sensors

MDPI

Article

An IoT-Oriented Offloading Method with Privacy Preservation for Cloudlet-Enabled Wireless Metropolitan Area Networks

Zhanyang Xu [1,2], Renhao Gu [1,2], Tao Huang [3], Haolong Xiang [4], Xuyun Zhang [4], Lianyong Qi [5,*] and Xiaolong Xu [1,2,*]

[1] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China; zhanyang_xu@nuist.edu.cn (Z.X.); renhaogu@gmail.com (R.G.)
[2] Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] School of Computer Science and Technology, Silicon Lake College, Suzhou 215332, China; nuisthuangtao@163.com
[4] Department of Electrical and Computer Engineering, University of Auckland, Auckland 1023, New Zealand; hlx6700@gmail.com (H.X.); xuyun.zhang@auckland.ac.nz (X.Z.)
[5] School of Information Science and Engineering, Qufu Normal University, Rizhao 276826, China
* Correspondence: lianyongqi@gmail.com (L.Q.); njuxlxu@gmail.com (X.X.); Tel.: +86-177-6814-3062 (X.X.)

Received: 16 August 2018; Accepted: 9 September 2018; Published: 10 September 2018

Abstract: With the development of the Internet of Things (IoT) technology, a vast amount of the IoT data is generated by mobile applications from mobile devices. Cloudlets provide a paradigm that allows the mobile applications and the generated IoT data to be offloaded from the mobile devices to the cloudlets for processing and storage through the access points (APs) in the Wireless Metropolitan Area Networks (WMANs). Since most of the IoT data is relevant to personal privacy, it is necessary to pay attention to data transmission security. However, it is still a challenge to realize the goal of optimizing the data transmission time, energy consumption and resource utilization with the privacy preservation considered for the cloudlet-enabled WMAN. In this paper, an IoT-oriented offloading method, named IOM, with privacy preservation is proposed to solve this problem. The task-offloading strategy with privacy preservation in WMANs is analyzed and modeled as a constrained multi-objective optimization problem. Then, the Dijkstra algorithm is employed to evaluate the shortest path between APs in WMANs, and the nondominated sorting differential evolution algorithm (NSDE) is adopted to optimize the proposed multi-objective problem. Finally, the experimental results demonstrate that the proposed method is both effective and efficient.

Keywords: IoT data; cloudlet; WMAN environment; privacy preservation

1. Introduction

1.1. Background

A Wireless Metropolitan Area Network (WMAN) is a kind of mobile broadband wireless network, launched as a computer communication network within a city, which provides users with more convenient wireless services [1]. Metropolitan areas have high-density populations, where there are intensive data produced by the mobile devices in people's daily lives. Mobile cloud computing provides a novel paradigm that allows the computing tasks and the data from the mobile devices to be offloaded to the remote cloud for processing and storage through access points (APs) in the WMAN [2]. With the increasing number of mobile devices and the rapid growth of mobile cloud computing technology, the WMAN draws a great development via its excellent features in transmission speed, coverage area and so on [3].

The rapid development of mobile and network technologies has resulted in the emergence of various intelligent mobile devices, including smartphones, tablets, Kindles, and so on [4]. However, the capacity of a mobile device is often limited by its battery, size and weight. Also, the mobile devices have fewer communication and computing resources compared with the desktop computers. Consequently, these devices are unable to perform the computing tasks that require high levels of processing and long working time. In order to solve this problem, mobile cloud computing technology is introduced, which migrates complex computing tasks for execution from the mobile devices to the cloud system [5,6]. In recent years, Internet of Things (IoT) technology has been developed rapidly, and puts forward improved requirements for task transmission delay [7,8].

Wearable technology, smart homes, smart medicals, smart car networks and so on are all parts of IoT technology which are connected to the sensors, controllers, machines, people and things via local networks or other communication technologies [9–11]. However, most mobile applications hosted in the mobile devices have vast data and high requirements for real-time processing [12]. Specifically, the distance between the cloud and the mobile devices is relatively long [13]. Therefore, the process of offloading may bring about high delay [14–16]. Such delay is intolerable for the mobile applications that require strict response time, such as HD live video, natural language processing and face recognition [17,18]. Thus, the cloudlet is hired to solve the above problems. Cloudlet is a cloud service enhancement technology which is composed by some small "clouds", deployed on the edge of mobile devices [19].

Generally, there are many APs in the WMAN environment that are employed by the mobile users to access the cloudlets for efficient cloud services [20]. When the data in the mobile devices is transmitted to the cloudlet, it can be processed and the results can be returned efficiently [21]. However, with the increasing amount of IoT data, the problem of privacy leaks is getting worse [22]. Privacy preservation refers to the scenario where a cloudlet should not contain privacy conflicting information. When offloading the computing tasks with privacy conflicts on the same cloudlet, the privacy information of the mobile user, including home address, telephone and financial details, may be easily hacked [23]. Currently, some researchers use data encryption to deal with such security issues, but they were unable to deal with the troubles when the data were used legitimately [24].

On the other hand, the energy consumption and the transmission time are the important parts to evaluate the quality of IoT-oriented offload methods [25,26]. If the IoT data is encrypted, it will cost much more time in data processing and transmission, since the size of the data is greater. At the same time, more energy will be required for bigger data scales [27]. Considering the cloudlet environment under WMANs, the utilization of resources needs to be taken into consideration to avoid the overload or underload of the cloudlets. The resources in the cloudlet are provided in the form of virtual machines (VMs), which are assigned to multiple tasks elastically and on demand [28].

1.2. Motivation

Currently, a QoS (Quality of Service) aware cloudlet load balancing method for IoT data in WMAN is proposed [29], but it does not take into consideration the data privacy preservation. As a result, some important IoT data leak easily, causing great loss to users. Therefore, the privacy preservation of the IoT data in cloudlet-based WMAN environments has become an urgent problem. Hence, we consider a separated data offloading for the IoT data which has privacy conflicts. Meanwhile, an IoT-oriented offloading method (IOM) with privacy preservation is proposed to lower the transmission time, save energy consumption and improve the resource utility.

1.3. Paper Contributions

The main contributions of this paper include the following:

- Construct a systematic model of the resource utilization, the energy consumption and the data transmission time of the cloudlets when offloading the IoT data to the cloudlets.
- Adopt the Dijkstra algorithm to calculate the shortest path between AP points in a WMAN in order to reduce the transmission time of data.
- Optimize the multi-objective problem model by the nondominated sorting differential evolution (NSDE) algorithm with privacy preservation considered, and finally the optimal offloading strategies are output.
- Conduct extensive experimental evaluations and comparison analysis to demonstrate the efficiency and effectiveness of the proposed method.

The rest of the paper is organized as follows. Section 2 introduces the system model and the problem definition. Section 3 proposes an IoT-oriented offloading method with privacy preservation. In Section 4, simulation experiments and a comparison analysis are presented. Section 5 reviews related works. Finally, conclusions and future work are drawn in Section 6.

2. System Model and Problem Formulation

In this section, we present a system model that closely approximates the cloudlet environment in the WMAN first. Then, three optimized models of IoT data, that is, the transmission time model, the energy consumption model and the resource utilization model, are formed. Some key notations and descriptions used in the paper are listed in Table 1.

Table 1. Key notations and descriptions.

Terms	Descriptions
C	The cloudlet collection
A	The AP(Access Point) collection
T	The computing task collection
D	The dataset collection of the computing tasks
X	The data offloading policy collection for T
P	The number of computing tasks
$l_n(X)$	The resource utilization rate of the cloudlet c_n
$\sigma(X)$	The number of the occupied cloudlets
$\psi(X)$	The average of $l_n(X)$
$TT(X)$	The propagation delay time of the computing tasks
$T(X)$	The average of propagation delay time
$\beta_{P,n}(X)$	The execution time of the task x_p in the cloudlet c_n
$ST(X)$	The maximal execution time of the task in the cloudlet c_n
$E_{VM}^{idle}(X)$	The energy consumption of the idle VMs (Virtual Machines)
$E_{VM}^{active}(X)$	The energy consumption of the active VMs
$E_c(X)$	The energy consumption of the cloudlets
$E(X)$	The total energy consumption

2.1. Resource Model

In this paper, we focus on the IoT-oriented data offloading with privacy preservation for the cloudlet-enabled WMAN. We consider a separated data offloading for the IoT data with privacy conflicts to optimize the transmission time, the energy consumption and the resource utilization. Suppose that there are N cloudlets, denoted as $C = \{c_1, c_2, \ldots, c_N\}$ (each cloudlet has one host), which are deployed in the WMAN. There are M APs, denoted as $A = \{a_1, a_2, \ldots, a_M\}$. The cloudlets are connected through the APs, and $M > N$. P computing tasks that should be offloaded to the cloudlets for processing are donated as $T = \{t_1, t_2, \ldots, t_P\}$. The datasets of the computing tasks are donated as $D = \{d_1, d_2, \ldots, d_P\}$.

Let $X = \{x_1, x_2, \ldots, x_P\}$ be the offloading policy for the IoT data of the computing task set T, where $x_{P\ \bar{I}}\in C$ ($p = \{1, 2, \ldots, P\}$) is the cloudlet that the computing task t_P is offloaded to.

Figure 1 shows an example of cloudlet layout in the WMAN. The cloudlets are connected by the APs and deployed in the WMAN. There are seven APs, three physical machines, three cloudlets and three computing tasks in the example. The cloudlet c_1 is connected to the cloudlet c_2 through four APs, named a_1, a_2, a_3 and a_4. If the IoT data of the computing task t_1 has privacy conflicts with the other data, t_1 will be migrated to the cloudlet c_2 for processing, through a_1, a_2, a_3 and a_4. In addition, the cloudlet c_1 is connected to the cloudlet c_3 through three APs, named a_1, a_6 and a_7 or through APs of a_1, a_5, a_6 and a_7.

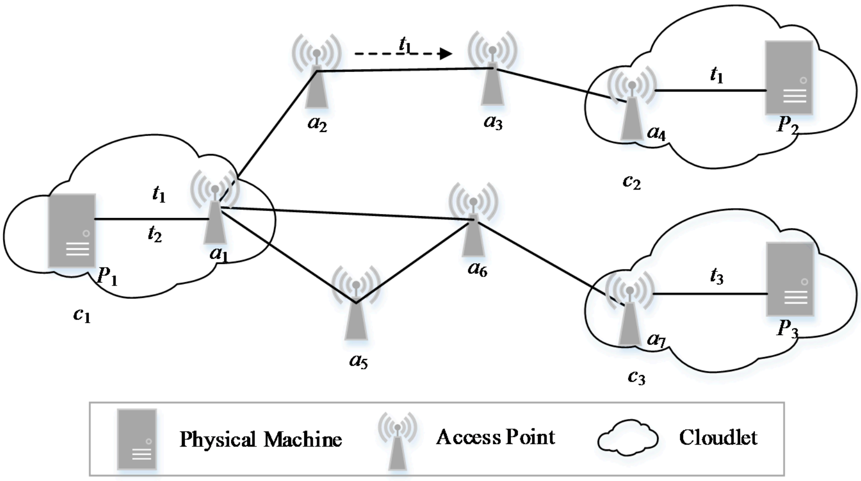


Figure 1. An example of cloudlet layout in the wireless metropolitan area network.

2.2. Resource Utilization Model

Compared to the mobile devices, the cloudlets have much greater physical resources including storage resources, computing resources and communication resources. When allocating the cloudlet resources to accommodate the IoT data, the resources are provided in the form of VMs. Assume that a cloudlet owns one physical machine. Let h_n be the capacity of the n-th cloudlet c_n and let $u_{p,n}$ be the requirements of the computing task t_p.

The resource utilization is an important part to evaluate the efficiency of the cloudlets. According to the data offloading policy in X, the resource utilization of the cloudlet c_n is $l_n(X)$, which is calculated by

$$l_n(X) = \frac{1}{h_n}\sum\nolimits_{p=1}^{P} u_{p,n} \cdot \theta_{p,n}(X), \tag{1}$$

where $\theta_{p,n}$ is a binary variable to judge whether t_p is offloaded on c_n, which is measured by

$$\theta_{p,n}(X) = \begin{cases} 1, \text{ if } c_n = x_p, \\ 0, \text{ Otherwise.} \end{cases} \tag{2}$$

Then, the number of the occupied hosts, denoted as $\sigma(X)$, is calculated by

$$\sigma(X) = \sum\nolimits_{p=1}^{P} \sum\nolimits_{n=1}^{N} \theta_{p,n}(X). \tag{3}$$

Finally, the average resource utilization of the cloudlet can be calculated by

$$\psi(X) = \frac{1}{\sigma(X)}\sum\nolimits_{n=1}^{N} l_n(X). \tag{4}$$

2.3. Data Transmission Model

The IoT data needs to be offloaded to the cloudlet for processing or the conflict data needs to be offloaded to another cloudlet. During the offloading, there is some transmission delay, which may influence the efficiency of the cloudlets. The transmission time for the IoT applications to offload datasets should be taken into account.

When the computing tasks are in the current cloudlet without offloading, the transmission delay is neglected. When the computing tasks are offloaded from one cloudlet to another, it will pass through multiple APs. Thus the transmission delay is added by the transmission time among APs.

Let $N_{p,q}$ be the number of halfway APs when the computing tasks are offloaded from the cloudlet c_p to c_q. Then, the transmission delay is calculated by

$$TT(X) = \begin{cases} 0, \text{ if } x_p = c_n, \\ (2 \cdot \frac{D_p}{\iota} + \frac{D_p}{\kappa} \cdot (N_{p,q} - 1)) \cdot \theta_{p,n}(X), \text{ Otherwise,} \end{cases}, \tag{5}$$

where D_p is the data scale of t_p. Then, the average data transmission time $T(X)$ can be calculated by

$$T(X) = \frac{1}{P} \cdot \sum_{p=1}^{P} \sum_{n=1}^{N} TT(X). \tag{6}$$

2.4. Energy Consumption Model

In this paper, the energy consumption is generated by the cloudlets, the active VMs and the idle VMs. The energy consumption is associated with the execution time of the computing tasks. Let $\beta_{P,n}(X)$ be the execution time of x_p in c_n, and it can be calculated by

$$\beta_{P,n}(X) = \frac{I_{p,n}}{\mu_{p,n} \cdot CPU_n}, \tag{7}$$

where $I_{p,n}$ is the instruction length of x_p in c_n, $\mu_{p,n}$ is the number of the occupied VMs x_p in c_n, and CPU_n is the running power of the physical machine in the cloudlet c_n.

The execution time of c_n is denoted as $ST(X)$, which is calculated by

$$ST(X) = \max_{n=1}^{N} (\beta_{P,n}(X) \cdot \theta_{p,n}(X)). \tag{8}$$

During the tracked execution period $ST(X)$, the energy consumption of the active VMs in the cloudlet is denoted as $E_{VM}^{active}(X)$, which is calculated by

$$E_{VM}^{active}(X) = \sum_{n=1}^{N} \sum_{p=1}^{P} u_p \theta_{p,n}(X) \cdot ST(X) \cdot \eta, \tag{9}$$

where η is the power rate for the running VM instances.

Before the data offloading, all the VMs are assumed as the idle VMs. When the IoT data are offloaded to the cloudlets for execution, some idle VMs are switched into active VMs. This switch also generates a certain amount of energy consumption. Hence the energy consumption of the initial idle VMs is denoted as $E_{VM}^{initial}(X)$, which is calculated by

$$E_{VM}^{initial}(X) = \sum_{p=1}^{p} u_p \cdot (ST(X) - \beta_P(X)) \cdot \theta_{p,n}(X) \cdot \tau, \tag{10}$$

where τ is the power rate for the idle VM instances.

The process by which an idle VM is changed into an active one also consumes some energy. The energy consumption to switch the situation of VMs is denoted as $E_{VM}^{switch}(X)$, which is calculated by

$$E_{VM}^{switch}(X) = \left(h_n - \sum_{p=1}^{P} u_{p,n}\right) \cdot ST(X) \cdot \theta_{p,n}(X) \cdot \tau. \tag{11}$$

In this way, the energy consumption of the idle VMs in the cloudlet is denoted as $E_{VM}^{idle}(X)$, which is calculated by

$$E_{VM}^{idle}(X) = \sum_{n=1}^{N} \left(E_{VM}^{initial}(X) + E_{VM}^{switch}(X)\right). \tag{12}$$

All the running cloudlets consume the baseline energy during the tracked execution period $ST(X)$. Such baseline energy consumption for the cloudlets is denoted as $E_C(X)$, which can be calculated by

$$E_C(X) = \sum_{n=1}^{N} \sum_{p=1}^{p} \theta_{p,n}(X) \cdot ST(X) \cdot \xi, \tag{13}$$

where ξ is the power rate for the cloudlets.

Then, the total energy consumption denoted as $E(X)$ is calculated by

$$E(X) = E_{VM}^{active}(X) + E_{VM}^{idle}(X) + E_C(X). \tag{14}$$

2.5. Data Privacy Preservation Model of the Comouting Tasks

The IoT data contained in the computing tasks have different attributes, which may generate privacy conflicts. Putting the IoT data with privacy conflicts on the same cloudlet may make the IoT data be easily attacked by hackers and it will cause data leakage. Hence, the IoT data with privacy conflicts under this condition are offloaded to the other cloudlets for processing.

The privacy conflicts of the datasets are modeled by a graph $G = (D, E)$, where D is the set of datasets and E represents the conflicting relations between two datasets in D. Additionally, $(d_i, d_j) \in E$ represents that there is a privacy conflict between the datasets d_i and d_j, which should be offloaded to different cloudlets for processing.

Then, the conflicting datasets of d_p are denoted as CD_p, which is obtained by

$$CD_p = \{d_j | (d_p, d_j) \in E, p = \{1, 2, \cdots, P\}\}. \tag{15}$$

Hence, the conflicting collection is obtained by $X = \{x_1, x_2, \cdots, x_P\}$, where $X = \{x_i | x_i \in CD_p, i = \{1, 2, \cdots, |CD_p|\}\}$.

3. An IoT-Oriented Offloading Method with Privacy Preservation

In this section, we mainly encode the IoT-oriented offloading model with data privacy conflicts in the WMAN. We aim to maximize the resource utilization in (4), minimize the data transmission time in (6) and minimize the energy consumption in (14) while satisfying the privacy constraints in (15). The formalized multi-objective problem is optimized by NSDE, and the diversity and convergence of the population are ensured through the mutation and crossover operations. In the individual selection phase, NSDE uses the fast nondominated sorting approach and the crowded-comparison operator to ensure that individuals with the relatively best fitness values in the current population can be preserved for the next generation.

3.1. Shortest Path Acquisition of APs in WMAN Based on Dijkstra Algorithm

To estimate the transmission time among the APs, we adopt the Dijkstra algorithm to calculate the shortest path between the APs. In order to reduce the transmission time of IoT data, each AP selects the shortest path for transmission.

In Figure 1, all the computing tasks and the IoT data are uploaded to the APs closest to them and offloaded to the appropriate cloudlet for processing. However, in the WMAN, there may be multiple transmission paths between two APs. In order to reduce the transmission time of data, each AP selects the shortest path for transmission.

Assuming that the transmission rate between the APs is the same, the WMAN can be regarded as a set of undirected unweighted graphs, and each AP is a node on this graph. The Dijkstra algorithm is used to calculate the shortest path between the mobile nodes.

3.2. Optimization Problem Model by NSDE

The problem model proposed in this paper is summarized as a constrained multi-objective optimization problem. NSDE is an essential multi-objective optimization algorithm using real number coding whose mutation vector is generated by the parent difference vector and intersects with the parent individual vector to generate a new individual vector. In the parent population and the offspring population, the fast nondominated sorting approach and the crowded-comparison operator are performed, and the individuals with better target values are preserved for the next generation. Compared with other algorithms, it is more effective in solving the approximation of the global optimal solution set in multidimensional space.

In this section, the problem model is real-coded, performing crossover, mutation and selection operations, and the problem is using the fast nondominated sorting approach and the crowded-comparison operator in the selection phase to preserve the individuals with better fitness for the next generation, and through continuous iteration, it is constantly approaching the optimal solution set.

(1) Encoding: A distribution strategy for all the computing tasks uploaded from each AP point is represented by a chromosome, and each gene in the chromosome represents the execution location of the task, which means that the task will be assigned to the corresponding cloudlet for execution. Therefore, the range of values for each gene depends on the number of the cloudlets that are used to perform the computing tasks.

Figure 2 shows the value of each gene in a chromosome. Suppose that the n-th task t_n will be assigned to M cloudlets for execution. Then, the length of this chromosome X_j is N, and each gene will be a real value between 0 and M. However, in the calculation, each real value will be converted into an integer that represents the position of the execution cloudlet. For example, in Figure 2, the second task t_1 has a gene value of 3.2, and adopts the "down rounding" method, so the task t_1 will be assigned to cloudlet 3 for execution.

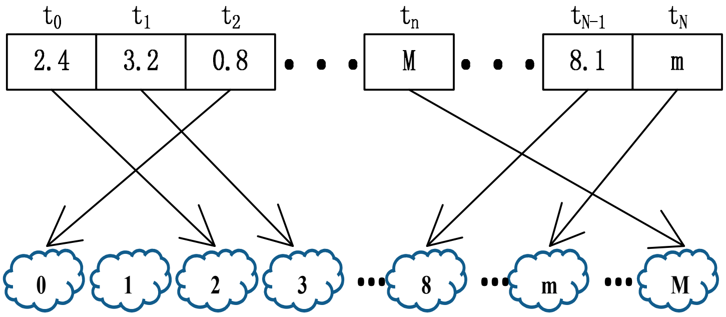


Figure 2. An encoding example of the computation offloading for the computing tasks.

(2) Fitness functions and constraints: A chromosome is an individual which represents an offloading strategy for all the computing tasks in the optimization problem. Multiple individuals constitute a population, and NSDE is used to optimize the population. The fitness functions are the criteria for evaluating each individual in the population, and the constraints are the conditions that each individual needs to satisfy during the problem optimization process.

There are three fitness functions for this optimization problem: the average resource utilization of the cloudlet, the data transmission time and the energy consumption, which are calculated by fitness functions (4), (6) and (14), respectively. In this optimization problem, the larger average resource utilization of the cloudlet with the smaller data transmission time and the lower energy consumption contributes to the better individual. Hence, the NSDE comprehensively evaluates all individuals in the population through these three fitness functions, not one or two of them.

However, during the evolution of the population, each individual also needs to meet two constraints: the total number of VMs requested by all the computing tasks on each cloudlet cannot be greater than the maximum number of VMs in the cloudlet, and the privacy preservation of the data is satisfied, which means the conflicting data cannot be offloaded to the same cloudlet.

(3) Initialization: Before the population is initialized, there are several algorithm parameters that need to be determined: the individual length N, which depends on the total number of computing tasks from all APs and each gene in the individual, represents the position of the cloudlet that the task is executed on, which has been introduced in the "Encoding" section; the size of population NP, which is usually set between $5N$ and $10N$, but not less than $4N$; additionally, in the evolution of the NSDE, three parameters are used, including the crossover factor CR, the mutation factor F and the mutation strategy, where CR and F mainly determine the optimization ability and the convergence speed of the NSDE. The mutation strategy selected in this paper is "*DE/rand/1*". After the parameters have been determined, the NSDE will generate a parent population P whose size is NP by initialization. The length of each individual is N, and the value of each gene in the individual is between 0 and M, where N and M represent the number of t computing tasks and cloudlets in the WMAN, respectively.

(4) Mutation and crossover: The mutation operation is performed by randomly selecting three individuals X_a, X_b and X_c from the parent population P, and generating a mutated individual H_i by combining the third individual X_a with the difference vector of X_b and X_c, which is scaled according to the variation factor F. The crossover operation generates every gene $V_{i,j}$ of the offspring individual V_i by crossing $X_{i,j}$ of the parent individual X_i and $H_{i,j}$ of the mutated individual H_i, where $X_{i,j}$, $H_{i,j}$ and $V_{i,j}$ respectively represent the j-th gene of the parent individual X_i, the mutated individual H_i and the offspring individual V_i.

(5) Selection: In the selection phase, based on the three fitness functions (4), (6) and (14), the NSDE performs the fast nondominated sorting approach and the crowded-comparison operator for the population O, which is composed of the parent population P and the offspring individual $\mathrm{V_i}$. The multiple nondominated layers $L_i(i = 0, 1, 2, \ldots)$ will be generated by the fast nondominated sorting approach, and the individuals in the nondominated layer with the lower nondominated level or the individuals with a better crowding distance in the same nondominated layer are preferentially populated into the parent population P of the next generation until the size of the population P is exactly equal to NP. The method of crowding distance calculation is described as follows:

$$D_j = D_j{}^U + D_j{}^T + D_j{}^E = |U^{j+1} - U^{j-1}| + |T^{j+1} - T^{j-1}| + |E^{j+1} - E^{j-1}|, \tag{16}$$

where D_j represents the crowding distance, $D_j{}^U$, $D_j{}^T$ and $D_j{}^E$ represent the crowding distances of the average resource utilization, the data transmission time and the energy consumption, respectively. Besides, in (16), U_j, T_j and E_j represent the objective function values of the average resource utilization, the data transmission time and the energy consumption, respectively, by the j-th offloading strategy X_j.

(6) Iteration: The NSDE takes the population P generated by performing selection operations as the parent population of the next generation, and combines the population P and the mutation population Q generated by performing mutation and crossover operations into a population whose size is $2NP$. The parental population P of the next generation is regenerated by performing the selection operations on the population O. This process iterates until the termination condition is met, and finally the better solutions set S of the optimization problems is obtained.

NSDE-based migration strategy acquisition is illustrated in Algorithm 1.

Algorithm 1. nondominated sorting differential evolution based offloading strategy acquisition.

```
Input: the size of the population $NP$, the number of the cloudlets $M$, the number of the iterations $G$, the mutation factor $F$, the crossover factor $CR$
Output: the better solution set $S$
$g = 0$
$P$ = Initialization ($NP$, $M$)
while $g < G$ do
    $Q$ = Crossover and mutation($P$, $F$, $CR$)
    $O = P + Q$)
    $L$ = nondominated sort($O$))
    $P = \varnothing$, $Q = \varnothing$, $i = 0$)
    while size($P$) + num ($L_i$) < $NP$ do
        $P$ += $L_i$
        $i$ ++
    end while
    Calculate crowding distance ($L_i$) by (16)
    $L_{i,j}$ = Sort ($L_i$) according to $D_j$ from large to small
    $j = 0$
    while size($P$) < $NP$ do
        $P$ += $L_{i,j}$
        $j$ ++
    end while
$g$ ++
end while
return the better solution set $S$
```

4. Experimental Evaluation

In this section, a set of comprehensive simulations and experiments are conducted to evaluate the performance of the proposed IOM method. Specifically, we first introduce the simulation setup, including the simulation parameter settings and the statement of the comparative methods. Then, the influence of different task scales on the performance of the optimization metrics is evaluated.

4.1. Simulation Setup

In our simulation, three datasets with different scales of the computing tasks are applied for our experiments, and the number of computing tasks is set to 100, 150 and 200, respectively. The transmission speed of the cloudlets and the power rate of the cloudlets are set to 1200 M/s and 300 W according to [30]. The system decides which data items have conflicts according to the requirement of information security defined by the users or the processing records, which are assumed as the known information in our simulation. The specified parameter settings in this experiment are illustrated in Table 2.

Table 2. Parameter settings.

Parameter Description	Value
The total number of cloudlets	50
The maximum number of VMs a cloudlet owns	10
The transmission speed of AP	540 M/s
The transmission speed of the cloudlet	1200 M/s
The execution speed of the VMs	2000 MHz
The power rate of the active VMs	50 W
The power rate of the idle VMs	30 W
The power rate of the cloudlets	300 W

To conduct the comparison analysis, we employ another basic offloaded method. The comparative method is briefly expounded as follows.

Benchmark: The task is offloaded to the nearest cloudlet first. If the task to be offloaded requires more resources than the current cloudlet owns or it has data conflicts with the computing tasks offloaded to the current cloudlet already, this task is offloaded to the cloudlet near the current one according to the Dijkstra algorithm. This process is repeated until all the computing tasks are offloaded to the cloudlets.

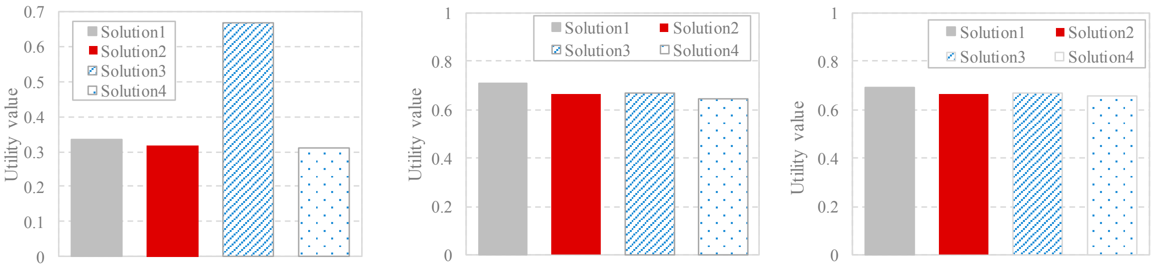
The methods are implemented under the simulation tools by CloudSim on a PC machine with two Intel Core i7-5500U 2.40 GHz processors and 8 GB RAM. The corresponding evaluation results are depicted in detail in the following sections.

4.2. Performance Evaluation of IOM

The proposed IOM is intended to achieve a trade-off between optimizing the resource utilization, shortening the data transmission time and reducing the energy consumption while taking privacy preservation into consideration. We conducted 50 replicates of the experiment in the case of convergence for each task scale, and multiple sets of results were obtained. To identify a relatively optimal solution, simple additive weighting (SAW) and multiple-criteria decision making (MCDM) were used, where the optimal function is measured as follows:

$$V(X) = \frac{1}{3} \cdot \frac{\psi(X) - \psi^{\min}}{\psi^{\max} - \psi^{\min}} + \frac{1}{3} \cdot \frac{T^{\max} - T(X)}{T^{\max} - T^{\min}} + \frac{1}{3} \cdot \frac{E^{\max} - E(X)}{E^{\max} - E^{\min}}, \tag{17}$$

where $\Psi(X)$, $T(X)$ and $E(X)$ represent the fitness of the data offloading strategy x_i regarding the three objective functions mentioned above, respectively. $\Psi^{\max}$ and $\Psi^{\min}$ represent the maximum and minimum fitness values for the resource utilization. If $\Psi^{\max} = \Psi^{\min}$, let $\frac{\Psi(X)-\Psi^{\min}}{\Psi^{\max}-\Psi^{\min}} = 1$. Analogously, $T^{\max}$ and $T^{\min}$ represent the maximum and minimum fitness for the data transmission time, and if $T^{\max} = T^{\min}$, let $\frac{T^{\max}-T(X)}{T^{\max}-T^{\min}} = 1$; $E^{\max}$ and $E^{\min}$ represent the maximum and minimum fitness for the energy consumption, and if $E^{\max} = E^{\min}$, let $\frac{E^{\max}-E(X)}{E^{\max}-E^{\min}} = 1$. Figure 3 shows the comparison of the utility value of the solutions generated by IOM with different task scales. It is illustrated that when the task scale is 100, 150 or 200, four solutions are generated by IOM. For the solutions generated by IOM, we attempt to obtain the most balanced data offloading strategy by evaluating the utility value given in (17). After statistics and analysis, the solution with the maximum utility value is considered as the most balanced strategy. For instance, in Figure 3a, the final selected strategy is solution 3 because it achieves the highest utility value.



(**a**) Number of computing tasks = 100 (**b**) Number of computing tasks = 150 (**c**) Number of computing tasks = 200

Figure 3. Comparison of the utility value of the solutions generated with different task scales by IOM.

4.3. Comparison Analysis

In this subsection, the comparisons of Benchmark and IOM with the same experimental context are analyzed in detail. The resource utilization, the data transmission time and the energy consumption are the main metrics for evaluating the performance of the data offloading methods. In addition, the number of employed cloudlets is presented to show the resource usage of all the cloudlets for offloading the computing tasks. The corresponding results are shown in Figures 4–8.

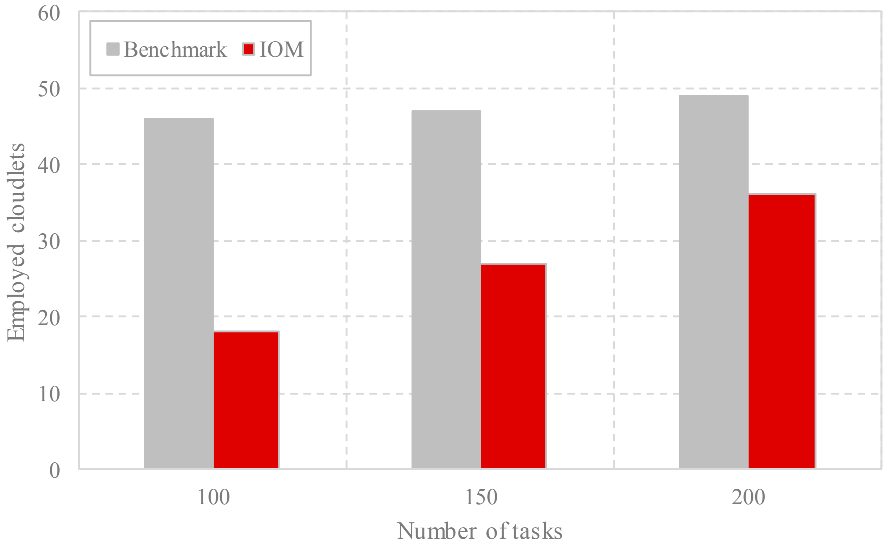


Figure 4. Comparison of the number of the employed cloudlets with different task scales by Benchmark and IOM.

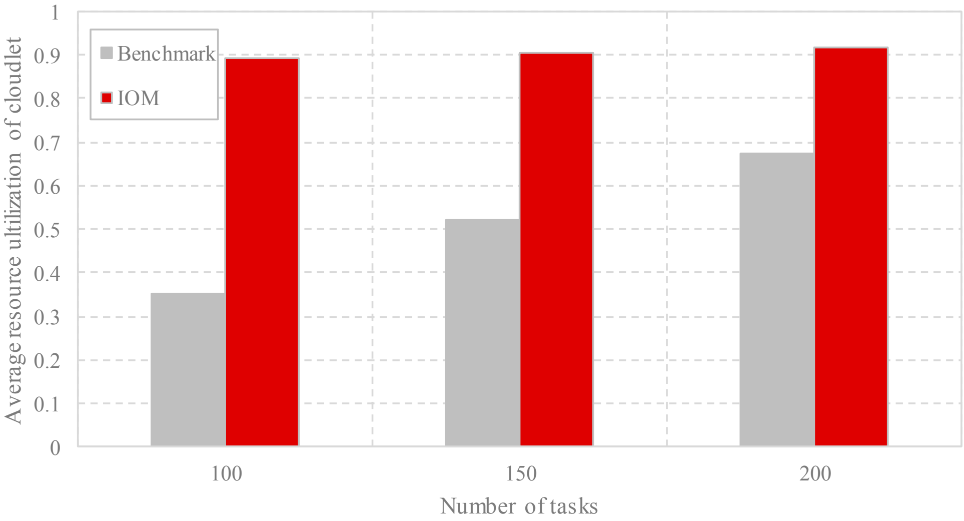


Figure 5. Comparison of the average resource utilization of the cloudlets with different task scales by Benchmark and IOM.

(1) Comparison of the number of employed cloudlets: Figure 4 illustrates the number of cloudlets employed by the data offloading methods. The total number of the cloudlets in our experiment is set to 50. As shown in Figure 4, IOM employs fewer cloudlets compared with Benchmark. Furthermore, as the number of the computing tasks increases, the number of the cloudlets employed by IOM increases.

(2) Comparison of the resource utilization: After offloading all the computing tasks to the cloudlets via the data offloading methods, the occupation of the VMs is achieved. Figure 5 shows the comparison of the resource utilization of the cloudlets by using Benchmark and IOM with different task scales. The resource utilization is calculated according to the number of occupied cloudlets and the employed VMs in each cloudlet. Fewer employed cloudlets with more employed VMs contribute to a higher resource utilization. It is intuitive from Figure 5 that IOM achieves higher and stable resource utilization. That is, IOM reduces the number of unemployed VMs and wastes less resources than Benchmark.

(3) Comparison of the data transmission time: In Figure 6, we compare the data transmission time of the different data offloading methods. It is intuitive that our proposed method IOM costs more time than Benchmark. With the increase of the task scales, the data transmission time is enlarged. This may be because our proposed method needs more transmission times to realize the goal of optimizing the resource utilization and the energy consumption, which may sacrifice some transmission time on the other hand.

(4) Comparison of the energy consumption: As outlined in Section 2, the energy consumption is composed of the energy consumption of the active VMs, the energy consumption of the idle VMs, and the energy consumption of the cloudlets. In Figure 7, we compare these three aspects respectively with different task scales. As shown in Figure 7a, both methods achieve the same energy consumption of the active VMs at the same task scale because the same number of VMs are employed by Benchmark and IOM. Figure 7b shows that as the number of computing tasks increases, both methods increase the energy consumption of the idle VMs, but IOM generates less energy of the idle VMs due to less unemployed VMs used compared with Benchmark by occupying fewer cloudlets. Figure 7c indicates that IOM consumes less energy of the cloudlets than the Benchmark. The comparison of energy consumption in Figure 8 shows that IOM has better performance. For example, when the number of computing tasks is 100, IOM achieves a power consumption of less than 3000 W.s, whereas Benchmark generates more than 5000 W.s energy.

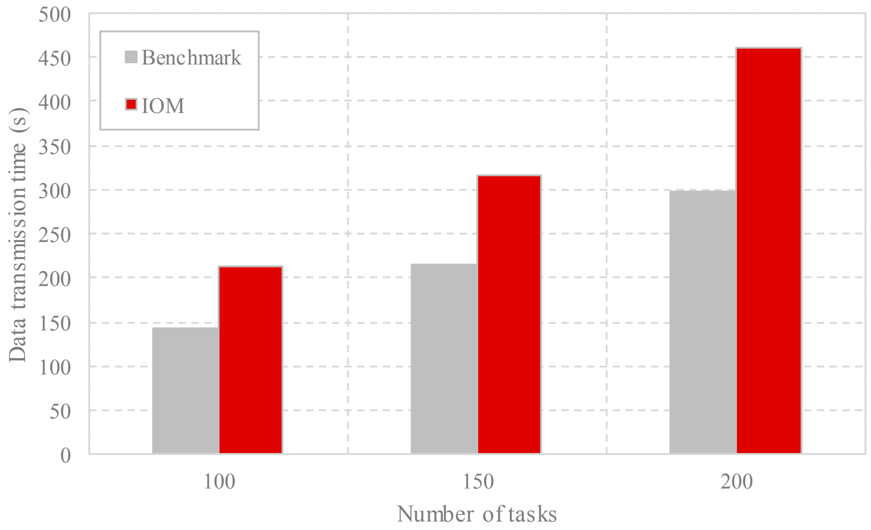
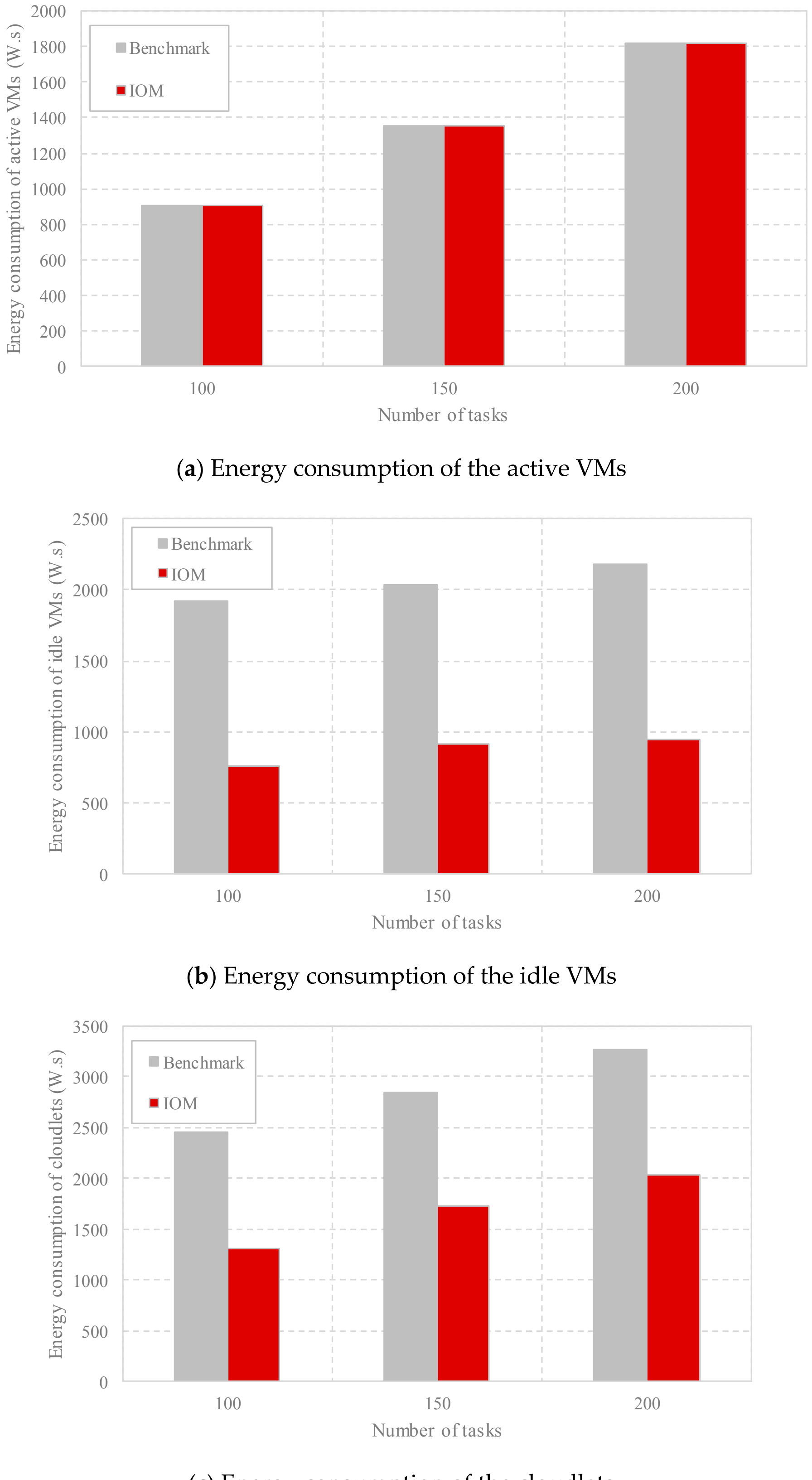


Figure 6. Comparison of the data transmission time with different task scales by Benchmark and IOM.



(**a**) Energy consumption of the active VMs

(**b**) Energy consumption of the idle VMs

(**c**) Energy consumption of the cloudlets

Figure 7. Comparison of the different components of the energy consumption with different task scales by Benchmark and IOM.

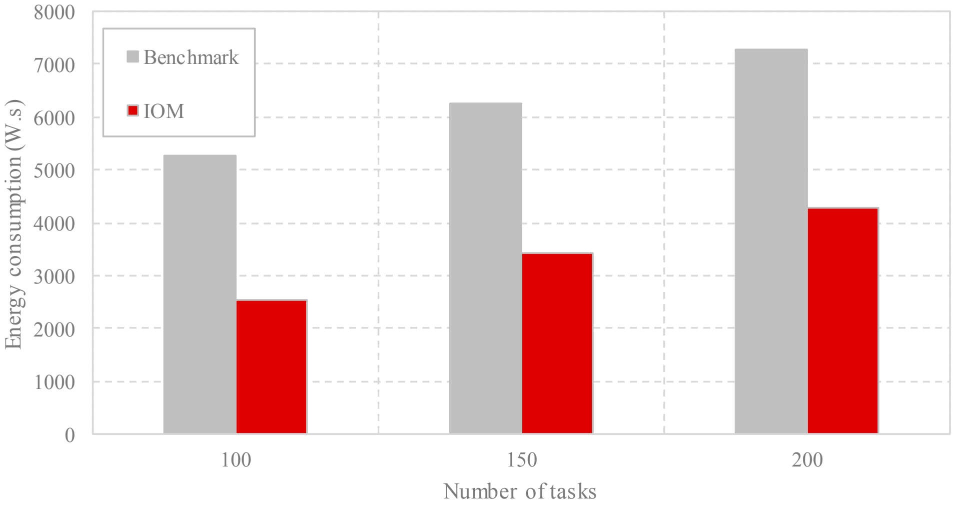


Figure 8. Comparison of the energy consumption with different task scales by Benchmark and IOM.

5. Related Work

With the development of the IoT technology, more IoT data is produced by mobile devices in daily life. Edge cloud computing developed rapidly to solve the transmission delay of the IoT data, providing high-speed processing in cloud service [31]. One of the hot technologies of edge cloud computing is the cloudlet, which is applied to get a shorter response time and reduce the energy consumption of mobile devices by alternating the offloading destinations, compared to the traditional mobile cloud computing paradigm [32–35]. There have been many studies about cloudlets, which were fully investigated in [36–42], to name a few.

In [36], the author studied the placement of the cloudlets in a large WMAN, consisting of many wireless APs. In order to realize the resource sharing of mobile users, Hoang et al. [37] used a cloudlet as a semi-Markov decision process (SMDP) to formalize a dynamic optimization problem. The SMDP is converted into a linear programming model to get the best solution. In the optimization model, mobile users need to consider different types of service quality under resource constraints. In [38], the author proposes a Performance-Enhancement Framework of Cloudlet (PEFC) to enhance the service performance of a cloudlet with limited resources. That paper aims to enhance the performance of the cloudlet and improve the experience of cloud service with limited resources. Artail et al. [39] proposed a general solution based on a mobile intelligent device to solve the service delay of the remote cloud. The author considered a cloud network, which distributes within a region and connects to the root server, to ensure resource availability. The framework is applicable to the environment where the cloudlet clients can sense networks and software services. Ciobanu et al. [40] introduced the drop computing paradigm, which proposes the concept of decentralized computing over multilayered networks, combining cloud and wireless technologies over a social crowd formed between mobile and edge devices. Mao et al. [41] jointly optimized task offload scheduling and transmission power allocation for mobile edge computing systems to reduce execution latency and device power consumption. The author proposed a low-complexity suboptimal algorithm to minimize the weighted sum of execution delay and device energy consumption, based on alternating minimization. Although the research on cloudlets is increasing, people often overlook the optimization of resource utilization, transmission delay and energy consumption when taking the privacy protection into account [43–46].

Current research mainly focuses on the capacitated cloudlets' placement to save energy or encrypt data to prevent data leakage. In [47], the author studied the cloudlet placement and mobile user allocation to the cloudlets in the WMAN. The author also designed a cloudlet placement algorithm, which placed the cloudlet in a user-intensive area of the wireless metropolitan area networks to balance the workload of WMAN. Mahadev et al. [48] introduced GigaSight, which is an internet-scale crowdsourced video content repository with powerful privacy preferences and access control features. The GigaSight architecture is a joint system of VM-based cloudlets that performs video analytics on the edge of the internet, reducing the need for cloudlet ingress bandwidth. Rahman et al. [49] proposed a mobile edge computing framework that provides real-time and location-aware personalized services for a large number of users. According to the new privacy policy paradigm, it can make a secure share of location. The framework uses server-side cloud blending and crowd edge fog computing terminals (FCTs) to switch tasks between FCTs and the cloud, based on network condition, geographic location and available resources. Chen et al. [50] used the flexibility of the cloudlets to create a novel healthcare system. Cloudlet features include privacy protection, data sharing and intrusion detection. In the data collection phase, the author used the Number Theory Research Unit (NTRU) method to encrypt data collected by wearable devices. Then, they proposed a new trust model to help users choose trusted partners to share stored data in the cloudlet and help similar patients to communicate with each other. Finally, in order to protect the medical system from malicious attacks, the author developed a new collaborative intrusion detection system (IDS) method based on cloud networks.

Generally speaking, researchers do not take into consideration the data privacy preservation when optimizing the energy consumption of the cloudlets in the WMAN, or ameliorate the transmission time, energy consumption and resource utility when encrypting data [51–53]. Thus, an IoT-oriented offloading method with privacy preservation is proposed in this paper to optimize the transmission time, the energy consumption and the resource utilization when considering data privacy preservation.

6. Conclusions and Future Work

With the rapid development of IoT technology, the computing tasks of mobile applications have become so complex that it is necessary to offload the computing tasks to the remote cloud. For some applications with low latency requirements, it is necessary to offload the computing tasks to the nearby cloudlets for execution. Meanwhile, we have to be considerate of data conflicts to realize privacy preservation. In order to tackle such problems that have happened in the cloudlet-based WMAN environment, an IoT-oriented offloading method with privacy preservation is proposed in this paper to optimize the transmission time, the energy consumption and the resource utilization. Concretely, the task-offloading strategy with privacy preservation in the WMAN is modeled as a constrained multi-objective optimization problem. In order to reduce the transmission time, the Dijkstra algorithm is adopted to calculate the shortest path among APs in the WMAN. The multi-objective optimization problem is solved by an NSDE algorithm, and finally the best task-offloading strategy in the WMAN is obtained.

In future work, we will attempt to adapt and extend our proposed method to a real-world scenario for cloudlet services in the WMAN environment. Additionally, the privacy preservation strategy will be updated on the basis of the IoT data. At the same time, more attributes of the real-world scenario will be added to confirm the accuracy of our experiment.

Author Contributions: Conceptualization, H.X. and Z.X.; Methodology, T.H.; Software, T.H.; Validation, Z.X., X.X. and R.G.; Formal Analysis, R.G.; Investigation, L.Q.; Resources, X.Z.; Data Curation, L.Q.; Writing-Original Draft Preparation, Z.X.; Writing-Review & Editing, R.G., H.X.; Visualization, X.X.; Supervision, Z.X.; Project Administration, Z.X.; Funding Acquisition, X.X.

Funding: This research was funded by [National Science Foundation of China] grant number [61702277], [61872219] and [61772283]. It is also supported by [The Startup Foundation for Introducing Talent of NUIST] grant number [KFKT2017B04].

Acknowledgments: This research is supported the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund, and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Baskaran, S.B.M.; Raja, G. Blind key distribution mechanism to secure wireless metropolitan area network. *CSI Trans. ICT* **2016**, *4*, 157–163. [CrossRef]
2. Zhou, B.; Dastjerdi, A.V.; Calheiros, R.N.; Srirama, S.N.; Buyya, R. mCloud: A context-aware offloading framework for heterogeneous mobile cloud. *Trans. Serv. Comput.* **2015**, *10*, 797–810. [CrossRef]
3. Karloff, H.; Subbaraman, R. Designing wireless metropolitan-area networks using mathematical optimization. In Proceedings of the Wireless Telecommunications Symposium (WTS), New York, NY, USA, 15–17 April 2015; pp. 1–8.
4. Al-Janabi, S.; Al-Shourbaji, I.; Shojafar, M.; Abdelhag, M. Mobile Cloud Computing: Challenges and Future Research Directions. In Proceedings of the 10th International Conference on Developments in eSystems Engineering (DeSE), Paris, France, 14–16 June 2017; pp. 62–67.
5. Zhaoqing, P.; Jianjun, L.; Yajuan, Z.; Wang, F.L. Adaptive fractional-pixel motion estimation skipped algorithm for efficient HEVC motion estimation. *ACM Trans. Multimed. Comput. Commun. Appl.* **2018**, *14*, 12–19.
6. Chen, X. Decentralized computation offloading game for mobile cloud computing. *Trans. Parallel Distrib. Syst.* **2015**, *26*, 974–983. [CrossRef]
7. Jara, A.J. Wearable internet: Powering personal devices with the internet of things capabilities. In Proceedings of the International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI), Beijing, China, 17–18 October 2014; pp. 1–7.
8. Li, P.; Zhao, S.; Zhang, R. A cluster analysis selection strategy for supersaturated designs. *Comput. Stat. Data Anal.* **2010**, *54*, 1605–1612. [CrossRef]
9. Keng, D.; Koo, S.G.M. Spatial standards for Internet of Things. In Proceedings of the Conference on Internet of Things (iThings), Taipei, Taiwan, 1–3 September 2014; pp. 284–287.
10. Tian, G.L.; Wang, M.; Song, L. Variable selection in the high-dimensional continuous generalized linear model with current status data. *J. Appl. Stat.* **2014**, *41*, 467–483. [CrossRef]
11. Li, S.; Zhang, Y. On-Line Scheduling on Parallel Machines to Minimize the Makespan. *J. Syst. Sci. Complex.* **2016**, *29*, 472–477. [CrossRef]
12. Wang, Y.; Sun, X.; Meng, F. On the conditional and partial trade credit policy with capital constraints: A Stackelberg model. *Appl. Math. Model.* **2016**, *40*, 1–18. [CrossRef]
13. Meng, S.; Wang, S.; Wu, T.; Li, D.; Huang, T.; Wu, X.; Xu, X.; Dou, W. An uncertainty-aware evolutionary scheduling method for cloud service provisioning. In Proceedings of the IEEE International Conference on Web Services (ICWS), San Francisco, CA, USA, 27 June–2 July 2016; pp. 506–513.
14. Wang, G.; Huang, X.X.; Zhang, J. Levitin-Polyak well-posedness in generalized equilibrium problems with functional constraints. *Pac. J. Optim.* **2010**, *6*, 441–453.
15. Qu, B.; Zhao, J. Methods for Solving Generalized Nash Equilibrium. *J. Appl. Math.* **2013**. [CrossRef]
16. Zhang, J.; Qu, B.; Xiu, N. Some projection-like methods for the generalized Nash equilibria. *Comput. Optim. Appl.* **2010**, *45*, 89–109. [CrossRef]
17. Li, Y.; Gao, W. Code offload with least context migration in the mobile cloud. In Proceedings of the Conference on Computer Communications (ICCC), Kowloon, Hong Kong, 26 April–1 May 2015; pp. 1876–1884.
18. Soyata, T.; Muraleedharan, R.; Funai, C.; Kwon, M.; Heinzelman, W. Cloud-vision: Real-time face recognition using a mobile cloudlet-cloud acceleration architecture. In Proceedings of the IEEE Symposium on Computers and Communications (ISCC), Cappadocia, Turkey, 1–4 July 2012; pp. 59–66.
19. Li, Y.; Anh, N.T.; Nooh, A.S.; Ra, K.; Jo, M. Dynamic Mobile Cloudlet Clustering for Fog Computing. In Proceedings of the International Conference on Electronics, Information, and Communication (CEIC), Honolulu, HI, USA, 24–27 January 2018; pp. 1–4.
20. Shen, J.; Tan, H.-W.; Wang, J.; Wang, J.-W.; Lee, S.-Y. A novel routing protocol providing good transmission reliability in underwater sensor networks. *J. Internet Technol.* **2015**, *16*, 171–178.
21. Xiang, H.; Xu, X.; Zheng, H.; Li, S.; Wu, T.; Dou, W.; Yu, S. An adaptive cloudlet placement method for mobile applications over cloud. In Proceedings of the Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016.
22. Li, H.; Wang, S. Partial condition number for the equality constrained linear least squares problem. *Calcolo* **2017**, *54*, 1121–1146. [CrossRef]

23. Li, P.; Li, T.; Ye, H.; Li, J.; Chen, X.; Xiang, Y. Privacy-preserving machine learning with multiple data providers. *Future Gener. Comput. Syst.* **2018**, *87*, 341–350. [CrossRef]
24. He, K.; Chen, J.; Du, R.; Wu, Q.; Xue, G.; Zhang, X. Deypos: Deduplicatable dynamic proof of storage for multi-user environments. *IEEE Trans. Comput.* **2016**, *65*, 3631–3645. [CrossRef]
25. Namjoshi, J.; Gupte, A. Service oriented architecture for cloud based travel reservation software as a service. In Proceedings of the 2009 IEEE International Conference on Cloud Computing (ICCC), Bangalore, India, 21–25 September 2009; pp. 147–150.
26. Xue, S.; Zhang, Y.; Xu, X.; Xing, G.; Xiang, H.; Ji, S. QET: A QoS-Based Energy-Aware Task Scheduling Method in Cloud Environment. *Clust. Comput.* **2017**. [CrossRef]
27. Yang, J.; Xu, X.; Dou, W. A Task Scheduling Method for Energy-Performance Trade-off in Clouds. In Proceedings of the 18th International Conference on High Performance Computing and Communications (HPCC), Sydney, Australia, 12–14 December 2016; pp. 1029–1036.
28. Su, M.; Zhang, L.; Wu, Y.; Chen, K.; Li, K. Systematic data placement optimization in multi-cloud storage for complex requirements. *IEEE Trans. Comput.* **2016**, *65*, 1964–1977. [CrossRef]
29. Jia, M.; Liang, W.; Xu, Z.; Huang, M.; Ma, Y. QoS-Aware Cloudlet Load Balancing in Wireless Metropolitan Area Networks. *IEEE Trans. Cloud Comput.* **2018**. [CrossRef]
30. Jia, M.; Liang, W.; Xu, Z.; Huang, M. Cloudlet load balancing in wireless metropolitan area networks. In Proceedings of the Conference on Computer Communications (INFOCOM), San Francisco, CA, USA, 10–14 April 2016; pp. 1–9.
31. Khan, A.M.; Freitag, F. On Edge Cloud Service Provision with Distributed Home Servers. In Proceedings of the IEEE International Conference on Cloud Computing Technology and Science (CloudCom), Hong Kong, China, 11–14 December 2017; pp. 223–226.
32. Barbera, M.; Kosta, S.; Mei, A.; Stefa, J. To offload or not to offload? the bandwidth and energy costs of mobile cloud computing. In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), Turin, Italy, 14–19 April 2013; pp. 1285–1293.
33. Zhang, Z.; Li, S. A Survey of Computational Offloading in Mobile Cloud Computing. In Proceedings of the IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), Oxford, UK, 29 March–1 April 2016; pp. 81–82.
34. Wang, P.; Zhao, L. Some geometrical properties of convex level sets of minimal graph on 2-dimensional Riemannian manifolds. *Nonlinear Anal.* **2016**, *130*, 1–17. [CrossRef]
35. Wang, S.; Huang, L.; Hsu, C.-H.; Yang, F. Collaboration reputation for trustworthy Web service selection in social networks. *J. Comput. Syst. Sci.* **2016**, *82*, 130–143. [CrossRef]
36. Xu, Z.; Liang, W.; Xu, W.; Jia, M.; Guo, S. Capacitated cloudlet placements in wireless metropolitan area networks. In Proceedings of the IEEE International Conference on Local Computer Networks (LCN), Clearwater Beach, FL, USA, 26–29 October 2015; pp. 570–578.
37. Hoang, D.T.; Niyato, D.; Wang, P. Optimal admission control policy for mobile cloud computing hotspot with cloudlet. In Proceedings of the Conference on Wireless Communications and Networking Conference (IWCNC), Shanghai, China, 1–4 April 2012; pp. 3145–3149.
38. Whaiduzzaman, M.; Gani, A.; Naveed, A. Pefc: Performance enhancement framework for cloudlet in mobile cloud computing. In Proceedings of the Symposium on Robotics and Manufacturing Automation, Kuala Lumpur, Malaysia, 15–16 December 2014; pp. 224–229.
39. Artail, A.; Frenn, K.; Safa, H.; Artail, H. A framework of mobile cloudlet centers based on the use of mobile devices as cloudlets. In Proceedings of the Conference on Advanced Information Networking and Applications (AINA), Gwangiu, Korea, 24–27 March 2015; pp. 777–784.
40. Ciobanu, R.I.; Negru, C.; Pop, F.; Dobre, C. Drop computing: Ad-hoc dynamic collaborative computing. *Future Gener. Comput. Syst.* **2017**. [CrossRef]
41. Mao, Y.; Zhang, J.; Letaief, K.B. Joint task offloading scheduling and transmit power allocation for mobile-edge computing systems. In Proceedings of the in Wireless Communications and Networking Conference (WCNC), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.
42. Lian, S.; Duan, Y. Smoothing of the lower-order exact penalty function for inequality constrained optimization. *J. Inequal. Appl.* **2016**, *2016*, 185. [CrossRef]
43. Wu, H.; Ren, Y.; Hu, F. Continuous dependence property of BSDE with constraints. *Appl. Math. Lett.* **2015**, *45*, 41–46. [CrossRef]

44. Wang, C.; Ma, C.; Zhou, J. A new class of exact penalty functions and penalty algorithms. *J. Glob. Optim.* **2014**, *58*, 51–73. [CrossRef]
45. Fan, Q.; Ansari, N.; Sun, X. Energy driven avatar migration in green cloudlet networks. *IEEE Commun. Lett.* **2017**, *21*, 1601–1604. [CrossRef]
46. Wang, S.; Lei, T.; Zhang, L.; Hsu, C.-H.; Yang, F. Offloading Mobile Data Traffic for QoS-aware Service Provision in Vehicular Cyber-Physical Systems. *Future Gener. Comput. Syst.* **2016**, *61*, 118–127. [CrossRef]
47. Jia, M.; Cao, J.; Liang, W. Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks. *IEEE Trans. Cloud Comput.* **2017**, *5*, 725–737. [CrossRef]
48. Satyanarayanan, M.; Simoens, P.; Xiao, Y.; Pillai, P.; Chen, Z.; Ha, K.; Hu, W.; Amos, B. Edge analytics in the internet of things. *IEEE Perv. Comput.* **2015**, *2*, 24–31. [CrossRef]
49. ARahman; Hassanain, E; Hossain, M.S. Towards a secure mobile edge computing framework for Hajj. *IEEE Access* **2017**, *5*, 11768–11781.
50. Chen, M.; Qian, Y.; Chen, J.; Hwang, K.; Mao, S.; Hu, L. Privacy protection and intrusion avoidance for cloudlet-based medical data sharing. *IEEE Trans. Cloud Comput.* **2016**. [CrossRef]
51. Liu, B.; Qu, B.; Zheng, N. A Successive Projection Algorithm for Solving the Multiple-Sets Split Feasibility Problem. *Numer. Funct. Anal. Optim.* **2014**, *35*, 1459–1466. [CrossRef]
52. Cai, J.Q.; Li, H. An implicit degree condition for relative length of long paths and cycles in graphs. *Acta Math. Appl. Sin.* **2016**, *32*, 365–372. [CrossRef]
53. Li, L.; Meng, F.; Ju, P. Some new integral inequalities and their applications in studying the stability of nonlinear integro-differential equations with time delay. *J. Math. Anal. Appl.* **2011**, *377*, 853–862. [CrossRef]