Cross-Layer Optimization for Industrial Internet of Things in Real Scene Digital Twins

Zhihan Lv, *Senior Member, IEEE*, Jingyi Wu, *Student Member, IEEE*, Yuxi Li, and Houbing Song, *Senior Member, IEEE*

***Abstract*—The development of the Industrial Internet of Things (IIoT) and digital twins (DTs) technology brings new opportunities and challenges to all walks of life. The work aims to study the cross-layer optimization of DTs in IIoT. The specific application scenarios of hazardous gas leakage boundary tracking in the industry is explored. The work proposes an industrial hazardous gas tracking algorithm based on a parallel optimization framework, establishes a three-layer network of distributed edge computing based on IIoT, and develops a two-stage industrial hazardous gas tracking algorithm based on a state transition model. The performance of different algorithms is analyzed. The results indicate that the tracking state transition and target wake-up module can effectively track the gas boundary and reduce the network energy consumption. The task success rate of the parallel optimization algorithm exceeds 0.9 in 5 s. When the number of network nodes in the state transition algorithm is N = 600, the energy consumption is only 2.11 J. The minimum tracking error is 0.31, which is at least 1.33 lower than that of the exact conditional tracking algorithm. Therefore, the three-layer network edge computing architecture proposed here has an excellent performance in industrial gas diffusion boundary tracking.**

***Index Terms*—Cross-layer optimization, digital twins (DTs), hazardous gas tracing, Industrial Internet of Things (IIoT), parallel optimization framework.**

I. Introduction

Internet of Things (IoT) is changing the way people interact with things around them. The emergence of low-cost microsensors and high-bandwidth wireless networks in the Industrial IoT (IIoT) means that even the smallest devices can be connected as long as there is a certain level of digital intelligence [1]. Shen *et al.* [2] reported that virtual reality had become a feasible alternative to traditional learning methods in various knowledge fields. Besides, there were usually some toxic gases harmful to the human body used as production raw materials in the manufacturing environment of an intelligent factory, which could pose a significant threat to the life safety of front-line workers if they were not correctly stored and leaked [3]. It is necessary to predict and locate the gas diffusion boundary and evacuate the staff in time. Edge computing can significantly improve the Quality of Service (QoS) of IIoT [4]. The term edge has taken on a new definition in the world of the IoT, referring to somewhere near the device end. Therefore, according to the literal definition, edge computing is the computing generated near the device end. Edge computing solves significant problems, including high latency, network instability, and low bandwidth in traditional cloud computing or central computing mode.

Digital twins (DTs) technology can build a complete separation in the digital world through digital means for the entities of the physical world. The virtual world can maintain a real-time interactive connection with the physical entities. It also realizes the understanding, analysis, and optimization of physical entities through simulation, verification, prediction, and control of the whole life cycle process of the physical entities and with the help of historical data, real-time data, and algorithm models [5]. The IoT applied in the industrial industry can be summarized into four levels: 1) data collection and display; 2) fundamental data analysis and management; 3) deep data analysis; and 4) application and industrial control [6]. Most tasks in IIoT are delay-sensitive tasks. If all IoT devices are connected to the cloud, the computer will face an excessive burden, and data congestion will easily occur in the transmission process. The core idea of edge computing is to move the task forward, that is, completing the calculation on the node close to the application end to meet the real-time and security requirements of the system [7]. Edge computing is particularly crucial for these scenarios where "cannot wait" for data to be uploaded to the cloud for decision making and interaction, such as gas diffusion monitoring in the industry. It calculates closer to the data to reduce the "commuting time" of the data [8]. In other words, edge computing can effectively deal with the cloud center load, data privacy protection, and other problems that the cloud computing model cannot solve in the era of edge big data processing. Edge computing is not marginal.

Data upload in intelligent network physical systems is facing new challenges in energy saving and privacy protection to provide fine-grained access to different dimensions of the physical world [9]. It is always vital for participants to consume as little energy as possible for data upload. However,

Manuscript received 2 December 2021; revised 14 January 2022; accepted 6 February 2022. Date of publication 18 February 2022; date of current version 24 August 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61902203. *(Corresponding author: Zhihan Lv.)*

Zhihan Lv is with the Department of Game design, Faculty of Arts, Uppsala University, 751 05 Uppsala, Sweden (e-mail: lvzhihan@gmail.com).

Jingyi Wu and Yuxi Li are with the College of Computer Science and Technology, Qingdao University, Qingdao 266071, China (e-mail: wjy_515151@163.com; 7117139@163.com).

Houbing Song is with the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114 USA (e-mail: h.song@ieee.org).

Digital Object Identifier 10.1109/JIOT.2022.3152634

2327-4662 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

the mere pursuit of energy efficiency may lead to extreme disclosure of private information, especially when participants upload more real-time content than ever before. Cai and Zheng [10] proposed a new mechanism for data upload in an intelligent network physical system, which simultaneously considered energy-saving and privacy protection.

Therefore, the IIoT architecture integrating edge computing is significant for gas diffusion tracking in intelligent factories. It is also an effective way to connect independent factories, workshops, and machines to improve production efficiency. The work adopts the methods of literature research and algorithm case analysis. The innovation lies in a Two-stage continuous object tracking algorithm based on state transition (STM-COT) model. The outstanding contribution is the STM-COT model that can realize gas diffusion boundary early warning, the tracking state conversion, and a target wake-up module, improving the target tracking efficiency while reducing the energy consumption.

II. Research Status of Gas Monitoring and Edge Location in IIoT

Driven IoT, gas sensors combined with various communication technologies and artificial intelligence (AI) technologies are gradually extensively used in IIoT harmful gas monitoring and edge positioning. The IIoT is also widely applied in all walks of life [11]. Interconnecting sensors with things over the Internet enables organizations to track real-time locations, monitor performance, improve workflow, and optimize utilization. With the development of computing technology and storage resources, the connotation and extension of DTs are becoming even more expansive. Conceived initially as mapping the natural world environment in the virtual space, DTs have increasingly better feasibility and availability.

DTs technology is a universal theory and technology system that can be applied in many fields, such as product design, product manufacturing, medical analysis, and engineering construction. Shah and Mishra [12] established a low-power IoT environment monitoring system to monitor ten important air quality parameters in real-time. Adamu *et al.* [13] detected various toxic industrial gases from 1480 to 1700 nm with a simple customized supercontinuum laser source. The laser source showed high responsiveness and selectivity for continuous detection in a 2.4-m long hollow-core photonic ribbon fiber with a 20-μm core diameter ray system. Wang *et al.* [14] proposed a formaldehyde gas sensor with excellent sensing characteristics based on Cu doped Sn to detect toxic gases by layered nanoflowers synthesized by a brief hydrothermal method. The combination of edge computing and AI, edge intelligence, is a hot research direction in academic circles. Edge intelligence mainly studies putting the AI model on the network's edge. For example, for deep neural networks, federated learning is an excellent training framework that could be tentatively placed on resource-constrained terminals and edge devices. Moreover, multitudes of data generated by IoT devices have greatly influenced sensory data query and processing [15]. Due to the limitation of computing and data transmission capacity in traditional wireless sensor networks, the current query processing methods are no longer effective. In addition, processing a considerable amount of sensory data will overload the cloud. Cai and Shi [16] studied the query processing in the edge assisted IoT data monitoring system to solve these problems.

It can be seen that in the research on the gas boundary location in the past few years, the gas diffusion model has matured, and most methods can predict the location of gas all the time. However, few studies are on the combination of the gas diffusion model and continuous object tracking based on IIoT, so more research data are needed for further exploration. The work provides a new idea for gas diffusion location in IIoT and can open a new research direction for cross-layer optimization of IIoT.

III. Research on DTs-Oriented Two-Stage Harmful Industrial Gas Diffusion Boundary Tracking

A. Harmful Industrial Gas Tracking and Detection in IIoT Based on DTs

DTs can maintain the digital matching of physical objects and operations using the data of intelligent devices such as sensors. The channel connecting the DTs and its physical matching part is called the digital thread. Industrial automation is experiencing a digital revolution that Industry 4.0 and DTs technology have introduced innovative methods to execute manufacturing processes. DTs technology allows industrial manufacturers to create digital models of actual machines. Then, they can combine digital and physical data to understand these physical machines' performance and potential value. This technology is suitable for all machine functions, such as production, maintenance, and user experience. Applying DTs to harmful industrial gas monitoring can effectively prevent heavy casualties caused by industrial gas leakage and is vital in promoting industrial automation. DTs have a wide range of advantages and applications. Through digital models, manufacturers can simulate plans, processes, and systems without risk, visualize various new strategies and results, and invest actual resources in their development.

IoT is the information carrier of the Internet and traditional telecommunication network, and it can accomplish the interaction between all the general objects that can exercise independent functions. There are many industrial hazardous gases in IIoT, such as toxic gases, nuclear radiation, and air suspended solids, which are badly destructive for the obscure boundary and covering an extensive range of targets [17].

As shown in Fig. 1, the tracking and positioning of industrial hazardous gases in IIoT are a process of obtaining the location information of the boundary of industrial hazardous gases through event-driven or regular collection and by the collaboration between sensing nodes [18]. The tracking process consists of four steps: 1) scilicet sensing by sensing nodes; 2) exchanging information between different nodes; 3) predicting future locations; and 4) uploading sensing results, to obtain coverage area and location information of the harmful gas [19].

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

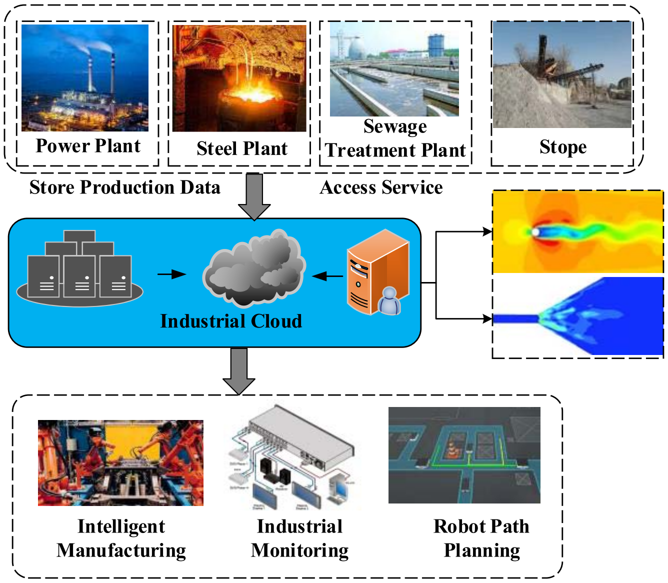


Fig. 1. Application scenario of IIoT—industrial gas boundary tracking.

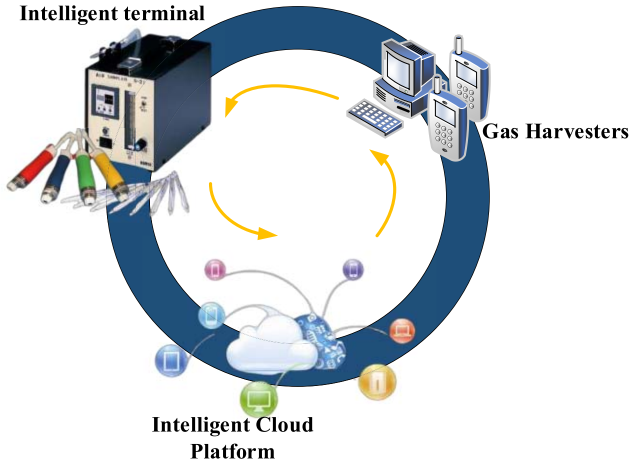


Fig. 2. Hazardous gas monitoring process in IIoT.

In the production process, the leakage of flammable and toxic gases often occurs, and the released flammable or toxic gases are quickly diluted by air. After a period of time, these gases accumulate in local areas or a dead corner. The combustible gas reaching a certain concentration in the air is prone to burn or even explode when exposed to fire. As shown in Fig. 2, by accessing various gas sensors, real-time data of toxic gas and combustible gas can be collected for real-time monitoring [20]. Compared with the traditional field detection method of handheld instruments by operating personnel, IoT remote gas monitoring system can realize real-time monitoring of hazardous gases without contact. As the core and foundation of applications of IoT, sensors are the bridge of various information and AI technologies [21]. The gas sensor is an electronic device that can convert the composition and concentration of the detected gas into identifiable signals, which integrates functional materials, microelectromechanical system technology [22], organic polymers, and electronic information [23].

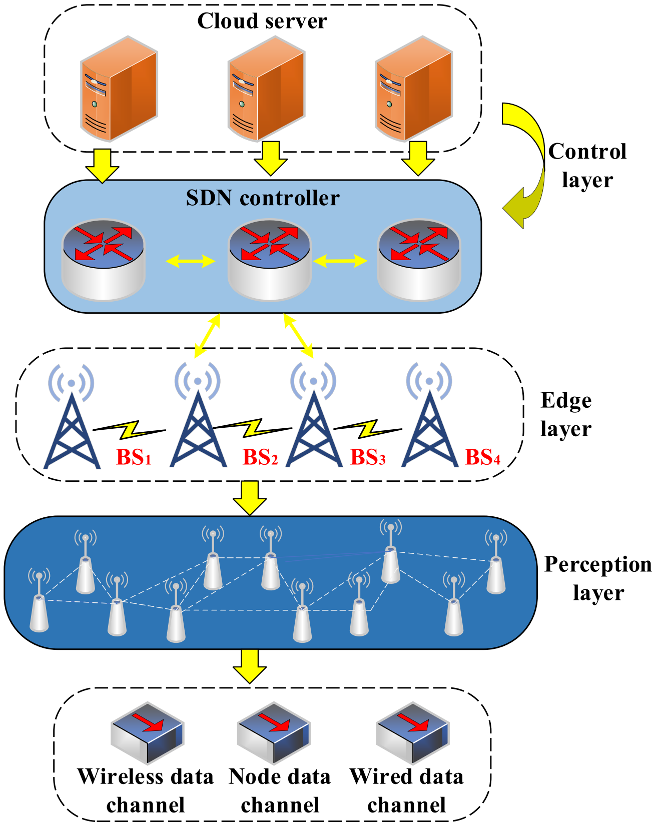


Fig. 3. Three-layer network architecture based on parallel optimization framework.

B. Industrial Hazardous Gas Tracing Algorithm Based on Parallel Optimization Framework

In IIoT, the sensing network composed of sensing nodes only has limited computing power, and industrial hazardous gas tracking cannot fully meet the requirements of industrial gas monitoring due to network delay, network energy consumption, and insufficient boundary tracking accuracy [24]. The work comprehensively considers gas's mobility and diffusion characteristics and the constraints of computing and network resources. A harmful industrial gas tracking algorithm for DTs is proposed based on the trajectory perception task unloading and distribution algorithm for harmful industrial gas, including control, edge, and perception layers. Fig. 3 shows the network architecture.

The three-layer network architecture depends on the distributed edge computing architecture of IIoT. When processing a tracking task, the first cluster node in Position 1 collects the information in the cluster, and calculates the boundary in the cluster to diffuse to Position 2 at the next reporting time. It is necessary to evacuate the staff in Position 2 in time and wake up and track the nodes around Position 2 at the next reporting time. This architecture can perform edge computing since it supports the software-defined network technology [25].

As an IoT device in the smart factory, the sensing layer can sense gas concentration and communicate with base stations [26]. The tracking method of boundary position of continuous objects [27] is adopted to collect the gas concentration information at the y node on the boundary of the targeted

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

toxic gas according to a certain period. The optimal boundary node acts as the cluster head after competing with other nodes. Then, the boundary nodes transmit the collected concentration information to the cluster head, and the cluster head uniformly transmits it to the nearest base station.

The edge layer is an intermediate device for transmitting messages to cloud nodes, in which different base stations are evenly distributed. The mutual communication between base stations forms an edge layer network responsible for processing the tasks in the edge layer. Specifically, it not only transmits the information submitted by the cluster head of the sensing node to the cloud but also deals with the tasks of the edge layer and manages the cooperative work between the edge layer and base stations.

The control layer is composed of multiple cloud servers, which can connect the base stations of the edge layer to the cloud, concurrently process the task information uploaded by the edge layer, schedule the tasks, and allocate resources reasonably.

The set of cluster heads and base stations are expressed as $N = \{i|i = 1, 2, \ldots, |N\}$ and $B = \{B_{Sl}|l = 1, 2, \ldots, |B\}$, respectively. Each cluster head has its own calculation task, which is processed by the perception layer, network layer, and control layer under the premise of meeting the allowable range of system delay. Suppose that A_i (l_i, k_i) represents the calculation task of cluster head i, l_i denotes the size of data in the task, and k_i refers to the number of machine cycles required to complete the task. Moreover, denote d_i^L, $d_{i,j}^E$, and d_i^C as the unloading mode indicator, and $d_i^L, d_{i,j}^E, d_i^C \in \{0, 1\}$. Then, $d_i^C = 1$ demonstrates that the calculation task is executed locally by the base station, $d_{i,j}^E = 1$ means that the calculation task is dispatched by the cloud to each base station passing through a section of the boundary corresponding to the cluster head. Besides, $d_i^L = 1$ indicates that the calculation task is executed by the cloud. In addition, (1) is satisfied

$$d_i^L + \sum_{j=1}^{|P_i|} d_{i,j}^E + d_i^C = 1. \tag{1}$$

Equation (1) demonstrates that there is only one way to execute the tasks of a node, and only one of d_i^L, $d_{i,j}^E$, and d_i^C equals 1.

When executing the calculation model locally, sensing node i first generates a computing task A_i (l_i, k_i). Then, it sends the task, link information, and base station information related to the gas boundary to the cloud server of the control layer through source base station (SBS) closest to the node. After scheduling, the cloud server returns the task decision to the node i. If the calculation is executed locally, the node i will calculate the position of the boundary within the cluster at the next moment.

Because the computing speed of the cloud server is very fast and there is only a small amount of decision task data, the delay of the scheduling process can be ignored. Then, the total delay of the calculation task A_i (l_i, k_i) can be expressed as

$$t_i^L = t_i^{\text{exec}}. \tag{2}$$

In (2), t_i^{exec} represents the latency of local execution of the node i.

f_i^L is marked as the number of machine cycles per second of node i, and then, the operational delay of a task can be presented as

$$t_i^{\text{exec}} = \frac{k_i}{f_i^L}. \tag{3}$$

In the edge computing model, the first cluster node i transmits the information to the SBS after collecting the perceived boundary information, and then the task A_i (l_i, k_i) is transmitted to the cloud server of the control layer by SBS. The cloud server carries out task scheduling, and selects the appropriate base station for execution among the base stations passing through the boundary of the cluster. Finally, the cloud server returns the results to the first cluster node i.

Assume that $t_{i,j}^E$ signifies the total delay of task A_i (l_i, k_i) unloading from node i to base station $BS_j (j \in P_i)$, and the base station BS_j is the base station assigned by the cloud server to task A_i (l_i, k_i). Denote p_i as a collection of base stations that node i passes along an inner cluster boundary. The total delay is divided into the operation delay $t_{i,j}^{\text{exec}}$, waiting delay $t_{i,j}^{\text{que}}$, and transmission delay t_i^{trans}. Then, the total delay of node computing task A_i is written as (4)

$$t_i^E = t_{i,j}^{\text{exec}} + t_{i,j}^{\text{que}} + t_i^{\text{trans}}. \tag{4}$$

Equation (5) describes the transmission delay t_i^{trans}

$$t_i^{\text{trans}} = \frac{l_i}{su_i} + \frac{\mu_i l_i}{sd_i} + \omega_{i,j}\left(\frac{l_i}{bu_{\text{j}}} + \frac{\mu_i l_i}{bd_j}\right). \tag{5}$$

In (5), t_i^{trans} represents the delay of uploading the task to the base station, $(l_i/su_i) + [(\mu_i l_i)/(sd_i)]$ refers to the delay of downloading the task from the SBS, and $\omega_{i,j}([l_i/bu_j] + [(\mu_i l_i)/bd_j])$ denotes the delay generated during downloading the task from SBS and the returning results by the base station. In addition, su_i indicates the upload speed of the node i, sd_i represents the download speed, and $\omega_{i,j}$ is the symbol of whether the base station is the SBS. $\omega_{i,j} = 1$ represents noncoincidence, and $\omega_{i,j} = 0$ signifies coincidence. Besides, bu_j and bd_j denote the upload and download rates of the base station.

The waiting delay $t_{i,j}^{\text{que}}$ is the sum of the operation time delay of all the tasks in the task queue of base station BS_j before the task A_i is completed, which can be written as

$$t_{i,j}^{\text{que}} = \sum_{n=1}^{idx(A_i)-1} t^E(n). \tag{6}$$

In (6), n represents the nth task queued in the base station BS_j, $idx(A_i)$ stands for the index of the task A_i, and $t^E(n)$ refers to the operation delay of the nth task in the base station BS_j.

Then, the operation delay of the node can be expressed as

$$t_{i,j}^{\text{exec}} = \frac{k_i}{f^C}. \tag{7}$$

In (7), f^c denotes the calculation power of the base station.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

Similarly, the total delay t_i^C of the cloud computing model contains the operation delay $t_{i,j}^{\text{exec}}$, waiting delay $t_{i,j}^{\text{que}}$, and transmission delay t_i^{trans}

$$t_i^C = t_{i,C}^{\text{que}} + t_{i,C}^{\text{trans}} + t_{i,C}^{\text{exec}}. \tag{8}$$

The transmission $t_{i,C}^{\text{trans}}$ refers to the process of sensing node i uploading the task from SBS to the cloud server and the process of the cloud server returning calculation results to the first cluster node i, as shown in

$$t_{i,C}^{\text{trans}} = \frac{l_i}{su_i} + \frac{\mu_i l_i}{sd_i} + \frac{l_i}{bu_s} + \frac{\mu_i l_i}{bd_s}. \tag{9}$$

The waiting delay $t_{i,C}^{\text{que}}$ is the total operation delay of tasks in the queue before task A_i (l_i, k_i), which is calculated according to the following:

$$t_{i,C}^{\text{que}} = \sum_{n=1}^{idx(A_i)-1} t^C(n). \tag{10}$$

In (10), n denotes the index of the task queue of the cloud server, $idx(A_i)$ represents the index of task A_i, and $t^C(n)$ represents the operation delay of the nth task in the cloud server.

The operation delay $t_{i,j}^{\text{exec}}$ can be written as

$$t_{i,C}^{\text{exec}} = \frac{k_i}{f^C}. \tag{11}$$

Since hazardous gas tracking requires high delay sensitivity, the task unloading and distribution are optimization in this report. The following equation displays the vector of the first cluster node i

$$D_i = \left\{d_i^L, d_{i,j}^E, d_i^C\right\}. \tag{12}$$

Then, the set of the whole task unload can be expressed as follows:

$$D = \{D_i | i \in N\}. \tag{13}$$

The following optimization objective models are proposed to minimize the delay:

$$P1: \ \min_D \ \frac{1}{N}\sum_{i=1}^{N} t_i \tag{14}$$

$$d_i^L, d_{i,j}^E, d_i^C \in \{0,1\}, i \in N, j \in P_i \tag{15}$$

$$t_i = d_i^L t_i^L + \sum_{j=1}^{|P_i|} d_{i,j}^E t_i^E + d_i^C t_i^C \tag{16}$$

$$d_i^L + \sum_{j=1}^{|P_i|} d_{i,j}^E + d_i^C = 1. \tag{17}$$

With the increase in the number of submitted tasks, optimizing the perception layer, edge layer, and control layer is necessary. First, D_i needs to be determined. A decision vector optimization method is proposed based on the simulated annealing algorithm, and the optimization process is introduced into this method so that the results gradually approach the optimal solution. The pseudocode of the vector decision optimization algorithm based on the simulated annealing algorithm is shown in Algorithm 1.

Algorithm 1 Vector Decision Optimization Algorithm

```
Start
Input: Initialization: initial temperature $T_0$, set the initial unloading decision vector set $D_0$, and the length of Markov chain $L$
Output: Optimal unloading decision vector set $D$
for $T < T_s$ $T < T_s$ do
    for $k < L$ do
        Random selection mutation method, two recombination method or three recombination method are used to produce new solutions $D'$
    if $\Delta C < 0$ then
        $D = D'$
    else
        if $\exp(\frac{-\Delta C}{T}) > random(0, 1)$ then
            $D = D'$
        end if
    end if
$k = k + 1$
end for
        $T = T_0 q$
End for
```

The node vector formula is transformed into the longitudinal form as follows:

$$D = \{x, y, z\}. \tag{18}$$

In (18), x, y, and z represent all tasks unloaded to the three-layer framework. Then, the tasks are decomposed into subproblem P_1 of the perception layer, subproblem P_2 of the edge layer, and subproblem P_3 of the control layer, which are expressed as (19)–(21)

$$P1' : \min_x \ f(x_i) \qquad x_i \in x, i \in N. \tag{19}$$

In (19), x_i denotes to the task assigned to the node i, $x = \{x_i | i = 1, 2, \ldots, |N|\}$, and f represents the calculation strategy of the perception layer. In the perception layer, the node has poor computing capacity and only performs a single task. If the computing capacity remains unchanged, the optimization method is just allocating tasks with a small amount of computation

$$P2' : \min_y \ g(y_l) \qquad y_l \in y, l \in B, l \in P_i. \tag{20}$$

In (20), y_l refers to the task subset assigned to the base station, $y = \{y_i | i = 1, 2, \ldots, |l|\}$, and g represents the calculation strategy of edge computing. The edge layer possesses a slightly stronger computing ability than the perception layer, so the task distribution strategy based on gas trajectory perception is introduced

$$P3' : \min_h \ h\big(z_{idx(A_i)}\big) \qquad z_{idx(A_i)} \in z, i \in N. \tag{21}$$

In (21), $z_{idx(A_i)}$stands for the index of the task unloaded to the cloud of the control layer, and h represents the computing

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

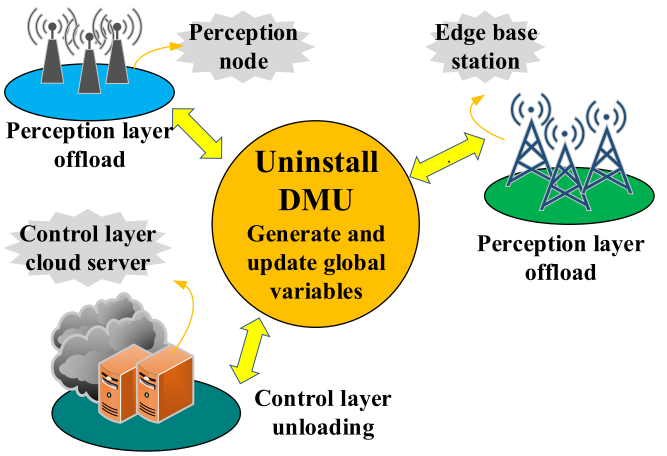


Fig. 4. Parallel optimization framework based on the three-layer architecture.

strategy of the control layer. The short task priority strategy is adopted for optimization. Fig. 4 reveals a parallel optimization framework based on the three-layer architecture.

C. Harmful Industrial Gas Tracing Algorithm Based on State Transition Model

The parallel optimization framework based on industrial hazardous gas tracing algorithm established here reduces the delay caused by the boundary position at the next reporting time of industrial harmful gas. In view of the fact that the probability of gas leakage in IIoT is generally not high, but once the leakage occurs or even triggers an explosion, it will pose an immeasurable safety threat to front-line staff and industrial losses to the factory. Chen *et al.* [28] pointed out that evacuation behavior was a critical factor that must be considered in the design of public structures. Therefore, it is essential to ensure the accuracy of detecting the gas position, and to ensure that the operation can be maintained for a long time. Correspondingly, the node should not be maintained in a long-term state of perception or wake-up, which increases energy consumption.

Consequently, the STM-COT is proposed considering the energy consumption of nodes, which not only can solve the problem of gas diffusion boundary warning but also can improve the efficiency of target tracking and reduce energy consumption through the tracing state-transition model and target awakening module. Because the gas diffusion is fast in the early stage, but the gas diffusion gradually slows down, and the boundary extends in the later stage. Therefore, in the early stage of diffusion, the gas is tracked in the unit of point, and in the later stage, it is tracked in the unit of cluster. Besides, the node state is divided into the sleep state, active state, and ready state. Only the boundary nodes that need to perform the tracking task are active, and the rest nodes are sleep. A wake-up area is established near the boundary, and the regional awakening algorithm is used to reduce the error.

In the deployment of the network environment, sensing nodes are equipped with gas sensors, temperature sensors, and pressure sensors. Each node has a unique identity document (ID), and it communicates by sending a "Hello" packet after a network is built. The node position is obtained by the satellite positioning of the global positioning system. In addition, the gas boundary adopts the 4σ principle of environmental engineering [29]. The sensing node is the general term of IIoT nodes with sensing ability for harmful gases. The event nodes and nonevent nodes are nodes within or outside the gas coverage. The boundary nodes are nodes around the leakage gas boundary. Furthermore, the temporary cluster is the temporary organization for the collection of the sensing data of the boundary nodes. The reporting cycle is the time for collecting the sensing data of the leakage gas boundary.

Here, the 2-D convection diffusion model is used as the gas diffusion model to calculate the gas concentration at a coordinate point t, as shown in (22)

$$c(x, y, t) = \frac{M}{4\pi Dt} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{4Dt}}. \tag{22}$$

In (22), M represents the total amount of diffusion material, and D denotes the diffusion coefficient, which represents the degree of gas diffusion. Meanwhile, (x_0, y_0) refers to the coordinates of the diffusion source point. Equation (23) displays the width of gas diffusion under the windless condition

$$L = 4\sigma = 4\sqrt{2Dt}. \tag{23}$$

In view of the energy consumption in the tracking process, the three states of the node are switched as follows. Nodes that are not notified of target approaching are in sleep state. The nodes in the sleep state are woken up regularly during sleep to check whether there is industrial harmful gas approaching. If there is no approaching gas, the node will return to sleep to save energy. If an object is detected, the node will be switched to the active state. If the node is notified by other nodes that there is industrial harmful gas approaching immediately, it wakes up in advance and changes to the ready state. The awakening node will switch to the active state if it detects the target. Otherwise, it will switch back to the sleep state. The active node collects, uploads, and predicts the boundary until it is no longer a boundary node. The node state-transition space can be expressed as $S = \{\text{Sleep, Active, Ready}\}$, and the behavior space is expressed as $V = \{\text{Transmission } (T), \text{ Listening } (L), \text{ Perception } (P), \text{ Calculation } (C)\}$, i.e., $V = \{T, L, P, C\}$. Moreover, the state-transition space $\text{SC} = \{\text{Undetected } (UD), \text{ Detected } (D), \text{ Woken } (W)\}$, that is $\text{SC} = \{UD, D, W\}$.

At the initial stage of gas diffusion, the discrete tracing algorithm is adopted because the diffusion speed is fast and the boundary is not obvious. Fig. 5 provides the process of the discrete tracing algorithm. After monitoring the gas boundary, the node first reports the sensing information, then the node directly transmits the sensing information to the sink node, and finally the node predicts and wakes up the boundary position at the next moment.

The node as a unit first determines the area to wake up, scilicet the area around the boundary in the next reporting period t_{np}. The nodes in this area need to be awoken for boundary tracking before the next reporting time comes. The nodes

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

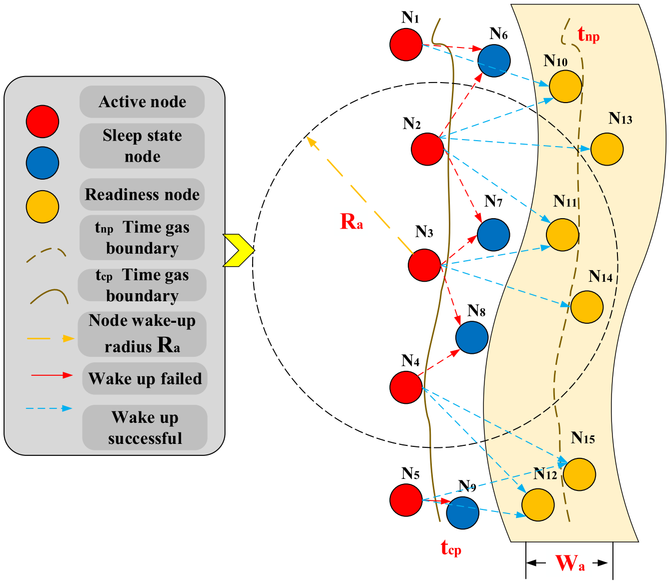


Fig. 5. Discrete tracing algorithm.

$N_{13}, N_{14}, \ldots, N_{21}$ in this area are ready for boundary perception and data collection when the next reporting time comes. The boundary nodes $N_1, N_2, \ldots, N_5$ at the current reporting period t_{cp} are at the active state, and nodes, $N_6, N_7, \ldots, N_{12}$ are at the ready state.

When node N_3 wakes up, it broadcasts Informing_Msg information to nodes N_7, N_8, N_{11}, and N_{14} within its communication range, and the node receiving the information judges whether its position is in the wake-up area. In Fig. 6, nodes N_{11} and N_{14} are within the wake-up area at the reporting period t_{np}, so they wake up to track the gas, while nodes N_7 and N_8 out of the area fail to wake up.

At the later stage of gas diffusion, the speed becomes slow, but the boundary lengthens, and the gas boundary becomes even wider with time. Besides, there is an excess of nodes. If the node is taken as the unit, the nodes will be repeatedly awakened. Therefore, the intercluster collaborative tracing algorithm is employed to reduce the energy consumption caused by repeatedly waking up nodes. The nodes first report the sensing information in clusters, and the cluster head directly uploads the sensing information to the sink node. Then, the cluster head predicts and awakens the nodes in the boundary position of the next moment according to the sensing information of the boundary nodes in the cluster. If a mass of boundary nodes upload information one by one, energy overconsumption will occur, or even network congestion will happen, so the information is aggregated and sent to the Sink node together. The aggregation takes the temporary cluster approach and is only responsible for current boundary information collection and future boundary prediction. The node first calculates the backoff time Tbackoff according to its own energy and the number of nodes in the communication range, as shown in

$$T_{\text{backoff}} = T_0 * \left[\alpha\left(1 - \frac{E_{\text{res}}}{E_0}\right) + \beta\left(1 - \frac{D_{\text{nei}}}{D_0}\right)\right]. \tag{24}$$

In (24), T_0 denotes the maximum backoff time, E_{res} refers to the residual energy of the node, and E_0 signifies the total energy of the node. In addition, D_{nei} represents the boundary node within the communication range of the node, and D_0 indicates the maximum number of nodes within the preset communication range of the node. Besides, α and β are the preset parameters of the system, responsible for adjusting the weights of the two items in the (24). This is conducive to selecting the node with more boundary node information and high energy as the cluster head.

The neighbor cluster is the cluster closest to the left and right of cluster C_j. Cluster C_j and these two clusters are neighbor clusters to each other, and they are defined as a set $C = \{C_j | j = 1, 2, \ldots, |C|\}$. Each node can only belong to a cluster, but there is an interval between two clusters which are neighbors. When the cluster head awakens the nodes around the boundary at the next reporting period, the nodes between neighbor cluster intervals will not be awakened, which will affect the accuracy of boundary tracking. Therefore, when awakening nodes, the distance between neighbor clusters is eliminated by exchanging information between neighbor clusters.

To eliminate the interval, it is necessary to determine a common node between two neighbor clusters. Therefore, the virtual partner node strategy is proposed. First, the neighbor cluster head table is established, containing the ID, coordinates of cluster heads, and the number of neighbor cluster boundary nodes. The set of nodes of the first cluster is defined as $\text{CH} = \{\text{CH}_l | l = 1, 2, \ldots, |\text{CH}\}$. Then, the node CH_l in the first cluster searches within the communication range. If it detects other clusters, the nearest cluster head is added to the neighbor cluster table. Moreover, it is essential to determine whether the newly searched cluster head CH_{new} is on the same side with the previous cluster head according to (25)

$$\cos\theta = \frac{\overrightarrow{V_{\text{ln}}}\overrightarrow{V_k}}{\left\|\overrightarrow{V_{\text{ln}}}\right\|\left\|\overrightarrow{V_k}\right\|}. \tag{25}$$

In (25), $\overrightarrow{V_{\text{ln}}}$ represents the vector obtained by the difference between the coordinates of the first cluster node CH_l and CH_{new}, and $\overrightarrow{V_k}$ denotes the vector obtained by the difference between the coordinates of the node CH_l and CH_{close}. Furthermore, θ represents the angle between two vectors

$$\theta = \left\langle \overrightarrow{V_{\text{ln}}}, \overrightarrow{V_k} \right\rangle. \tag{26}$$

If $\cos\theta$ is less than 0, CH_{new} will be added to the neighbor cluster table, and the search ends. If $\cos\theta$ is more than 0, CH_{new} will be abandoned. If the nearest cluster head is not found, CH_l will expand the communication radius.

After searching the neighbor clusters, CH_l checks the cluster heads in the neighbor cluster table. The left and right neighbor clusters are set to CH_L and CH_R. If there are clusters more CH_l than clusters CH_L, the nearest node to CH_L will be found and sent to CH_l. If there are more clusters CH_L than clusters CH_l, some nodes will be assigned by CH_L to CH_l. The right neighbor clusters are allocated in the same way. This allocation approach can effectively avoid the problem of uneven

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

distribution of nodes. If only neighbor clusters on one side are found, the nodes on one side are processed, and the virtual partner node strategy is used for allocation on the other side. If no cluster on either side is found, the virtual partner nodes on both sides will be established.

After eliminating the interval between clusters, it is essential to fit the gas boundary. Because the gas boundary covers a wide range with fluctuation, it is difficult to obtain the real boundary shape during tracking, so the idea of virtual node algorithm is used to improve the fitting degree of the boundary. The gas concentration value sensed by the sensing nodes is combined with the gas diffusion model to find out the node coordinates with the concentration of 0 and at the boundary.

Taking the boundary node in the reporting period t_{cp} as an example, the boundary width is first calculated according to the following:

$$L_{cp} = 4\sqrt{2Dt_{cp}}. \tag{27}$$

Then, the equation between the two points is calculated to solve the coordinates of two points away from the leakage source ($L_{cp}/2$), and the point closer to the boundary node is selected as the virtual node. The set Visual_Table of virtual node tables is defined as follows:

$$\mathrm{VP} = \{\mathrm{VP}_k(vx_k, vy_k) | k = 1, 2, \ldots, |\mathrm{VP}|\}. \tag{28}$$

Suppose the fitting equation of the virtual node is

$$p(x) = a + bx. \tag{29}$$

Then, a and b are calculated according to (30) and (31) when the mean square error Q (a, b) of the virtual node is the minimum

$$a = \frac{\sum_{k=1}^{|\mathrm{VP}|} y_k \sum_{k=1}^{|\mathrm{VP}|} {x_k}^2 - \sum_{k=1}^{|\mathrm{VP}|} x_k \sum_{k=1}^{|\mathrm{VP}|} x_k y_k}{|\mathrm{VP}| \sum_{k=1}^{|\mathrm{VP}|} {x_k}^2 - \left(\sum_{k=1}^{|\mathrm{VP}|} x_k\right)^2} \tag{30}$$

$$b = \frac{m \sum_{k=1}^{|\mathrm{VP}|} x_k y_k - \sum_{k=1}^{|\mathrm{VP}|} x_k \sum_{k=1}^{|\mathrm{VP}|} y_k}{|\mathrm{VP}| \sum_{k=1}^{|\mathrm{VP}|} {x_k}^2 - \left(\sum_{k=1}^{|\mathrm{VP}|} x_k\right)^2}. \tag{31}$$

Finally, the diffusion region is solved after the area is awaken.

IV. Simulation and Evaluation of Industrial Hazardous Gas Tracing Algorithm

A. *Simulation and Evaluation of Industrial Hazardous Gas Tracing Algorithm Based on Parallel Optimization Framework*

The simulation experiment is performed to compare the performance of different gas tracing algorithms. The perception layer nodes and edge layer nodes are distributed in a 1 km × 1 km square, the boundary is defined as a circle, the gas diffusion speed is 5 m/s, and the uplink and downlink speeds of the base station are 80 Mb/s. The transmission medium is cable, and the transmission speed is 1000 Mb/s the same as the uplink and downlink speeds of the cloud server. The computing power of nodes and the cloud server is set to 0.05 and 20 GHz, the unit of the data packet is MB, and the value range is (1, 30). Besides, the unit of task cycles is cycles per task (cpt), and the value range is (0.25, 0.75).

TABLE I
Parameter Setting of the DTs Mechanism Model

Network Deployment	Square Area of 200m × 200m	Total energy of nodes	0.5J
Power Consumption at Active State	3mW	Power Consumption at Sleep State	15u W/0.5m W
Power Consumption at Ready State	0.5m W	Number of Deployment Nodes	600 (Sparse) 900 (Medium) 1200 (Intensive)
Communication Radius	15m 20m	Coordinates of the Diffusion Center	(100,100)
Gas Diffusion Source Intensity	M=100kg/m^2	Width of the Wake-up Area	2m
Time Interval for Boundary Nodes to Report Perception Information	2s	Weight Coefficients α and β of Residual energy and node degree	α=β=0.5

Compare the performance of the following six algorithms.

1) *Random Algorithm:* That is, the task random allocation algorithm, tasks are randomly distributed to each layer.
2) *Vertical Algorithm:* This algorithm only adopts the global optimization algorithm of decision vector, and the three-layer architecture is vertical cooperation.
3) *Horizontal Algorithm:* This algorithm only utilizes the heuristic algorithm for optimization, which is not applicable to global optimization. Tasks are still distributed randomly, and the base station cooperates horizontally.
4) *Simulated Annealing-Continuous Object Tracking (SA-COT) Algorithm:* This algorithm adopts the decision vector algorithm for global optimization and also employs the heuristic edge layer cooperation algorithm.
5) This algorithm is the industrial harmful gas tracing algorithm studied in [30], marked as Literature [30].
6) This algorithm is the industrial harmful gas tracing algorithm studied in [31], denoted as Literature [31].

The evaluation indicators contain the average delay of task execution, the ratio of tasks completed within the specified time, i.e., the success rate of task execution, and the longest delay required to execute a single task within a reporting period.

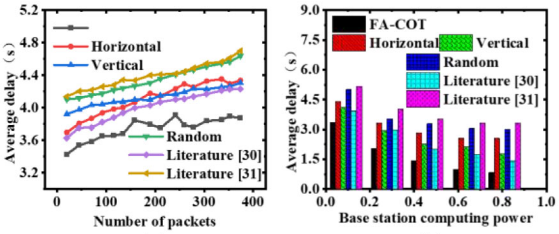
B. *Simulation and Evaluation of STM-COT*

Table I illustrates the parameter setting of the simulation experiment.

Compared STM-COT reported here with other target tracing algorithms proposed in related literature.

1) STM-COT reported here.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 6. Average delay of the six algorithms under different influence factors. (a) When the computing power is 0.2 GHz, the influence of different data number on the average delay. (b) When the number of tasks is 200, the influence of computing capabilities of different base stations on the average delay.

2) Yu *et al.* [32] proposed a target tracking system with multiplicative noise, and put forward an optimal filtering algorithm to solve the tracking system contaminated by the Gaussian white and color multiplicative noise. This system is marked as optimal filtering algorithm-continuous object tracking (OFA-COT).
3) Li *et al.* [33] built a joint trajectory smoothing and tracking framework, which modeled the target trajectory through the continuous function of time, denoted as denoted as Function of Time (FOT).
4) Jondhale and Deshpande [34] constructed a generalized regression neural network as an alternative to the traditional target localization and tracking method based on received signal strength indicator, labeled as Unscented Kalman Filter.
5) Li *et al.* [35] constructed a continuous tracking system based on radio frequency identification used for random moving targets in an unconstrained indoor environment, marked as RFID.
6) Yi *et al.* [36] proposed an effective solution for joint beam and power scheduling in distributed multitarget tracking (MTT) mesh radar system, marked as MTT.

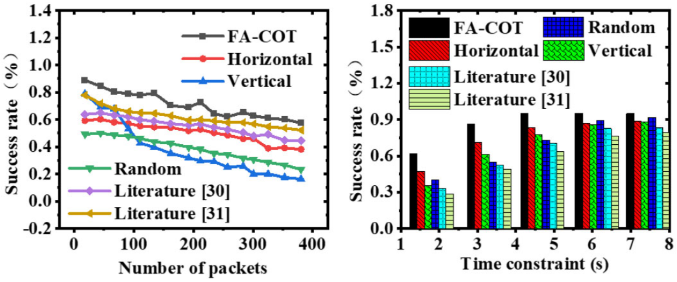
There are three evaluation indices as the performance evaluation criterion, including the sum of energy consumption of all nodes in a single time cycle, the sum of all communication information packets in a single cycle, and the tracking error, namely, the average distance between the awakened node in a single cycle and the boundary at the next reporting period.

V. Results and Discussion

A. Results of Hazardous Gas Tracing Algorithms Based on Parallel Optimization Framework

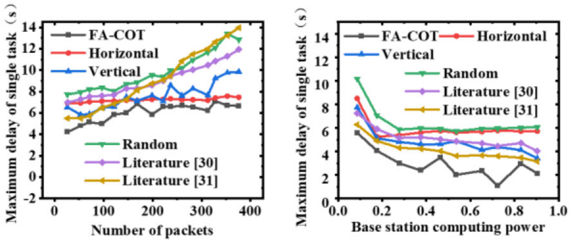
Fig. 6 displays the comparison of the average delay of task execution of six algorithms. The smaller the average delay, the better the effect of the algorithm on the utilization of computing resources.

Through Fig. 6(a), when there is a small number of tasks, the Horizontal algorithm can allocate tasks to the appropriate calculation unit. When there are numerous tasks, the calculation load of the edge layer increases, and the Horizontal algorithm shows more obvious advantages than the Vertical algorithm. SA-COT based on the parallel optimization algorithm combines the advantages of the two algorithms, so the average delay is the lowest. Fig. 6(b) indicates that under the premise of the same number of tasks, SA-COT based on the parallel optimization algorithm uses the Horizontal algorithm and Vertical algorithm, and the average delay is the lowest.



(a) (b)

Fig. 7. Influence of different number of particles on the fusion accuracy of different tool materials. (a) When the confinement time is 5 s, the task execution rate of the algorithm under different packet numbers. (b) Success rate of task execution under different confinement times when the task amount is 200.



(a) (b)

Fig. 8. Comparison of different resampling methods. (a) Influence of the number of packets in different algorithms on the maximum delay of a single task when the base station's computing capacity is 0.2 GHz. (b) Influence of base station computing capacity with different algorithms on the maximum delay when the number of network tasks is 200.

The number of packets has different effects on the evaluation index of success rate of task execution, which reflects the ability of different algorithms to deal with tasks. Fig. 7 signifies the comparison results of the above six algorithms.

From Fig. 7(a), SA-COT via a parallel optimization framework combining the advantages of the Horizontal algorithm and Vertical algorithm improves the computational ability of the edge layer through the Horizontal algorithm, and realizes the optimal deployment of tasks through the Vertical algorithm. Consequently, the success rate of task execution is the highest. In Fig. 7(b), when the confinement time is equal to 5 s, the success rate of the parallel optimization algorithm for the execution of tasks is more than 0.9, which is higher than that of the other five algorithms.

The maximum delay of a single task assessment standard reflects the ability of different algorithms to deal with large amounts of data and large amounts of computation. Fig. 8 illustrates the comparison results.

According to Fig. 8(a), SA-COT via a parallel optimization algorithm has lower delay than the other algorithms, and the

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

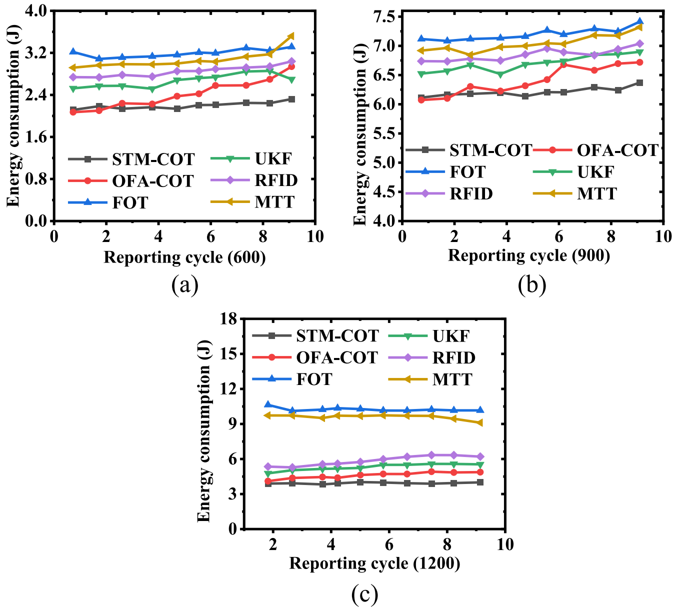


Fig. 9. Total energy consumption of different algorithms in a reporting cycle with different amounts of nodes deployed in the network. (a) Energy consumption comparison of different algorithms when the number of nodes is 600. (b) Energy consumption comparison of different algorithms when the number of nodes is 900. (c) Energy consumption comparison of different algorithms when the number of nodes is 1200.

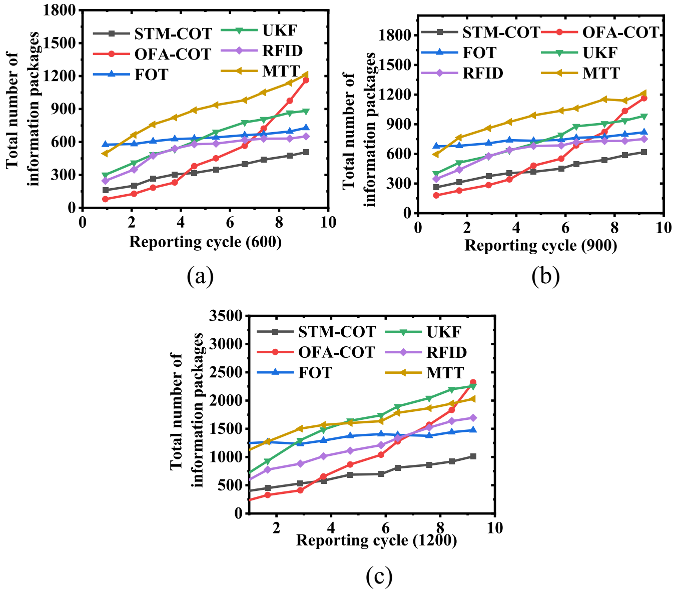


Fig. 10. Comparison of the change results of packet number of different algorithms in the reporting period with different number of nodes. (a) Comparison of packet numbers of different algorithms when the number of nodes is 600. (b) Comparison of packet numbers of different algorithms when the number of nodes is 900. (c) Comparison of packet numbers of different algorithms when the number of nodes is 1200.

maximum delay of task execution increases slowly with the increase of the number of packets. Fig. 8(b) shows that SA-COT via parallel optimization algorithm uses both Horizontal cooperation strategy and Vertical cooperation strategy, so the maximum execution delay of a single task is the lowest.

The above results suggest that because the task random allocation algorithm has no optimization strategy, the execution success rate of the task is the lowest. In contrast, the horizontal cooperation algorithm adopts the task distribution strategy in the edge layer to reduce the task delay, so its task execution success rate is higher than that of the random distribution algorithm. With few tasks, the vertical cooperation algorithm can realize the unloading strategy with a low delay of the horizontal cooperation algorithm through decision vector optimization. When there are many tasks, the success rate decreases due to the limitation of the computing power of the edge layer. The FA-COT algorithm proposed here improves the computing power of the edge layer through the horizontal cooperation algorithm and realizes the optimal deployment of tasks through the vertical cooperation algorithm, achieving the highest success rate.

B. Simulation Results of STM-COT for Industrial Hazardous Gases

Fig. 9 reveals the comparison of the total energy consumption of different algorithms in a reporting cycle with different amounts of nodes deployed in the network.

From Fig. 9, as the number of nodes increases, the total energy consumption of nodes in a single reporting cycle also increases. Although the overall energy consumption of STM-COT for industrial hazardous gases increases with the growth of the number of nodes, it always consumes the least energy with the same number of nodes among the six comparative algorithms. When the number of nodes is 600, the lowest energy consumption of STM-COT is only 2.11 J, the curve is gentle throughout the reporting period, and the change is small, which further explains the stability of the algorithm performance.

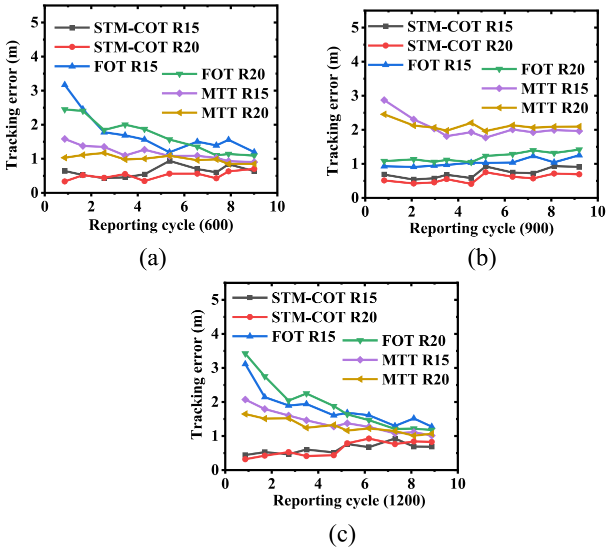
The number of packets reflects the busy state of communication links in the network. The more packets there are, the easier the network is blocked. Fig. 10 indicates the relationship between the number of information packets of different network nodes and the reporting period.

Fig. 10 shows that STM-COT avoids sending wake-up information to nodes far from the boundary by creating a wake-up region to. Besides, it further reduces the redundant wake-up information generated in the process of repeatedly waking up nodes in the later diffusion stage by tracking in clusters. Therefore, the packets increase stably in the network when the nodes increase.

The tracking error reflects the accuracy of the wake-up strategy in the algorithm. The larger the error, the farther the node to be awakened is from the boundary. Arousing points far away from the boundary is unnecessary and it may result in energy consumption. Fig. 11 displays the tracking errors of different algorithms when the communication radius R is 15 and 20, respectively.

Fig. 11 demonstrates that when the number of network nodes is $N = 600$, $N = 900$, and $N = 1200$, respectively, the tracking error of is the lowest, and the transformation trend is relatively stable. Meanwhile, the tracing effect of STM-COT is better when communication radius $R = 20$ than that when $R = 15$. The minimum tracking error is 0.31, which is 1.33 less than that of the other algorithms under the same

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.



(a)

(b)

(c)

Fig. 11. Comparison of tracking errors of different algorithms with various number of nodes under different communication radius. (a) Comparison of tracking errors of different algorithms when the number of nodes is 600. (b) Comparison of tracking errors of different algorithms when the number of nodes is 900. (c) Comparison of tracking errors of different algorithms when the number of nodes is 1200.

TABLE II
ABBREVIATION INDEX

Abbreviation	Full name
IoT	Internet of Things
IIoT	IIoT
QoS	Quality of Service
STM-COT	Two-stage Continuous Object Tracking Algorithm based on State Transition Model
MEMS	Micro-Electro-Mechanical System
SDN	Software Defined Network
SBS	Source base station
ID	Identity Document
GPS	Global Positioning System
cpt	cycles per task
SA-COT	Simulated annealing- Continuous Object Tracking
OFA-COT	Optimal Filtering Algorithm- Continuous Object Tracking
FOT	Function Of Time
UKF	Unscented Kalman Filter
RFID	Radio Frequency Identification
MTT	Multi-Target Tracking

condition. Obviously, STM-COT for industrial harmful gases has superior performance.

In conclusion, in the parameter setting, the first five and the last five reporting cycles belong to the pretracking and post tracking. Thus, the tracking strategies are different in these two stages so that the tracking error will increase slightly in the last five cycles compared with the pretracking. In the later stage of tracking, although some accuracy is sacrificed, repeated wake-up information to nodes is avoided, saving energy and reducing traffic. The full names of the abbreviations mentioned above are shown in Table II.

VI. CONCLUSION

With the expansion of the manufacturing scale, the IIoT has become the fastest-growing field of the IoT industry. DTs have proved particularly useful when integrating IoT systems in industrial manufacturing. Once the harmful gases are released into the industrial environment, they are incredibly harmful. According to the characteristics of the gas diffusion boundary, a two-stage gas boundary pursuit algorithm based on the DTs state-transition model is proposed, improving the accuracy of boundary location and saving energy consumption. However, some deficiencies need to be improved in the research. Some external factors affecting gas flow, such as ambient temperature and gas pressure, may increase the error of node perception data. In the future, it is considered to establish a gas diffusion model based on federated learning in each edge base station based on the existing three-tier architecture. The model in each edge base station will be continuously updated according to the perceived data of nodes to adapt to the impact of environmental changes on gas movement and track the boundary of continuous targets in an unsupervised manner. In addition, considering the existing gas diffusion model, the model's parameters will be modified using the sensing data of nodes to adapt to the changes of temperature, pressure, and surface roughness, constructing a continuous target tracking algorithm based on incremental learning.

REFERENCES

[1] M. M. Hassan, S. Huda, S. Sharmeen, J. Abawajy, and G. Fortino, "An adaptive trust boundary protection for IIoT networks using deep-learning feature-extraction-based semisupervised model," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2860–2870, Apr. 2021.

[2] C.-W. Shen, J.-T. Ho, P. T. M. Ly, and T.-C. Kuo, "Behavioural intentions of using virtual reality in learning: Perspectives of acceptance of information technology and learning style," *Virtual Reality*, vol. 23, no. 3, pp. 313–324, 2019.

[3] J. Wang, D. Li, and Y. Hu, "Fog nodes deployment based on space–time characteristics in smart factory," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3534–3543, May 2021.

[4] T. Xu, G. Han, X. Qi, J. Du, C. Lin, and L. Shu, "A hybrid machine learning model for demand prediction of edge-computing-based bike-sharing system using Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7345–7356, Aug. 2020.

[5] Y. Jiang, S. Yin, K. Li, H. Luo, and O. Kaynak, "Industrial applications of digital twins," *Philosoph. Trans. Royal Soc. A*, vol. 379, no. 2207, 2021, Art. no. 20200360.

[6] F. Niaz, M. Khalid, Z. Ullah, N. Aslam, M. Raza, and M. K. Priyan, "A bonded channel in cognitive wireless body area network based on IEEE 802.15.6 and Internet of Things," *Comput. Commun.*, vol. 150, pp. 131–143, Jan. 2020.

[7] H. Lu, X. He, M. Du, X. Ruan, Y. Sun, and K. Wang, "Edge QoE: Computation offloading with deep reinforcement learning for Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9255–9265, Oct. 2020.

[8] T. Wang, L. Qiu, A. K. Sangaiah, A. Liu, M. Z. A. Bhuiyan, and Y. Ma, "Edge-computing-based trustworthy data collection model in the Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4218–4227, May 2020.

[9] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 144–153.

[10] Z. Cai and X. Zheng. "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 766–775, Apr.–Jun. 2020.

[11] C. K. Wu, K. F. Tsang, Y. Liu, H. Zhu, H. Wang, and Y. Wei, "Critical Internet of Things: An interworking solution to improve service reliability," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 74–79, Jan. 2020.

[12] J. Shah and B. Mishra, "IoT-enabled low power environment monitoring system for prediction of PM2.5," *Pervasive Mobile Comput.*, vol. 67, Sep. 2020, Art. no. 101175.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.

[13] A. I. Adamu, M. K. Dasa, O. Bang, and C. Markos, "Multispecies continuous gas detection with supercontinuum laser at telecommunication wavelength," *IEEE Sensors J.*, vol. 20, no. 18, pp. 10591–10597, Sep. 2020.

[14] L. Wang, Y. Li, W. Yue, S. Gao, C. Zhang, and Z. Chen, "High-performance formaldehyde gas sensor based on Cu-doped Sn3 O4 hierarchical nanoflowers," *IEEE Sensors J.*, vol. 20, no. 13, pp. 6945–6953, Jul. 2020.

[15] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 968–979, May 2020.

[16] Z. Cai and T. Shi, "Distributed query processing in the edge assisted IoT data monitoring system," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12679–12693, Aug. 2021.

[17] Z. Lv and A. K. Singh, "Big data analysis of Internet of Things system," *ACM Trans. Internet Technol.*, vol. 21, no. 2, pp. 1–15, 2021.

[18] P. Vaishnav and A. Antra, "Continuous human activity classification with unscented Kalman filter tracking using FMCW radar," *IEEE Sens. Lett.*, vol. 4, no. 5, pp. 1–4, May 2020.

[19] W. Yi, Z. Fang, W. Li, R. Hoseinnezhad, and L. Kong, "Multi-frame track-before-detect algorithm for maneuvering target tracking," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4104–4118, Apr. 2020.

[20] S. Das, S. Roy, T. S. Bhattacharya, and C. K. Sarkar, "Efficient room temperature hydrogen gas sensor using ZnO nanoparticles-reduced graphene oxide nanohybrid," *IEEE Sensors J.*, vol. 21, no. 2, pp. 1264–1272, Jan. 2021.

[21] Z. Yi, W. Shang, T. Xu, S. Guo, and X. Wu, "Local discriminant subspace learning for gas sensor drift problem," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 1, pp. 247–259, Jan. 2022.

[22] A. Mohammadzadeh and R. H. Vafaie, "A deep learned fuzzy control for inertial sensing: Micro electro mechanical systems," *Appl. Soft Comput.*, vol. 109, Sep. 2021, Art. no. 107597.

[23] J. Wang *et al.*, "A fully integrated gas detection system with programmable heating voltage and digital output rate for gas sensor array," *IEEE Sensors J.*, vol. 21, no. 5, pp. 6821–6829, Mar. 2021.

[24] R. Cao *et al.*, "Multiplexable intrinsic Fabry–Perot interferometric fiber sensors for multipoint hydrogen gas monitoring," *Opt. Lett.*, vol. 45, no. 11, pp. 3163–3166, 2020.

[25] D. V. Medhane, A. K. Sangaiah, M. S. Hossain, G. Muhammad, and J. Wang, "Blockchain-enabled distributed security framework for next-generation IoT: An edge cloud and software-defined network-integrated approach," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6143–6149, Jul. 2020.

[26] M. M. Nasralla, I. García-Magariño, and J. Lloret, "Defenses against perception-layer attacks on iot smart furniture for impaired people," *IEEE Access*, vol. 8, pp. 119795–119805, 2020.

[27] C. Fu, J. Xu, F. Lin, F. Guo, T. Liu, and Z. Zhang, "Object saliency-aware dual regularized correlation filter for real-time aerial tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8940–8951, Dec. 2020.

[28] Y. Chen, S. Hu, H. Mao, W. Deng, and X. Gao, "Application of the best evacuation model of deep learning in the design of public structures," *Image Vis. Comput.*, vol. 102, Oct. 2020, Art. no. 103975.

[29] Y. Liu and C. Zhang, "Application of dueling DQN and DECGA for parameter estimation in variogram models," *IEEE Access*, vol. 8, pp. 38112–38122, 2020.

[30] W. Xia, M. Sun, and Q. Wang, "Direct target tracking by distributed Gaussian particle filtering for heterogeneous networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 1361–1373, Feb. 2020, [Online]. Available: https://ieeexplore.ieee.org/document/8982031, doi: 10.1109/TSP.2020.2971449.

[31] J. Yan, J. Dai, W. Pu, S. Zhou, H. Liu, and Z. Bao, "Quality of service constrained-resource allocation scheme for multiple target tracking in radar sensor network," *IEEE Syst. J.*, vol. 15, no. 1, pp. 771–779, Mar. 2021.

[32] X. Yu, G. Jin, and J. Li, "Target tracking algorithm for system with Gaussian/non-Gaussian multiplicative noise," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 90–100, Jan. 2020.

[33] T. Li, H. Chen, S. Sun, and J. M. Corchado, "Joint smoothing and tracking based on continuous-time target trajectory function fitting," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1476–1483, Jul. 2019.

[34] S. R. Jondhale and R. S. Deshpande, "Kalman filtering framework-based real time target tracking in wireless sensor networks using generalized regression neural networks," *IEEE Sensors J.*, vol. 19, no. 1, pp. 224–233, Jan. 2019.

[35] J. Li *et al.*, "PSOTrack: A RFID-based system for random moving objects tracking in unconstrained indoor environment," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4632–4641, Dec. 2018.

[36] W. Yi, Y. Yuan, R. Hoseinnezhad, and L. Kong "Resource scheduling for distributed multi-target tracking in netted colocated MIMO radar systems," *IEEE Trans. Signal Process.*, vol. 68, pp. 1602–1617, Feb. 2020, [Online]. Available: https://ieeexplore.ieee.org/document/9018085, doi: 10.1109/TSP.2020.2976587.

Zhihan Lv (Senior Member, IEEE) received the Ph.D. degree from Paris7 University, Paris, France, and the Ocean University of China, Qingdao, China, in 2012.

He has received more than 20 awards from China, Europe, and IEEE. He has received 1.5 million U.S. $ funding as PI. He has given more than 80 invited talks for universities and companies in Europe and China. He has given 15 keynote speech for international conferences. He has contributed 300 papers, including more than 50 papers on IEEE/ACM Transactions. He is the Editor-in-Chief of *Internet of Things and Cyber-Physical Systems* (KeAi), an Associate Editor of 18 journals, including IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEM, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, and *IEEE Technology Policy and Ethics Newsletter*, and the Leading Guest Editor for 40 special issues, including nine IEEE. He is the Co-Chair or TPC Member of 50 conferences, including ACM MM 2021 and ACM IUI 2015–2022.

Jingyi Wu (Student Member, IEEE) is currently pursuing the bachelor's degree in big data technology with the College of Computer Science and Technology, Qingdao University, Qingdao, China.

She has worked as an intern with the Institute of Automation, Chinese Academy of Sciences, Beijing, China. She has four soft copyrights, two published patents, and four papers.

Ms. Wu won the National Second Prize in the National Computer Design Competition as the Person in Charge, the First Prize in the "Hua Jiao Cup" National Mathematics Competition, and the Provincial Second Prize in the "Challenge Cup" Competition.

Yuxi Li received the bachelor's degree from Qingdao University, Qingdao, China, in 2020, where he is currently pursuing the master's degree in software engineering with the College of Computer Science and Technology.

He has extensive experience in software development and algorithm design. His main research direction is blockchain and Internet of Things.

Houbing Song (Senior Member, IEEE) received the M.S. degree in civil engineering from the University of Texas, Austin, TX, USA, in December 2006, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in August 2012.

In August 2017, he joined the Department of Electrical, Computer, Software, and Systems Engineering, Embry–Riddle Aeronautical University, Daytona Beach, FL, USA, where he is currently an Assistant Professor and the Director of the Security and Optimization for Networked Globe Laboratory. His research interests include cyber–physical systems, cybersecurity and privacy, Internet of Things, edge computing, AI/machine learning, and big data analytics.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on August 07,2023 at 12:49:36 UTC from IEEE Xplore. Restrictions apply.