

Research Article

Design of the Wireless Network Hierarchy System of Intelligent City Industrial Data Management Based on SDN Network Architecture

Wenken Tan¹ and Jianmin Hu ²

¹College of Architecture and Urban Planning, Tongji University, Shanghai 200092, China

²Shanghai Tongqian Construction Planning and Design Co. Ltd., Shanghai 200433, China

Correspondence should be addressed to Jianmin Hu; 49405890@qq.com

Received 13 August 2021; Accepted 4 October 2021; Published 10 November 2021

Academic Editor: Xuyun Zhang

Copyright © 2021 Wenken Tan and Jianmin Hu. 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 rapid development of the industrial Internet of Things and the comprehensive popularization of mobile intelligent devices, the construction of smart city and economic development of wireless network demand are increasingly high. SDN has the advantages of control separation, programmable interface, and centralized control logic. Therefore, integrating this technical concept into the smart city data management WLAN network not only can effectively solve the problems existing in the previous wireless network operation but also provide more functions according to different user needs. In this case, the traditional WLAN network is of low cost and is simple to operate, but it cannot guarantee network compatibility and performance. From a practical perspective, further network compatibility and security are a key part of industrial IoT applications. This paper designs the network architecture of smart city industrial IoT based on SDN, summarizes the access control requirements and research status of industrial IoT, and puts forward the access control requirements and objectives of industrial IoT based on SDN. The characteristics of the industrial Internet of Things are regularly associated with data resources. In the framework of SDN industrial Internet of Things, gateway protocol is simplified and topology discovery algorithm is designed. The access control policy is configured on the gateway. The access control rule can be dynamically adjusted in real time. An SDN-based intelligent city industrial Internet of Things access control function test platform was built, and the system was simulated. The proposed method is compared with other methods in terms of extension protocol and channel allocation algorithm. Experimental results verify the feasibility of the proposed scheme. Finally, on the basis of performance analysis, the practical significance of the design of a smart city wireless network hierarchical data management system based on SDN industrial Internet of Things architecture is expanded.

1. Introduction

According to the analysis of “Made in China 2025” proposed under the background of the new era, the development of the industrial Internet of Things has received the attention of the whole society, especially the information and communication technology and intelligent manufacturing have now become the core content of China’s urban construction and development. From the perspective of the Internet of Things practice, the Internet of Things can complete big data integration services more efficiently through the use of work. Among them, the Industrial Internet of Things, as the basis

of Internet content, will integrate cloud computing, big data, and sensors into the entire process. The emergence of the Industrial Internet of Things uses advanced technology to gradually transform traditional industries into intelligent industries, thereby improving product quality and production efficiency. At the same time, during the development of the Industry 4.0 era, the biggest feature is the network physical system, also known as CPS. The whole physical process needs to be controlled by a computer, and relevant data and commands are transmitted to the CPS controller. With the comprehensive promotion of subsequent Ethernet technology, it can bring clear real-time

attributes and can transmit real-time or non-real-time traffic in the same media, but because of their incompatibility, it is impossible to run different technologies in the same physical media in the development of practice. In view of this situation, researchers have strengthened the research on time-sensitive network (TSN), especially as the key content of which the smart city data management algorithm has become the focus of scientific research and exploration, which will play a positive role in the future industrial communication and automation technology innovation. Abosata et al. [1] further verified the importance of TSN research in Industry 4.0 in their research on the development of China's industrial Internet of Things. Wiers et al. [2] also put forward new conclusions about time-sensitive network technology and its application in an industrial network, especially it has been widely used in many fields of industrial network and attracted more scholars to participate in the research. Nowadays, with the deepening of practical research tasks, effective countermeasures have been obtained to solve some TSN data processing problems [3]. For example, Cerrato et al. [4] put forward new views on the South Industrial Control Security Gateway on the basis of ensuring the security of sensitive data in the industrial control system. Pinheiro et al. [5] proposed the TT traffic data processing method based on the operation characteristics of TSN and completed various TT communication data processing work accordingly. At the same time, Xu et al. [6] proposed the tabu search heuristic algorithm to clarify the TT frame data processing so as to minimize the WCD of the RC frame. In addition, Smirnov et al. [7] calculated and analyzed the constraint conditions required for offline data processing according to the clear characteristics of time-sensitive traffic in TSN and the general configuration of behavioral function data so as to avoid the delay phenomenon of key communication flow from port to port. Chen et al. [8]. Mapped the actual system data processing problems to the job-shop data processing problems that did not need to wait under the condition of ensuring time-sensitive traffic to minimize network delay and then used the tabu search algorithm to deal with the problems. As the core content of urban economic construction and development, Internet of Things technology is gradually integrated into the innovation work of various industries and fields but also shows a very broad development prospect and brings more terminal equipment management problems. Combined with the above design and analysis of the smart city data management wireless networking hierarchical system based on SDN network architecture, it can be seen that the future technology research and development of the Internet of Things must pay attention to wireless network access control, and combined with CPS, the TSN data processing algorithm is deeply explored [9].

2. Methods

Similar to the industrial Internet of Things, it can be applied to the development of various industries and can provide more opportunities and challenges for the realization of automation and intelligent industrialization. Considering

the integration of SDN network architecture and wireless network sharing system design in smart city construction, it can be seen that when the virtual WLAN design simplifies network management operation, it does not put forward effective solutions for resource allocation, so it is difficult to improve the overall performance in the practical development. Due to the open network architecture and industrial Internet of Things sharing data information in big data era, network information security issues are easy to affect information operations, so real-time networks have advantages in R&D application. First is large data transmission capacity. The second is to give priority to functional design. At this time, this problem can be solved by controlling the real-time communication network. This provides the basic guarantee for the actual data transmission [10] as shown in Figure 1.

2.1. Architecture Analysis. According to the analysis of Figure 2, the overall design framework is mainly divided into three layers: the first layer is the application layer; the second is the network control layer; and the third is the network infrastructure layer. Take the network infrastructure layer as an example, all the wireless access points (APs) contained in it need to build virtualized APs, also known as VAPs, which belong to virtual machines of physical APs and can be used to virtualize network resources. At the same time, all APs have the ability to virtual multiple VAPs, and all VAPs exist independently not only to provide system users with required network services but also to provide relay services for various APs. In addition, any switch with a wire terminal in the network system can run the OpenFlow protocol service [11].

2.2. Functional Design. According to the analysis of Figure 3, the structure of SD-AP contains three layers: (1) VAP: this level refers to the abstract materialization of SD-AP, which is mainly used to deal with channel detection, user management, state detection and other work. (2) Wireless network card driver: this level is mainly used to support the transmission of WLAN protocol signals and collect the information of the state of the wireless channel. (3) The south interface agent: it is mainly used to expand the protocol interpretation, to execute the commands proposed by the controller, and on the basis of encapsulating the network events, the relevant contents are transmitted to the controller through the southward protocol.

- (1) *User Management.* System users should connect with the network through the SD-AP wireless port and conduct comprehensive control over the operation of the wireless network. This effort is user-centric, not port-centric. In this study, in order to better control all associated users, a lightweight virtual access point (LVAP), which represents the user's proprietary state set, is proposed to represent the specific state of the user's connection with the network. In essence, LVAP can be regarded as the interface provided by SD-AP for all users, mainly used

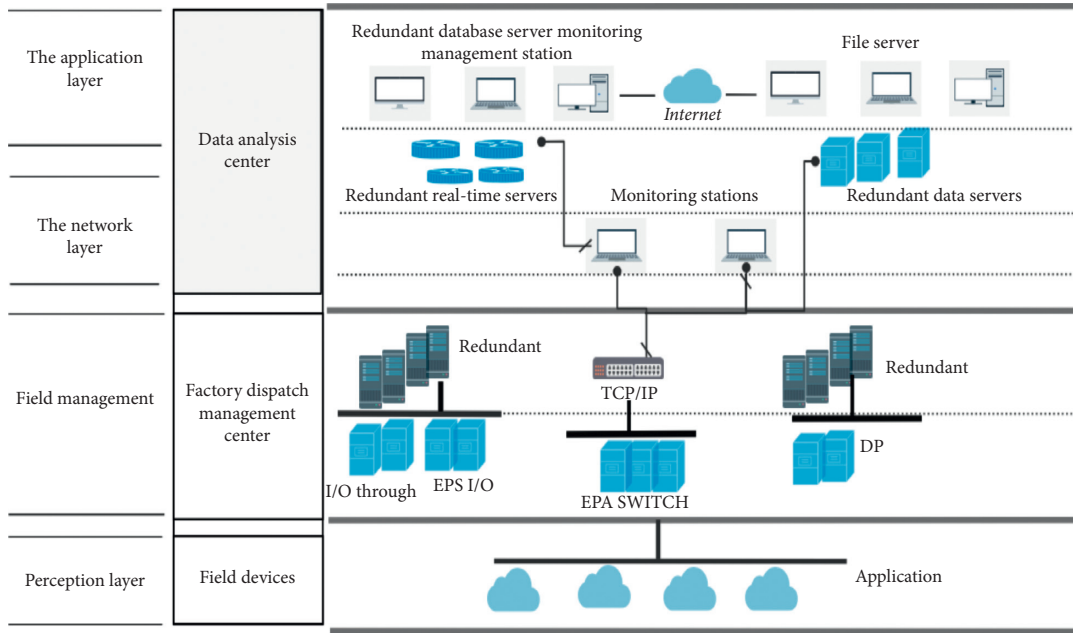


FIGURE 1: Industrial Internet of Things architecture.

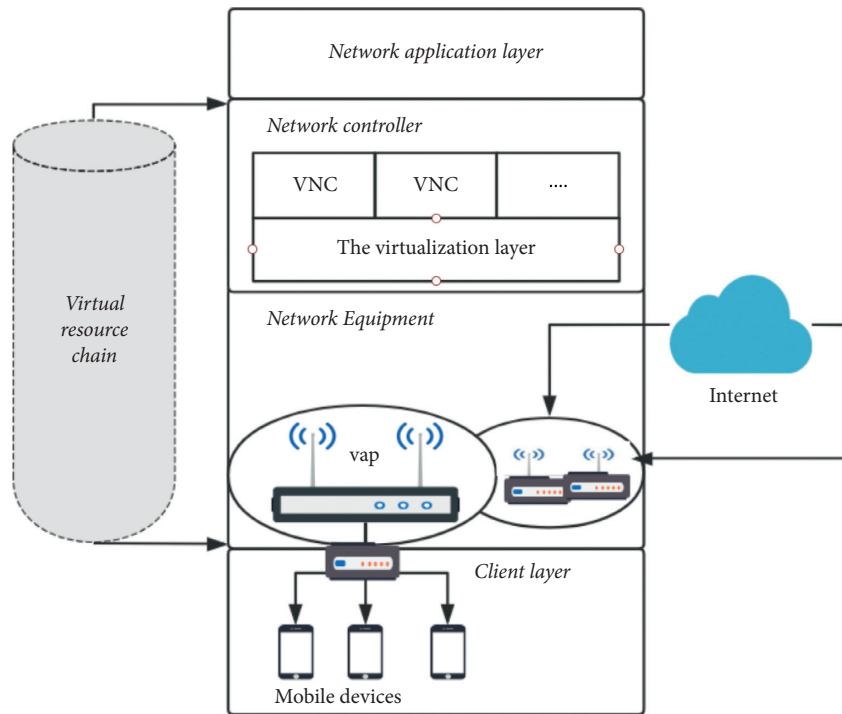


FIGURE 2: Network architecture diagram based on SDN.

to control the specific situation of users. When all users connect to the network for the first time, this module will form the LVAP, and all LVAPs will have a unique basic service set identifier (BSSID), which needs to be formed from the MAC address of the system user. At the same time, because LVAP reserves the authentication information of all users, such as MAC address, IP address, and so on, in the process of

communication with SD-AP, it is equivalent to communicating with LVAP. In other words, as long as LVAPs are formed in SD-AP, users can be effectively linked to the network. In this module, users not only can manage the scheduling problems transformed into LVAP after abstract processing but also can put forward appropriate security countermeasures according to different user needs. [12].

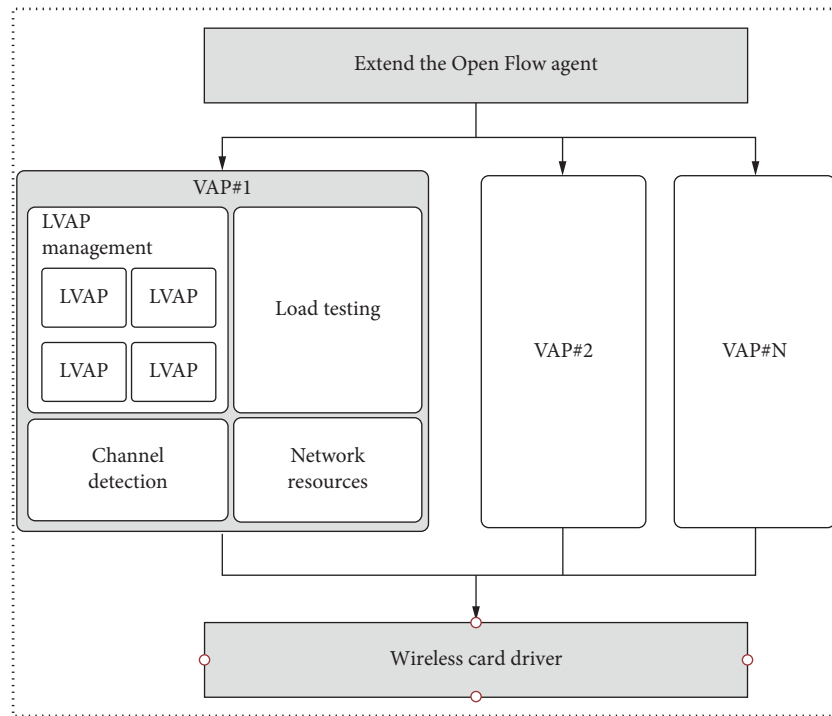


FIGURE 3: The SD-AP structural analysis diagram.

- (2) *Channel Detection*. For all users, the most important concern is how to obtain a QoE or how to provide a corresponding QoS. And this mapping to the physical layer is the transmission quality of wireless communication. From the practical point of view, the construction of the channel monitoring module is to better guarantee the operation management quality of the application layer, avoid the interference of various factors during the application period, and thus improve the effectiveness of network resource allocation and the experience of receiving services. Generally speaking, the channel monitoring module is mainly used to control layer transmission. There is a relationship between the user and the signal-to-interference ratio, the overall strength of the received signal, the duration, and the average packet loss rate associating all the parameters between the SD-AP. These contents are uploaded to the network controller after encapsulation and processing, which can facilitate the network management personnel to implement resource scheduling according to the specific operation state [13].
- (3) *Resource Description*. The network is distributed equally among all users, and the smallest unit of resources is called a resource block. And the user will obtain the network resources according to the acquisition of resource blocks. In the wireless network system, because the access mechanism of random access has been fully promoted. So user access to resources is competitive. However, in a large-scale network, factors such as adjacent channels and transmitting power will affect user services, and each

SD-AP will involve computing storage resources, real-time performance and other related issues, which need to be dealt with directly in SD-AP. In other words, on the basis of collecting and analyzing network, SD-AP resources can implement global optimization processing to the whole network. It can be seen that the purpose of this module design is to obtain the resources in SD-AP and bring the specific view of network resources to the control layer so as to help the system application carry out the resource scheduling work in an orderly manner [14].

- (4) *Status Monitoring*. In order to better provide users with quality services, the network will put forward corresponding optimization mechanisms according to the system operation, such as interference management and load balancing, so as to ensure that users can obtain a better service experience inside the system. In the process of popularizing these optimization countermeasures, it is necessary to combine the current network state analysis results to conduct in-depth exploration. Therefore, it is necessary to obtain the state information of SD-AP in the control layer, so as to bring an effective basis for the actual application layer. In practice, this module is mainly used to obtain the information related to computing storage and network resource application contained in SD-AP, package and process it into state frame, and transmit the relevant information to the controller under the condition that the controller makes a request. Because all SD-APs can involve multiple VAPs, in order to ensure the stable operation of all VAPs, it is necessary to balance all SD-AP

resources occupied by all VAPs in the control layer and report the application of resources in real time.

2.3. Network Controller. According to the analysis of Figure 3 above, this module is mainly divided into two parts: (1) the network virtualization module and (2) the network controller set (VNCs). The former is mainly used to virtualize the physical networks, while the latter is mainly used to control the virtualization resources within the network. This paper studies thinking from the user perspective, mapping the transmit power included in SD-AP into the letter drying ratio (SINR) of all users, then the mapping between the resource and the user signal quality is obtained, and the following formula is obtained:

$$\text{SINR}_{ij} = \frac{g_{ij}P_j}{\sum_{k \in A_i \cap k \neq j} g_{ik}P_k + N_0}, \quad (1)$$

where g_{ij} represents the link gain case existing by the SD-AP user i , P_j represents the emission power of the SD-AP, N_0 represents additive Gaussian white noise, and A_i represents the cover range, involving all SD-AP sets contained by the user i .

According to the network controller, the wireless network system design must include two forms: one is resource management (RVNC) and the other is state management (SVNC). For the former, its work content is to classify the internal resources according to the resource view described by the network virtual module so as to facilitate the subsequent resource scheduling. The latter needs to collect various states of the network, such as resource occupation and network operation, so as to lay a foundation guarantee for the orderly allocation of subsequent resources [15].

The biggest advantage of the southbound interface expansion protocol of OpenFlow is centralized processing control and convenient fine-grained flow-based flow control. Therefore, the number of associated user terminals for all VAPs can be accurately calculated in a specific topological view. The average channel utilization probability formula for VAP _{j} is as follows:

$$CU_j = \frac{\sum_{i=1}^N t_i}{T} = \frac{\sum_{i=1}^N l_i/r_{ij}}{T}, \quad (2)$$

Formula (2) expresses the transmission speed of real-time network. In the formula, T represents the unit test time, t_i represents the user I channel time, l_i represents the user I 's unit test time range, and r_{ij} represents the user I 's VAP _{j} probability of data transmission speed. Unit test includes general data collection and transmission times; the specific calculation formula is as follows:

$$r_{ij} = f(G, P), \quad (3)$$

where P represents the transmitted power, G represents the factor of link gain, and F represents the mapping function

SDN integrates existing virtual WLAN and time-sensitive networks (TSN) to obtain the TSSDN architecture, as shown in Figure 4. It can isolate the time-triggered flow from

the spatial and temporal perspectives and thus provide a basic guarantee for data clarity.

Combined with the analysis of the OpenFlow architecture diagram shown in Figure 5, it is assumed that the source of the time-triggered flow will deliver unicast packets to the specified area at a fixed bit rate, and the time period is set to be an integer multiple of the minimum transmission period that can be supported. This time-triggered mode is more suitable for sensors with fixed sampling cycles or actuators that transmit commands at specified time intervals so that time-triggered traffic can transmit data in high-priority UDP packets and is stronger than other priority traffic. Because the end system needs to synchronize accurately using the precise time protocol (PTP) and all event-triggered traffic has the same priority, an additional scheduling mechanism needs to be proposed to optimize the time-triggered traffic when conflicts occur.

According to the Figure 5 architecture, derived by the above formula, the trigger time can be determined. $G \equiv (V, E)$.

Consider the trigger flow of time as a tuple $ts_i \equiv (s_i, d_i)$. And, V represents a collection of nodes, and derive $E \equiv \{(i, j) | i, j \in V\}$, where i and j represent network link connections as a set of network connections. At the same time, $V \equiv (S \cup H)$, where S and H are collections of switches and hosts $s_i, d_i \in H$, where s_i and d_i represent the source and final destination region of the stream, respectively. Then you design the input variables for the related problem, where the set of streams triggered by scheduled events is as follows:

$$TS : TS \equiv \{ts_i\}. \quad (4)$$

The specific map to the network link is as follows:

$$SL : SL \equiv \{f_{i,j}\}, \forall i \in TS, \forall j \in E. \quad (5)$$

Suppose the stream i is transferred to the destination area via the link j , otherwise, it will become 0.

The specific map of the flow to the time slot is as follows:

$$ST : ST \equiv \{t_{i,k}\}, \quad \forall i \in TS, \forall k \in T. \quad (6)$$

The corresponding variables are as follows:

$$SLT : SLT \equiv \{y_{i,j,k}\}, \quad \forall i \in TS, \forall j \in E, \forall k \in T. \quad (7)$$

The stream i is obtained assuming that the link j is transferred to the destination area and is assigned to the time slot k , $y_{i,j,k} = 1$. Otherwise, it will become 0. This gives the best scheduling countermeasures to define the delivery schedule of the event trigger flow, as follows:

$$\begin{aligned} \text{subjectto: } u_n = \text{minimize } & \sum_{i \in TS} \sum_{j \in E} f_{i,j}, \\ & \sum_{k \in T} t_{i,k} = 1 \forall i \in TS. \end{aligned} \quad (8)$$

Among them, the shortest path formula for the optimization objective is as follows:

$$\sum_{j \in \text{in}(\text{src}(i))} f_{i,j} = 0 \quad \sum_{j \in \text{out}(\text{src}(i))} f_{i,j} = 1. \quad (9)$$

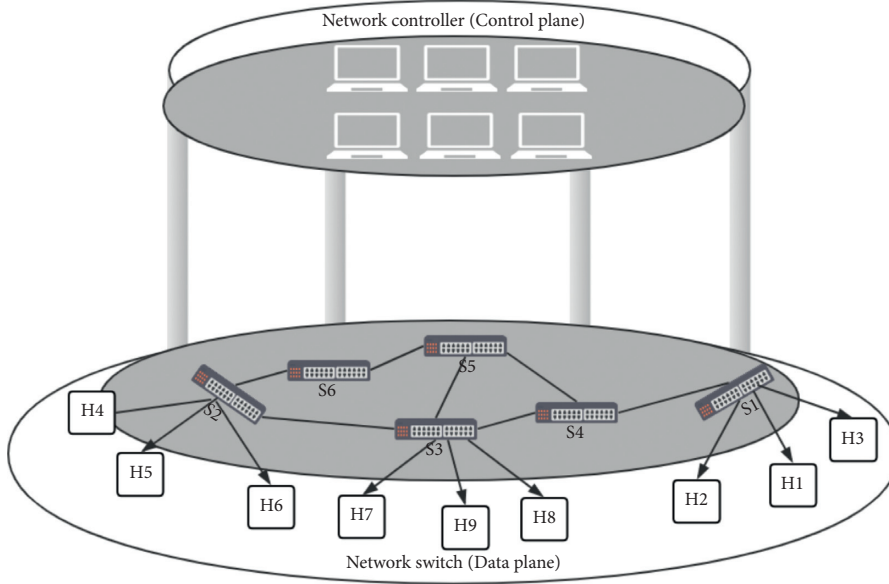


FIGURE 4: Network architecture diagram of TSSDN.

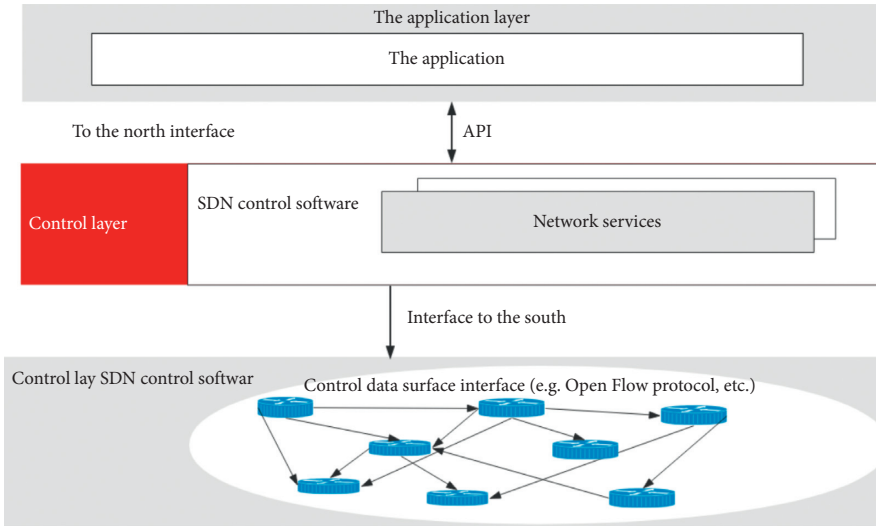


FIGURE 5: Network architecture diagram of OpenFlow.

The calculation formula for the time-slot constraints is as follows:

$$\sum_{\forall j \in m(\text{dst}(i))} f_{i,j} = 1 \quad \sum_{\forall j \in \text{out}(\text{dst}(i))} f_{i,j} = 0. \quad (10)$$

The numerical formula for variable consistency is as follows:

$$y_{i,j,k} = f_{i,j} \times t_{i,k} \forall i \in TS, \quad \forall j \in E, \forall k \in T, \quad (11)$$

where in (src (i)) represents the input of the source host, out (src (i)) represents the source host output, in (dst (i)) represents the input of the destination host, and out (dst (i)) represents the output of the destination host.

2.4. Transmission Scheduling. In studying the scheduling of time-triggered flows in the industrial Internet of Things, the transmission delay of link packets can be seen as a binary multibackpack phenomenon and is studied using the minimum ant system (MMAS) algorithm. Assuming you want n items into different weights of m backpacks and all the items in the backpack have different qualities, then you need to consider the backpack in which they should be placed. The multiple pack problem is more complex than the 0-1 pack problem and can be described as a backpack with m capacity is c_1, c_2, \dots, c_m . And there are n items whose value is P_i ; in the case of putting the i -th item into the j -th item, the corresponding weight is W_{ij} and conforms to $1 \leq i \leq n$ and $1 \leq j \leq m$. This problem involves calculating two cases: one

where a single item goes into the backpack and the other is not to put a certain item into the backpack at all; in the backpack, capacity does not meet the constraints on the basis of maximizing the total value of the items into the backpack.

The problem of binary multiple knapsacks can be expressed as follows:

$$\text{maximize } P(\bar{x}) = \sum_{i=1}^n x_i p_i, i \in \{1, \dots, n\}. \quad (12)$$

The formula for maximizing the total value of items in the backpack of the optimization target is as follows:

$$\sum_{i=1}^n w_{ij} x_i \leq c_j, j \in \{1, \dots, m\}. \quad (13)$$

The capacity limit formula of each backpack is as follows:

$$x_i = 1 \text{ or } 0. \quad (14)$$

Among them, $x_i = 1$. Item I is selected into the backpack, and 0 is not selected into the backpack. The formula for selecting a game for a bunch of items is as follows:

$$\bar{x} = (x_1, x_2, \dots, x_n). \quad (15)$$

This paper studies the use of a directed graph $G(V, E)$ to represent the network, where V represents the set of nodes and conforms to $V \equiv (S \cup H)$. This condition. The KTH bandwidth provided by link (I, j) is assumed to be consistent $x_{ij}^k = 1 \text{ or } 0$. A stream can go through a link, so it is 1, or vice versa. In addition, combining with the conservation constraint analysis of flow, we can get

$$\begin{cases} \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = 0, & k \in TS, i \neq s_i, d_i, \\ \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = 1, & k \in TS, i = s_i, \\ \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = -1, & k \in TS, i = d_i. \end{cases} \quad (16)$$

It is proved that the traffic of the KTH stream of the network entering the transmission node should be consistent with the traffic proposed by this node.

At the same time, the link capacity should be restricted and meet the constraint conditions, as follows:

$$\sum_{k \in TS} d_k x_{ij}^k \leq c_{ij}, (i, j) \in E. \quad (17)$$

In addition, the formula of service self-similar traffic is used to study the average packet delay of the flow through the i -th link. The calculation formula is as follows:

$$\tau_{ij} = \frac{\rho_{ij}^{1/2(1-H)}}{d_k x_{ij}^k (1 - \rho_{ij})^{1/(1-H)}}, \quad (18)$$

where ρ_{ij} conforms to the $\rho_{ij} = d_k x_{ij}^k / c_{ij}$. This condition, and H is a constant.

According to the binary backpack problem, the number of the link through which the event-triggered flow passes is determined so as to avoid queuing phenomena during the transmission of packets and to minimize the actual delay phenomenon so as to achieve optimization processing goal. The specific mathematical model is as follows:

$$\text{Subject to: } u(n) = \text{Minimize } \sum_{(i,j) \in E} x_{ij}^k \tau_{ij}, k \in TS,$$

$$x_{ij}^k = 1 \text{ or } 0,$$

$$\begin{cases} \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = 0, & k \in TS, i \neq s_i, d_i, \\ \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = 1, & k \in TS, i = s_i, \\ \sum_{j:(i,j) \in E} x_{ij}^k - \sum_{j:(j,i) \in E} x_{ji}^k = -1, & k \in TS, i = d_i, \end{cases}$$

$$\sum_{k \in TS} d_k x_{ij}^k \leq c_{ij}, (i, j) \in E.$$

(19)

From the perspective of ant colony optimization algorithm, it is assumed that the number of nodes is n ; the total number of ants is m ; and the distance between node I and node j needs to be expressed as $d_{ij}(i, j = 1, 2, \dots, n)$. Then, the residual information intensity at time t on the link between the two can be expressed as $\tau_{ij}(t)$. Ant k looks for the next path according to the residual pheromone intensity; then the probability that it moves from node I at time t to node j is $p_{ij}^k(t)$. Then, we can get

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{s \in J_k(i)} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta}, & j \in J_k(i), 0, \text{ other} \end{cases} \quad (20)$$

where $J_k(i) (i = 1, 2, \dots, n) - \text{tabu}_k$ represents the node set that ant k can walk, and the taboos can be expressed as tabu_k ; Nodes that pass through are recorded in the taboos table. After the ant passes through all nodes and returns to the initial point, n cities are recorded tabu_k . And the path is the feasible solution. $\eta_{ij}(t)$ represents the degree of expectation of ant K from node I to I and represents the heuristic factor, usually $1/d_{ij}$. The corresponding importance needs to be utilized α, β . s represents the neighboring nodes of the current time t . After the completion of all ants in each link, the global update formula is as follows:

$$\begin{aligned} \tau_{ij}(t+1) &= (1 - \rho) \tau_{ij}(t) + \Delta \tau_{ij}, \\ \Delta \tau_{ij} &= \sum_{k=1}^m \Delta \tau_{ij}^k, \end{aligned} \quad (21)$$

where $\rho (0 < \rho < 1)$ represents the volatility coefficient, $1 - \rho$ represents the permanent coefficient of pheromone, using the information increment in the ant week model $\Delta \tau_{ij}$. After the above derivation, as shown in the figure, we show the basic model of the Ant (ACO) algorithm:

$$\Delta\tau_{ij} = \begin{cases} \frac{Q}{L_k}, & \text{ant } k \text{ passes a link between } I \text{ and } j, \\ 0, & \text{other,} \end{cases} \quad (22)$$

where Q represents a constant, which refers to the intensity of pheromone, and L_k represents the length of the path traveled by the KTH ant.

Through optimization and improvement analysis of TSN scheduling algorithm combined with ant colony system, the improved form of the initial pheromone is shown as follows:

$$\tau_0 = \frac{1}{A + \text{count}(U_{J_i(i)})}, \quad (23)$$

where A represents the parameter, U represents the complement of the node set, and $\text{count}(U_{J_i(i)})$ represents the number of currently impassable nodes in the vicinity of the node.

In order to improve the level of actual random search, the adaptive pseudo-random ratio should be used to obtain the next node j . The specific formula is as follows:

$$j = \begin{cases} \arg \max \left\{ \tau_{ij} [\eta_{ij}]^\beta \right\}, & q \leq q_0, \\ s, & q > q_0, \end{cases} \quad (24)$$

where q_0 will usually get a constant in the range of 0 to 1 and q can also be selected at random. The probability of getting the next node needs to be determined by q . Assuming that the number goes down gradually, the corresponding probability goes up. When the amount of information and heuristic factors between nodes are strengthened, the results are as follows:

$$\tau_{ij}(t+1) = (1-\rho)^{\sqrt{m+clm}} \tau_{ij}(t) + \Delta\tau_{ij}^{bs}, \quad (25)$$

in $\tau < \tau_{\max}$. At this time, it is derived as

$$\tau_{ij}(t+1) = (1-\rho)^{\sqrt{m-clm}} \tau_{ij}(t) + \Delta\tau_{ij}^{bs}, \quad (26)$$

where $\rho \in (0, 1)$ represents the volatility coefficient of pheromone, while t represents the specific parameter. The corresponding algorithm flow chart is shown in Algorithm 1.

3. Results

3.1. Performance Analysis of Transmission Scheduling. First, the application performance of the proposed improved algorithm shows that the impact of the number of event-triggered flows in the industrial IoT should be studied. The improved algorithm MMAS uses the network topology of the switch and the host to detect the working hours of industrial data processing at different times; the algorithm is compared with 10 traditional simulated annealing algorithms (TSA) and 11 traditional genetic algorithms (TGA). The final results are shown in Figures 6 and 7.

According to the analysis of the picture presentation results, the working time needed to outline the algorithm is less than the other two algorithms [16].

The actual number is controlled between 13 and 86, and the 60 event trigger streams are used for smart city transmission data processing, and finally, the working hours are carefully detected under different network topologies. Based on the linear optimization of topology size, the improved algorithm proposed in this paper is significantly lower than the results of other algorithms [17].

This paper designs two standard forms, one with 4 switches and 12 hosts, 6 switches with 24 hosts, and 60 event trigger streams to process and transmit data at any time. Compared with the above three algorithms, 30 operations are performed, and the results are shown in Table 1. At this time, it is shown to improve the convergence of the algorithm under different network scales. This shows that the target value and average value of the improved algorithm are better than the other two algorithms, and the number of optimal solutions in the 30-run process is more than that of the other two algorithms [18].

Combined with the performance comparison results shown in Table 1, we study the target convergence results of three algorithms at two sizes, as shown in Figure 8 [18].

Compared with the convergence at two scales, the improved algorithm is superior and the optimal solution can be obtained quickly. It proves that the MMAS algorithm can show positive advantages in the intelligent urban data management based on SDN network architecture, can effectively deal with the TSN data processing problems, appear no falling problems similar to the traditional algorithm, and thus improve the computational efficiency and quality of the actual optimal solution. [19].

3.2. Controller Performance Analysis. Considering the jitter change of the network time continuation, the observation and analysis show that when the ants look for the initial route, the number of pheromones in the link is very small, so the initial jitter change is very large. On the basis of increasing iterations, the jitter level of the design algorithm began to reach stability. On the basis of increasing the number of iterations, it can be lower than 22% from the perspective of parameters, so the jitter level of the design algorithm begins to stabilize. In addition, when studying the convergence of the target value obtained by triggering the flow algorithm at different times, 6 switches and 24 hosts are selected to build a network scenario, and the control quantity value is between 12 and 68. It is concluded that the actual number of iterations of the simulation results reaches 50 times, and the final results are shown in Table 2.

It is proved that the algorithm selected in this paper has better performance. In order to prove that the algorithm selected in this paper has better performance, the influence on the simulation is studied in combination with the number of network connections. Based on the continued increase in the number of network links, the objective function value will be increased, and the results show that the increase in the improved algorithm is significantly lower than the original algorithm. Therefore, the improved algorithm can meet the needs of system design. Through observation and analysis, it can be seen that the working time of the two


```

Input:  $G(V, E)$ 
Initialize:  $\tau_0, [\tau_{ij\min}, \tau_{ij\max}], \alpha, \beta, \rho, \text{MacGen}, \text{MaxN}$ 
Output:  $u(n)$ 
Procedure SET_TABU_INFORMATION
For  $\forall k \in M$ 
Build  $J_k(i) (i = 1, 2, \dots, n) - \text{tabu}_k$ 
End for
End procedure
Procedure CONSTRUCT_ROUTES
For  $i, j \in V$ 
For  $k \in M$  do
Select the next node according to the following formula

$$j = \begin{cases} \arg \max\{\tau_{ij}[\eta_{ij}]^\beta\}, & q \leq q_n \\ s, & q > q_n \end{cases}$$

End for
End for
For  $\forall k \in M$  do
Continue to pathfinding
End for
End procedure
Procedure UPDATE_PHEROMONES
To calculate  $L_{\text{hen}}$ 
update  $\tau_{ij}(t+1) = \begin{cases} (1-\rho)^{\sqrt{m+c/m}} \tau_{ij}(t) + \Delta\tau_{ij}^{bc}, & \tau > \tau_{\max} \\ (1-\rho)^{\sqrt{m-c/m}} \tau_{ij}(t) + \Delta\tau_{ij}^{bc}, & \tau < \tau_{\max} \end{cases}$ 
End procedure
Procedure MAIN
For  $\forall (i, j) \in E$  do
 $t_{ij} \leftarrow \tau_0$ 
 $\eta_{ij} \leftarrow 1/d_{ij}$ 
End for
While the termination condition is not met do
SET_TABU_INFORMATION
CONSTRUCT_ROUTES
UPDATE_PHEROMONES
End while
End procedure

```

ALGORITHM 1: MMAS algorithm.

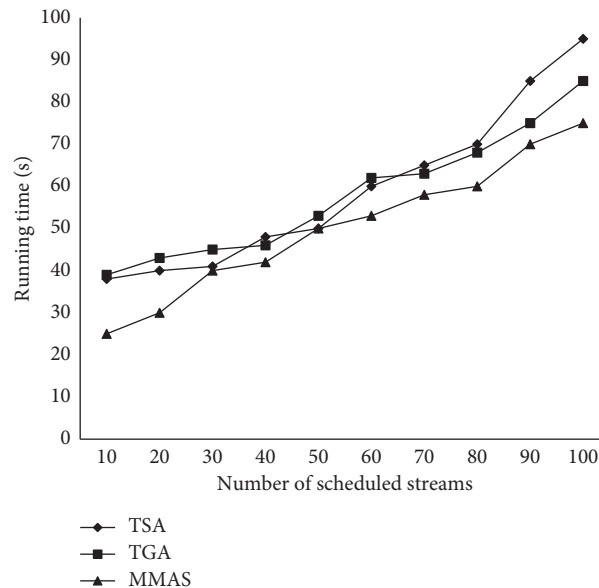


FIGURE 6: Research algorithm running time based on the number of trigger flows.

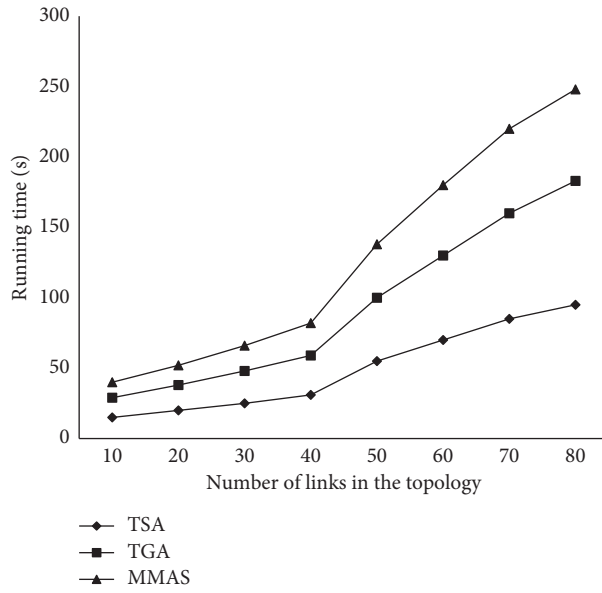


FIGURE 7: Run time according to the network topology research algorithm.

TABLE 1: Performance comparison results of the two network sizes.

Network size	Algorithm	Number of optimal solutions	Optimal solution	Average
Size 1	TSA	12	33	35.6
Size 1	TGA	13	32	34.2
Size 2	MMAS	19	29	29
Size 1	TSA	15	136	151.3
Size 2	TGA	11	145	160.7
Size 2	MMAS	18	126	135.2

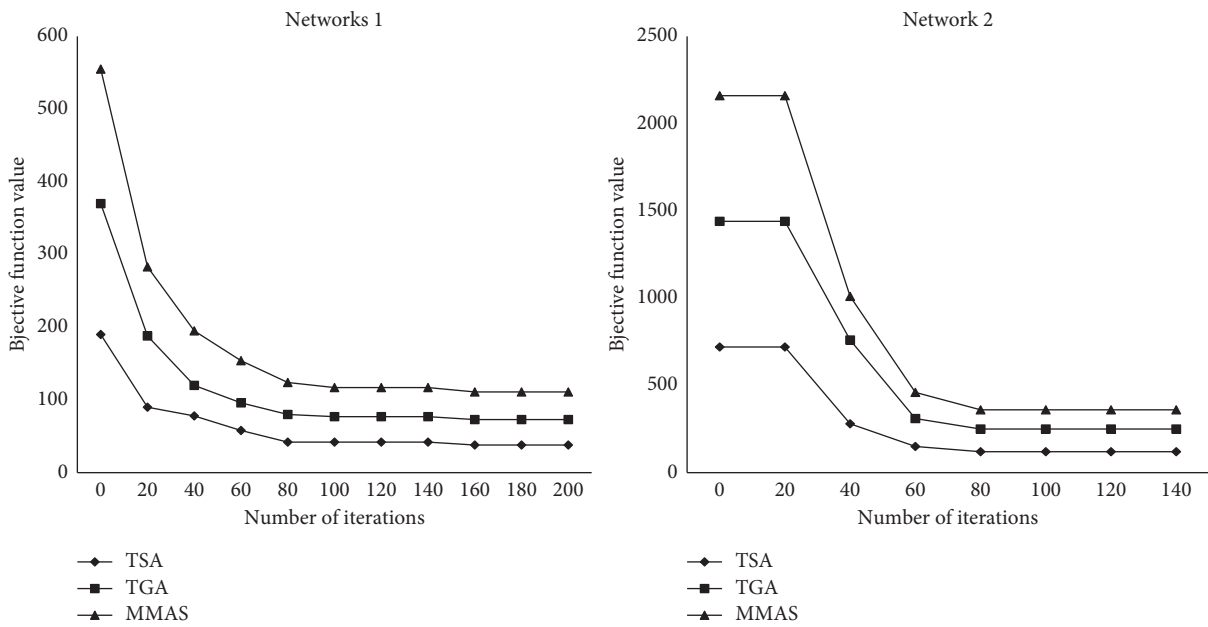


FIGURE 8: The algorithm target convergence situation.

TABLE 2: Comparison results of the algorithm performance.

Purpose of the flow	Optimal solution obtained	Average solution	Average number of iterations
12	126.2	133.4	21.23
	118.5	124.6	16.45
26	132.1	136.1	22.21
	121.3	125.7	16.36
40	136.4	140.2	18.77
	126.5	128.6	15.12
68	137.1	139.8	16.15
	125.9	128.2	14.2

algorithms will increase along with the continuous linear increase of the network topology. For smaller network structures, the incremental ant colony algorithm takes a longer time than the original one. However, from the perspective of large-scale network structure, the incremental ant colony algorithm has a shorter running time. Therefore, the incremental ant colony algorithm can effectively deal with large-scale network scheduling problems in the SDN network architecture.

4. Conclusion

To sum up, with the in-depth research on industrial Internet of Things technology concept in recent years, domestic and foreign researchers gradually realize the positive role of ant colony algorithm in the overall technical research and start to use the soft-considered network (SDN) framework to provide basic guarantee for data processing transmitted by a smart city. In this study, the scheduling problems in smart city data management design are static scheduling. It changes the possibility that some unpredictable disturbances may interfere with our schedule due to the complexity of the intelligent industry environment and uses the minimum ant system (MMAS) algorithm to design the system framework. For better improvement, dynamic scheduling is selected instead. At the same time, the data processing problem is discussed deeply. In this way, we not only can find effective optimization countermeasures but also further improve the urban intelligent construction system to meet the needs of urban residents and industrial development [20, 21].

Data Availability

This article did not collect relevant data but designed the algorithm and added parameters for simulation. The data have been reflected in the table.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the present study.

References

- [1] N. Abosata, S. Al-Rubaye, G. Inalhan, and C. Emmanouilidis, "Internet of things for system integrity: a comprehensive survey on security, attacks and countermeasures for industrial applications," *Sensors*, vol. 21, no. 11, p. 3654, 2021.
- [2] R. W. Wiers, S. L. Ames, W. Hofmann, M. Krank, and A. W. Stacy, "Impulsivity, impulsive and reflective processes and the development of alcohol use and misuse in adolescents and young adults," *Frontiers in Psychology*, vol. 1, no. 6, p. 144, 2010.
- [3] X. Xu, D. Zhu, X. Yang, S. Wang, L. Qi, and W. Dou, "Concurrent practical byzantine fault tolerance for integration of blockchain and supply chain," *ACM Transactions on Internet Technology*, vol. 21, no. 1, pp. 1–17, 2021.
- [4] I. Cerrato, F. Risso, R. Bonafiglia, K. Pentikousis, G. Pongrácz, and H. Woesner, "COMPOSER: a compact open-source service platform," *Computer Networks*, vol. 139, pp. 151–174, 2018.
- [5] A. J. Pinheiro, E. B. Gondim, and D. R. Campelo, "An efficient architecture for dynamic middlebox policy enforcement in SDN networks," *Computer Networks*, vol. 122, pp. 153–162, 2017.
- [6] X. Xu, Q. Huang, H. Zhu et al., "Secure service offloading for internet of vehicles in SDN-enabled mobile edge computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3720–3729, 2021.
- [7] F. Smirnov, M. Glas, and F. Reimann, "Formal timing analysis of non-scheduled traffic in automotive scheduled TSN networks," in *Proceedings of the Conference on Design. European Design and Automation Association*, pp. 1643–1646, Lausanne, Switzerland, September 2017.
- [8] M.-H. Chen, Y.-C. Tien, Y.-T. Huang, I.-H. Chung, and C.-F. Chou, "A low-latency two-tier measurement and control platform for commodity SDN," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 98–104, 2016.
- [9] X. Xu, Z. Fang, J. Zhang et al., "Edge content caching with deep spatiotemporal residual network for IoV in smart city," *ACM Transactions on Sensor Networks*, vol. 17, no. 3, pp. 1–33, 2021.
- [10] R. De Donno, A. Ghidoni, G. Noventa, and S. Rebay, "Shape optimization of the ERCOFTAC centrifugal pump impeller using open-source software," *Optimization and Engineering*, vol. 20, no. 3, pp. 929–953, 2019.
- [11] X. Luo, J. Liu, D. Zhang, and X. Chang, "A large-scale web QoS prediction scheme for the industrial internet of things based on a kernel machine learning algorithm," *Computer Networks*, vol. 101, pp. 81–89, 2016.
- [12] B. Rahmani, "Industrial internet of things: design and stabilization of nonlinear automation systems," *Journal of Intelligent and Robotic Systems*, vol. 86, no. 3-4, 2017.
- [13] T. Miyashita, T. Suzuki, T. Soumiya, and A. Yamada, "SDN solution for wide area networks," *Fujitsu Scientific & Technical Journal*, vol. 52, no. 2, pp. 28–34, 2016.
- [14] Y. Han, T. Vachuska, A. Al-Shabibi, and J. Li, "ONVisor: towards a scalable and flexible SDN-based network

- virtualization platform on ONOS,” *International Journal of Network Management*, vol. 28, no. 2, pp. 1–20, 2018.
- [15] J. Lian and M. Gao, “Design and Implementation of network information service platform based on microservice architecture,” *Journal of Physics: Conference Series*, vol. 1678, p. 012094, 2020.
- [16] F. Dongqing and X. Zhu, “Design and implementation of 6LOWPAN smart city data acquisition system,” *Computer Engineering*, vol. 11, pp. 286–291, 2017.
- [17] Y. Zhang, P. Tan, H. Ma, and M. Don, “Improving the seismic performance of staircases in building structures with a novel isolator,” *Computer Modeling in Engineering and Sciences*, vol. 124, no. 2, pp. 415–431, 2020.
- [18] L. Zhang, L. He, and J. Huang, “Research on secure device routing based on SDN network,” *Computer Engineering and Applications*, vol. 54, no. 4, pp. 103–109, 2018.
- [19] P. Gao, F. Zhang, and D. Zhang, “High deterministic traffic control method for cloud architecture networks based on SDN,” *Computer Engineering*, vol. 44, no. 12, pp. 80–84+90, 2018.
- [20] Y. Jin, Y. Liu, and X. Wang, “Multi-path traffic scheduling algorithm for data center network based on SDN,” *Computer Science*, vol. 6, pp. 90–99, 2019.
- [21] F. Li, Z. Yu, and C. Qin, “Constructive texture steganography based on compression mapping of secret messages,” *Computer Modeling in Engineering and Sciences*, vol. 124, no. 1, pp. 393–410, 2020.