Research Article

Task Offloading of Edge Computing Network and Energy Saving of Passive House for Smart City

Yanfang Li, Xiaorong He, and Yuzhu Bian

Hebei University of Architecture, Zhangjiakou, 075000 Hebei, China

Correspondence should be addressed to Yuzhu Bian; byz1546@hebiace.edu.cn

Received 15 February 2022; Revised 4 March 2022; Accepted 7 March 2022; Published 15 April 2022

Academic Editor: Hasan Ali Khattak

Copyright © 2022 Yanfang Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the advent of technologies such as the Internet of Things, edge computing, and 5G, a tremendous amount of structured and unstructured data is being generated from different applications in the smart city environment. In this study, the current problems to be overcome by edge computing (EC) and the basic framework of edge computing are investigated to enhance the sustainable development of the smart city. Three aspects of the edge computing offloading technology are explored including the software-defined network (SDN) controller, offload decision, and resource allocation, and a task offloading model of an edge computing network for a smart city is designed based on data preprocessing. Moreover, the energy-saving design and analysis of passive houses are carried out with Chengdu city as an example. The results reveal that the deployment scheme is feasible. In the passive house design scheme with natural ventilation, external shading, wall heat transfer coefficient of 0.63 W/(m² K), and water storage roof, the annual energy consumption per unit area is the lowest, 18.97 kWh/m², and its energy-saving rate is the highest, 0.77. The findings of the study provide some research experience for increasing the efficiency of smart city edge computing and boosting the smart city’s long-term development.

1. Introduction

The Internet of Things (IoT) has become increasingly significant in our daily lives as information technology has progressed. Interconnected devices can collect and exchange different data amongst themselves through recent communication networks connected by various IoT nodes [1]. Then, a range of IoT applications can provide users with more precise and fine-grained network services. In this situation, IoT techniques are connecting an increasing number of sensors and devices, and these sensors and devices will generate vast amounts of data that will require further processing, offering intelligence to both customers and service providers [2]. All data must be uploaded to centralized servers in traditional cloud computing, and the results must be transmitted back to the sensors and devices after processing. This procedure puts a lot of pressure on the network, especially in terms of bandwidth and resource costs for data transmission. Furthermore, when the data amount grows larger, the network’s performance will deteriorate [3].

The development of a smart city is based on information technologies such as the Internet of Things and cloud computing [4]. The existing cloud computing scenario has high latency [2], making it unable to meet the needs of various emerging businesses. Therefore, edge computing (EC) technology [3] emerges. Edge computing is computing that is close to the source of practice, data, or action, allowing for real-time analysis. It efficiently reduces the burden on the cloud and provides users with faster responses. Task offloading of an EC network is to process the computing offloading task of the mobile terminal in the edge network, which can effectively improve equipment’s performance and energy efficiency [5].

In the literature, various edge computing offloading techniques are proposed. Naourri et al. [6] proposed a three-layer task offloading framework composed of the device layer, cloudlet layer, and cloud layer. Tasks with high computing requirements were assigned to the cloudlet layer and the cloud layer, while tasks with low computing and high communication cost were executed at the device layer, which avoids transmitting a large amount of data to the cloud and effectively
reduces the processing delay. Rodrigues et al. [7] developed a mathematical model of total service delay of the multiaccess EC system based on cyber twin. The system included user mobility, virtual server migration, multiple physical servers in different network layers, front-end and backhaul communication, processing, and content request/cache. Nguyen et al. [8] studied the backscattering-assisted data offloading in the mobile EC of the Internet of Things system based on OFDMA and maximized and optimized the calculation rate through the transmission power, backscatter coefficient, time splitting ratio, and binary decision matrix at the joint gateway. The authors in [9] proposed a computing offloading mode aided by unmanned aerial vehicle (UAV). The problem of minimizing the average mission response time was established and modeled as a Markov decision process by considering the interdependence of UAV missions, dynamic network states, and energy constraints. In [10–12], some mobile edge computing architectures and offloading strategies for remote task computing have been investigated, which deals separately with different minimizing or maximizing objectives, such as minimizing energy consumption or execution delay, maximizing the offloaded tasks ratio, or the system profit within computing devices or fog nodes. The existing task offloading models of EC networks take the EC offloading decision and resource allocation as the research object to reduce the delay of the task offloading model of EC. However, there is little research on the deployment of edge servers in practical application scenarios.

In this study, a task offloading model of EC for the smart city is designed through research on EC and EC offloading tasks. Moreover, a delay model and a cost model are established. The edge server deployment scheme suitable for the smart city is studied. Finally, the passive house design is carried out with Chengdu as an example. The offloading model of EC for the collaborative processing of mobile edge servers, edge cloud servers, and local devices has been innovatively established, providing a reference experience for reducing the delay and cost of EC.

The rest of the manuscript is organized into three sections: Section 2 is about material and methods and provides a complete description of the data collection and experimental process. Section 3 demonstrates the results and illustrates the different parameters. Finally, the conclusion is presented in Section 4.

2. Materials and Methods

2.1. Edge Computing. Edge computing is a distributed information technology (IT) architecture in which client data is processed as close to the source as possible at the network’s periphery. The essence of EC is relative to cloud computing. Cloud computing is a computing method that puts data computing into the cloud server [13], while EC uses edge devices to calculate and process data near the data source, which reduces the transfer process of data in the network, speeds up the processing and transmission speed of data, and reduces the delay [14]. EC industry alliance jointly initiated and established by Huawei, China Academy of Information and Communications Technology, Intel, and other institutions in 2016 is an influential alliance in EC. Its alliance with the industrial Internet of Things industry points out the challenges to be faced by the current intelligent development environment, as shown in Figure 1.

On this basis, the EC industry alliance also puts forward four major problems that need to be solved by EC technology. These problems are depicted in Figure 2.

Edge computing is a three-layer architecture with field layer, edge layer, and cloud computing layer, as shown in Figure 3.

2.2. Edge Computing Offloading Technology. Computing offloading technology is primarily divided into two categories. The first is the offloading decision, which includes the offloading technique, offloading address, and the data size and type of the device’s offloading task. The second method is resource allocation, which distributes computer and communication resources for offloading activities. The offloading decision and resource allocation for executing the offloading task are carried out with the assistance of the software-defined network (SDN) controller [15].

SDN separates the network control plane from the data forwarding plane, realizes the programmable control of the underlying hardware through the software platform in the centralized controller, and realizes the flexible on-demand allocation of network resources. The SDN network architecture includes three layers: application, control, and forwarding, as shown in Figure 4.

During the task offloading of EC, with the assistance of the SDN controller, the local terminal device transmits the offloading decision and resource allocation of the offloading task to the EC server or cloud server for execution through a wireless network or cellular network. After the task is completed, the local terminal will receive the processing result of the task. Figure 5 is an explanatory diagram of computing offloading.

The offloading decision system is mainly used to judge the task generated by key equipment and judge whether the task needs to be calculated and unloaded. Figure 6 is the flow chart. Computing resource allocation is to allocate resources to the tasks to be unloaded according to the network bandwidth and resource utilization.

2.3. Task Offloading Model of EC Based on Data Preprocessing. In this study, an EC system model based on data preprocessing is proposed to study the impact of data preprocessing on the performance of the EC model. In the EC system model, the user equipment connects with the base station through wireless connection transmission. Then, it performs data preprocessing on the mobile edge preprocessing server, obtains the corresponding services on the mobile edge application server, or accesses the cloud server through the core network to obtain the corresponding services. Users can also directly obtain the corresponding application services on the mobile edge application server. Both wireless and wired transmissions have a certain time delay in the transmission process [16]. The application switching in the edge server will not impact its delay. The delay of wireless transmission is only affected by the user’s service plan and the bandwidth allocation strategy of mobile operators. Therefore, only the impact on the wired delay of the user equipment from the base station to the cloud or edge server is considered.
Challenges to intelligent development

- Interoperability between operational technology & information communication technology
- Data integration issues
- Knowledge modeling problem
- The end-to-end flow of data

Figure 1: Challenges to be met in the current intelligent development environment.

Through digital twins, the real-time characteristics of the physical world are established in the digital world

By integrating cooperation among people, things, local systems, and cloud, a model-driven intelligent distributed architecture platform is built

Realize the industrial platform, provide an end-to-end service framework that can iterate development, operation and maintenance deployment, and online operation

Improve the processing capacity and realize the synergy between cloud computing and edge computing

Figure 2: Four major problems to be solved by current EC technology.

Figure 3: The basic architecture of EC.
The number of the user device in the model is set as \( i \). When the number is \( N \), the number of base stations is 1, and the number of mobile edges preprocessing servers is \( M_1 \). When the number is \( J \), the number of mobile edge application servers is \( M_2 \). When the number is \( K \), one \( M_1 \) can provide preprocessing services for \( S \) users simultaneously, and one \( M_2 \) can provide up to \( R \) virtual resource blocks. \( C \) represents the cloud server. In definition \( M_{1i} \in [0, 1], M_{1i} = 1 \) indicates that user device \( i \) uses the preprocessing services in \( M_1 \). In definition \( M_{2i} \in [0, 1], M_{2i} = 1 \) indicates that user device \( i \) uses the virtual resources in \( M_2 \). In definition \( C_i \in [0, 1], C_i = 1 \) indicates that user device \( i \) uses the relevant services in the cloud server. \( D_i \) represents the wired delay \( D_i \) of the user device \( i \) from the base station to the cloud or edge server. The task offloading model is expressed as \( (A_i, r_i) \). \( A_i \) represents the size of the task, and \( r_i \) represents the number of virtual resource blocks in \( M_2 \) that need to be used to process the task. It is assumed that the wired delay has \( \delta \) linear relationship with the size of the task.

The delay of the cloud server is about 8-12 times that of the edge server. Here, it is set that the delay multiple \( \rho \) obeys the normal distribution with the mean value of 10. When all user devices obtain services through the cloud server, the delay of the system model reaches the maximum \( D_{\text{max}} \). The following is the delay model of user device \( i \).

\[
\begin{align*}
\rho & \sim N(10, 1) \quad 8 \leq \rho \leq 12 \\
D_{\text{max}} & = 12\delta A \\
D_i & = \delta A_i [\rho C_i (\eta M_{1i} + \overline{M_{1i}}) + (\eta M_{1i} + \overline{M_{1i}}) M_{2i}]
\end{align*}
\]

\( \eta \) represents the number of the cloud server in the model.\( \overline{M_{1i}} \) represents the average number of virtual resource blocks used by the user device \( i \) in the model.
where $\eta$ represents the compression degree of the data volume and $M_{1i} + M_{1t} = 1$. $\alpha_1$ and $\alpha_2$ are the fixed deployment cost of $M_1$ and $M_2$, respectively, and $\beta_1$ and $\beta_2$ represent the flexible cost of $M_1$ and $M_2$. $Q_i$ shows the flexible cost. $Q$ represents the total cost, that is, the sum of fixed cost and flexible cost. $Q_{\text{max}}$ represents the maximum cost of the system. The following is the system cost model.

$$
\begin{align*}
Q_i &= \beta_1 M_{1i} + \beta_2 M_{2i} \\
Q &= \alpha_1 J + \alpha_2 K + \sum_{i=1}^{N} Q_i \\
Q_{\text{max}} &= \alpha_1 J_{\text{max}} + \alpha_2 K_{\text{max}} + \beta_1 N + 2 \beta_2 N,
\end{align*}
$$

where $J_{\text{max}}$ and $K_{\text{max}}$ meet the following conditions:

$$
\begin{align*}
J_{\text{max}} &= \frac{N}{S} \\
K_{\text{max}} &= 2N.
\end{align*}
$$

The weight of the system cost is set as $V$ in the system objective function to better balance the system delay and cost and achieve optimal system performance. The expression of the objective problem of the system reads

$$
\begin{align*}
P_1 &\min_{J, K, M_{1i}, M_{2i}, C_i} \sum_{i=1}^{N} D_i + VQ \\
\text{s.t.} & M_{1i}, M_{2i}, C_i \in [0, 1] \\
& M_{2i} + C_i = 1 \\
& \sum_{i=1}^{N} M_{1i} \leq SJ \\
& \sum_{i=1}^{N} r_i M_{2i} \leq RK.
\end{align*}
$$

The weight $V$ of system cost is calculated as follows:

$$
V = \frac{D_{\text{max}}}{Q_{\text{max}}} \times \frac{1 - w}{w},
$$

where $w$ is the weight coefficient.

After introducing the system cost weight $V$, the expression of the calculation of the system’s fixed cost reads

$$
Q_{JK} = V(a_1 J + a_2 K).
$$

In practical applications, the edge server will be overloaded or idle [17]. The deployment number of edge servers needs to be redetermined to make the system run stably. The target values of $M_1$ and $M_2$ when the deployment quantity is fixed to the fixed values of $J$ and $K$ can be obtained by solving the $P2$ problem. At this time, the fixed cost is evenly distributed to each user’s device. The equation of the $P2$ problem is as follows:

$$
\begin{align*}
P_2 &: \min_{J, K} \sum_{i=1}^{N} \delta A_i [\rho C_i (\eta M_{1i} + M_{1t}) + (\eta M_{1i} + M_{1t}) M_{2i}] + V \left( \beta_1 M_{1i} + \beta_2 r_i M_{2i} + \frac{Q_{JK}}{N} \right) \\
\text{s.t.} & M_{1i}, M_{2i}, C_i \in [0, 1] \\
& M_{2i} + C_i = 1 \\
& \sum_{i=1}^{N} M_{1i} \leq SJ \\
& \sum_{i=1}^{N} r_i M_{2i} \leq RK.
\end{align*}
$$
The greedy algorithm [14] is used to solve the minimum value of the deployment number of $M_1$ and $M_2$ under the $P_2$ problem. The value obtained is used as the fixed cost of the deployment scheme for a certain time. Then, the trade-off problem $P_3$ between delay and flexible cost is solved according to the actual situation. The expression of the $P_3$ problem reads

$$
\begin{align*}
P_3: & \min_{X_i} \sum_{i=1}^{N} \delta A_i \left[ \rho C_i (\eta M_{1i} + \overline{M}_{1i}) + (\eta M_{1i} + \overline{M}_{1i}) M_{2i} \right] + V (\beta_1 M_{1i} + \beta_2 r_i M_{2i}) \\
& \text{s.t.} M_{1i}, M_{2i}, C_i \in [0, 1] \\
& \quad M_{2i} + C_i = 1 \\
& \quad \sum_{i=1}^{N} M_{1i} \leq SJ \\
& \quad \sum_{i=1}^{N} r_i M_{2i} \leq RK, \\
\end{align*}
$$

(8)

where

$$
X_i = \left[ C_i, \overline{M}_{1i}, \overline{M}_{1i}, M_{1i} + C_i, M_{1i}, M_{2i} \right]^T. 
$$

(9)

To better solve $X_i$, the target values of users using cloud services, using $M_1$, and cloud services, and using $M_1$ and cloud services, and using $M_1$ and $M_2$ are solved. The target values of the four methods are $O_{ic}, O_{i2}, O_{i12}, O_{i1}$, and the calculation method is

$$
\begin{align*}
O_{ic} &= \delta A_i \rho \\
O_{i2} &= \delta A_i + V \beta_2 r_i \\
O_{i1} &= \delta A_i \rho \eta + V \beta_1 \\
O_{i12} &= \delta A_i \eta + V (\beta_1 + \beta_2 r_i).
\end{align*}
$$

(10)

$P_3$ is converted to $P_4$ to obtain the following equation:

$$
\begin{align*}
P_4: & \min_{X_i} \sum_{i=1}^{N} O_i X_i \\
& \text{s.t.} \sum_{i=1}^{N} O_i X_i = 1 \\
& \quad \sum_{i=1}^{N} M_{1i} + C_i + M_{1i} + M_{2i} \leq SJ \\
& \quad \sum_{i=1}^{N} r_i (M_{2i} + \overline{M}_{1i} + M_{1i} + M_{2i}) \leq RK, \\
\end{align*}
$$

(11)

where the coefficient matrix $O_i$ is

$$
O_i = \begin{bmatrix}
O_{ic} & O_{i2} & O_{i1} & O_{i12}
\end{bmatrix}.
$$

(12)

The problem of converting $P_4$ to $P_5$ linear integer programming is as follows:

$$
\begin{align*}
P_5: & \min OX \\
& \text{s.t.} \sum_{i=1}^{N} X_i = 1 \\
& \quad \sum_{i=1}^{N} M_{1i} + C_i + M_{1i} + M_{2i} \leq SJ \\
& \quad \sum_{i=1}^{N} r_i (M_{2i} + \overline{M}_{1i} + M_{1i} + M_{2i}) \leq RK, \\
\end{align*}
$$

(13)

The coefficient matrix $O$ can be expressed as

$$
O = \begin{bmatrix}
O_{1} & O_{2} & \cdots & O_{N}
\end{bmatrix}.
$$

(14)

The vector solution $X$ is represented as

$$
X = [X_1, X_2, \cdots, X_N]^T.
$$

(15)

The decision path for each user device to obtain application services can be obtained by solving $X$.

2.4. Building Energy Conservation and Passive House

2.4.1. Building Energy Conservation. Comprehensive building energy conservation refers to the use of energy-saving building materials and equipment, the rational design of the thermal performance of the building envelope, and the improvement of the operation efficiency of heating, refrigeration, water supply, drainage, lighting, and other systems in the whole process of site selection, planning, design, construction, use, and demolition within the whole life cycle of the building to achieve the purpose of saving energy and reducing building energy consumption [18]. The energy consumption during building operation accounts for about
80% of all energy consumption in the whole life cycle of the building. Therefore, this study mainly studies building energy conservation to reduce building energy consumption in the process of building operation.

2.4.2. Passive House. Around 1920, people studied the application of solar energy in buildings. The buildings that use mechanical equipment to use solar energy are called active buildings [19]. On the contrary, the houses that do not use mechanical equipment and only rely on design to make the buildings use solar energy are called passive houses. Passive design means that in the process of architectural scheme design, according to the regional climate characteristics of the building location, the basic principles of building environment control technology are followed, and the needs of building functions and forms are integrated. Besides, the building elements are reasonably organized and processed to give the building strong climate adaptation and regulation ability [20]. The focus of the passive design is to complete the design of the building by considering the building function and form and following the basic principles of building environment control based on the climate conditions and geographical characteristics in the study area [21].

2.5. Research on Energy-Saving Design of Passive House—A Case Study of Chengdu. Chengdu is located in the southwest of China, in the northwest of Sichuan Basin and the hinterland of Chengdu Plain. The terrain is flat, with an altitude of 1000~3000 meters. Chengdu has a subtropical monsoon humid climate with hot and humid summer. It is cloudy and cold in winter and hot in summer.

Chengdu has a large population, compact building layout, and serious building shelters. Natural lighting and natural ventilation are difficult [22]. The exterior of the building is mostly paved with floor tiles or directly hardened ground, and the greenery and greening levels of natural plants are low. Southwest wind and northeast wind prevail in Chengdu in summer. Therefore, in the passive design in summer, the building opening needs to avoid the southwest and northeast directions. If they cannot be avoided, greening and landscape water bodies can be set at the opening to cool down and realize heat insulation. Shading is also an effective measure of heat insulation [23]. Building shading is mainly divided into three ways: external shading, internal shading, and middle shading. Most residential buildings in Chengdu adopt the form of casement windows, so the external shading method is more suitable.

In terms of ventilation, natural ventilation is mainly adopted to meet the requirements [24]. Natural ventilation in summer needs to focus on two aspects. The first is to discharge the indoor heat and humidity outdoors, and the second is to avoid bringing the outdoor overheated and humid air into the room. In summer, generally, the dominant wind direction is also the wind direction with the highest temperature. Therefore, the air inlet should form a certain angle with the incoming wind direction, and the air inlet direction can be determined through the Computational Fluid Dynamics (CFD) software [25]. The outdoor temperature in Chengdu from 9:00 p.m. to 5:30 a.m. of the next day is low, so it is suitable for natural ventilation. Greening can reduce the environment’s temperature through the thermal infiltration of vegetation, block solar radiation, and change the flow direction of airflow. The reasonable setting of vegetation can effectively improve the comfort of buildings.

Passive evaporative cooling technology [26, 27] is a way of passive cooling through the comprehensive utilization of various renewable energy sources. In summer, most of the heat in the house is transmitted through the roof. Hence, the passive evaporative cooling design of the roof is studied. There are three kinds of roofs designed by passive evaporative cooling: free water surface evaporation roof, porous material aquifer roof, and hygroscopic roof. The free water surface evaporation roof is suitable for areas with low relative humidity and large temperature difference between day and night. The porous material aquifer roof is suitable for areas with low relative humidity. The hygroscopic roof dissipates moisture and stores heat during the day and absorbs moisture and regenerates at night according to the change of air humidity. The hygroscopic roof is more suitable for Chengdu. The northwest wind is the predominant wind direction throughout the winter season. In the cold season, the rarely used area can be placed on the windward side, while the frequently used space can be placed on the leeward side. In addition, thermal insulation measures of envelop enclosure can also be taken to improve the comfort of the indoor environment. The roof can be insulated with an extruded polystyrene insulation board [28–31]. The external wall insulation technology can adopt external and internal insulation technology, and the cold bridge can be subject to certain structural treatment. The windows should be double glazed and face south to reduce the heat loss through the windows. The air humidity in Chengdu is high and low in winter, so it is easy to produce condensation inside the wall, which destroys the thermal insulation performance of the wall. Setting a damp proof layer on the layer near the wall with high indoor temperature is conducive to protecting the thermal insulation performance of the wall.

3. Simulation Results

3.1. Experimental Setup. To accomplish the experimental process, we employ the MATLAB software. We used 100 user devices for communication, the size of user-generated tasks was Mbits, and the maximum number of devices connected was 10. Table 1 shows the parameter settings for the experimental process.

3.2. Analysis of the Relationship between Weight Coefficient \( w \) and System Delay and Cost. The delay and cost when the weight coefficient \( w \) is 0.1, 0.3, 0.5, 0.7, and 0.9 are tested by simulation, and the average value is taken through ten tests. Figure 7 shows that when \( \eta = 0.1 \), the time delay is the maximum when \( w = 0.1 \), and the minimum when \( w = 0.9 \); the cost is the smallest when \( w = 0.1 \), the largest when \( w = 0.9 \), and there is no change between 0.3 and 0.7. It is because the user device adopts the processing method of mobile edge processing server \( M_1 \) + edge cloud computing when \( w = 0.1 \sim 0.7 \). However, when \( w \) is 0.9, the processing
Table 1: Parameter setting of a simulation experiment.

<table>
<thead>
<tr>
<th>Items</th>
<th>Set value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of user devices</td>
<td>100</td>
</tr>
<tr>
<td>User-generated task size</td>
<td>1Mbits</td>
</tr>
<tr>
<td>Multiple delays to task size</td>
<td>10</td>
</tr>
<tr>
<td>Maximum number of devices that can be connected with $M_1$</td>
<td>10</td>
</tr>
<tr>
<td>The maximum number of virtual resource blocks that $M_2$ can provide</td>
<td>5</td>
</tr>
<tr>
<td>Fixed cost of $M_1$</td>
<td>500</td>
</tr>
<tr>
<td>Flexible cost of $M_1$</td>
<td>30</td>
</tr>
<tr>
<td>Fixed cost of $M_2$</td>
<td>600</td>
</tr>
<tr>
<td>Flexible cost of $M_2$</td>
<td>50</td>
</tr>
</tbody>
</table>

![Figure 7: Relationship between weight coefficient $w$ and delay and cost.](image-url)

(a) Relationship between weight coefficient $w$ and time delay. (b) Relationship between weight coefficient $w$ and cost.
method of using mobile edge application server \( M_2 \) + edge cloud computing is adopted to reduce the delay, and its cost is relatively high. When \( w = 0.1 \), the edge server is not deployed because the model target has low requirements for the delay, so the cost at this time is 0. When \( \eta = 0.5 \), the model delay decreases with the increase of \( w \), and the model cost increases with the increase of weight coefficient \( w \). Likewise, when \( \eta = 0.9 \) and \( w \) increases from 0.7 to 0.9, the model delay does not change, and the model cost is reduced to a certain extent. It shows that when \( \eta = 0.1 \) and \( \eta = 0.5 \) and the weight coefficient \( w = 0.9 \), replacing \( M_1 \) with \( M_2 \) can reduce the model delay, but it will also increase the cost of the model. When \( w = 0.9 \), using edge application server \( M_2 \) cannot effectively reduce the delay of the model, but it will reduce a certain cost, so the deployment scheme is still worth taking.

3.3. Analysis on Energy-Saving Performance of Passive House Design. Two groups of experiments were carried out separately. One group adopted the traditional architectural design scheme, and the other group adopted the passive house design scheme. The top floor of a typical residential building in Chengdu was taken as the calculation model and simplified by Energy Plus simulation software. Its energy consumption under five same working conditions was studied, as shown in Figure 8. Different working conditions were considered as follows:

- **Working condition 1**: natural ventilation, external shading, a wall heat transfer coefficient of 0.22 W/(m² K), and a planted roof
- **Working condition 2**: natural ventilation, no shading, a wall heat transfer coefficient of 0.83 W/(m² K), and a water storage roof
- **Working condition 3**: only ventilation at night, external shading, a wall heat transfer coefficient of 0.83 W/(m² K), and a water storage roof
- **Working condition 4**: only ventilation at night, no shading, a wall heat transfer coefficient of 0.3 W/(m² K), and a planted roof
- **Working condition 5**: natural ventilation, external shading, a wall heat transfer coefficient of 0.63 W/(m² K), and a water storage roof

Figure 8 shows that the passive design under condition 5 has the lowest annual energy consumption per unit area, which is 18.97 kWh/m². Moreover, its energy-saving rate is the highest, which is 0.77. From an energy consumption perspective, working condition 5 has the highest energy-saving efficiency and is the best scheme.

4. Conclusions

With the advancement of new-generation information technologies such as the Internet of Things and cloud computing, smart city building has reached a new level. However, in traditional clouding computing, all data must be uploaded to centralized servers which put a lot of pressure on the network. In this study, the task offloading model of an EC network for a smart city was designed based on data preprocessing. The energy-saving design of passive houses in Chengdu was carried out to reduce the energy consumption of urban buildings in the process of operation. The results showed that when \( \eta = 0.1 \), the delay and cost of the model have no change between \( w = 0.3 \) and 0.7. The delay reaches the minimum at 0.9, and the cost reaches the maximum at 0.9. When \( \eta = 0.5 \), with the increase of the weight coefficient \( w \), the model delay decreases, and the model cost increases. When \( \eta = 0.9 \), the model delay has no obvious change between \( w = 0.1 - 0.3 \) and \( w = 0.7 - 0.9 \), and when \( w = 0.3 - 0.7 \), the model delay decreases. The cost of the model has no obvious change between \( w = 0.1 \) and 0.3. However, in the growth process of \( w = 0.3 - 0.7 \), the cost of the model shows an increasing trend. In the process of
Changing $\omega$ to 0.9, the cost has decreased to a certain extent, indicating that this deployment method has certain practical value. When natural ventilation, external shading, and water storage roof are adopted and the heat transfer coefficient of the wall was 0.63 W/(m² K), the passive design scheme has the lowest energy consumption per unit area in the whole year, which was 18.97 kWh/m². Its energy-saving rate is the highest. From the perspective of energy consumption, its energy-saving efficiency is the highest. The research results provide research experience for promoting the construction of a smart city with sustainable development. However, the cost of the deployment scheme is still high. In future work, how to reduce the cost of the task offloading model of the EC network will be investigated.

Data Availability

The data used to support this study are included in the paper or available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This article is a research project sponsored by the Department of Science and Technology of Hebei Province: research on the passive ultralow energy consumption construction industry promotion path based on related policies (Project No.: 21557641D).

References


