A Privacy-preserving and Energy-efficient Offloading Algorithm based on Lyapunov Optimization

Lu Chen, Hongbo Tang*, Yu Zhao, Wei You and Kai Wang
Information Engineering University, Zhengzhou 450002, China
[e-mail: luchenmec@163.com, tahobo@sina.com]
*Corresponding author: Hongbo Tang

Received December 30, 2021; revised March 13, 2022; revised April 18, 2022; accepted August 4, 2022; published August 31, 2022

Abstract

In Mobile Edge Computing (MEC), attackers can speculate and mine sensitive user information by eavesdropping wireless channel status and offloading usage pattern, leading to user privacy leakage. To solve this problem, this paper proposes a Privacy-preserving and Energy-efficient Offloading Algorithm (PEOA) based on Lyapunov optimization. In this method, a continuous Markov process offloading model with a buffer queue strategy is built first. Then the amount of privacy of offloading usage pattern in wireless channel is defined. Finally, by introducing the Lyapunov optimization, the problem of minimum average energy consumption in continuous state transition process with privacy constraints in the infinite time domain is transformed into the minimum value problem of each timeslot, which reduces the complexity of algorithms and helps obtain the optimal solution while maintaining low energy consumption. The experimental results show that, compared with other methods, PEOA can maintain the amount of privacy accumulation in the system near zero, while sustaining low average energy consumption costs. This makes it difficult for attackers to infer sensitive user information through offloading usage patterns, thus effectively protecting user privacy and safety.

Keywords: Mobile edge computing, Computing offloading, Usage pattern, Privacy protection, Lyapunov optimization.

http://doi.org/10.3837/tiis.2022.08.002 **ISSN : 1976-7277**

1. Introduction

With the rapid development of science and technology, mobile terminal devices carry more functions and roles. Due to the limited computing power and storage capacity of mobile devices, which cannot meet the needs of some emerging applications (e.g., auto-driving vehicular networks [1] and augmented reality (AR) [2]), Mobile Edge Computing (MEC) [3] has emerged. MEC offers computing power guarantee for the three types of scenarios in 5G era, enhanced Mobile BroadBand (eMBB), massive Machine Type of Communication (mMTC), as well as Ultra-Reliable and Low Latency Communications (URLLC).

However, the localized deployment of MEC expose them to the edge of network, they facilitate attackers to attack MEC physical facilities and mine user privacy. In addition, MEC allows operators to open their wireless access networks to authorized third parties through MEC for channel listening and illegal access, resulting in security threats such as privacy leakage and information tampering [4]. For example, malicious users can use illegal access methods (e.g.,untrustworthy service providers) to silently monitor offloading patterns of specific users to infer or track their privacy [5]. In particular, since the patterns of computation offloading are highly correlated with personal characteristics, such as the time characteristics of user terminals generating computation tasks are strongly correlated with individual characteristics, attackers can mine privacy and infer sensitive user information by eavesdropping on wireless channels. Therefore, offloading methods with privacy protection need to be designed to avoid attackers from mining user privacy through channel eavesdropping.

Existing research on privacy protection in MEC is relatively small and mostly focuses on traditional privacy security issues such as authentication [6], trust security [7], intrusion detection [8]. Therefore, in this work, a privacy-preserving and energy-efficient offloading algorithm (PEOA) method based on Lyapunov optimization is designed to protect offloading usage pattern. The approach seeks to maintain a low average energy cost while providing privacy protection. The main contributions of this work are as follows. 1) An offloading model for continuous state MEC systems with privacy constraints is developed based on a Markov Decision Process (MDP) by defining the amount of privacy. 2) An offloading method PEOA based on Lyapunov drift considering privacy preservation and energy cost is designed. 3) Finally, the experiments are conducted to verify the effectiveness of the method.

2. Related Work

In this section, we briefly introduce the privacy protection techniques in MEC.

In the study of MEC systems, a large amount of research work has focused on studying the problem of optimizing dynamic resource allocation. For example, in [9], a dual time-frame associative offloading resource allocation scheme is designed with the goal of reducing latency, and different resource allocation policies are specified for users at each time slot by Lyapunov's online optimization algorithm. In [10], resource allocation algorithms that provide high quality services by maintaining queue length stability are investigated. A series of game-theory-based resource scheduling approaches in multi-access edge computing are outlined in [11]. However, these studies mainly elaborated on the optimization problem of resource allocation on energy consumption and latency, with little

attention paid to the security issues in the offloading process.

As for the offloading security issues, MEC privacy protection techniques in MEC are less studied and mainly focus on traditional security protection methods, such as data encryption and access control. Research on privacy security strategies related to the offloading process focused on data masking, i.e., hiding the real information of users against background knowledge attacks during data distribution. For example, in [12], the authors obfuscate the offloading pattern by adding fake tasks to defend against private information detected by attackers through eavesdropping. In [13], the authors used differential privacy based on Voronoi diagram to scramble the data so that attackers cannot easily obtain the location information, thus protecting privacy. In [14], a privacy-preserving computation offloading method is proposed by changing user offloading frequency based on k-anonymity. In [15], the authors reduce the relevance of task offloading and users by increasing the cache hit rate or randomly selecting edge servers to protect user privacy and security. In [16], the authors propose a two-phase offload optimization strategy to improve the reliability of the system by jointly optimizing the resource utilization efficiency and privacy goals of edge computing units. However, the above studies did not to analyze the correlation between offloading usage pattern and channel status in continuous dynamic channel status. Since users have to minimize energy consumption when offloading through wireless channel, they will generally perform MEC offloading when the channel status is good and resort to local computation processing when the channel status is poor. The average channel gain is highly correlated with the distance between users and the MEC server, and hence attackers may easily locate users and infer their sensitive information [17]. Therefore, this paper characterizes the amount of privacy accumulation of offloading usage pattern using the statistical information observed by attackers in the wireless channel status. The concept of amount of privacy is proposed to show the correlation between offloading usage pattern observed through channel eavesdropping and original offloading usage pattern. When the channel status is good while users choose to offload, amount of privacy will increase; conversely, when the channel status is poor while users choose to offload, amount of privacy will drop because this breaks the original offloading pattern in general, which makes it more difficult for attackers to infer sensitive information. In particular, in [18], the impact of different scheduling strategies on delay performance under different delay requirements in 5G wireless networks is discussed. Here, we exploited the buffer queue of user terminals to change the original offloading behavior patterns with delay constraint in order to limit the amount of privacy accumulation in MEC nodes, which can ensure user privacy. Since MDP is more adaptable in dynamic environments, it has been applied in algorithms for optimal scheduling of MEC systems [19-20] and energy consumption analysis [21]. Therefore, in this paper, a PEOA method based on Lyapunov optimization in continuous dynamic channel status is proposed. Specifically, an MEC offloading model based on Markov process with privacy constraints is built first. Then, by introducing the Lyapunov optimization, the problem of minimum average energy consumption in continuous state transition process with privacy constraints in the infinite time domain is transformed into the minimum value problem of each timeslot, which reduces the complexity of the algorithms and helps obtain the optimal solution. The experimental results show that PEOA can maintain the amount of privacy accumulation in the system near zero, while sustaining low average energy consumption costs, which makes it difficult for attackers to infer sensitive information through offloading usage pattern, thus effectively protecting user privacy and safety.

3. System model and privacy issues

3.1 System model

It is assumed that the system model in the MEC consists of three parts: the user terminal, the wireless channel, and the MEC server, as shown in **Fig. 1**. The number of tasks generated by users in each timeslot is $d_n \in \{0,1,...,\mathrm{d}_{\max}\}$, and each task contains M bits. At each timeslot, users may specify the offloading policy based on the channel status and their own decision,

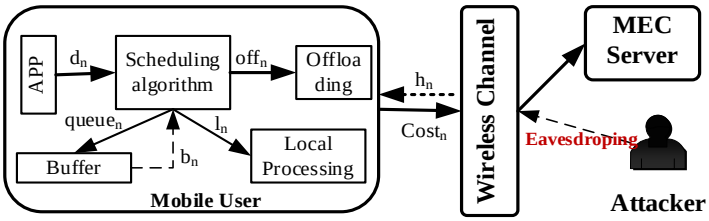

Fig. 1. MEC system model with buffer.

and this process is considered as a MDP [22]. The channel status admits a second-order Markov process $\mathrm{H} \in \{0,1\}$ whose transition probability is $P_H\{h_{n+1} \mid h_n\}$. The status of each timeslot is $S_n = \{d_n, h_n, b_n\}$, where $\{h_n\}_{n\geq 0}$ denotes the status of the communication channel between the user terminal and the MEC server, $d_n \in \{0,1,...,\mathrm{d}_{\max}\}$ denotes the number of tasks generated in n-th timeslot randomly selected within the value range, and $b_n \in \{0,1,...,\mathrm{b}_{\max}\}$ means the number of tasks in a certain timeslot buffer. The offloading policy for each timeslot is $a_n = \{off_n, queue_n, l_n\}, a_n \in \pi$, where $off_n \in [0, d_{\max} + b_{\max}]$ denotes the number of tasks offloaded to the MEC server, $queue_n \in [0, b_{\max}]$ represents the number of tasks buffered, and l_n denotes the number of tasks processed locally. According to the offloading policy of a timeslot, the next state is obtained as $S_{n+1} = \{d_{n+1}, h_{n+1}, b_{n+1}\}$, where $b_{n+1} = queue_n$. Therefore, the cost of taking the action policy a_n in state S_n is given by:

$$Cost(S_n, a_n) = w_q \cdot queue_n + E_n(off_n, h_n) + E_l(l_n) \tag{1}$$

In (1), w_q denotes the relative importance of user terminal time delay for energy consumption, E_n represents the energy consumption for offloading to the MEC server and E_l denotes the energy consumption for local processing. Based on the above model, the average energy consumption in the initial state S_0 of offloading policy π is given by:

$$\overline{Cost_{\pi,s_0}} = \lim_{n\to+\infty} \frac{1}{n} \mathrm{E}_n[\sum_{t=1}^{n} Cost(S_t, a_t) \mid S_0] \tag{2}$$

In the above equation, E_n denotes the expected value of the costs incurred by the corresponding offloading policy. Therefore, the system state is a finite-state Markov process under the initial state S_0 of the offloading strategy π, and the optimal solution can be found through the MDP framework [22]. The main symbols and descriptions involved in the article are shown in **Table 1**.

Table 1. Main symbols and description.

Symbol	Description
H	Channel status with good or poor
d_n	The number of tasks generated in timeslot n with an upper bound of $d_{\max}$
h_n	The channel status in timeslot n
b_n	The number of tasks in timeslot n in buffer with an upper bound of $b_{\max}$
S_n	The status of timeslot n in MEC system
$P_H\{h_{n+1} \mid h$	Channel status transition probability from timeslot n to $n+1$
a_n	Offloading policy in timeslot n
a_n^*	The optimal offloading policy in timeslot n
off_n	The number of tasks offloaded to the MEC server in timeslot n
$queue_n$	The number of tasks buffered in mobile user in timeslot n
l_n	The number of tasks processed locally in timeslot n
π	The set of all possible strategies for a_n
w_q	The relative importance of user terminal time delay for energy consumption
q_n	The privacy contained under strategy a_n in timeslot n
E_n	The energy consumption for offloading to the MEC server in timeslot n
E_l	The energy consumption for local processing
S_0	The initial state of MEC system
$Cost(S_n, a$	The cost of taking the action policy a_n in state S_n
$\overline{Cost_{\pi,s_0}}$	The average energy consumption in the initial state S_0 of offloading policy π
$\Gamma_{\{\cdot\}}$	An indicator function
$Task_{fake}$	The number of fake task generated by user
$Q_{\pi,s_0}(n)$	The amount of privacy accumulation at timeslot n with the initial state S_0 of offloading policy π
V	Lyapunov adjustment parameter
$L(n)$	Lyapunov function in timeslot n
$\Delta L(n)$	Lyapunov drift
θ	Privacy threshold
$\Lambda_{\pi,S_0}(n)$	The optimal solution at timeslot n with the initial state S_0 of offloading policy π

3.2 Privacy issues

Since MEC systems transition tasks over wireless channels, malicious attackers can infer sensitive information about users by eavesdropping on the channel state, thus posing privacy threats. For example, in general, user offloading behaviors obeys this rule: when the channel status is good ($h_n=1$), all the generated tasks are chosen to offload to the MEC servers; when the channel status is poor ($h_n=0$), all the generated tasks are chosen to be processed locally, as shown in **Fig. 2**. Attackers can obtain users' offloading characteristics by listening to the channel state, and speculate sensitive information, thus causing privacy leakage.

Therefore, we use the offloading behavior that eavesdroppers may tap through the wireless channel as a privacy feature.

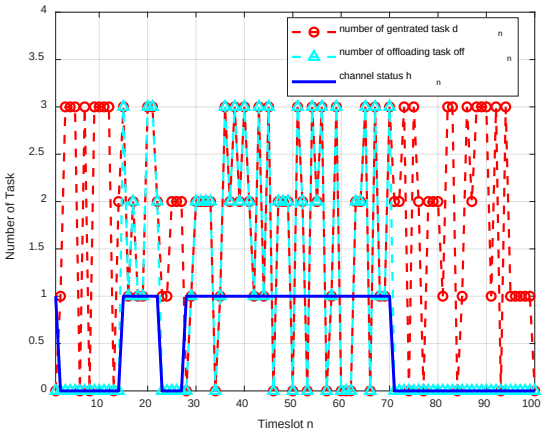

Fig. 2. Task offloading strategy without privacy constraints.

The amount of privacy contained in action strategy a_n taken in a certain timeslot $S_{n\text{-}1}$ is given by (3).

$$q_n(S_{n\text{-}1}, a_n) = off_n \cdot \Gamma_{\{h_n=1\}} - off_n \cdot \Gamma_{\{off_n>0\}} \cdot \Gamma_{\{h_n=0\}} \tag{3}$$

Here, $\Gamma_{\{\cdot\}}$ is an indicator function. In (3), the user terminal needs to appropriately reduce the number of tasks that are offloaded to MEC server when the channel status is good in order to protect privacy. At the same time, it is necessary to upload tasks to MEC server appropriately when the channel condition is poor, which can be achieved by buffering tasks that satisfy the delay constraint. Breaking the original user offloading rule, this new strategy protects user privacy by reducing the amount of privacy accumulation that attackers can monitor through channel eavesdropping, making it more difficult to mine sensitive user information. In particular, the amount of privacy can also be reduced by randomly generating fake tasks if the current privacy accumulation exceeds a threshold θ while the newly generated task for that timeslot is 0 and the buffer is 0. Here, θ is a positive constant and its magnitude indicates the stringency of the user's privacy requirements. The smaller θ indicates a more stringent privacy requirement.

In the specific implementation, false tasks can be achieved by user-initiated randomly generated offload request tasks, where $Task_{fake} \in [1, d_{\max}]$ and $off_n = Task_{fake}$. The fake tasks only perform offloading requests and do not occupy the computing resources of MEC servers. Thus they are not the complete task computing offloading process, so as to reduce the energy consumption. Therefore, the amount of privacy accumulation at timeslot n with the initial state S_0 of offloading policy π is given by (4):

$$Q_{\pi,s_0}(n) = \sum_{t=1}^{n} q_t(S_t, a_t) \mid S_0 \tag{4}$$

4. Proposed solution

In this section, we first construct the drift function by introducing the Lyapunov function and prove the stability of the drift; then, we equilibrate the relationship between energy consumption and privacy by introducing the parameter V; finally, we design a privacy-preserving and energy-cost offloading method based on Lyapunov optimization.

4.1 Lyapunov optimization

According to MEC system model and the privacy amount determination method, the PEOA model in the initial state S_0 following strategy π at the user terminal is given by:

$$\begin{aligned} &\min_{\pi,S_0} \lim_{n\to+\infty} \frac{1}{n} E_n[\sum_{t=1}^{n} Cost(S_t,a_t) \mid S_0] \\ &s.t. \quad a: Q_{\pi,S_0}(n) \to 0 \ \& \ |Q_{\pi,S_0}(n)| \le \theta \\ &\quad b: \forall a_n = \{off_n, queue_n, l_n\} \in \pi \mid \xi(n) \\ &\quad c: 0 \le d_n \le d_{\max} \\ &\quad d: 0 \le off_n \le (d_{\max} + b_{\max}) \\ &\quad e: 0 \le queue_n \le b_{\max} \\ &\quad f: b_{n+1} = queue_n \end{aligned} \tag{5}$$

Equation (5) represents the objective of minimizing the average energy cost based on the simultaneous incorporation of privacy constraints. Equation (5a) describes the privacy constraints; Equation (5b) represents the offloading strategies that users can adopt under the time delay constraint; Equations (5c), (5d), and (5e) represent the constraints of each indicator function of the user terminal; and Equation (5f) clarifies the status transition relationship. When the amount of privacy accumulated by eavesdroppers is closer to zero, the less likely users' sensitive information will be mined. The objective function is a MDP. If the function is solved using the MDP framework, transforming the system into finite deterministic states can be complex, and a large amount of storage space is required at the terminal to preserve the optimal policy space for each status.

Therefore, the Lyapunov optimization method [23] is used in this paper for the optimal solution. This method can transform the minimum average energy consumption problem in the continuous state transition process with privacy constraints in the infinite time domain into the minimum value problem for each timeslot, seeking the optimal solution by simplify the algorithm. This paper considers $Q_{\pi,S_0}(n)$ the amount of privacy accumulation that can be monitored by attackers at the timeslot n, as a virtual queue, which leads to (6):

$$Q_{\pi,S_0}(n) = Q_{\pi,S_0}(n-1) + off_n \cdot \Gamma_{\{h_n=1\}} - off_n \cdot \Gamma_{\{off_n>0\}} \cdot \Gamma_{\{h_n=0\}} \tag{6}$$

The Lyapunov function is added to describe the stability of the queue, and for each timeslot n,

$$L(n) = \frac{1}{2} Q_{\pi,S_0}(n)^2 \tag{7}$$

Then the Lyapunov drift of two adjacent timeslots is given by:

$$\begin{aligned}\Delta L(n) &= L(n+1) - L(n) \\ &= \frac{1}{2}(off_n \cdot \Gamma_{\{h_n=1\}} - off_n \cdot \Gamma_{\{off_n>0\}} \cdot \Gamma_{\{h_n=1\}})^2 \\ &\quad + Q_{\pi,S_0}(n) \cdot (off_n \cdot \Gamma_{\{h_n=1\}} - off_n \cdot \Gamma_{\{off_n>0\}} \cdot \Gamma_{\{h_n=1\}}) \\ &\le B + Q_{\pi,S_0}(n) \cdot (off_n \cdot \Gamma_{\{h_n=1\}} - off_n \cdot \Gamma_{\{off_n>0\}} \cdot \Gamma_{\{h_n=1\}}) \\ &= \tilde{\Delta} L(n)\end{aligned} \tag{8}$$

Here, B is a positive constant, with an upper bound of $off_n^{\,2}/2$, and $\tilde{\Delta} L(n)$ has an approximate value of $\Delta L(n)$. To avoid focusing only on drift minimization and ignoring energy consumption, which may cause all tasks offloaded locally, the adjustment parameter V is added. With energy consumption as a penalty mechanism, V is used to equilibrate between privacy and energy consumption. Further, the optimal solution for the offloading decision in each timeslot is obtained, as shown in (9):

$$\Lambda_{\pi,S_0}(n) : \min_{n\to\infty} \tilde{\Delta} L(n) + V \cdot Cost(S_n, a_n) \tag{9}$$

Here, V is a positive constant. The size of V indicates how much each timeslot is penalized for energy consumption. The larger the V, the lower the average energy costs. Larger V means that more amounts of privacy accumulation $Q_{\pi,S_0}(n)$ cannot be stabilized near zero. Therefore, V needs to be determined in line with the specific channel status and user requirements, so as to strike a balance between energy consumption and privacy constraints.

4.2 PEOA design

To further prove the PEOA method based on Lyapunov optimization, the PEOA algorithm is designed. The specific algorithm description is shown below.

The objective of the algorithm proposed in this paper is to find the offloading strategy for each timeslot for a given MEC system status while minimizing the average energy consumption and protecting privacy. In each timeslot n, alternative offloading policies a_n are first determined based on the system state S_0 (**Step 1**). Then, the amount of privacy accumulation and drift function are calculated according to (1) and (3) (**Step 2**). When the privacy accumulation is less than the threshold or the privacy accumulation exceeds the threshold meanwhile the number of newly generated tasks or the number of buffer queuing tasks in this timeslot is not zero, the optimal offloading policy a_n^* is determined according to the Lyapunov drift function (**Step 3**). If the amount of privacy exceeds the threshold and the number of newly generated tasks and the number of buffer queuing tasks are both zero for that timeslot, random fake tasks are generated to reduce privacy accumulation (**Step 4**). Finally, the action policy is recorded and the system state is updated (**Step 5**). Thus, PEOA can minimize the average energy consumption of the MEC system in a continuous Markov status by finding the optimal offloading policy at each timeslot. Experiments show that average costs converge when the timeslot reaches $a_n = a_n^*$.

Algorithm 1: PEOA

Initialization:

1) Initialize the system state S_0 and action strategy space a_n and generate channel status and corresponding tasks by timeslot.

2) Set iteration parameters n, privacy constraints θ, and MEC system model communication parameters, e.g. , $d_{\max}$, $b_{\max}$, w_q.

Repeat Iterations:

Step 1: Obtain alternative action strategy combinations based on system status.

Step 2: The amount of privacy accumulation and the drift function are calculated based on the combination of each action strategy.

Step 3: If the current privacy accumulation is less than the threshold or if the current privacy accumulation exceeds the threshold meanwhile the number of newly generated tasks in that timeslot or the buffer queuing tasks is not zero, the optimal action policy selected according to (9) is the policy $a_n = a_n^*$.

Step 4: If the current privacy accumulation exceeds the threshold and both the number of newly generated tasks and buffer queuing tasks for that timeslot are zero, then random fake tasks are generated at that timeslot $a_n = \{Task_{fake}, 0, 0\}$.

Step 5: Record action strategies and update system state $b_{n+1} = queue_n$.

End

The algorithm will terminate when the index reaches the maximum number of iterations, and return to the selected action strategy.

4.3 Complexity Analysis

In this section, we analyze the complexity of the PEOA algorithm.

In the MEC system, we assume that the maximum number of spatial states brought by the policy action a_n is $S_{\max}$, and the maximum number of possible policy actions under the state space S_n is $a_{\max}$. Then the complexity of the Basic MDP algorithm in timeslot t would be $O(S_{\max} * a_{\max} * t)$. However, with the introduction of Lyapunov, the range of alternative policy action space can be narrowed by calculating the minimum value of timeslot drift, and the problem of minimum average cost in finite time domain is transformed into the problem of minimum value per timeslot. In this case, the complexity of PEOA algorithm would be $O(a_{\max} * t)$ which can effectively reduce the complexity of the algorithm and save the computational space.

5. Numerical results

5.1 Experimental design

In this section, to corroborate the proposed algorithm, Matlab 2018b is employed for numerical simulations. In the simulation analysis, the transition probability of the channel status is assumed to be $P_H(h_{n+1} = 1 \mid h_n = 1) = P_H(h_{n+1} = 0 \mid h_n = 0) = 0.95 / 0.5$. When users perform computational offloading, the resulting energy consumption $E_n(off_n, h_n) = off_n \cdot e_{h_n}$, where $e_{h_n} \in \{e_{h_n=0}, e_{h_n=1}\}$ represents the transition energy consumption of a single task at a certain timeslot in the channel status e_{h_n}. Assuming that a single task size $M = 500K$ bits,

bandwidth $B = 5MHz$, the transition time for each task is $T_{off} = 0.1s$, time delay tolerance per task $\xi(n)$ is 2 timeslots, and the values of channel power gain to noise power $N_0 \cdot B$ in the case of good and poor channel status are 0.2/0.05. Based on $e_{off}(n) = p_{h_n}(n) \cdot T_{off} = (2^{M/(B \cdot T_{off})} - 1) \cdot (B \cdot N_0) / h_n^2 \cdot T_{off}$, there is $e_{h_n=1} = 0.5J$, $e_{h_n=0} = 2J$. Furthermore, energy consumption for local computing $E_l(l_n) = l_n \cdot e_l$, where e_l indicates the energy consumption for processing a single task locally. It is assumed that the local CPU operating frequency $f_l = 2GHz$, energy consumption factor $\kappa = 1 \times 10^{-27}$ [24], local energy consumption $e_l = Task \cdot \eta / f_l \times (\kappa \cdot f_l^3) = 1J$, where $\eta = 500 cycles/bit$ [25]. The queuing weight factor is used to adjust users' sensitivity to latency and energy consumption, and in this experiment this value is set as $w_q = 0.8$, $S_0 = \{1,1,0\}$. In the experiment, we assume that $d_{\max} = 3, b_{\max} = 4$, and timeslot $t = 10^4$. The experimental setup is shown in **Table 2**.

To validate the proposed algorithm, three methods were selected for comparison, Basic, Naive and Chaff [12]. In the Basic method, users choose the least energy consuming method for local computation or computational offloading at each timeslot without considering privacy constraints; in the Naive method, users find the least energy consuming offloading method without exceeding privacy constraints on the basis of the Basic algorithm; in the Chaff algorithm, users reduce the amount of privacy accumulation by adding false tasks, where the cost of setting up a single fake task is $e_{fake} = 2J$ and that of dropping a task is $e_{drop} = 5J$. All experiments were judged by the mean value of the results of 100 Monte Carlo runs.

5.2 Results analysis

In the numerical analysis, **Fig. 3** first shows the actual offloading strategy adopted by the user in the PEOA approach proposed in this paper. Subsequently, **Figs. 4** to **7** validate the effectiveness of the algorithm in this paper in terms of privacy accumulation, average energy consumption, stability of the drift function, and variation in the values of the parameter *V*.

Table 2. Experiment Settings

Symbol	Description	Value
$P_H(h_{n+1}=1 \mid h_n=1)$	The transition probability of the channel status from $h_n=1$ to $h_{n+1}=1$	0.95/0.5
$P_H(h_{n+1}=0 \mid h_n=0)$	The transition probability of the channel status from $h_n=0$ to $h_{n+1}=0$	0.95/0.5
M	A single task size	500K bits
B	Bandwidth	5 MHz
T_{off}	The transition time of offloading	0.1s
$\xi(n)$	User time delay tolerance	2 timeslot
f_l	The local CPU operating frequency	2GHz
κ	Energy consumption coefficient	1×10^{-27} [24]
η	The number of CPU cycles required to process 1 bit of data	500 cycles/bit [25]

e_l	The energy consumption for processing a single task locally	1J
$h_n^{\ 2}/(N_0 \cdot B)$	The channel power gain to noise power $N_0 \cdot B$ ratio	0.2/0.05
$e_{h_n=1}$	The transition energy consumption of a single task at timeslot n in $h_n = 1$	0.5J
$e_{h_n=0}$	The transition energy consumption of a single task at timeslot n in $h_n = 0$	2J
w_q	The relative importance of user terminal time delay for energy consumption	0.8
S_0	The initial state of MEC system	{1,1,0}
$d_{\max}$	The upper bound of task number generated per timeslot	3
$b_{\max}$	The upper bound of buffer	4
t	The experiment time slot	10^4
e_{fake}	The cost of setting up a single fake task	2J
e_{drop}	The cost of dropping a task	5J

Fig. 3 illustrates the offloading policy of users under the PEOA algorithm ($d_{\max}=3, b_{\max}=4$). Compared to the Basic algorithm, PEOA offloads more tasks to the MEC server to protect user privacy when the channel status $h_n = 0$, such as the timeslot between $t = 60s$ and $t = 70s$. When the channel status $h_n = 1$, users select tasks to be computed locally, such as the timeslot between $t = 20s$ and $t = 30s$. This differs from the original offloading method in that not all computations are performed locally when $h_n = 0$ and all computations are offloaded when $h_n = 1$.

Fig. 4 presents the changes in the amount of privacy accumulation $Q_{\pi,S_0}(t)$ for the four algorithms when the timeslot is varied ($\theta = 10,\ \mathrm{V} = 10$). As can be seen from the figure, the privacy accumulation in the Basic method is much higher than the privacy threshold (according to the right coordinate), which facilitates attackers to infer user information and thus increases the risk of user privacy leakage; the privacy accumulation in the Naive and Chaff algorithms is concentrated around the threshold (according to the left coordinate), while that of PEOA floats up and down in a smaller range around 0, which means that it can protect user privacy by adjusting the offloading or making tasks processed locally.

To further illustrate the energy consumption of these algorithms under privacy constraints, **Fig. 5** is presented to show the changes in the four algorithms ($\theta = 10,\ \mathrm{V} = 10$). As can be seen, with the increase of timeslot, the average energy consumption of all four algorithms stabilizes at $t = 10^4$. Except for the Basic algorithm which does not consider privacy constraints, the PEOA algorithm has the lowest average energy consumption among the other three algorithms, saving about 62.4% and 25.2% compared to Naive and Chaff methods. PEOA algorithm can keep lower average energy consumption costs while offering privacy protection.

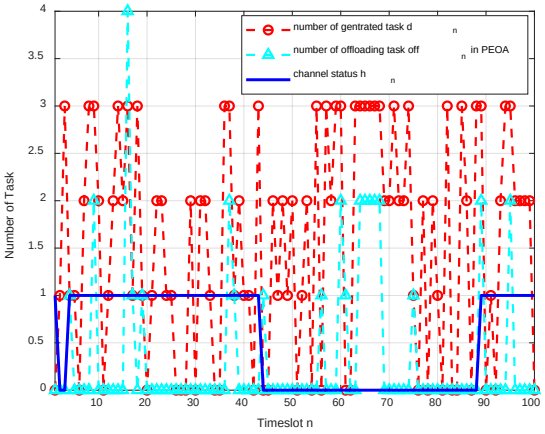

Fig. 3. Actual offloading decisions with privacy constraints.

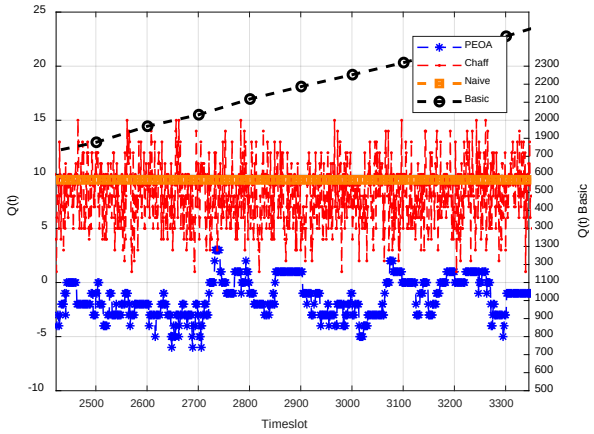

Fig. 4. Changes in the amount of privacy accumulation.

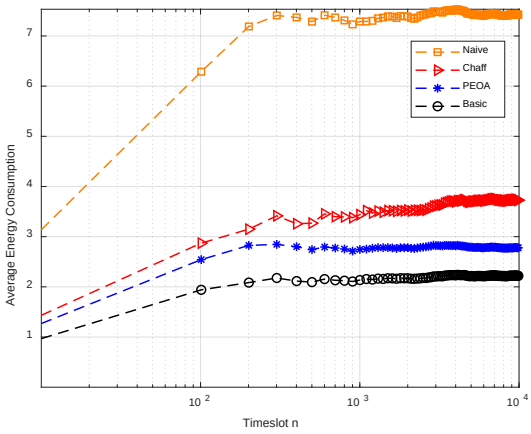

Fig. 5. Average energy consumption analysis.

Table 3. Analysis of false task and drop task strategies with different privacy thresholds

Privacy thresholds	False task strategies	Drop task strategies
$\theta = 6$ (PEOA)	0.0041	0.0357
$\theta = 6$ (Chaff)	0.0357	0.4162
$\theta = 10$ (PEOA)	0.0035	0.0308
$\theta = 10$ (Chaff)	0.0456	0.4204
$\theta = 50$ (PEOA)	0	0
$\theta = 50$ (Chaff)	0.0453	0.4467

Table 3 shows the proportions of false task strategies and drop task strategies for both PEOA and Chaff algorithms under different privacy threshold constraints. When the privacy threshold is low; in other words, when the privacy constraint is more stringent, the false task rate and drop task rate of PEOA are lower than those of Chaff algorithm. This is due to the fact that PEOA can reduce the privacy accumulation and satisfy the privacy constraint by selecting buffer tasks with the optimal policy within the latency constraint by user terminal. When privacy constraint is more relaxed, the false task rate and task drop rate of PEOA can be reduced to 0. The analysis also explains why average energy consumption of PEOA is low when the privacy constraint is relaxed.

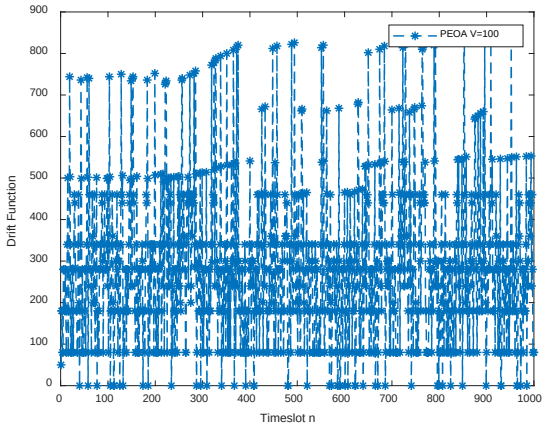

Fig. 6. Drift function analysis.

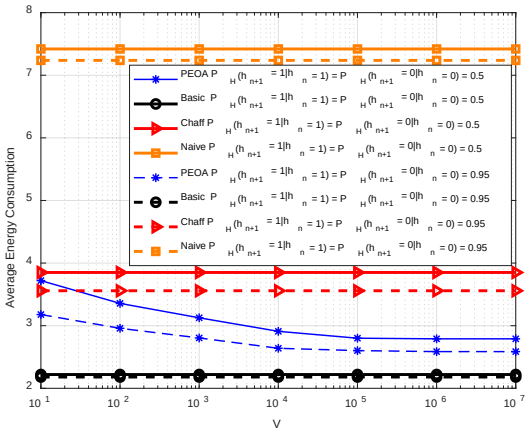

Fig. 7. Parameter V analysis.

Fig. 6 illustrates how the drift function $\Lambda_{\pi,S_0}(n)$ varies with timeslot ($V=100$). As can be seen, the value of $\Lambda_{\pi,S_0}(n)$ is fixed within a certain area whenever the timeslot changes. This result verifies the existence of the upper bound of (9) and indirectly verifies the stability of the Lyapunov drift. **Fig. 7** shows the changes in the average energy consumption with parameter V for different channel status when $t=10^4$ ($P_H(h_{n+1}=1|h_n=1)=P_H(h_{n+1}=0|h_n=0)=0.95/0.5$). In different channel states, the average energy consumption first decreases as V increases; when the value of V is greater than 10^5, the average energy consumption will not vary with V, but will remain at a stable stale. At the same time, the increase of V affects the amount of privacy accumulation, so that it cannot be stabilized near 0. Therefore, the size of V can be chosen within the range of $V<10^5$ based on energy consumption and privacy requirements. Moreover, PEOA can provide stable privacy protection and keep low energy consumption costs in different channel status.

6. Conclusion

In this paper, a PEOA method based on Lyapunov optimization is designed in response to the privacy leakage that may arise from behavioral characteristics of user computing offloading in MEC scenarios. Upon defining the amount of privacy, the method reduces the probability that attackers obtain sensitive user information through channels eavesdropping by changing the relationship between different channel status and offloading usage decisions. This paper exploits the buffer queue strategy of user terminals and integrates energy consumption overhead so as to limit the amount of privacy at MEC nodes and protect user privacy. Simulation experiments show that the PEOA algorithm can effectively protect user privacy while ensuring low energy consumption.

Acknowledgement

This work was supported by National Natural Science Foundation of China (No. 61941114) and (No.61801515).

References

[1] Z. Xiao, X. Dai, H. Jiang, D. Wang, H. Chen, L. Yang, and F. Zeng, “Vehicular task offloading via heat-aware mec cooperation using game-theoretic method,” *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2038-2052, 2020. Article (CrossRef Link)

[2] X. Liu and Y. Deng, “Learning-based prediction, rendering and association optimization for MEC-enabled wireless virtual reality (vr) network,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 10, pp. 6356-6370, 2021. Article (CrossRef Link)

[3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” *IEEE Internet of Things Journa*l, vol.5, no.1, pp. 450-465, 2018. Article (CrossRef Link)

[4] R. Roman, J. Lopez, and M. Mambo, “Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges,” *Future Generation Computer Systems*, vol. 78, pp. 680-698, 2018. Article (CrossRef Link)

[5] J. Yang and T. Johansson, "An overview of cryptographic primitives for possible use in 5G and beyond," *Science China Information Sciences*, vol. 63, no. 12, pp. 5-26, 2020. Article (CrossRef Link)
[6] K. Kaur, S. Garg, G. Kaddoum, M. Guizani, and D. N. K. Jayakody, "A lightweight and privacy-preserving authentication protocol for mobile edge computing," in *Proc. of the 2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa Village, USA, pp. 1-6, 2019. Article (CrossRef Link)
[7] Y. Li, X. Wang, X. Gan, H. Jin, L. Fu, and X. Wang, "Learning-aided computation offloading for trusted collaborative mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 12, pp. 2833-2849, 2020. Article (CrossRef Link)
[8] T. Gopalakrishnan, D. Ruby, F. Al-Turjman, D. Gupta, I. V. Pustokhina, D. A. Pustokhin, and K. Shankar, "Deep learning enabled data offloading with cyber attack detection model in mobile edge computing systems," *IEEE Access*, vol. 8, pp. 185938-185949, 2020. Article (CrossRef Link)
[9] C. F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4132-4150, 2019. Article (CrossRef Link)
[10] S. W. Ko, K. Han, and K. Huang, "Wireless networks for mobile edge computing: spatial modeling and latency analysis," *IEEE Transactions on Wireless Communications*, vol. 17, no. 8, pp.5225-5240, 2018. Article (CrossRef Link)
[11] J. Moura, and D. Hutchison, "Game theory for multi-access edge computing: Survey, use cases, and future trends," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 260-288, 2019. Article (CrossRef Link)
[12] T. He, E. N. Ciftcioglu, S. Wang, and K. S. Chan, "Location privacy in mobile edge clouds: A chaff-based approach," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp.2625-2636, 2017. Article (CrossRef Link)
[13] M. Bi, Y. Wang, Z. Cai, and X. Tong, "A privacy-preserving mechanism based on local differential privacy in edge computing," *China Communications*, vol. 17, no. 9, pp. 50-65, 2020. Article (CrossRef Link)
[14] X. ZHAO, J. PENG, W. YOU, and L. CHEN, "A Privacy-preserving Computation Offloading Method Based on k-Anonymity," *Journal of Electronics & Information Technology*, vol. 43, no.4, pp.892-899, 2021. Article (CrossRef Link)
[15] H. Ko, H. Lee, T. Kim, and S. Pack, "LPGA: Location Privacy-Guaranteed Offloading Algorithm in Cache-Enabled Edge Clouds," *IEEE Transactions on Cloud Computing*, pp. 1-1, 2020. Article (CrossRef Link)
[16] X. Xu, C. He, Z. Xu, L. Qi, S. Wan, and M. Z. A. Bhuiyan, "Joint optimization of offloading utility and privacy for edge computing enabled IoT," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2622-2629, 2020. Article (CrossRef Link)
[17] X. He, R. Jin, and H. Dai, "Deep PDS-learning for privacy-aware offloading in MEC-enabled IoT," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4547-4555, 2019. Article (CrossRef Link)
[18] Y. Zhong, T. Q. Quek, and X. Ge, "Heterogeneous cellular networks with spatio-temporal traffic: Delay analysis and scheduling," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6 pp. 1373-1386, 2017. Article (CrossRef Link)
[19] X. He, J. Liu, R. Jin, and H. Dai, "Privacy-aware offloading in mobile-edge computing," in *Proc. of the IEEE Global Communications Conference*, Singapore, pp. 1-6, 2017. Article (CrossRef Link)
[20] J. Xu, and S. Ren, "Online learning for offloading and auto scaling in renewable-powered mobile edge computing," in *Proc. of the 2016 IEEE Global Communications Conference (GLOBECOM)*, Washington, DC USA, 2016. Article (CrossRef Link)
[21] X. Ge, B. Yang, J. Ye, G. Mao, C. X. Wang, and T. Han, "Spatial spectrum and energy efficiency of random cellular networks," *IEEE Transactions on Communications*, vol. 63, no. 3, pp.1019-1030, 2015. Article (CrossRef Link)

[22] S. Huang, B. Lv, and R. Wang, "Mdp-based scheduling design for mobile-edge computing systems with random user arrival," in *Proc. of the 2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa Village, USA, pp.1-6, Dec, 2019. Article (CrossRef Link)
[23] Y. Cui, V. K. Lau, R. Wang, H. Huang, and S. Zhang, "A survey on delay-aware resource control for wireless systems—Large deviation theory, stochastic Lyapunov drift, and distributed stochastic learning," *IEEE Transactions on Information Theory*, vol. 58, no. 3, pp. 1677-1701, 2012. Article (CrossRef Link)
[24] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay trade off in multi-user mobile-edge computing systems," in *Proc. of the 2016 IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, USA, pp. 1-6, 2016. Article (CrossRef Link)
[25] A. P. Miettinen, and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proc. of the 2nd USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 10)*, 2010. Article (CrossRef Link)

Lu Chen received the B.S. and M.S. degrees from Information Engineering University, China, in 2011 and 2014 respectively. She is currently working towards the Ph.D degree in Information Engineering University. Her research interest includes mobile communication network security, mobile edge computing security technology.

Hongbo Tang received the B.E. degree from Huazhong University of Science and Technology in 1989, and M.E. degree in Engineering from National University of Defense Technology in 1995. He is now a professor of Information Engineering University, China. His current interest includes Next-generation on mobile communication and mobile communication network security.

Yu Zhao received the M.S. and Ph.D degrees in Information and Communication Systems from Information Engineering University, China. He is now a lecturer of Information Engineering University. His current interest includes Next-generation on mobile communication and mobile communication network security.

Wei You received the M.S. and Ph.D degrees in cryptology from Information Engineering University, China. He is currently an associate professor of Information Engineering University. His major research interests include New-generation mobile communication systems, and mobile communication network security.

Kai Wang received the B.S., M.S. and Ph.D degrees from Information Engineering University, China, in 2002, 2005 and 2019 respectively. He is currently an instructor of Information Engineering University. His major research interests include cyberspace Security and network Data Processing.