mathematics

MDPI

Article

Optimization Method of Fog Computing High Offloading Service Based on Frame of Reference

Deng Li , Chengqin Yu, Ying Tan and Jiaqi Liu *

School of Electronic Information, Central South University, Changsha 410075, China; d.li@csu.edu.cn (L.D.); 214711003@csu.edu.cn (C.Y.); 234711028@csu.edu.cn (Y.T.)
* Correspondence: liujiaqi@csu.edu.cn

Abstract: The cost of offloading tasks is a crucial parameter that influences the task selection of fog nodes. Low-cost tasks can be completed quickly, while high-cost tasks are rarely chosen. Therefore, it is essential to design an effective incentive mechanism to encourage fog nodes to actively participate in high-cost offloading tasks. Current incentive mechanisms generally increase remuneration to enhance the probability of participants selecting high-cost tasks, which inevitably leads to increased platform costs. To improve the likelihood of choosing high-cost tasks, we introduce a frame of reference into fog computing offloading and design a Reference Incentive Mechanism (RIM) by incorporating reference objects. Leveraging the characteristics of the frame of reference, we set an appropriate reference task as the reference point that influences the attraction of offloading tasks to fog nodes and motivates them towards choosing high-cost tasks. Finally, simulation results demonstrate that our proposed mechanism outperforms existing algorithms in enhancing the selection probability of high-cost tasks and improving platform utility.

Keywords: fog computing, optimization method, frame of reference, task offloading rate, platform utility

MSC: 68W15

check for updates

Citation: Li, D.; Yu, C.; Tan, Y.; Liu, J. Optimization Method of Fog Computing High Offloading Service Based on Frame of Reference. *Mathematics* **2024**, *12*, 621. https://doi.org/10.3390/math12050621

Academic Editor: Jonathan Blackledge

Received: 3 January 2024
Revised: 5 February 2024
Accepted: 6 February 2024
Published: 20 February 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Fog computing is a computing model that provides users with computing, storage, and network services through Fog Node (FN) links between user terminal devices and cloud servers [1,2]. It finds wide applications in various domains, such as Augmented Reality [3] and Wise Information Technology of med [4,5]. In these fields, fog nodes must provide high-computational offloading services actively. For example, in augmented reality [3], in order to provide users with good real-time interaction capabilities, most of their tasks require a large amount of memory and computing resources, and are sensitive to delay. This means that fog nodes must actively choose such tasks to provide offloading services. However, due to the sensitivity of fog nodes to offloading costs, low-cost tasks are more likely to be selected for completion under the same cost-remuneration ratio. Consequently, high-cost tasks may be left unattended leading to service interruptions for applications with significant computational requirements [6], adversely affecting users' experience. Therefore, it is necessary to design a reasonable incentive mechanism to encourage fog nodes to select high-cost computing offloading tasks actively [7].

In order to address this issue, the existing incentive mechanisms in fog computing, such as auction mechanism [8] and Game Theory methods [9], generally believe that the selection probability of a specific type of task can only be enhanced by increasing its remuneration [10]. This will inevitably increase the platform budget and lead to a decrease in the final utility. The frame of reference refers to another object selected as a standard to determine the position of the research object and describe its motion [11]. Research on frame of reference [12] indicates that different frames of reference yield different states of affairs, and the attraction of an option to an individual is affected by the reference

point. Therefore, incorporating a specific reference item to the selected set will increase the attraction of an original option.

The above frame of reference theory is incorporated into fog computing offloading in this paper to establish the incentive mechanism RIM with reference objects. Based on the frame of reference, we establish an attraction function for tasks assigned to fog nodes. Aiming at high-cost tasks, we analyze the reference effect of different tasks to enhance the selection probability of target tasks. Additionally, we establish the tendency model of fog nodes and different reference tasks are designed for different types of fog nodes to modify their tendency, so as to improve the overall task offloading rate and reduce the platform utility.

2. Related Work

At present, there have been several studies conducted on the incentive mechanism of computing offloading, which can be primarily categorized into monetary and non-monetary approaches. The next section will provide an introduction to each of these mechanisms.

2.1. Monetary Incentive Mechanism

Reference [13] proposed a double auction mechanism to study the video caching problem in dense heterogeneous networks, which maximizes social welfare by encouraging video service providers and mobile operators to reflect real caching needs and bids. In [14], a resource allocation mechanism for mobile edge computing of industrial Internet of Things based on double auction is proposed, which improves the utility of fog nodes by double auction pricing. In [9], the task offloading problem of mobile edge computing in ultra-dense networks is studied. Game Theory is used to design the task offloading strategy scheme, which minimizes the delay in the network and saves the battery life of the mobile user device. Reference [15] proposed to use Game Theory to reduce the execution cost of social groups in fog computing, that is, energy consumption and delay.

2.2. Non-Monetary Incentive Mechanism

Reference [16] proposed an enhanced framework combined with a reputation mechanism. Mobile phone users who help others will receive a lot of virtual currency. When a client user wants to uninstall a task, it broadcasts a request to all adjacent users. Neighbors can calculate the number of floppy disk coins according to the resources required to perform the task. The client user selects a user as the offloading node to perform the task according to the neighbor's bid and their reliability or reputation. Reference [17] proposed a similar incentive mechanism based on reputation and credit. When the user needs to offload the computing task to the offloading node, the reputation value of the node will be reduced, and the reduced part will be added to the reputation value of the offloading node. After each offloading task is completed, the user will receive updated reputation and credit information from the interactive peer.

The main problem in the current research on the incentive mechanism of fog computing lies in the assumption that the attraction of the task to the fog nodes of the auction or Game subject is unaffected by external factors. In other words, the higher the remuneration, the higher the attraction, the stronger the incentive effect [18]. The research on frame of reference indicates that individuals will be influenced by the reference point when making choices, and introducing a reference item among alternatives will increase the attraction of the original choice. By applying this theory, these problems above can be effectively addressed.

3. Design and Analysis of RIM

This section introduces the frame of reference for the fog computing offloading system. Firstly, the system model composed of the physical model and the logical model is introduced, and then the incentive mechanism RIM based on the frame of reference is introduced in detail.

3.1. System Model

3.1.1. Physical Model

In the fog computing offloading scenario, the fog network contains I fog nodes ($I \geq 2$) and J tasks ($J \geq 2$). The communication between the fog node and the user terminal device is relayed through the intermediate platform. The computing offloading platform supports multiple tasks. Each task can only be performed on one fog node, and each fog node can perform multiple offloading tasks at the same time. The cost of some tasks is high, while the cost of other tasks is low. Because most fog nodes are more willing to complete tasks with low cost, the completion rate of tasks with high cost is usually low. Table 1 lists the parameters and related descriptions involved in this paper.

Table 1. Common symbols and meanings.

Variable	Description
I	Total number of fog nodes
J	Total number of tasks
c_j	The cost of task τ_j
r_j	The remuneration of task τ_j
χ	Cost reference factor
γ	Remuneration reference factor
n_{type}	The total number of fog nodes participating in τ_type tasks
$O_i(\tau_{\text{type}} \mid m)$	The attraction value of task to fog node i when the reference point is m
δ_i	The tendency coefficient of fog node i to task remuneration

Figure 1 shows the physical model before and after introducing the incentive mechanism. The completion rate of the high-cost tasks in the left part of Figure 1 is significantly lower compared to that of the low-cost tasks. The right part of Figure 1 demonstrates the impact of introducing an incentive mechanism, resulting in a significant improvement in the completion rate of the high-cost task after incorporating a reference task. The specific process of fog computing offloading is defined as follows:

(1) Task publishing: According to the task information of this round, the platform classifies the computing tasks, designs the reference tasks for the target tasks, and publishes the information of the computing tasks and the reference tasks to the edge nodes together. The information includes the unloading cost of the computing task and the remuneration available for completing the task;
(2) Task selection: the fog node selects the calculation task according to the attraction of the task and submits the quotation application to the platform;
(3) Task allocation: the platform selects the winning node and assigns the tasks;
(4) Performing the offloading task: the fog node performs the offloading task assigned by the platform and returns the result;
(5) Payment: the platform pays the task remuneration to the fog node that completes tasks.

In this paper, the clock frequency is used to quantify the computing power of fog nodes and mobile devices. The tasks of mobile device τ_j are represented by (r_j, c_j), where r_j represents the remuneration of computing tasks, and c_j represents the cost required to calculate tasks. The resource state of fog node i is represented by a binary group (f_i, δ_i), where f_i denotes the clock frequency of fog node i, and δ_i denotes the tendency of fog node i to task remuneration.

The goal of incentive mechanism design based on frame of reference in this paper is to make the mechanism meet the following three expected characteristics:

(1) Authenticity: no matter how much other fog nodes bid, if no fog node can improve its utility by submitting a bid different from its true valuation, the mechanism is authentic.
(2) Individual rationality: in the mechanism, no fog node performs tasks with a return lower than its cost, that is, the utility of the fog node is not negative.
(3) Computational efficiency: the mechanism can be run in a polynomial time.

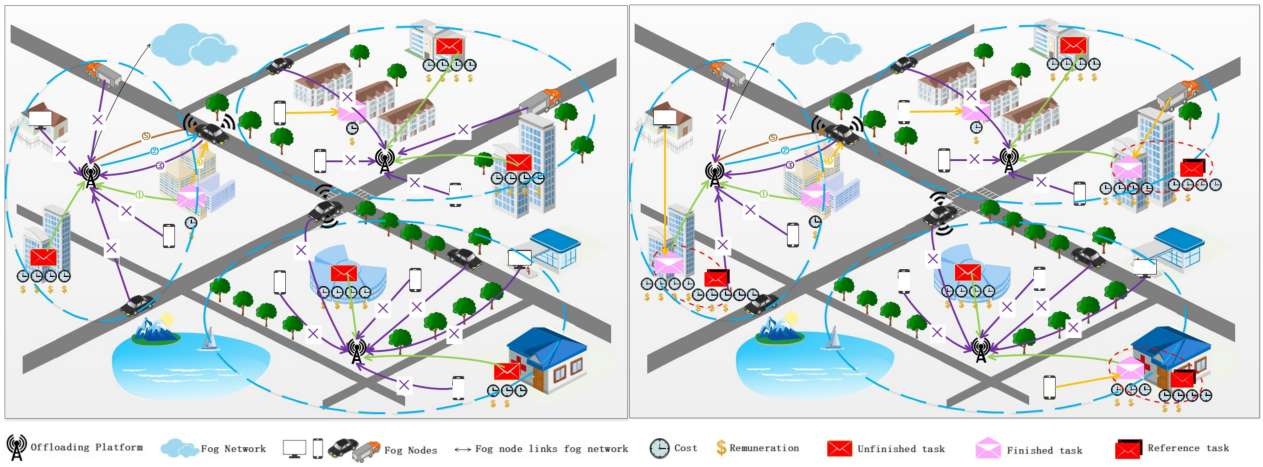


Figure 1. Physical model.

3.1.2. Logical Model

Figure 2 shows the specific incentive steps of the RIM mechanism. The platform takes the average remuneration value and average cost value of all tasks as the reference point.

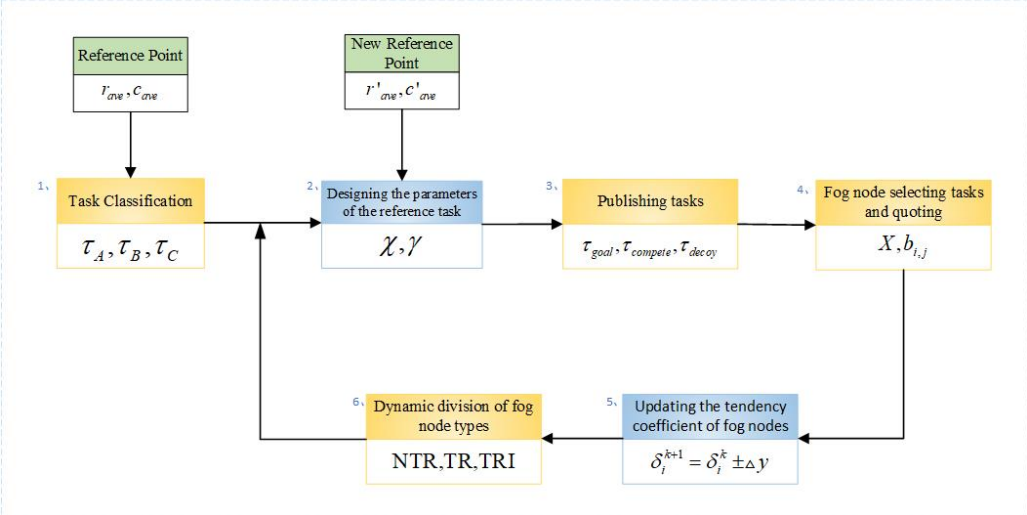


Figure 2. Logical model.

Step 1 involves the platform categorizing tasks into three distinct categories based on their reference point: high-cost and high-reward, low-cost and low-reward, and other miscellaneous tasks. In Step 2, the platform designs reference task parameters with optimal reference effect by taking the high-cost and high-reward task as its target. Following this, in Step 3, the platform publishes all tasks including the reference task to fog nodes. Subsequently, in Step 4, fog nodes select a task to complete and make an offer to the platform while updating their propensity coefficient in Step 5. Finally, in Step 6, dynamic division of fog nodes occurs according to changing tendency coefficients which affects different types of fog node's reference task design.

3.2. The Design of Incentive Mechanism Based on Frame of Reference

3.2.1. Task Classification

This section classifies the computing tasks that need to be offloaded in the environment. Firstly, the average remuneration $r_{ave} = \frac{1}{J}\sum_{j=1}^{J} r_j$ and the average cost $c_{ave} = \frac{1}{J}\sum_{j=1}^{J} c_j$ of j tasks in the environment are calculated, and then the tasks are divided into three categories based on the average value: if the remuneration of the task is higher than the average value, and the cost of the task is also higher than the average value, then the task is assigned to the

class A with high remuneration and high cost. The tasks whose task remuneration and cost are lower than the average value are assigned to class B with low remuneration and low cost. The remaining tasks will be assigned to class C. The specific process is shown in Algorithm 1.

Algorithm 1 Task classification

Input: $\Gamma = \{\tau_1, \tau_2, \cdots, \tau_J\}$
Output: τ_A, τ_B, τ_C

$r_{ave} = \frac{1}{J}\sum_{j=1}^{J} r_j$,$c_{ave} = \frac{1}{J}\sum_{j=1}^{J} c_j$;
$\tau_A, \tau_B, \tau_C \leftarrow \emptyset$;
for $j = 1 : J$ **do**
    if $r_j > r_{ave}$&&$c_j > c_{ave}$ **then**
        $\tau_A \leftarrow \tau_j$;
    else if $r_j < r_{ave}$&&$c_j < c_{ave}$ **then**
        $\tau_B \leftarrow \tau_j$;
    else $\tau_C \leftarrow \tau_j$
    end if
end for
return τ_A, τ_B, τ_c;

Algorithm 1 divides the tasks into three categories according to the task information published by mobile devices in the environment. Each type of task $\tau_{\text{type}}(\text{type} \in \{A, B, C\})$ contains two attributes $\tau_{\text{type}}(r_{\text{type}}, c_{\text{type}})$ of remuneration r_{type} and cost c_{type} . The average value of this kind of task is taken as its attribute value. At this time, the reference point of the task in the environment is $m(r_{\text{ave}}, c_{\text{ave}})$.

In order to reflect the comprehensive attraction of computing tasks to fog nodes, the attraction function of class type$(\text{type} \in \{A, B, C\})$ tasks based on reference points is constructed. The fog node determines whether to participate in the offloading task according to the attraction value of each type of task. When the attraction value of one type of task is greater than that of other types of tasks, the fog node chooses to participate in the offloading task.

In order to eliminate the difference in the dimension of the values of the two attributes, it is necessary first to scale the original data so that the dimensions of the two attributes are on the same scale. This paper uses linear normalization to scale the value of the remuneration to the same dimension as the cost, as shown in Formula (1):

$$r_i = (c_{\max} - c_{\min})\left(\frac{r_i{}' - r_{\min}}{r_{\max} - r_{\min}}\right) + c_{\min}, \tag{1}$$

where $r_i{}'$ and r_i represent the values before and after data normalization, respectively; $r_{\max}$, $r_{\min}$, $c_{\max}$ and $c_{\min}$ represent the maximum and minimum values of the two attributes in the sample data, respectively.

In this paper, the attraction values of the task on the two attributes are added to obtain the overall attraction value of the task. The higher the attraction value of the task, the more likely it is to be selected by the fog node. The attraction function of τ_{type} based on the reference point is shown in Formula (2):

$$O(\tau_{\text{type}} \mid m) = M(r_{\text{type}} \mid r_{\text{ave}}) + N(c_{\text{type}} \mid c_{\text{ave}}), \tag{2}$$

where $M(r_{\text{type}} \mid r_{\text{ave}})$ denotes the remuneration attraction of task τ_{type} based on the reference point's remuneration r_{ave}; $N(c_{\text{type}} \mid c_{\text{ave}})$ denotes the cost attraction of task τ_{type} based on the reference point's cost c_{ave}. According to the definition of frame of reference, the attractiveness of the remuneration above the reference point to the task is positive, and the attractiveness of the remuneration below the reference point to the task is negative. The time cost attribute is the opposite of the remuneration attribute. Therefore, we design the remuneration and time cost attraction functions as shown in Formulas (3) and (4), respectively:

$$M(r_{\text{type}} \mid r_{\text{ave}}) = \begin{cases} (r_{\text{type}} - r_{\text{ave}})^{\alpha} & r_{\text{type}} - r_{\text{ave}} \geq 0 \\ -\lambda(r_{\text{ave}} - r_{\text{type}})^{\beta} & r_{\text{type}} - r_{\text{ave}} < 0 \end{cases}, \tag{3}$$

$$N(c_{\text{type}} \mid c_{\text{ave}}) = \begin{cases} -\lambda(c_{\text{type}} - c_{\text{ave}})^{\beta} & c_{\text{type}} - c_{\text{ave}} \geq 0 \\ (c_{\text{ave}} - c_{\text{type}})^{\alpha} & c_{\text{type}} - c_{\text{ave}} < 0 \end{cases}, \tag{4}$$

where $0 < \alpha = \beta < 1$ denotes the marginal diminishing sensitivity of returns and losses to the attraction. $\lambda > 1$ indicates loss aversion, that is, the change in the loss interval is steeper than the gain interval. This is because, compared with the reference point, the more the remuneration, the more the income, and the less the remuneration, the greater the loss. The acquisition of the same unit is not as affected by the loss of the same unit. Similarly, the higher the cost than the reference point, the greater the loss.

3.2.2. Design of Reference Task

In this paper, the task in class A is the target task $\tau_{\text{goal}}(r_{\text{goal}}, c_{\text{goal}}) = \tau_A(r_A, c_A)$, then the task in class B is its competitive task $\tau_{\text{compete}}(r_{\text{compete}}, c_{\text{compete}}) = \tau_B(r_B, c_B)$, and the reference task $\tau_{\text{decoy}}(r_{\text{decoy}}, c_{\text{decoy}})$ is to be designed. Since the reference task needs to form a reference relationship with τ_{goal}, the remuneration reference factor χ and the cost reference factor γ are used to reflect the relationship between the two types of tasks in terms of attributes. And the attribute values of the reference task are designed, where $\chi = \frac{r_{\text{decoy}}}{r_{\text{goal}}} \in (0, 1]$, $\gamma = \frac{c_{\text{decoy}}}{c_{\text{goal}}} \in [1, +\infty)$. After adding the reference item, the reference point is $m'(r'_{\text{ave}}, c'_{\text{ave}})$, where $r'_{\text{ave}} = \frac{r_{\text{goal}} + r_{\text{compete}} + r_{\text{decoy}}}{3}$, $c'_{\text{ave}} = \frac{c_{\text{goal}} + c_{\text{compete}} + c_{\text{decoy}}}{3}$.

Definition 1. *(Reference effect). The effect of the reference effect is defined by the intensity coefficient is the difference between the attraction difference between the target task and the competitive task after adding the reference task and the attraction difference between the target task and the competitive task before adding the reference task, as shown in Equation (5):*

$$E = (O'(\tau_{goal} \mid m') - O'(\tau_{compete} \mid m')) - (O(\tau_{goal} \mid m) - O(\tau_{compete} \mid m)), \tag{5}$$

According to the definition of the formula, $E > 0$ indicates that a positive reference effect has occurred, and the larger the value of E, the better the reference effect on the target task. $E < 0$ indicates that a negative reference effect has occurred, and the smaller the value of E, the better the reference effect on the competitive task. $E = 0$ indicates that there is no reference effect.

Theorem 1. *When the positive reference function $E > 0$ is obtained, the range of the effective reference factor is $\chi \in [\frac{3\frac{1}{J}\sum_{j=1}^{J} r_j - (r_A + r_B)}{r_A}, 1]$,$\gamma \in [1, \frac{2c_A - c_B}{c_A}]$.*

Proof of Theorem 1. After adding the reference task $\tau_{\text{decoy}}(r_{\text{decoy}}, c_{\text{decoy}})$, $r_A > r'_{ave} > r_B$, $c_A > c'_{ave} > c_B$, expand and simplify to obtain $\frac{2r_B - r_A}{r_A} \leq \chi \leq \frac{2r_A - r_B}{r_A}$, $\frac{2c_B - c_A}{c_A} \leq \gamma \leq \frac{2c_A - c_B}{c_A}$. Because of $r_A > r_B$ and $c_A > c_B$, $\frac{2r_B - r_A}{r_A} < 1$, $\frac{2r_A - r_B}{r_A} > 1$, $\frac{2c_B - c_A}{c_A} < 1$, $\frac{2c_A - c_B}{c_A} > 1$. And because of $\chi \in (0, 1]$ and $\gamma \in [1, +\infty)$, we can obtain $\chi \in [\frac{2r_B - r_A}{r_A}, 1]$, $\gamma \in [1, \frac{2c_A - c_B}{c_A}]$. Then, according to Definition 1, the attraction Formula (2) is substituted into Formula (5) to expand $E = ((M(r_A \mid r'_{ave}) + N(c_A \mid c'_{ave})) - (M(r_B \mid r'_{ave}) + N(c_B \mid c'_{ave}))) - ((M(r_A \mid r_{ave}) + N(c_A \mid c_{ave})) - (M(r_B \mid r_{ave}) + N(c_B \mid c_{ave}))) > 0$, and the reference point formula is simplified to obtain $\chi \in [\frac{3\frac{1}{J}\sum_{j=1}^{J} r_j - (r_A + r_B)}{r_A}, 1]$, $\gamma \in [\frac{3\frac{1}{J}\sum_{j=1}^{J} c_j - (c_A + c_B)}{c_A}, +\infty)$. Taking the intersection, we can, respectively, obtain the range of remuneration reference factor and cost reference factor that produce positive reference effect $\chi \in [\frac{3\frac{1}{J}\sum_{j=1}^{J} r_j - (r_A + r_B)}{r_A}, 1]$, $\gamma \in [1, \frac{2c_A - c_B}{c_A}]$. □

Theorem 1 discusses the value range of the effective reference factor when obtaining the positive reference effect combined with the reference effect discussed in this paper, that is, the reference task is equal to the target task in terms of remuneration, and the cost is higher than the target task. When the reference factors are $\chi = 1$ and $\gamma = frac2c_A - c_Bc_A$, respectively, the maximum reference effect is obtained.

According to the reference factor obtained by Theorem 1, the reference task is designed and added to the task set. The specific process is as shown in Algorithm 2:

Algorithm 2 Design of reference task

Input: $\Gamma = \{\tau_A, \tau_B, \tau_C\}$

Output: $\Gamma = \left\{\tau_{goal}, \tau_{compete}, \tau_{decoy}\right\}$

1: Let $\tau_{goal} = \tau_A$,$\tau_{compete} = \tau_B$;
2: Create τ_{decoy};
3: $\chi = 1, \gamma = \frac{2c_A - c_B}{c_A}$;
4: Let $r_{decoy} = \chi r_{goal}, c_{decoy} = \gamma c_{goal}$;
5: **if** $\tau_C \neq \emptyset$ **then**
6: $\quad \tau_C = \tau_{decoy}$;
7: **else**
8: $\quad$ add task τ_{decoy} to Γ;
9: **end if**
10: return $\Gamma = \left\{\tau_{goal}, \tau_{compete}, \tau_{decoy}\right\}$;

Algorithm 2 first calculates the time reference factor value and the remuneration reference factor value to obtain the maximum reference effect according to the task attribute, and then designs the reference task (lines 2–4) according to the value. Finally, the reference task is added to the original task set and published in the form of three types of tasks (lines 5–9).

3.2.3. The Tendency Change in Fog Nodes

In this model, we only consider the two attributes of remuneration and cost of computing tasks. Therefore, for fog nodes, only their remuneration tendency and cost tendency are considered. By defining the remuneration tendency coefficient $\delta_i \in [0, 1]$ of the fog node, the cost tendency coefficient of the fog node is $1 - \delta_i$. $\delta_i > 0.5$ means the remuneration-oriented node; $\delta_i < 0.5$ means the cost-oriented node; $\delta_i = 0.5$ means the non-oriented node. The tendency value of the second round node to the class task is shown in Formula (6):

$$P^k_{i,\,\text{type}} = \delta^k_i M\left(r_{\text{type}} \mid r_{\text{ave}}\right) + \left(1 - \delta^k_i\right) N\left(c_{\text{type}} \mid c_{\text{ave}}\right), \tag{6}$$

where $P^k_{i,\,\text{type}}$ represents the tendency value of fog node i to task $\tau_{\text{type}}(\text{type} \in \{A, B\})$, the task remuneration attraction value $M\left(r_{\text{type}} \mid r_{\text{ave}}\right)$ and the task cost attraction value $N\left(c_{\text{type}} \mid c_{\text{ave}}\right)$ are obtained by Formulas (3) and (4), respectively.

Next, the tendency model of fog nodes is established. The fog node will dynamically adjust its tendency coefficient according to each round of the task set. If the fog node is more inclined to the target task than the competitive task, it will increase its tendency coefficient. That is to say, it is more inclined to remuneration-based tasks. On the contrary, it will reduce their tendency coefficient. That is, it is more inclined to cost-based tasks. If the tendency degree of the fog node to the target task is equal to the competitive task, or when the tendency coefficient of the fog node has reached 0 or 1, the tendency coefficient will not be changed. The specific update rules are shown in Formula (7):

$$\delta_i^{k+1} = \begin{cases} \delta_i^k + \Delta y & P_{i,A}^k > P_{i,B}^k \\ \delta_i^k - \Delta y & P_{i,A}^k < P_{i,B}^k \\ \delta_i^k & P_{i,A}^k = P_{i,B}^k \Big\| \delta_i^k = 0 \Big\| \delta_i^k = 1 \end{cases}, \tag{7}$$

where δ_i^{k+1} represents the tendency coefficient of fog node i in round $k+1 (k \geq 0)$, and Δy is the increase or decrease in the tendency coefficient. The influence of the reference effect on the tendency is reflected in the degree of tendency change, so Δy is obtained by the tendency ratio of the fog node on the two attributes of remuneration and cost, as shown in Formula (8):

$$\Delta y_i = \frac{\delta_i (M_A - M_B)}{(1 - \delta_i)(N_B - N_A)}, \tag{8}$$

According to the results of task classification in Section 3.2.1, $\delta_i (M_A - M_B)$ indicates the tendency change in fog nodes from B to A in remuneration attribute. $(1 - \delta_i)(N_B - N_A)$ represents the tendency change caused by the improvement in the fog node from class A to class B in cost attribute.

Lemma 1. *Under the RIM mechanism, the total number of selected target tasks increases.*

Proof of Lemma 1. According to the derivation of Theorem 1, after adding the reference task τ_{decoy}, $O(\tau_{\text{goal}} \mid m') > O(\tau_{\text{goal}} \mid m)$, that is, the change of the reference point makes the attraction of the target task increase. Therefore, the tendency value of the fog node to the target task is increased, and the total number of selected target tasks is increased. □

Next, we discuss how fog nodes select tasks. In order to make more fog nodes participate in the offloading of high-cost tasks (i.e., tasks in class A), the tendency value of fog nodes to tasks in class A should be increased as much as possible. For the remuneration-oriented node, because the tendency coefficient $\delta_i > 0.5$ is applicable to the situation discussed in Section 3.2.2 of this paper, according to Theorem 1, $P_{i,A}^k > P_{i,B}^k$ is valid obviously. Therefore, the remuneration-oriented node will choose the tasks in class A. Through Lemma 1, it can be seen that after adding the reference task, under the action of the RIM mechanism, the tendency value of the non-oriented node to the tasks in class A will be greater than the tasks in class B. For the cost-oriented node, the range of its tendency coefficient for tendency reversal is discussed by Theorem 2.

Theorem 2. *If the cost-oriented node is to be reversed to the remuneration-oriented node after a round of task selection, the value range of the tendency coefficient δ_i^k must satisfy $\delta_i^k \in \left(\frac{3}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)} - \sqrt{\left[\frac{1}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2 + \left[\frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2}, 0.5 \right)$.*

Proof of Theorem 2. It is assumed that the type of task selected by the fog node in the two rounds changes, that is, the tendency value of the fog node in the round $k-1$ is $P_{i,A}^{k-1} < P_{i,B}^{k-1}$ and in the round k is $P_{i,A}^k > P_{i,B}^k$. According to Formula (7), the tendency coefficient changes to $\delta_i^{k+1} = \delta_i^k + \Delta y > 0.5$ in the round $k+1$, indicating that the tendency reversal occurs. The inequality is expanded according to Formula (8): $\delta_i^k + \frac{\delta_i^k (M_A - M_B)}{(1 - \delta_i^k)(N_B - N_A)} > 0.5$. According to the design of the reference task, $\gamma > 1$, then $c'_{\text{ave}} > c_{\text{ave}}$. Then, $N_B - N_A > 0$, the same as $M_A - M_B > 0$. The simplified inequality is shown in Formula (9):

$$(N_B - N_A)\left(\delta_i^k\right)^2 - [1.5(N_B - N_A) + (M_A - M_B)]\delta_i^k + 0.5(N_B - N_A) < 0, \tag{9}$$

The left side of Formula (9) is a quadratic function about δ_i^k. Let $a = (N_B - N_A)$, $b = -[1.5(N_B - N_A) + (M_A - M_B)]$, $c = 0.5(N_B - N_A)$, solve the inequality according

to the properties of the quadratic function. Firstly, it is known that $a = (N_B - N_A) > 0$ indicates that the function opening is upward, so the solution of the original inequality is $\frac{-b-\sqrt{b^2-4ac}}{2a} < \delta_i < \frac{-b+\sqrt{b^2-4ac}}{2a}$. Substitute into the expression of a, b, c, because $\frac{-b+\sqrt{b^2-4ac}}{2a} = \frac{[1.5(N_B-N_A)+(M_A-M_B)]+\sqrt{b^2-4ac}}{2(N_B-N_A)} > \frac{1.5(N_B-N_A)}{2(N_B-N_A)} > 0.5$ does not meet, discard. Simplifying another solution, the value range of δ_i^k is: $\delta_i^k \in \left(\frac{3}{4} + \frac{(M_A-M_B)}{2(N_B-N_A)} - \sqrt{\left[\frac{1}{4} + \frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2 + \left[\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2}, 0.5\right)$. □

Theorem 2 discusses the range of the tendency coefficient of the cost-oriented node that can reverse under the reference effect. On the whole, when the tendency coefficient of the fog node satisfies $\delta_i^k > \frac{3}{4} + \frac{(M_A-M_B)}{2(N_B-N_A)} - \sqrt{\left[\frac{1}{4} + \frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2 + \left[\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2}$, the fog node can improve the tendency to the target task under the influence of the reference task, so as to select the target task.

3.2.4. Analysis of Node Quotation

After selecting the task, the fog node determines the quotation of the task according to the cost and its tendency to the task. $C_{i,j}(c_j, f_i)$ represents the cost of the fog node i selecting task τ_j, and $C_{i,j}(c_j, f_i) = \frac{c_j}{f_i}$, where c_j represents the time required for the task τ_j to be offloaded, and f_i represents the clock frequency of the fog node i.

According to the above analysis, the quotation model of the fog node is established. The quotation $b_{i,j}$ of the computing task τ_j by the fog node i is related to the cost of completing the task and the tendency to the task, as shown in Formula (10):

$$b_{i,j} = C_{i,j}(c_j, f_i) - \max(P_{i,type^*} - P_{i,T-type^*}), j \in \text{type}^*, \tag{10}$$

where T represents all task types, $T = \{A, B, decoy\}$. $type^*$ represent the task types that the node chooses to offload, and $T - \text{type}^*$ represents other class tasks. The selection of fog nodes and tasks is represented by the matching matrix X, as shown in Equation (11):

$$X = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1J} \\ a_{21} & a_{22} & \dots & a_{2J} \\ \dots & \dots & \dots & \dots \\ a_{I1} & a_{I2} & \dots & a_{IJ} \end{bmatrix}, \tag{11}$$

All elements of the first round are zero. After the fog node quotes, the matrix element corresponding to the fog node i and task τ_j is $a_{i,j} = 1$, otherwise it is $a_{i,j} = 0$. If the matching matrix has at most one element of 1 in each column, the matching matrix is output. Otherwise, the task τ_j is selected by multiple fog nodes, and fog nodes are counter-selected. This process determines the fog node that wins the task and the final transaction price of the task through the Vickrey auction. That is, the fog node with the lowest quotation becomes the task winner, and the second lowest quotation $b'_{i,j}$ of the round is the final remuneration $fb_{i,j}$ to be paid for the task. At the same time, the matching matrix elements corresponding to the successfully matched fog nodes and tasks are set to $a_{i,j} = 1$, and the other aspects of the column are modified to 0.

Therefore, if the fog node participates in the quotation and finally obtains the task, the utility of the fog node is defined as the difference between the transaction price of the task and its quotation for the task; otherwise, the utility of the fog node is 0, as shown in Formula (12):

$$u_i = a_{i,j}\left(fb_{i,j} - b_{i,j}\right), \tag{12}$$

According to the analysis of the behavior pattern of fog nodes in Section 3.2.3, firstly, the tendency of fog nodes to tasks will change with the addition of reference tasks. Then, the fog node selects the task type according to the tendency value, and quotes the selected

task according to the cost and tendency. Based on this, this paper proposes a fog node selection algorithm under the RIM mechanism, as shown in Algorithm 3.

Algorithm 3 Task selection

Input: $\Gamma = \left\{\tau_{goal}, \tau_{compete}, \tau_{decoy}\right\}, I$

Output: $X, fb_{i,j}, \Gamma^k$

1: 1. Initialize:$X = 0; k = 1;$
2: **for** $i = 1 : I$ **do**
3: $P_{i,A}^k = \delta_i^k M(r_A \mid r_{\text{ave}}) + \left(1 - \delta_i^k\right) N(c_A \mid c_{\text{ave}})$
4: $P_{i,B}^k = \delta_i^k M(r_B \mid r_{\text{ave}}) + \left(1 - \delta_i^k\right) N(c_B \mid c_{\text{ave}})$
5: **if** $P_{i,A}^k > P_{i,B}^k$ **then**
6: $\delta_i^{k+1} = \delta_i^k + \Delta y$; type$^* = A$;
7: **else if** $P_{i,A}^k < P_{i,B}^k$ **then**
8: $\delta_i^{k+1} = \delta_i^k - \Delta y$; type$^* = B$;
9: **else if** $P_{i,A}^k = P_{i,B}^k \parallel \delta_i^k = 0 \parallel \delta_i^k = 1$ **then**
10: $\delta_i^{k+1} = \delta_i^k$;
11: **end if**
12: **for** $j = 1 : J$ **do**
13: $b_{i,j} = C_{i,j}(c_j, f_i) - max(P_{i,\text{type}^*} - P_{i,T-\text{type}^*})$;
14: **end for**
15: Select max $b_{i,j}$ in type* as the final task quotation of the node i
16: **end for**
17: **for** $j = 1 : J$ **do**
18: select the optimal bid $b_{i,j}$ and set $a_{i,j} = 1$ in X;
19: **if** $\sum_{i=1}^{I} a_{i,j} \geq 1$ **then**
20: select the sub-optimal bid $b'_{i,j} \rightarrow fb_{i,j}$;
21: **else**
22: add τ_j to Γ^k;
23: **end if**
24: **end for**
25: return $X, fb_{i,j}, \Gamma^k$;

In Algorithm 3, all elements of the matching matrix are initialized to 0. Then, update the tendency coefficient of all fog nodes according to the Section 3.2.3 fog node tendency update strategy (lines 2–11) . Then, according to the fog node quotation formula in this section, the quotation of each fog node for each task is calculated, and the highest quotation is selected as the final task quotation of the fog node (lines 12–16). Then, traverse all tasks and the matching matrix elements corresponding to the task and the fog node with the lowest quotation are set to 1 (lines 17–18). When the number of fog nodes for the task quotation is greater than 1, the second-lowest quotation is used as the final task remuneration (lines 19–20). Otherwise, the failed task is added to the failed task set to wait for the next round to be selected (lines 21–24).

3.2.5. Task Pushing

According to the range of the tendency coefficient of the cost-oriented nodes that reverse tendency under the reference effect obtained by Theorem 2, we divide the fog nodes into the following three groups:

(1) No Trend Reversal group (NTR): the node tend to neutral and remuneration, that is, $\delta_i^k \in [0.5, 1]$.

(2) Tendency reversal group (TR): the node that can change from the cost-oriented to the remuneration-oriented under the influence of the reference task, that is, $\delta_i^k \in \left(\frac{3}{4}+\frac{(M_A-M_B)}{2(N_B-N_A)}-\sqrt{\left[\frac{1}{4}+\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2+\left[\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2}, 0.5\right)$.

(3) Tendency reversal influence group (TRI): the node that always is remuneration-oriented no matter how to design the reference task, that is, $\delta_i^k \in [0, \frac{3}{4}+\frac{(M_A-M_B)}{2(N_B-N_A)}-\sqrt{\left[\frac{1}{4}+\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2+\left[\frac{(M_A-M_B)}{2(N_B-N_A)}\right]^2}]$.

Because the tendency coefficient is variable, the group of fog nodes also changes dynamically. n_{type} represents the total number of three types of fog nodes, type $\in \{NTR, TR, TRI\}$.

According to the classification, we mainly design reference tasks for TR and TRI to reduce the quotation of nodes, and discuss the design of reference factors through Theorem 3.

Theorem 3. *For TR, the best result is obtained when* $\chi = 1, \gamma = \frac{2c_A - c_B}{c_A}$. *For TRI, the best result is obtained when* $\chi = 1, \gamma = \frac{2c_A - c_B}{c_B}$.

Proof of Theorem 3. In order to reduce the quotation $b_{i,j}$ of nodes in TR, it is necessary to increase $P_{i,A} - P_{i,B}$. Let $L = P_{i,A} - P_{i,B}$, bringing this into Formula (6) can obtain $L = \delta_i^k[M(r_A \mid r_{ave}) - M(r_B \mid r_{ave})] + (1-\delta_i^k)[N(c_A \mid c_{ave}) - N(c_B \mid c_{ave})]$. Expansion can obtain $L = \delta_i^k[(r_A - r_{\text{ave}})^\alpha + \lambda(r_{\text{ave}} - r_B)^\beta] + (1-\delta_i^k)[-\lambda(c_A - \ldots c_{\text{ave}})^\beta - (c_{\text{ave}} - c_B)^\alpha] = \delta_i^k[(\frac{2-\chi}{3}r_A - \frac{1}{3}r_B)^\alpha + \lambda(\frac{1+\chi}{3}r_A - \frac{2}{3}r_B)^\beta] + (1-\delta_i^k)[-\lambda(\frac{2-\gamma}{3}c_A - \frac{1}{3}c_B)^\beta - (\frac{1+\gamma}{3}c_A - \frac{2}{3}c_B)^\alpha]$. $L = \delta_i^k(\frac{\chi+4}{3}r_A - \frac{5}{3}r_B) + (1-\delta_i^k)(\frac{\gamma-5}{3}c_A + \frac{4}{3}c_B)$ can be obtained by approximate treatment $\alpha = \beta = 1, \lambda = 2$. Therefore, it can be seen that the reference factor χ and γ are positively correlated with $L = P_{i,A} - P_{i,B}$. Combined with Theorem 1, it can be seen that the best effect is obtained when $\chi = 1, \gamma = \frac{2c_A - c_B}{c_A}$.

Similarly, for the nodes in TRI, it is necessary to increase $L = P_{i,A} - P_{i,B}$. According to the discussion of Theorem 1, it can be inferred that the task in class B is the target task, and the best effect is obtained when $\chi = 1, \gamma = \frac{2c_A - c_B}{c_B}$. □

Next, we add different reference tasks to push tasks to nodes for different types of fog nodes. The specific process is as shown in Algorithm 4.

In Algorithm 4, the fog nodes are divided into three groups: NTR, TR and TRI, according to the tendency coefficient (lines 1–9). If there is a task in the failure task set, the failed task is first reclassified by Algorithm 1. Then, the bait task attributes are redesigned according to the characteristics of three groups of fog nodes. Then, the bait task is added to the task set to push the three groups of fog nodes (lines 10–27).

Algorithm 4 Task pushing

Input: $\Gamma^k, b^k_{i,j}, \delta^k_i$

Output: $\Gamma^{k+1}_{\text{type}} = \left\{\tau_{goal}, \tau_{compete}, \tau_{decoy}\right\}$

for $i = 1 : I$ **do**
  if $\delta^k_i > 0.5$ **then**
    $i \rightarrow n_{NTR}$;
  else if $\frac{3}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)} - \sqrt{\left[\frac{1}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2 + \left[\frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2} < \delta^k_i < 0.5$ **then**
    $i \rightarrow n_{TR}$;
  else if $\delta^k_i < \frac{3}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)} - \sqrt{\left[\frac{1}{4} + \frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2 + \left[\frac{(M_A - M_B)}{2(N_B - N_A)}\right]^2}$ **then**
    $i \rightarrow n_{TRI}$;
  end if
end for
while $\Gamma^k_A \neq \emptyset$ **do**
  $\Gamma^k \rightarrow Algorithm1$;
  for $i = 1 : i$ **do**
    if $i \in NTR$ **then**
      set $\chi_{k+1} = \chi_k, \gamma_{k+1} = \gamma_k$;
      let $r_{decoy} = \chi_{k+1} r_A, c_{decoy} = \gamma_{k+1} c_A$;
      add task τ_{decoy} to Γ^{k+1}_A;
    else if $i \in TR$ **then**
      set $\chi_{k+1} = 1, \gamma_{k+1} = \frac{2c_A - c_B}{c_A}$;
      let $r_{decoy} = \chi_{k+1} r_A, c_{decoy} = \gamma_{k+1} c_A$;
      add task τ_{decoy} to Γ^{k+1}_A;
    else if $i \in TRI$ **then**
      set $\chi_{k+1} = 1, \gamma_{k+1} = \frac{2c_A - c_B}{c_B}$;
      let $r_{decoy} = \chi_{k+1} r_B, c_{decoy} = \gamma_{k+1} c_B$;
      add task τ_{decoy} to Γ^{k+1}_B;
    end if
  end for
end while
return $\Gamma^{k+1}_{\text{type}}$

3.2.6. Example

Next, this paper illustrates the situation of fog node selection task and the change in the tendency coefficient before and after adding reference task through an example. The parameter settings of offloading tasks and fog nodes in the environment are shown in Tables 2 and 3.

Table 2. Offloading task attributes.

Offloading Task	τ_1	τ_2	τ_3
The cost of task	10	4	16
The remuneration of task	10	4	10

Table 3. Fog node attributes.

Fog Node	n_1	n_2	n_3	n_4	n_5	n_6
The tendency coefficient of fog node	0.15	0.63	0.61	0.58	0.45	0.2

First of all, τ_3 is the reference task, and the reference points before and after the reference task are $m(r_{ave}, c_{ave}) = m(7,7)$ and $m'(r'_{ave}, c'_{ave}) = m'(8,10)$, respectively. The task attributes are $r_A = r_{goal} = 10$,$c_A = c_{goal} = 10$,$r_B = r_{compete} = 4$,$c_B = c_{compete} = 4$.

According to Theorem 2, the tendency coefficient range of the cost tendency node with tendency reversal is $\delta_i^k \in (0.35, 0.5)$. Therefore, only the fog nodes with the tendency coefficient falling in this interval will have a tendency reversal. Next, the fog node is calculated as an example.

The tendency values of n_1 for τ_1 and τ_2 are as follows, according to Formula (6):

$P_{1,A} = 0.15 \times (10-7)^{0.88} - 0.85 \times 2.25 \times (10-7)^{0.88} \approx -4.634$,
$P_{1,B} = -0.15 \times 2.25 \times (7-4)^{0.88} + 0.85 \times (7-4)^{0.88} \approx 1.348$;
$P_{1,\text{goal}} = 0.15 \times (10-8)^{0.88} - 0.85 \times 2.25 \times (10-10)^{0.88} \approx 0.276$,
$P_{1,\text{compete}} = -0.15 \times 2.25 \times (8-4)^{0.88} + 0.85 \times (10-4)^{0.88} \approx 2.970$.

From the results, $P_{1,A} < P_{1,B}$, before adding the reference task, n_1 selects τ_2. $P_{1,goal} < P_{1,compete}$, after adding the reference task, n_1 still selects τ_2. For the rest of the fog nodes, the same as above, the selection of n_2,n_3,n_4 and n_6 before and after adding the reference task is unchanged. n_2,n_3 and n_4 select τ_1, n_6 selects τ_2. The propensity values of n_5 for τ_1 and τ_2 are calculated as follows:

$P_{5,A} = 0.45 \times (10-7)^{0.88} - 0.55 \times 2.25 \times (10-7)^{0.88} \approx -2.071$,
$P_{5,B} = -0.45 \times 2.25 \times (7-4)^{0.88} + 0.55 \times (7-4)^{0.88} \approx -1.216$;
$P_{5,\text{goal}} = 0.45 \times (10-8)^{0.88} - 0.55 \times 2.25 \times (10-10)^{0.88} \approx 0.828$,
$P_{5,\text{compete}} = -0.45 \times 2.25 \times (8-4)^{0.88} + 0.55 \times (10-4)^{0.88} \approx -0.768$.

Before adding the reference task, $P_{5,A} < P_{5,B}$, n_5 selects τ_2. After adding the reference task, $P_{5,goal} > P_{5,compete}$, n_5 selects τ_1.

3.3. Utility Analysis and Mechanism Evaluation

This section will theoretically evaluate the RIM mechanism proposed in this paper from two aspects: utility and mechanism. Section 3.3.1 analyzes the transaction price of the task, the number of offloading tasks and the utility of the platform. Section 3.3.2 evaluates the performance of RIM mechanism.

3.3.1. Utility Analysis

This section compares the utility of the mechanism of this paper. Firstly, it is proved theoretically by Lemma 2 that under the mechanism of this paper, the final transaction price of the task will be reduced. Then, through Lemma 4, it is proved that the mechanism of this paper improves the platform utility.

The utility of the platform is the cost of local execution of all offloading tasks minus the compensation required to offload to the fog node, as shown in Formula (13):

$$U = \sum_{i=1}^{I} \sum_{j=1}^{J} a_{i,j} \left(C_j(c_j, f_j) - fb_{i,j} \right), \tag{13}$$

where f_j represents the clock frequency of the task τ_j, $C_j(c_j, f_j)$ represents the cost of the task executed locally, and $fb_{i,j}$ is the remuneration that the task ultimately needs to pay.

Before calculating the utility of the platform, Lemma 2 is used to analyze the change in $fb_{i,j}$ before and after the RIM mechanism.

Lemma 2. *The task transaction price under the RIM mechanism is lower than that without the RIM mechanism.*

Proof of Lemma 2. In lines 15 to 17 of Algorithm 3, the final transaction price of the task is the second lowest quotation of the fog node in this round, and the quotation of the fog node is given by Formula (10). The cost of the task and the clock frequency of the fog node are the objective environment, and the effect of the RIM mechanism is the same. Therefore,

the quotation discussed here is only determined by the tendency of the fog node to the task. The higher the tendency value, the lower the quotation of the fog node.

(1) For NTR, the added reference term increases the tendency coefficient of the fog node, thereby increasing the tendency value of the task in class A, so that $P^k_{i,A} - P^k_{i,B} > 0$.
(2) For TR, the reference term is added to reverse the fog node from the cost-oriented to the remuneration-oriented, thereby increasing the tendency value of the task in class A, so that $P^k_{i,A} - P^k_{i,B} > 0$.
(3) For TRI, the added reference item reduces the tendency coefficient of fog nodes, thereby increasing the tendency value for the task in class B, so that $P^k_{i,A} - P^k_{i,B} > 0$.

In summary, the RIM mechanism overall reduces the fog node 's bid for the task. Therefore, the final transaction price of the task determined at the second low price is also reduced. □

Lemma 3. *The platform utility under the RIM mechanism is more significant than that without the RIM mechanism.*

Proof of Lemma 3. According to the analysis results of Theorem 3, it can be seen that under the task pushing algorithm of RIM mechanism, the number of offloaded tasks of each type of fog node will be improved. As stated in Lemma 2, the remuneration paid to fog nodes under RIM is lower than the case without RIM. According to Formula (13), the platform utility under the RIM mechanism is more significant than that without the RIM mechanism. □

It can be concluded that the RIM mechanism improves the platform utility while improving the task offloading ratio.

3.3.2. Mechanism Evaluation

The previous section analyzes and compares the utility of the RIM mechanism in this paper. This section theoretically evaluates the performance of the mechanism in this paper, including authenticity, individual rationality and computational efficiency, which are proved by Lemmas 4, 5 and 6, respectively.

Lemma 4. *The RIM mechanism is authentic.*

Proof of Lemma 4. In the design process of the RIM mechanism, this paper uses a third-party service platform as an auctioneer to collect information, match and determine the final transaction remuneration. More importantly, the auctioneer has publishing and supervision functions. When the auctioneer detects that the bid of the fog node is higher than the actual utility, it has the right to cancel the bid of the fog node, that is, $b_{i,j} = 0$. Therefore, the fog node benefits the most only when submitting real task selection and price, effectively ensuring the authenticity of the RIM mechanism. □

Lemma 5. *The RIM mechanism is individually rational.*

Proof of Lemma 5. According to Formula (12) and Algorithm 3, this paper sets the sub-low price as the final transaction price, so the quotation of fog node must not be higher than the final transaction price, that is, $b_{i,j} \leq fb_{i,j} = b'_{i,j}$. Therefore, the utility is non-negative, ensuring the individual rationality of the RIM mechanism. □

Lemma 6. *The RIM mechanism is computationally efficient.*

Proof of Lemma 6. The algorithm with the highest time complexity in RIM is Algorithm 3. First, of all, for each fog node, it is necessary to calculate its tendency value to the target task and the competitive task, so as to update its tendency coefficient. Then, for each task

fog node, the corresponding quotation is calculated and the highest value is selected, so the time complexity of the fog node is $O(IJ)$. Similarly, when the platform selects fog nodes, it also requires sorting their offers, leading to a time complexity of $O(JIlogI)$ on the platform side. In summary, considering that the number of fog nodes is usually greater than 2, the overall time complexity of Algorithm 3 is $O(IJlogI)$. This means that the RIM algorithm converges in polynomial time relative to I and J, and the lemma is proved. □

4. Simulation

In order to evaluate the effect of the reference-based RIM mechanism, we model the fog computing offloading system and simulate the mechanism. This paper analyzes the impact of the RIM mechanism through experiments and compares it with the PMMRA mechanism [19]. PMMRA is a classical incentive mechanism considering multi-node and multi-task in computational offloading system. This experiment uses Java language to simulate on the IntelliJ IDEA platform in the OSX environment.

4.1. Experiment Setting

In order to ensure the fairness of the experimental comparison, the basic parameter settings of RIM are the same as the values of PMMRA. The parameter settings of the experiment are shown in Table 4.

Table 4. Parameter setting.

Variable	Values	Description
I	[10, 60]	Total number of fog nodes
J	50	Total number of tasks
f_i, f_j	[1, 1.5] GHz	The clock frequency of fog node i and mobile device j
r_j	[2, 20]	The remuneration of task τ_j
c_j	[200, 2000]	The cost of task τ_j
δ_i	(0, 1)	The tendency coefficient of fog node i to task remuneration

In the experiment, we consider scenarios where the number of offloading tasks ranges from 5 to 50, and the number of fog nodes ranges from 10 to 60. The CPU clock frequency of each fog node and mobile device is randomly set to 1 to 1.5 GHz. Since different tasks have different execution features, the expected pay for each task varies from 2 to 20, and the time cost varies from 200 to 2000. The fog node tendency coefficient is 0.5 in the range of 0–1, and 1 is the normal distribution of the variance.

4.2. Analysis of Reference Parameters

This section analyzes the impact of the remuneration reference factor χ and the cost reference factor γ on the RIM mechanism. By referring to the effect intensity coefficient and the target task selection rate, the pros and cons of the mechanism are reflected, that is, the ratio of the number of fog nodes of the target task to the total number of fog nodes is selected. These two represent the effect of the frame of reference theoretically and practically. Here, the parameter χ increases from 0 to 1, increasing by 0.1; the parameter γ is from 1 to 2, also increasing by 0.1. At this time, the number of unloading tasks in the environment is 50, and the number of fog nodes is 30.

As shown in Figure 3, according to Theorem 1, as the remuneration reference factor χ increases, the negative reference effect gradually decreases, and then the positive reference effect gradually increases. With the rise of the cost reference factor γ, the reference effect rises first and then decreases, and the reference term fails when $\gamma > 1.7$. The positive reference effect will increase with the increase in the remuneration reference factor within a certain range. This is because the increase in the remuneration reference factor leads to decreased distance between the target item and the reference point in the remuneration dimension, while the distance between the competition item and the reference point in the remuneration dimension increases. According to the definition of task attraction in

Section 3.2.1, the attraction of the target item is slightly lower than that before the reference item is added, while the attraction of the competition item is significantly lower than that before the reference item is added. Finally, the attraction of the target item is higher than that of the competition item. On the whole, when $\chi = 1, \gamma > 1.6$ is optimal.

Figure 4 is the selection rate of the target task in the experiment. It can be seen that the trend is basically consistent with the intensity coefficient value, indicating that the model in this paper is correct.

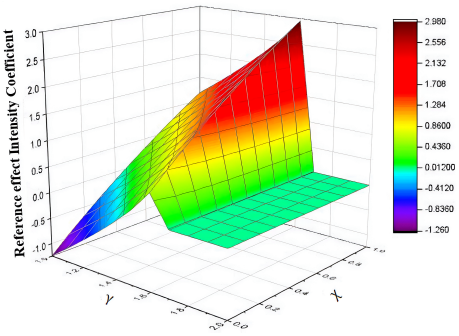


Figure 3. The influence of the reference factor on intensity coefficient.

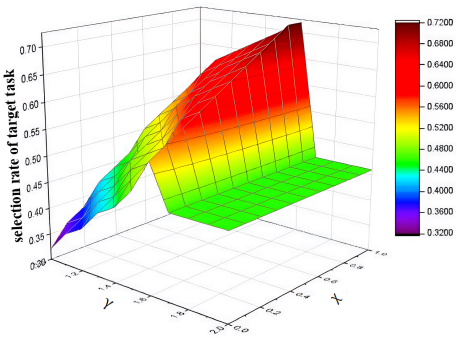


Figure 4. The influence of the reference factor on the selection rate of target task.

4.3. Evaluation of Task Selection

When the number of unloading tasks is constant, with an increase in the number of fog nodes, the selected cases of τ_A and τ_B, and the selected cases before and after adding τ_{decoy}, are represented by τ_A, τ_B, τ_{goal} and $\tau_{compete}$, respectively. As shown in Figure 5, the number of fog nodes selected τ_A and τ_B is basically the same before adding the reference task. However, following the addition of the reference task, there is a significant increase in the number of fog nodes selecting τ_{goal} compared to those selecting $\tau_{compete}$. And, the number of fog nodes selected τ_{decoy} is close to 0, primarily because τ_{decoy}, being the reference task, lacks significant attraction compared to τ_{goal}. So, few fog nodes will select it. It can be seen from Figure 5 that the RIM mechanism promotes the fog node to choose to offload high-cost tasks and the overall task offloading rate.

As shown in Figure 6, the distribution of the time cost of successfully unloaded tasks is shown. After adding the reference task, the task unloading rate with high time cost is higher. The black and red points represent the distribution under the PMMRA mechanism and the RIM mechanism, respectively. It can be seen that the number of task offloading with high time cost under the RIM mechanism is more than that under the PMMRA mechanism. This is because the reference item changes the reference point of the node, and the number of fog nodes selected to participate in the high time cost increases.

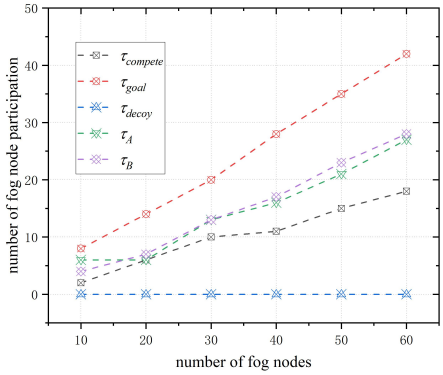


Figure 5. Comparison of the number of fog node selections for each type of task.

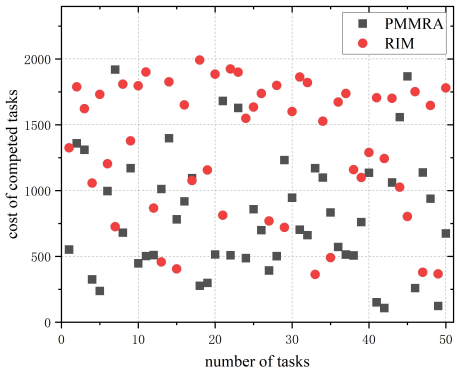


Figure 6. Comparison of the cost distribution of completing tasks under different mechanisms.

4.4. Number of Completed Tasks

This section compares the changes in the number of completed tasks when the total number of fog nodes in the environment remains 30 under the RIM mechanism and the PMMRA mechanism.

It can be seen from Figure 7 that the RIM mechanism is significantly better than the PMMRA mechanism in the number of completed tasks.

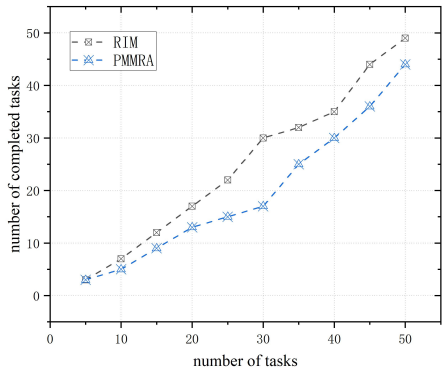


Figure 7. Comparison of the number of completed tasks under different mechanisms.

4.5. Platform Utility

The following subsection presents a comparative analysis of the impact of RIM and PMMRA on platform utility, considering 25 and 50 offloading tasks, respectively. In this scenario, the fog node propensity coefficient follows a normal distribution with a mean value of 0.5 bits within the range of 0–1, while maintaining a variance of 1.

RIM exhibits significantly higher platform utility than PMMRA for different total numbers of fog nodes, as illustrated in Figure 8, when there are 25 offloading tasks. Under the influence of RIM, computing tasks align with their tendency, leading to increased

participation of fog nodes and reduced transaction rewards for successful offloading tasks. Consequently, the overall utility is enhanced. Figure 9 demonstrates that when there are 50 offloading tasks, the platform utility of RIM surpasses that of PMMRA. From Figures 8 and 9, it can be seen that the RIM mechanism can improve the high-cost task offloading rate and the overall task offloading rate without increasing the remuneration of fog nodes.

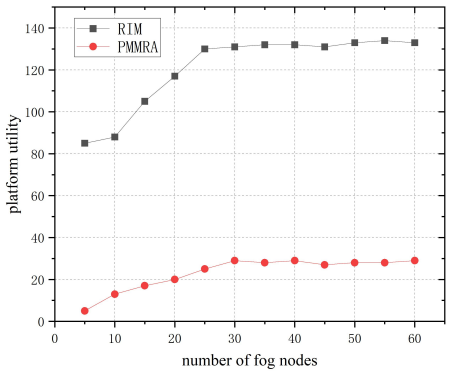


Figure 8. Comparison of platform utility under different mechanisms when task number is 25.

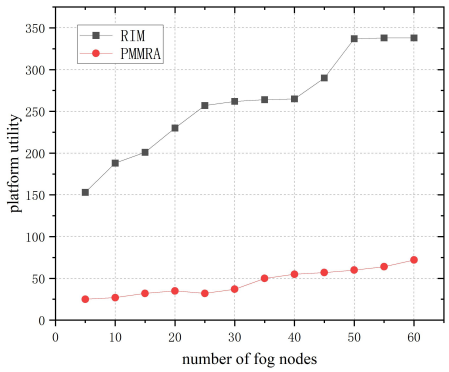


Figure 9. Comparison of platform utility under different mechanisms when task number is 50.

5. Conclusions

This paper primarily investigates the integration of reference frame theory into fog computing offloading. Subsequently, an incentive mechanism called RIM is established based on the characteristics of frame of reference to enhance the selection process for target tasks in fog nodes. Initially, tasks are categorized based on reference points, and task attractiveness is modeled from a task attribute perspective. Next, the impact of reference tasks on offloading numbers for target tasks is analyzed, and a design incorporating reference tasks into the original task set is proposed. Furthermore, a propensity model for fog nodes is developed by classifying nodes according to price and propensity value, followed by pushing new task sets with different types of reference tasks. The experimental results show that RIM can improve the offloading rate of the overall task without increasing the platform's expenditure and ensuring the platform's utility.

In this study, we implement the RIM mechanism in the fog node selection task stage.By adding reference tasks to the task set, the node's tendency to remuneration and cost is changed to promote the offloading rate of the overall task and ensure the utility of the platform. In the near future, we will explore the different characteristics of nodes and expand other attributes of computing tasks, and conduct in-depth analysis of the node selection task model to achieve more accurate node segmentation and task matching.

Author Contributions: D.L., C.Y. and Y.T. designed the project and drafted the manuscript. C.Y. and Y.T. wrote the code. J.L. performed the analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Saha, S.; Habib, M.A.; Adhikary, T.; Razzaque, M.A.; Rahman, M.M.; Altaf, M.; Hassan, M.M. Quality-of-experience-aware incentive mechanism for workers in mobile device cloud. *IEEE Access* **2021**, *9*, 95162–95179. [CrossRef]
2. Rayes, A.; Salam, S. *Internet of Things from Hype to Reality*; Springer: Berlin/Heidelberg, Germany, 2017.
3. Vaquero, L.M.; Rodero-Merino, L. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 27–32. [CrossRef]
4. Tuli, S.; Basumatary, N.; Gill, S.S.; Kahani, M.; Arya, R.C.; Wander, G.S.; Buyya, R. HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments. *Future Gener. Comput. Syst.* **2020**, *104*, 187–200. [CrossRef]
5. Stojmenovic, I.; Wen, S.; Huang, X.; Luan, H. An overview of fog computing and its security issues. *Concurr. Comput. Pract. Exp.* **2016**, *28*, 2991–3005. [CrossRef]
6. Zeng, D.; Gu, L.; Guo, S.; Cheng, Z.; Yu, S. Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. *IEEE Trans. Comput.* **2016**, *65*, 3702–3712. [CrossRef]
7. Qiu, H.; Zhu, K.; Luong, N.C.; Yi, C.; Niyato, D.; Kim, D.I. Applications of auction and mechanism design in edge computing: A survey. *IEEE Trans. Cogn. Commun. Netw.* **2022**, *8*, 1034–1058. [CrossRef]
8. Zhang, Y.; Wang, C.Y.; Wei, H.Y. Parking reservation auction for parked vehicle assistance in vehicular fog computing. *IEEE Trans. Veh. Technol.* **2019**, *68*, 3126–3139. [CrossRef]
9. Chen, M.; Hao, Y. Task offloading for mobile edge computing in software defined ultra-dense network. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 587–597. [CrossRef]
10. Mai, L.; Pan, D.; Song, H.; Wang, C. A T2T-based offloading method: Virtual bank with movement prediction. *IEEE Access* **2018**, *6*, 16408–16422. [CrossRef]
11. Feynman, R. *The Character of Physical Law (1965)*; Cox and Wyman Ltd.: London, UK, 1967.
12. Chernev, A. Context effects without a context: Attribute balance as a reason for choice. *J. Consum. Res.* **2005**, *32*, 213–223. [CrossRef]
13. Du, J.; Jiang, C.; Gelenbe, E.; Zhang, H.; Ren, Y.; Quek, T.Q. Double auction mechanism design for video caching in heterogeneous ultra-dense networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 1669–1683. [CrossRef]
14. Sun, W.; Liu, J.; Yue, Y.; Zhang, H. Double auction-based resource allocation for mobile edge computing in industrial internet of things. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4692–4701. [CrossRef]
15. Liu, L.; Chang, Z.; Guo, X. Socially aware dynamic computation offloading scheme for fog computing system with energy harvesting devices. *IEEE Internet Things J.* **2018**, *5*, 1869–1879. [CrossRef]
16. Chatzopoulos, D.; Ahmadi, M.; Kosta, S.; Hui, P. Have you asked your neighbors? A hidden market approach for device-to-device offloading. In Proceedings of the 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Coimbra, Portugal, 21–24 June 2016; pp. 1–9.
17. Flores, H.; Sharma, R.; Ferreira, D.; Kostakos, V.; Manner, J.; Tarkoma, S.; Hui, P.; Li, Y. Social-aware hybrid mobile offloading. *Pervasive Mob. Comput.* **2017**, *36*, 25–43. [CrossRef]
18. Wijnen, K. Gone But Not Forgotten: The Role of Unacceptable Options in Decision Making. Ph.D. Thesis, Ghent University, Ghent, Belgium, 2007.
19. Wang, Q.; Guo, S.; Liu, J.; Pan, C.; Yang, L. Profit maximization incentive mechanism for resource providers in mobile edge computing. *IEEE Trans. Serv. Comput.* **2019**, *15*, 138–149. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.