$\mathbf{C}$	State for denoting available networks
$\mathbf{R}$	State for denoting the time remaining until the deadline
$\mathbf{A}$	Action set
ω_1	Weighted factor to balance $f(s,a)$ and $g(s,a)$
ω_2	Weighted factor to balance $g_T(s,a)$ and $g_E(s,a)$
ω_3	Weighted factor to balance $g_S(s,a)$ and $g_R(s,a)$
κ_m	Coefficient to reflect the transmission cost of the mth networks
R_O	Unit reward
ζ_C (or ζ_E)	Transmission cost to reflect the distance between the central cloud (or the edge cloud) and the mobile device
ϵ	Energy consumption to process the task in the mobile device

A. DECISION EPOCH

A sequence $T_e = \{1, 2, 3, \ldots\}$ represents the time epochs in which successive decisions are made. Random variables s_t and a_t denote the state and the action chosen at decision epoch $t \in T_e$, respectively. τ represents the duration of each decision epoch.

B. STATE

We define the state space $\mathbf{S}$ as

$$\mathbf{S} = \mathbf{T} \times \mathbf{L} \times \mathbf{C} \times \mathbf{R} \tag{1}$$

where $\mathbf{T}$ describes the task phase, and $\mathbf{L}$ is the vector set that represents the current location of the mobile device and the adjacency information of each location. In addition, $\mathbf{C}$ is the vector set that represents the available networks, and $\mathbf{R}$ is the time remaining until the deadline.

[4]The MDP model represents a mathematical framework to model decision making in situations where outcomes are partially random and partially under the control of the decision maker [29]. Therefore, the MDP model is suitable for deciding where and when a task is offloaded or processed.

T can be defined as

$$\mathbf{T} = \{0, 1, 2, 3, 4, 5, 6, 7, 8\} \tag{2}$$

where t ($\in$ **T**) denotes the phase of the task. If there is no task, $t = 0$. On the other hand, $t = 1$ refers to the situation immediately after the task occurs. $t = 2$ represents the situation when the task is in the buffer (i.e., when task processing is not in progress). Meanwhile, $t = 3$, $t = 4$, and $t = 5$ represent the situations where the task is processed in the central cloud, the edge cloud, and the mobile device, respectively. Additionally, $t = 6$, $t = 7$, and $t = 8$ represent situations immediately after the task processing is completed at the central cloud, the edge cloud, and the mobile device, respectively.

L can be constructed by a map segmentation technique [32]. In the map segmentation technique, when the transmission power, the target signal to interference plus noise ratio (SINR), and the location of base stations/access points are given, we can analytically obtain the contours of radio access technologies [33], [34]. Then, we can segment a geographical region into a number of locations depending on the obtained contours. We describe **L** by

$$\mathbf{L} = \{L_1, L_2, L_3, \ldots, L_{N_L}\} \tag{3}$$

where N_L represents the total number of locations at which a mobile device can be located, and L_i is the vector representing the adjacency between location i and other locations. That is, L_i is given by

$$L_i = \left[l_i^1, l_i^2, l_i^3, \ldots, l_i^{N_L}\right] \tag{4}$$

where l_i^j represents whether location j is adjacent to location i, i.e.,

$$l_i^j = \begin{cases} 1, & \text{if location } j \text{ is adjacent to location } i \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

For example, if N_L is 5 and location 1 is adjacent to locations 2 and 4, L_1 is given by [0, 1, 0, 1, 0].

Meanwhile, **C** is described by

$$\mathbf{C} = \{C_1, C_2, \ldots, C_{N_{P,C}}\} \tag{6}$$

where $N_{P,C}$ denotes the total number of possible combinations of K heterogeneous wireless networks, i.e., $N_{P,C} = 2^K$. In addition, C_χ represents the χth possible combination, which is represented by

$$C_\chi = [c_1, c_2, \ldots, c_K] \tag{7}$$

where c_ξ is an indicator variable. That is, if the ξth network is available, $c_\xi = 1$; otherwise, $c_\xi = 0$. For example, if the total number of networks is 5 and the first and fourth networks are available, $C_\chi = [1, 0, 0, 1, 0]$.

R is represented by

$$\mathbf{R} = \{Inf, 0, 1, 2, \ldots, N_R\} \tag{8}$$

where r ($\in$ **R**) denotes the time remaining until the task deadline. In addition, N_R represents the deadline of the task. Note that when the task does not occur, $r = Inf$.

C. ACTION

When a task is in the buffer (i.e., $t = 2$), the mobile device decides whether to delay the task processing, to process the task by itself, or to offload the task to the external cloud (i.e., the mobile device takes an action for the task in its buffer) based on the current state information. Then, the action set can be described by

$$\mathbf{A} = \{O_C, O_E, O_S, D\} \tag{9}$$

where O_C and O_E indicate that the mobile device offloads the task to the central cloud and the edge cloud, respectively. Meanwhile, O_S denotes the action that the task is processed at the mobile device, and D is the action to delay the task processing.

D. TRANSITION PROBABILITY

The task phase can be changed by the chosen action. That is, **T** is influenced by the chosen action a. Meanwhile, when the location of a mobile device is given, the available networks in that location are also obtained. Therefore, **L** and **C** states are dependent on each other. On the one hand, when the task occurs (or the task processing is completed), the deadline is set (or reset). That is, **R** is influenced by **T**. Therefore, for the chosen action a, the transition probability from the current state, $s = [t, L_i, C_\chi, r]$, to the next state, $s' = [t', L_j, C_{\chi'}, r']$, can be described by

$$P[s'|s, a] = P[t'|t, a] \times P[L_j, C_{\chi'}|L_i, C_\chi] \times P[r'|r, t]. \tag{10}$$

The transition probability of **T** can be derived as follows. We assume that the inter-arrival rate of the task follows an exponential distribution with mean $1/\lambda_T$. Then, the transition probability from $t = 0$ to $t = 1$ is given by $\lambda_T \tau$ [35]. Therefore, when $t = 0$, the transition probability from t to t' is given by

$$P[t'|t = 0, a] = \begin{cases} \lambda_T \tau, & \text{if } t' = 1 \\ 1 - \lambda_T \tau, & \text{if } t' = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

On the other hand, when $t = 1$, t' is always 2. Therefore, we have

$$P[t'|t = 1, a] = \begin{cases} 1, & \text{if } t' = 2 \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

When $t = 2$, t' is dependent on the chosen action a. That is, when the chosen action is D, the task phase is not changed since the task remains in the buffer; therefore, $P[t'|t = 2, a = D]$ can be represented as

$$P[t'|t = 2, a = D] = \begin{cases} 1, & \text{if } t' = 2 \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

On the other hand, when the mobile device decides to offload the task or process it by itself, the task phase can be changed according to the chosen action. Since $t = 3$, $t = 4$, and $t = 5$ indicate that the task is processed in the central cloud, the edge cloud, and the mobile device, respectively, the corresponding transition probabilities can be denoted by

$$P[t'|t=2, a=O_C] = \begin{cases} 1, & \text{if } t'=3 \\ 0, & \text{otherwise}, \end{cases} \tag{14}$$

$$P[t'|t=2, a=O_E] = \begin{cases} 1, & \text{if } t'=4 \\ 0, & \text{otherwise}, \end{cases} \tag{15}$$

and

$$P[t'|t=2, a=O_S] = \begin{cases} 1, & \text{if } t'=5 \\ 0, & \text{otherwise}. \end{cases} \tag{16}$$

When the task is processed by the mobile device or the cloud (i.e., $3 \le t \le 5$), the state transition is determined by the service rates of the central cloud, the edge cloud, and the mobile device. We assume that the service rates of the central cloud, the edge cloud, and the mobile device for the task are μ_C, μ_E, and μ_S.[5] Then, the probabilities that the task processing is completed at the central cloud, the edge cloud, and the mobile device are $\mu_C\tau$, $\mu_E\tau$, and $\mu_S\tau$ [35]. Therefore, the corresponding state transition probabilities can be derived as

$$P[t'|t=3, a] = \begin{cases} \mu_C\tau, & \text{if } t'=6 \\ 1-\mu_C\tau, & \text{if } t'=3 \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

$$P[t_{k'}|t=4, a] = \begin{cases} \mu_E\tau, & \text{if } t'=7 \\ 1-\mu_E\tau, & \text{if } t'=4 \\ 0, & \text{otherwise}, \end{cases} \tag{18}$$

and

$$P[t_{k'}|t=5, a] = \begin{cases} \mu_S\tau, & \text{if } t'=8 \\ 1-\mu_S\tau, & \text{if } t'=5 \\ 0, & \text{otherwise}. \end{cases} \tag{19}$$

After the task processing is completed (i.e., $6 \le t \le 8$), t' is always 0. Therefore, we have

$$P[t'|6 \le t \le 8, a] = \begin{cases} 1, & \text{if } t'=0 \\ 0, & \text{otherwise}. \end{cases} \tag{20}$$

When the network topology and the residence time in each location are given, $P[L_j, C_{\chi'}|L_i, C_\chi]$ can be numerically obtained [29]. We assume that the residence time in L_i follows an exponential distribution with mean $1/\eta_i$. Then, the probability that a mobile device moves to another location is $\eta_i\tau$. Therefore, $P[L_j, C_{\chi'}|L_i, C_\chi]$ can be derived as

$$P[L_j, C_{\chi'}|L_i, C_\chi] = \begin{cases} P_{ij}\eta_i\tau, & \text{if } L_i^j = 1, \quad L_j \ne L_i \\ 1-\eta_i\tau, & \text{if } L_j = L_i \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

[5]The service rates are determined by the computation power of the computation node and the complexity of the task. Moreover, the background loads of each cloud (or mobile device) can influence the service rates. If the task cannot be processed in a certain computation node due to task properties (e.g., requirements for high computing power), the service rate of the computation node is set to 0.

where P_{ij} is the probability that the mobile device moves from location i to another location j. P_{ij} can be obtained by counting the number of movements between locations in the empirical data [36], [37].

Meanwhile, $P[r'|r, t]$ can be defined as follows. If there is no task (i.e., $t = 0$), r' remains Inf. On the other hand, when a task occurs (i.e., $t = 1$), r' is set to the deadline N_R. Therefore, the corresponding transition probabilities can be denoted by

$$P[r'|r, t=0] = \begin{cases} 1, \text{if } r' = Inf \\ 0, \text{ otherwise} \end{cases} \tag{22}$$

and

$$P[r'|r, t=1] = \begin{cases} 1, & \text{if } r' = N_R \\ 0, & \text{otherwise}. \end{cases} \tag{23}$$

When the processing of the task is not completed (i.e., $2 \le t \le 5$), the remaining deadline decreases until reaching 0 and then remains constant at 0 (i.e., $r' = 0$). Therefore, the corresponding transition probabilities can be given as

$$P[r'|r \ne 0, 2 \le t \le 5] = \begin{cases} 1, & \text{if } r' = r-1 \\ 0, & \text{otherwise} \end{cases} \tag{24}$$

and

$$P[r'|r = 0, 2 \le t \le 5] = \begin{cases} 1, & \text{if } r' = 0 \\ 0, & \text{otherwise}. \end{cases} \tag{25}$$

After the task processing is completed (i.e., $6 \le t \le 8$), the mobile device does not need to consider the remaining deadline since the mobile device initializes the remaining deadline. That is, r' is always Inf. Therefore, $P[r'|r, 6 \le t \le 8]$ can be defined as

$$P[r'|r, 6 \le t \le 8] = \begin{cases} 1, & \text{if } r' = Inf \\ 0, & \text{otherwise}. \end{cases} \tag{26}$$

E. REWARD AND COST FUNCTIONS

To define the reward and cost functions, we consider task completion within the deadline, the transmission cost, and the energy consumed to process the task in the mobile device. First, we define the total reward function, $r(s, a)$, as

$$r(s, a) = \omega_1 f(s, a) - (1-\omega_1) g(s, a), \tag{27}$$

where $f(s, a)$ and $g(s, a)$ are the reward function for task completion within the deadline and the cost function for the transmission cost and the energy consumption, respectively. ω_1 is a weighted factor to determine the importance of the reward and cost functions.

If task processing is completed (i.e., $6 \le t \le 8$) before the deadline (i.e., $r > 0$), a unit reward R_O can be obtained; thus, $f(s, a)$ is defined as

$$f(s, a) = \begin{cases} R_O, & \text{if } 6 \le t \le 8, \quad r > 0 \\ 0, & \text{otherwise}. \end{cases} \tag{28}$$

The cost function with respect to the transmission cost and the energy consumption, $g(s, a)$, can be written as

$$g(s, a) = \omega_2 g_T(s, a) + (1 - \omega_2) g_E(s, a), \quad (29)$$

where $g_T(s, a)$ is the cost function for transmitting the task to the cloud and receiving the result from the cloud. Meanwhile, $g_E(s, a)$ is the cost function for the energy consumption. In addition, ω_2 is a weighted factor to balance the two cost functions.

$g_T(s, a)$ can be defined as

$$g_T(s, a) = \omega_3 g_S(s, a) + (1 - \omega_3) g_R(s, a), \quad (30)$$

where $g_S(s, a)$ and $g_R(s, a)$ are the cost functions for transmitting the task to the cloud and receiving the result from the cloud, respectively. In addition, ω_3 is a weighted factor to balance $g_S(s, a)$ and $g_R(s, a)$.

The mobile device decides whether to offload to the cloud, process the task itself, or delay processing when the task is in the buffer (i.e., $t = 2$). When the task is offloaded to the central or edge cloud, there is a transmission cost; thus, $g_S(s, a)$ can be described by

$$g_S(s, a) = \begin{cases} \kappa_m \zeta_C, & \text{if } t = 2,\ a = O_C \\ \kappa_m \zeta_E, & \text{if } t = 2,\ a = O_E \\ 0, & \text{otherwise,} \end{cases} \quad (31)$$

where κ_m is a coefficient that reflects the transmission cost of the mth network. Intuitively, the mobile device prefers to use a wireless access network with lower transmission cost. Therefore, κ_m is determined as the coefficient of the wireless access network with the lowest transmission cost among the available wireless access networks. On the other hand, ζ_C (or ζ_E) is the transmission cost representing the distance between the central cloud (or the edge cloud) and the mobile device.

When the task is completed in the central cloud or the edge cloud (i.e., $t = 6$ or $t = 7$), the results must be returned to the mobile device. Therefore, $g_R(s, a)$ can be defined as

$$g_R(s, a) = \begin{cases} \kappa_m \zeta_C, & \text{if } t = 6 \\ \kappa_m \zeta_E, & \text{if } t = 7 \\ 0, & \text{otherwise.} \end{cases} \quad (32)$$

When the mobile device decides to process the task by itself (i.e., $a = O_S$), energy is consumed to handle the task. Note that this decision is made only when the task is in the buffer (i.e., $t = 2$); therefore, $g_E(s, a)$ can be written as

$$g_E(s, a) = \begin{cases} \epsilon, & \text{if } t = 2,\ a = O_S \\ 0, & \text{otherwise,} \end{cases} \quad (33)$$

where ϵ is the energy consumed to process the task in the mobile device.

F. OPTIMALITY EQUATION

To maximize the expected total reward and obtain the optimal policy, we choose the expected total discount reward optimality criterion [39], [40] as our objective function. Let $v(s)$ be the maximum expected total reward when the initial state is s. Then, we can describe $v(s)$ as

$$v(s) = \max_{\pi \in \Pi} v^{\pi}(s) \quad (34)$$

where $v^{\pi}(s)$ is the expected total reward when the policy π with an initial state s is given. Note that the expected total reward can be maximized when the mobile device takes the most beneficial action a, and such an optimal action a in each state s can be obtained by solving the formulated objective function.

The optimality equation is given by [30]

$$v(s) = \max_{a \in A} \left\{ r(s, a) + \sum_{s' \in S} \lambda P[s'|s, a] v(s') \right\} \quad (35)$$

where λ is a discount factor in the MDP model. λ closer to 1 gives greater weight to future rewards. The solution of the optimality equations corresponds to the maximum expected total reward and the optimal policy. To solve the optimality equation and to obtain the optimal policy δ, we use a value iteration algorithm, as shown in Algorithm 1, where $|v| = \max[v(s)]$ for $s \in \mathbf{S}$.

Algorithm 1 Value Iteration Algorithm

1: Set $v^0(s) = 0$ for each state s. Specify $\varepsilon > 0$, and set $\chi = 0$.

2: For each state s, compute $v^{\chi+1}(s)$ by
$$v^{\chi+1}(s) = \max_{a \in A} \left\{ r(s, a) + \sum_{s' \in S} \lambda P[s'|s, a] v^{\chi}(s') \right\}$$

3: If $|v^{\chi+1}(S) - v^{\chi}(S)| < \varepsilon(1 - \lambda)/2\lambda$, go to step 4. Otherwise, increase χ by 1 and return to step 2.

4: For each state $s \in \mathbf{S}$, compute the stationary optimal policy
$$\delta(s) = \arg\max_{a \in A} \left\{ r(s, a) + \sum_{s' \in S} \lambda P[s'|s, a] v^{\chi+1}(s') \right\}$$
and stop.

Generally, each iteration in the value iteration algorithm is performed in a polynomial time (i.e., $O(|A||S|^2)$) [41], [42]. Since this complexity cannot be neglected, the mobile device uses a table to store the optimal policy of where and when tasks are offloaded or processed (i.e., whether to offload to the central or edge cloud, process itself, or delay processing). This table includes the state and the decision in each state and can be pre-computed by the value iteration algorithm. In this way, ST-CODA can be applied to the mobile device without high computational overhead [43].

V. EVALUATION RESULTS

For the performance evaluation, we compare the proposed scheme, $S_{ST-CODA}$, with five schemes: 1) S_{CENT} where the mobile device always offloads its tasks to the central cloud; 2) S_{EDGE} where the mobile device always offloads its tasks to the edge cloud; 3) S_{MOBILE} where tasks are always processed at the mobile device; 4) S_{MINTX} where the mobile device offloads its tasks to the cloud that provides the lowest transmission cost; and 5) S_{NOHET} where the mobile device determines the optimal cloud without the consideration of the network heterogeneity (i.e., the mobile device does not choose the delay action).

In terms of network topology, we consider heterogeneous networks where Wi-Fi APs are sparsely deployed within a cellular area [38], [44]. In other words, $C_\chi = [c_1, c_2]$, where c_1 and c_2 indicate the availabilities of Wi-Fi and cellular networks, respectively, and P_{ij} is 1. Generally, users reside in the cellular networks for longer than in Wi-Fi networks [44]. Therefore, we conduct performance evaluations in various situations in which the average residence time in cellular networks is greater than that in Wi-Fi networks. However, different evaluation results show similar tendencies. Therefore, only a case where the average residence time in the cellular and WiFi networks, $1/\eta_{c_1}$ and $1/\eta_{c_2}$, are set to 10/6 and 10/4, respectively, is included in this paper. On the other hand, due to the open access property of Wi-Fi networks, we can assume that the coefficient κ_1 for Wi-Fi networks is smaller than κ_2 for cellular networks. Meanwhile, since the central cloud is located further from the mobile device than the edge cloud, ζ_C is larger than ζ_E. Even though extensive evaluations are conducted with several parameter settings of $\kappa_1 < \kappa_2$ and $\zeta_C > \zeta_E$, similar tendencies are observed; thus, only one case (i.e., $\kappa_1 = 2$, $\kappa_2 = 3$, $\zeta_C = 6$, and $\zeta_E = 1$) is included in this paper.

TABLE 2. Service rates of each cloud for each class task and inter arrival rate of each class task.

	task 1	task 2	task 3
μ_C	0.8	0.65	0.6
μ_E	0.5	0.4	0.1
μ_S	0.4	0	0
λ_T	0.3	0.3	0.3

Since tasks have different complexities, they should be processed by different criteria. To demonstrate this fact, we consider 3 representative task types: 1) tasks with low complexity (i.e., task 1); 2) tasks with moderate complexity (i.e., task 2); and 3) tasks with high complexity (i.e., task 3). The default service rates of the central cloud, the edge cloud, and the mobile device for each class and the inter-arrival rate of each task are summarized in Table 2. Note that the default service rates are determined by the complexity. In addition, $\mu = 0$ means that the task cannot be processed in that cloud. Meanwhile, ϵ is set to 1 for all tasks, all weighted factors (i.e., ω_1, ω_2, and ω_3) are set to 0.5, R_O is set to 10, and the time slot length τ is set to 1. For the value iteration algorithm, λ and ε are set to 0.98 and 0.001, respectively.

A. EFFECT OF R_O

Figure 4 shows the expected total reward as a function of the unit reward R_O that can be received when the task processing is completed before the deadline. The expected total reward of $S_{ST-CODA}$ is the highest among all comparison schemes.

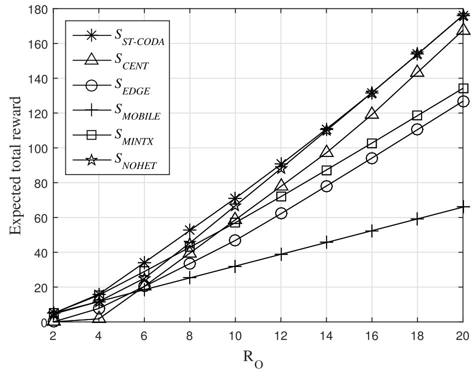


FIGURE 4. Effect of R_O.

This result can be explained as follows. $S_{ST-CODA}$ offloads tasks by considering the advantages and disadvantages of the computation nodes, as well as the transmission costs in heterogeneous wireless networks; that is, $S_{ST-CODA}$ offloads the task to an appropriate cloud with high probability that the task can be completely processed within the deadline. Specifically, when the mobile device is in a low-cost access network (i.e., a Wi-Fi network), it actively offloads its tasks to the cloud. Moreover, the mobile device can delay handling the task to fully utilize the advantages of heterogeneous networks when it expects to move to a low-cost access network within a short time. By contrast, S_{CENT}, S_{EDGE}, S_{MOBILE}, and S_{MINTX} have insufficient alternative actions (i.e., a subset of the whole action set); therefore, they cannot sufficiently utilize the advantages of the computation nodes' diversity and/or network heterogeneity.

B. EFFECT OF ζ_C

The effects of the transmission cost to the central cloud, ζ_C, are described in Figure 5. From these results, it can be shown that $S_{ST-CODA}$ operates adaptively even when ζ_C is changed. Specifically, when ζ_C is small (i.e., $\zeta_C = 1$), all tasks are offloaded to the central cloud.[6] This can be explained as follows. A small ζ_C means that the mobile device can offload the task to the central cloud with a low transmission cost. Moreover, when the task is offloaded to the central cloud, the probability that the task is processed within the deadline is the highest, which results in a higher expected total reward. Note that the central cloud has the greatest computation power. Meanwhile, when ζ_C is $3 \sim 6$, $S_{ST-CODA}$ offloads each task to an appropriate cloud or processes it in the mobile device; that is, since task 1 (i.e., a task with low complexity) can be processed in the mobile device within the deadline with high probability and there is no transmission cost when the task is processed in the mobile device, $S_{ST-CODA}$ processes the task in the mobile device (see Figure 5(b)).[7] By contrast, the probability that

[6]In Figure 5(b), (c), and (d), the expected total reward of $S_{ST-CODA}$ is the same with that of S_{CENT} when $\zeta_C = 1$.

[7]In Figure 5(b), when ζ_C is $3 \sim 6$, the expected total reward of $S_{ST-CODA}$ is the same as that of S_{MOBILE}.

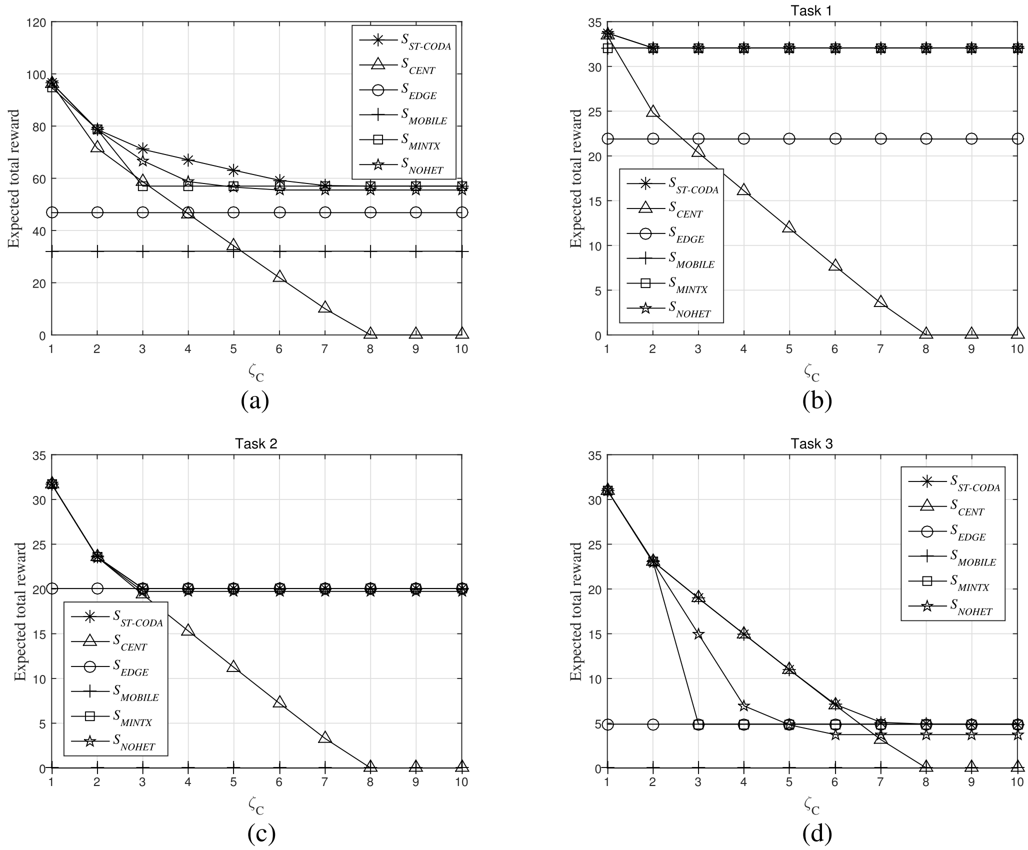


FIGURE 5. Effect of ζ_C. (a) For all task. (b) For task 1. (c) For task 2. (d) For task 3.

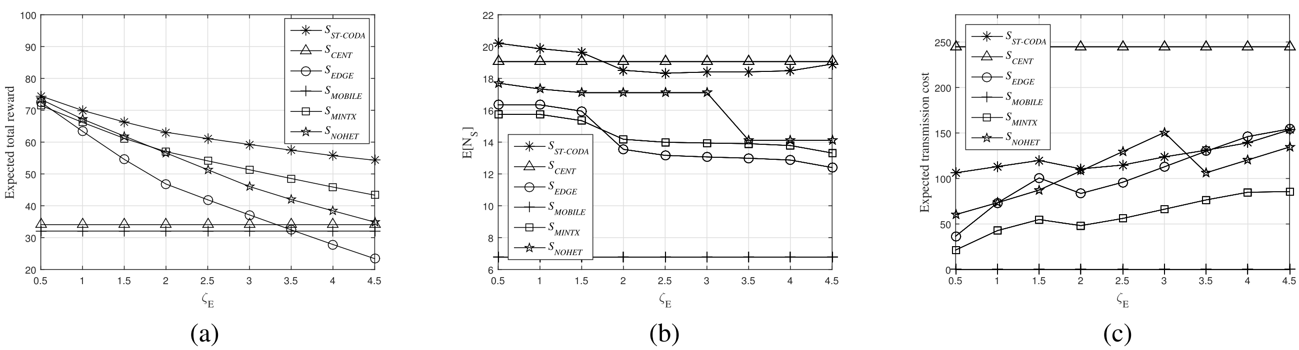


FIGURE 6. Effect of ζ_E. (a) Expected total reward. (b) $E[N_S]$. (c) Expected transmission cost.

tasks 2 and 3 (i.e., tasks with moderate and high complexity) cannot be processed in the mobile device within the deadline is high. Therefore, tasks 2 and 3 are offloaded to the edge and the central cloud, respectively (see Figure 5(c) and (d)). Meanwhile, when ζ_C is large (i.e., $7 \leq \zeta_C \leq 10$), $S_{ST-CODA}$ does not offload any tasks to the central cloud to reduce the transmission cost.

C. EFFECT OF ζ_E

Figure 6(a) shows the expected total reward as a function of the transmission cost to the edge cloud, ζ_E. In this result, ζ_C is set to 5. The expected total rewards of all schemes except S_{CENT} and S_{MOBILE} decrease as ζ_E increases, which can be explained as follows. A larger ζ_E indicates a higher transmission cost when the task is offloaded to the edge cloud. Moreover, in S_{CENT} and S_{MOBILE}, the mobile device does not offload any tasks to the edge cloud.

$S_{ST-CODA}$ has the smallest decreasing rate of the expected total reward among the schemes that can offload tasks to the edge cloud. This can be explained by Figure 6(b) and (c). Figure 6(b) and (c) represent the expected number of tasks that are successfully processed (i.e., the task processing is completed before the deadline), $E[N_S]$, and the expected transmission cost, respectively. In Figure 6(b), it can be seen that since $S_{ST-CODA}$ offloads the task to the appropriate cloud which can process the offloaded task within the deadline with high probability, $E[N_S]$ of $S_{ST-CODA}$ is maintained at a high level regardless of ζ_E, which can give the unit reward, R_O. Meanwhile, as shown in Figure 6(c), the increasing rate of the expected transmission cost of $S_{ST-CODA}$ is the smallest among the schemes that can offload tasks to the edge cloud. This is because the number of states where $S_{ST-CODA}$ does not use the edge cloud to offload tasks increases to reduce the transmission cost as ζ_E increases.

On the other hand, in Figure 6(a), it can be seen that the expected total reward of S_{EDGE} is smaller than that of S_{CENT} when ζ_E is greater than 3.5, which means that the advantage of the edge cloud disappears when the edge cloud is located farther than a certain distance from the mobile device. That is, the edge cloud should be installed sufficiently closer to the mobile device to obtain its effectiveness.

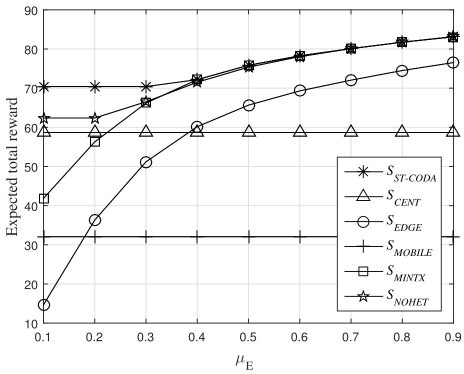


FIGURE 7. Effect of μ_E.

D. EFFECT OF μ_E

Figure 7 shows the expected total reward as a function of the service rate of the edge cloud, μ_E. It can be seen that the expected total rewards of all schemes except S_{CENT} and S_{MOBILE} increase as μ_E increases, which can be explained as follows. When the edge cloud has high computational power (i.e., μ_E is large), the probability that tasks offloaded to the edge cloud can be processed before their deadlines is high. Therefore, more rewards can be obtained when μ_E is large and tasks are offloaded to the edge cloud. Note that the cost to transmit tasks to the edge cloud is lower than the cost to transmit tasks to the central cloud. On the other hand, since S_{CENT} and S_{MOBILE} do not offload any tasks to the edge cloud, the expected total rewards of these schemes do not increase with increasing μ_E.

E. EFFECT OF ϵ

The effects of the energy consumed to process the task in the mobile device, ϵ, are illustrated in Figure 8. As ϵ increases, the cost to process the task in the mobile device increases. Therefore, in Figure 8(a), it can be seen that the expected total rewards of all the schemes, except S_{CENT} and S_{EDGE}, decrease with increasing ϵ.[8] However, the expected total reward of $S_{ST-CODA}$ is still the greatest, which can be explained by Figure 8(b). As ϵ increases, more tasks of $S_{ST-CODA}$ are not processed in the mobile device to reduce energy consumption. Specifically, when ϵ is large (i.e., $9 \leq \epsilon \leq 10$), $S_{ST-CODA}$ does not process any tasks in the mobile device; therefore, the expected energy consumption of $S_{ST-CODA}$ is 0 (see Figure 8(b)).

F. EFFECT OF ω

Figure 9 shows the effect of weighted factor ω on the expected total reward. From Figure 9(a), it can be shown that the expected total rewards of all schemes increase as ω_1 increases. This is because a large ω_1 means less transmission and energy consumption costs (see (27)).

Meanwhile, with the increase of ω_2, higher transmission cost is imposed (see (29)) and thus the decreased expected total rewards in all schemes except S_{MOBILE} are observed. Note that S_{MOBILE} does not offload any task to the external cloud and therefore no transmission cost occurs.

From Figure 9(c), it can be seen that a large ω_3 provides higher expected total reward in $S_{ST-CODA}$. This can be explained as follows. Even though a mobile device in $S_{ST-CODA}$ can transmit its tasks to the cloud when low-cost networks are available, it is not easy to predict a future available network to receive the result accurately. Hence, the result reception cost has negative impact on the expected total reward. On one hand, if ω_3 is large, the task transmission cost is more influential to the expected total reward than the result reception cost (see (30)). Consequently, the expected total reward of $S_{ST-CODA}$ increases with the increase of ω_3. On the other hand, since S_{MOBILE} processes their tasks locally and other comparison schemes (i.e., S_{CENT}, S_{EDGE}, S_{MINTX}, and S_{NOHET}) transmit their tasks to the external cloud immediately, their expected total rewards are constant regardless of ω_3.

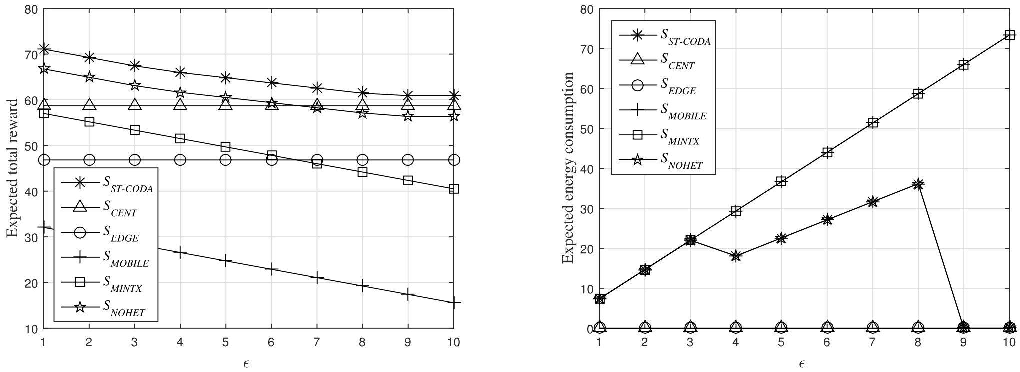
TABLE 3. Execution time (s).

Scheme \ λ	0.9	0.93	0.96	0.99
$S_{ST-CODA}$	0.173	0.200	0.311	1.095
S_{NOHET}	0.109	0.152	0.241	0.853

G. EFFECT OF λ

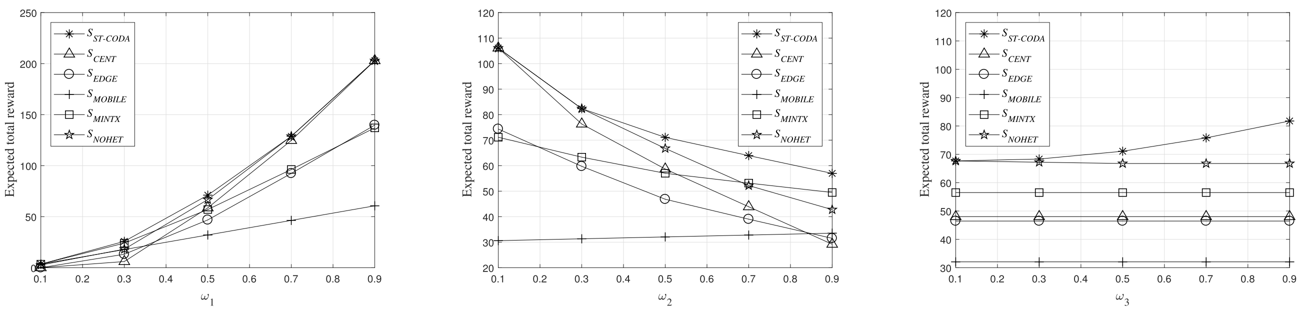
Table 3 shows the effect of the discount factor λ on the execution time to obtain the optimal policy through the value iteration algorithm. The value iteration algorithm is executed by Intel Core i7-7500U processor and the execution time is measured by using MATLAB R2017a. Note that comparison schemes except S_{NOHET} follow a fixed policy and they are not considered in the execution time comparison.

[8]Note that in S_{CENT} and S_{EDGE}, the mobile device does not process any tasks by itself.



(a) (b)

FIGURE 8. Effect of ϵ. (a) Expected total reward. (b) Expected energy consumption.



(a) (b) (c)

FIGURE 9. Effect of ω. (a) ω_1. (b) ω_2. (c) ω_3

From Table 3, it can be found that the execution times of $S_{ST-CODA}$ and S_{NOHET} increase as λ increases. This is because a larger discount factor means that the value iteration algorithm has to consider further future [30]. Meanwhile, S_{NOHET} determines the optimal policy without consideration of the network heterogeneity. Therefore, it can be seen that the execution time of S_{NOHET} is slightly shorter than that of $S_{ST-CODA}$. However, the mobile device can store its policy in a form of table, and thus the value iteration algorithm needs to be executed only once. After storing its policy in a table, the mobile device simply searches and takes an action matched to the current state. Note that the time complexity to search an element in the table is not high (e.g., $O(1)$ in a hash table and $O(|S|)$ in a binary search tree). Therefore, we believe that ST-CODA can be applied to the mobile device without high computational overhead.

VI. CONCLUSION

In this paper, we propose a spatial and temporal computation offloading decision algorithm (ST-CODA) in edge cloud-enabled heterogeneous networks. In ST-CODA, the mobile device determines where and when tasks are processed by considering the advantages and disadvantages of the computation nodes and the different transmission costs in heterogeneous networks. To optimize the performance of ST-CODA, a Markov decision process (MDP) problem is formulated, and the optimal policy for where and when to process tasks is obtained by a value iteration algorithm. Extensive evaluation results are given to demonstrate the effectiveness of ST-CODA with the optimal policy compared to alternative schemes in terms of the transmission cost, the energy efficiency of the mobile device, and the number of tasks processed before their deadline. In addition, ST-CODA operates adaptively, even the operating environment, such as the transmission cost and the service rate, changes. In our future work, we will extend the proposed scheme to consider task offloading between mobile devices by means of device to device (D2D) communication, and we will introduce an incentive mechanism to encourage mobile devices to process tasks.

REFERENCES

[1] W. Zhang, Y. Wen, and D. O. Wu, "Collaborative task execution in mobile cloud computing under a stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 81–93, Jan. 2015.

[2] K. Ha, Z. Chen, W. Hu, W. Richter, P. Pillai, and M. Satyanarayanan, "Towards wearable cognitive assistance," in *Proc. ACM Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, Jun. 2014, pp. 68–81.

[3] K. Lee *et al.*, "Outatime: Using speculation to enable low-latency continuous interaction for mobile cloud gaming," in *Proc. ACM Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, May 2015, pp. 151–165.

[4] M. Shahzad, A. Liu, and A. Samuel, "Secure unlocking of mobile touch screen devices by simple gestures—You can see it but you can not do it," in *Proc. ACM Int. Conf. Mobile Comput. Netw. (MobiCom)*, Sep./Oct. 2013, pp. 39–50.

[5] A. R. Khan, M. Othman, S. A. Madani, and S. U. Khan, "A survey of mobile cloud computing application models," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 393–413, Feb. 2014.
[6] X. Ma, Y. Zhao, L. Zhang, H. Wang, and L. Peng, "When mobile terminals meet the cloud: Computation offloading as the bridge," *IEEE Netw.*, vol. 27, no. 5, pp. 28–33, Sep./Oct. 2013.
[7] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Netw. Appl.*, vol. 18, no. 1, pp. 129–140, Feb. 2013.
[8] *Cisco System, Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are*. Accessed: Mar. 23, 2018. [Online]. Available: http://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf
[9] A. Damnjanovic *et al.*, "A survey on 3GPP heterogeneous networks," *IEEE Wireless Commun.*, vol. 18, no. 3, pp. 10–21, Jun. 2011.
[10] Amazon. *EC2*. Accessed: Mar. 23, 2018. [Online]. Available: https://aws.amazon.com/ec2/
[11] Microsoft. *Azure: Cloud Computing Platform and Services*. Accessed: Mar. 23, 2018. [Online]. Available: https://azure.microsoft.com/en-in/
[12] 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.
[13] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct. 2009.
[14] J. Li, K. Bu, X. Liu, and B. Xiao, "ENDA: Embracing network inconsistency for dynamic application offloading in mobile cloud computing," in *Proc. ACM Special Interest Group Data Commun. (SIGCOMM) Workshop Mobile Cloud Comput. (MCC)*, Aug. 2013, pp. 39–44.
[15] R. Shukla and A. Munir, "An efficient computation offloading architecture for the Internet of Things (IoT) devices," in *Proc. IEEE Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2017, pp. .
[16] D. Kovachev, T. Yu, and R. Klamma, "Adaptive Computation Offloading from Mobile Device into the Cloud," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl. (ISPA)*, Jul. 2012, pp. 728–731.
[17] L. Tong, Y. Li, and W. Gao, "A hierarchical edge cloud architecture for mobile computing," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.
[18] M. A. Puente, Z. Becvar, M. Rohlik, F. Lobillo, and E. C. Strinati, "A seamless integration of computationally-enhanced base stations into mobile networks towards 5G," in *Proc. IEEE Veh. Technol. Conf. (VTC) Spring*, May 2015, pp. 1–5.
[19] F. Lobillo *et al.*, "An architecture for mobile computation offloading on cloud-enabled LTE small cells," in *Proc. IEEE Workshop Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2014, pp. 1–6.
[20] Y. Zhang, D. Niyato, P. Wang, and C. Tham, "Dynamic offloading algorithm in intermittently connected mobile cloudlet systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 4190–4195.
[21] D. T. Hoang, D. Niyato, and P. Wang, "Optimal admission control policy for mobile cloud computing hotspot with cloudlet," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 1–11.
[22] Y. Geng, W. Hu, Y. Yang, W. Gao, and G. Cao, "Energy-efficient computation offloading in cellular networks," in *Proc. IEEE Int. Conf. Netw. Protocols (ICNP)*, Nov. 2015, pp. 145–155.
[23] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.
[24] R. Wolski, S. Gurun, C. Krintz, and D. Nurmi, "Using bandwidth data to make computation offloading decisions," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. (IPDPS)*, Apr. 2008, pp. 1–8.
[25] T. Truong-Huu, C.-K. Than, and D. Niyato, "To offload or to wait: An opportunistic offloading algorithm for parallel tasks in a mobile cloud," in *Proc. IEEE Int. Conf. Cloud Comput. Technol. Sci. (CloudCom)*, Dec. 2014, pp. 182–189.
[26] T. Zhao, S. Zhou, X. Guo, and Z. Niu, "Tasks scheduling and resource allocation in heterogeneous cloud for delay-bounded mobile RDGE computing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.
[27] L. Tang and X. Chen, "An efficient social-aware computation offloading algorithm in cloudlet system," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.
[28] H. Al-Shatri, S. Muller, and A. Klein, "Distributed algorithm for energy efficient multi-hop computation offloading," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.
[29] H. Ko, G. Lee, D. Suh, S. Pack, and X. Shen, "An optimized and distributed data packet forwarding in LTE/LTE-A networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3462–3473, May 2016.
[30] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 1994.
[31] E. A. Feinberg and A. Shwartz, *Handbook of Markov Decision Processes: Methods and Applications*. Norwell, MA, USA: Kluwer, 2002.
[32] J. Krumm, "A Markov model for driver turn prediction," in *Proc. Soc. Autom. Eng. (SAE) World Congr.*, Apr. 2008, pp. 1–7.
[33] H. S. Dhillon, R. K. Ganti, F. Baccelli, and J. G. Andrews, "Modeling and analysis of K-tier downlink heterogeneous cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 550–560, Apr. 2012.
[34] A. Ghosh *et al.*, "Heterogeneous cellular networks: From theory to practice," *IEEE Commun. Mag.*, vol. 50, no. 6, pp. 54–64, Jun. 2012.
[35] T. Guo, A. Ul Quddus, N. Wang, and R. Tafazolli, "Local mobility management for networked femtocells based on X2 traffic forwarding," *IEEE Trans. Veh. Technol.*, vol. 62, no. 1, pp. 326–340, Jan. 2013.
[36] S. Pack and Y. Choi, "Fast handoff scheme based on mobility prediction in public wireless LAN systems," *IEE Proc.-Commun.*, vol. 151, no. 5, pp. 489–495, Oct. 2004.
[37] L. Zhang, B. Yu, and J. Pan, "GeoMob: A mobility-aware geocast scheme in metropolitans via taxicabs and buses," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr./May 2014, pp. 1279–1287.
[38] H. Ko, J. Lee, and S. Pack, "An opportunistic push scheme for online social networking services in heterogeneous wireless networks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 2, pp. 416–428, Jun. 2017.
[39] J. Pan and W. Zhang, "An MDP-based handover decision algorithm in hierarchical LTE networks," in *Proc. IEEE Veh. Technol. Conf. (VTC-Fall)*, Sep. 2012, pp. 1–5.
[40] H. Tabrizi, G. Farhadi, and J. Cioffi, "A learning-based network selection method in heterogeneous wireless systems," in *Proc. IEEE Global Telecommun. Conf. (Globecom)*, Dec. 2011, pp. 1–5.
[41] E. Stevens-Navarro, Y. Lin, and V. W. S. Wong, "An MDP-based vertical handoff decision algorithm for heterogeneous wireless networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 2, pp. 1243–1254, Mar. 2008.
[42] M. Littman, T. Dean, and L. Kaelbling, "On the complexity of solving Markov decision problems," in *Proc. Int. Conf. Uncertainty Artif. Intell. (UAI)*, Aug. 1995, p. 1.
[43] M. Abu Alsheikh, D. T. Hoang, D. Niyato, H.-P. Tan, and S. Lin, "Markov decision processes with applications in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1239–1267, 3rd Quart., 2015.
[44] S. Pack, G. Park, and H. Ko, "An SIP-based location management framework in opportunistic WiFi networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 11, pp. 5269–5274, Nov. 2015.

HANEUL KO received the B.S. and Ph.D. degrees from the School of Electrical Engineering, Korea University, Seoul, South Korea, in 2011 and 2016, respectively. From 2016 to 2017, he was a Post-Doctoral Fellow in mobile network and communications, Korea University. He is currently with the Smart Quantum Communication Research Center, Korea University, where he is also a Visiting Post-Doctoral Fellow with The University of British Columbia, Vancouver, BC, Canada. His research interests include 5G networks, mobility management, mobile cloud computing, SDN/NFV, and future Internet.

JAEWOOK LEE received the B.S. degree from Korea University, Seoul, South Korea, in 2014. He is currently pursuing the integrated M.S. and Ph.D. degree with the School of Electrical Engineering, Korea University. His research interests include 5G networks, mobility management, and SDN/NFV.

SANGHEON PACK (SM'11) received the B.S. and Ph.D. degrees in computer engineering from Seoul National University, Seoul, South Korea, in 2000 and 2005, respectively. From 2005 to 2006, he was a Post-Doctoral Fellow with the Broadband Communications Research Group, University of Waterloo, Waterloo, ON, Canada. In 2007, he joined the Faculty of Korea University, where he is currently a Full Professor with the School of Electrical Engineering. His research interests include future internet, softwarized networking (SDN/NFV), information-centric networking/delay tolerant networking, and vehicular networks. He was a recipient of the IEEE/Institute of Electronics and Information Engineers Joint Award for IT Young Engineers Award in 2017, the Korean Institute of Information Scientists and Engineers Young Information Scientist Award in 2017, the Korean Institute of Communications and Information Sciences Haedong Young Scholar Award in 2013, the LG Yonam Foundation Overseas Research Professor Program in 2012, and the IEEE ComSoc APB Outstanding Young Researcher Award in 2009. He served as the TPC Chair for EAI Qshine 2016, a Publication Co-Chair for the IEEE INFOCOM 2014 and ACM MobiHoc 2015, a Co-Chair for IEEE VTC 2010-fall transportation track, a Co-Chair for IEEE WCSP 2013 Wireless Networking Symposium, and a Publicity Co-Chair for IEEE SECON 2012. He is an Editor of the *Journal of Communications Networks*, *IET Communications*, and he is a Guest Editor of the IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING.