A Cyber-Physical Information System for Smart Buildings with Collaborative Information Fusion

Qing Liu, and Lanlan Li*
[1] Chuzhou Polytechnic, Department of Information Engineering, Chuzhou 239000, China
[e-mail: liuqing@chzc.edu.cn, Lilanlan@chzc.edu.cn]
*Corresponding author: Lanlan Li

Received March 31, 2021; revised October 29, 2022; accepted May 12, 2022;
published May 31, 2022

Abstract

This article shows a set of physical information fusion IoT systems that we designed for smart buildings. Its essence is a computer system that combines physical quantities in buildings with quantitative analysis and control. In the part of the Internet of Things, its mechanism is controlled by a monitoring system based on sensor networks and computer-based algorithms. Based on the design idea of the agent, we have realized human-machine interaction (HMI) and machine-machine interaction (MMI). Among them, HMI is realized through human-machine interaction, while MMI is realized through embedded computing, sensors, controllers, and execution. Device and wireless communication network. This article mainly focuses on the function of wireless sensor networks and MMI in environmental monitoring. This function plays a fundamental role in building security, environmental control, HVAC, and other smart building control systems. The article not only discusses various network applications and their implementation based on agent design but also demonstrates our collaborative information fusion strategy. This strategy can provide a stable incentive method for the system through collaborative information fusion when the sensor system is unstable in the physical measurements, thereby preventing system jitter and unstable response caused by uncertain disturbances and environmental factors. This article also gives the results of the system test. The results show that through the CPS interaction of HMI and MMI, the intelligent building IoT system can achieve comprehensive monitoring, thereby providing support and expansion for advanced automation management.

Keywords: CPS(CPS), Internet of Things (IoT), Human-Machine Interaction (HMI), Machine-Machine Interaction (MMI), Collaborative Information Fusion, Smart building.

http://doi.org/10.3837/tiis.2022.05.006 **ISSN : 1976-7277**

1. Introduction

Smart buildings improve the level of urban intelligence of a smart city [1][2][3], enabling users the ability to interact with surrounding environmental facilities. With the ambient intelligence and ubiquitous networks, the building can interact with people, systems, and external factors surrounding them, which can be used for building management, including management systems such as ventilation, air conditioning, security, and energy.

In practice, the current construction of intelligent buildings mainly faces the following challenges. First of all, the various systems used in intelligent buildings are still closed and heterogenous systems [4][5], and each system is independent of each other and cannot be linked to each other to support intelligent decision-making. Secondly, each system is independent of each other, so maintenance is also independent, resulting in high and decentralized maintenance costs. Thirdly, there is a lack of sharing data between buildings and systems, so there are still information islands in management and control, which cannot support big data analysis [6][7]. Therefore, how to realize the interconnection and intercommunication between different systems and different devices and ensure the data security after the closed system becomes open, thus improving the integrity and reliability of cross-platform and cross-regional data, is an urgent problem for smart buildings.

Cyber Physics System (CPS) provides an integrated solution for management and control under the integration of cyber-physical information for the above problems. The concept of a CPS was proposed by the National Science Foundation of the United States in 2006 [11], and the origination of its thought principle can be traced back to the control and communication science in cybernetics and Human-Machine Interaction (HMI) [12][13][14][15]. In terms of engineering realization, CPS combines and builds on elements of different scientific theories and engineering disciplines, including, embedded systems [16], distributed computing [17], sensor networks [18], etc. With the development of science and technology such as the Internet of Things [19], embedded computing [20], wireless sensor networks [21], and ubiquitous computing [22], the development of CPS has also incorporated new content. These theories and disciplines support a better integration of cyber-physical components, which can improve the synergy between the subjects and techniques, further achieving significant optimization in terms of cost, performance, and sustainability throughout the life cycle of CPS functionalities.

This paper presents a CPS system with HMI-MMI physical information modeling under information fusion for smart building and its monitoring. In the article, the application scenarios, technical analysis, implementation plan, and the corresponding technical details of software and hardware are described and discussed in detail. In terms of the physical quantities to be monitored, we select the three most important typical physical quantities for buildings, namely temperature, humidity and smoke. Considering that in the process of physical quantity acquisition, the jitter caused by noise and sensor perturbation has a corresponding vicious influence on the instability of the system, we have introduced a strategy of collaborative information fusion in the system to increase the robustness the system and improve the smart building responsiveness of control management.This article also discusses the problems in the development of the CPS system and provides corresponding solutions in system design and implementation. After collecting these physical quantities, the system can gradually establish a physical information model of the building based on the accumulated data monitoring. With the HMI-MMI collaborative fusion, the solution in this paper with intelligent IoT information system support can be extended to more comprehensive autonomous CPS applications for smart building or agriculture applications.

2. Physical Scenario

2.1 Overview

The application scenario of the CPS is a smart building with physical sensing support. The task of the system can be abstracted as an environmental monitoring system for management. The environmental monitoring system of this design mainly detects the following environmental parameters in the smart building environment to achieve comprehensive monitoring of environmental information, thereby providing a reference for management decision-making.

As a virtual mapping of the real physical world, CPS needs to build a physical information model in real-time based on actual physical quantity collection to reflect the digitalized scene of the real-world. Therefore, the system needs to collect physical quantities to reflect the scenario in a targeted manner. We selected the three most typical physical quantities in building control as the main parameters of the system to establish a virtual data model of the physical information system.

2.2 Physical Measurements

2.2.1 Temperature

We regard temperature as an important parameter in the data model of the virtual physical information system, collected by temperature sensors in the physical section. The sensor can collect the temperature information of the area at the frequency set by the user, and send it to the coordinator node for processing, and then the coordinator sends the processing result data to the station through the serial port. At this time, the station can be set by the user before Good parameters and procedures control the air-conditioning system, to realize the control of the indoor temperature. These are follow-up controls and are beyond the scope of this paper. Each room in the buiding can put several such similar nodes, which can realize multi-point temperature information collection in the same room to improve the accuracy of temperature measurement.

2.2.2 Humidity

Humidity is one of the important parameters describing the physical information environment. Especially for building management, the humidity not only affects the human body but also affects the pipelines and facilities in the building. Humidity can be adjusted by controlling the ventilation system and air conditioning system, and the indoor humidity can also be adjusted by a humidifier. The control part and the sensing part in this system form a closed loop with each other. After the station is processed, the humidifier can be controlled to achieve quantitative control.

2.2.3 Smoke

Gas is the main source of the smoke. When gas leaks, it will threaten the lives of crew in the buildings. Therefore, monitoring the concentration of carbon monoxide gas is also an essential part. When the system detects that the smoke concentration is greater than the initial value set by the user, the station will immediately send an alarm signal to the alarm device or the mobile phone of a crew in the building or directly alarm. The station may automatically control the opening of the window while activating the alarm device. The effect of indoor air circulation ensures the life and property safety of the building.

3. CPS System Design

3.1 Analysis of Implementation

The ideal autonomous CPS is a system that can make decisions and operate independently, i.e., CPS can evaluate the situation according to its own perception of the surrounding environment and situation, and determine the corresponding processing strategy to implement it based on its own intelligence and knowledge. However, the development of cyber-physical systems is still mostly in semi-autonomous systems (e.g., example, surgical CPS, smart grid and industrial production application scenarios [23][24][25]), some of which are due to technical applications, but to a greater extent, the system’s autonomous cognition and coordination of the physical world are insufficient to appoint autonomous management of the entire system.

Therefore, in actual applications, these systems only run independently under predefined conditions (such as semi-automatic monitoring). For example, the administrator sets the operating instruction set after scheduling design, and then the real-time monitoring system will execute according to the set conditions, so there is no need to repeat manual operations. If the CPS has the operating experience and knowledge equivalent to the operator, the system may replace the operator's scheduling instructions and perform operations without being on duty. Based on the argument, the problem of system implementation is focused on how to enable the system to acquire these experiences and knowledge.

Considering the situation above, we introduce intelligent agents to distribute various complex calculations and functions among multiple agents. Each agent has certain intelligent computing capabilities. Therefore, the CPS system can be seen as a multi-agent system (MAS) from the perspective of logic operations [8][9][10]. We design and implement artificial intelligence modules, which are implanted in each agent through embedded computing. With the combination of sensor networks with embedded computing and interaction with agents to realize the closed-loop monitoring and control of the physical environment in the cyber-physical system, the system builds perception-computation-feedback loops, which enables this external stimulus to self-activate communication, calculation, and corresponding adjustment actions.

3.2 System Design

In order to achieve a high level of autonomy in the CPS system, one of the main implementations of the CPS in this article is a system that seamlessly integrates software and hardware components to perform clearly defined tasks, and achieves a high degree of intelligence. Therefore, we propose a closed-loop human-machine system or cyber-physical human system supported by human-machine collaborative learning, which combines human-machine interaction (HMI) with autonomous machine-machine interaction (MMI) management and control. Operators can interact with other elements of the system only when and when necessary. The cognitive subsystem of the system can learn real-time decision-making methods from the environment, humans, and itself, and then gradually form its own knowledge. The similar works can be seen as a collaborative CPS operations (e.g., [26][27]). Although human beings are still an indispensable part of the system’s decision-making process, in unattended situations, the CPS system can replace part of human work, especially in emergency situations, to respond in a short period of time, buying time for the follow-up process.

This CPS system for smart building monitoring and management is a Multi-Agent System (MAS) based on the Internet of Things technology. The nodes in this IoT system are divided into two categories. The node of the first type of Internet of Things is the machine-machine-interaction (MMI) node, which is defined as a data transceiver device that can sense the environment and perform wireless data communication. The system nodes are connected to various sensors and actuators with controllers and are self-organized in the network through a certain protocol. The second category is human-machine-interaction (HMI) nodes, which are mainly used for intelligent interaction under human-machine interfaces. In the implementation, we adopted the development and implementation of the mobile terminal based on the Android operating system. Based on the above-mentioned MMI and HMI mutual operation, the main functions realized by the whole system simulation are: one is home environment detection (temperature, humidity, smoke, human body induction, etc.), and the other is home device control (curtains, doors, lights, etc.).

Taking into account the possible coverage blind areas of the network, we deploy the mobile network by means of multi-point monitoring. The amount of data collected and transmitted may be not large, but the data transmission requires high reliability and high security. Considering the deployment and cost factors, the equipment volume of the system needs to be as small as possible and does not require a large power module. All nodes are powered by batteries or powerlines.

3.3 System Features

This system is mainly aimed at realizing various functions in the building environment. Taking into account the reliability of monitoring in the application scenarios and the effectiveness of long-term work, we also considered the design requirements of the system during the design process, including low power consumption, reliability, and scalability.

3.3.1 Low Power Consumption

Because it is a wireless sensor network with many nodes, it can only be powered by batteries. Therefore, low power consumption is required to extend the use and reduce the number of battery replacements.

3.3.2 Reliability

This system is used for the preliminary data collection of the home environment control system. If there is an error, it may cause an error in the judgment of the station and lead to error control (e.g., alarm while no carbon monoxide leaking occurs). Therefore, the Reliability of the system is required.

3.3.3 Scalability

According to the different needs of users, the node settings can be increased or decreased at any time. The software requires the modular design of the program, which can make the system upgrade convenient to modify one of the modules when adding nodes without changing the other parts; the program design should be simple and the data transmission format should be unified.

4. System Implementation

4.1 Architecture

The system in this article uses a wireless sensor network based on Zigbee technology to implement the environmental monitoring system. We designed a system consisting of multi-sensor nodes, coordinator nodes, and station computers according to the standards and characteristics of Zigbee technology. Among them, the sensor node exchanges information with the coordinator through wireless technology: the coordinator communicates through the serial port. The designed system structure is shown in **Fig. 1**.

Taking into account the specific application scenarios of CPS, the design and implementation of the system adopted two modes, including machine-machine interation (MMI) mode, which serves as an autonomous CPS system, and human-machine interaction (HMI) mode, which serves as a human-machine-closed-loop CPS system. The first mode is mainly used to handle tasks in unattended scenarios, such as monitoring, self-checking, self-healing, and autonomous energy-saving sleep, while the second mode is for some operations that require operators to participate in monitoring (e.g., forced sleep and forced sleep. Restart, manual trigger monitoring, online maintenance and command operation, etc.) A typical application scenario is to activate the HMI mode during the day and activate the MMI mode at night. Of course, the two modes can also be used together. For example, MMI is used to process the scene under normal working conditions, and the HMI is only responsible for monitoring. When adjusting the emergency treatment, the HMI will master the dispatching instruction system, and the MMI will execute the corresponding instructions and operations.

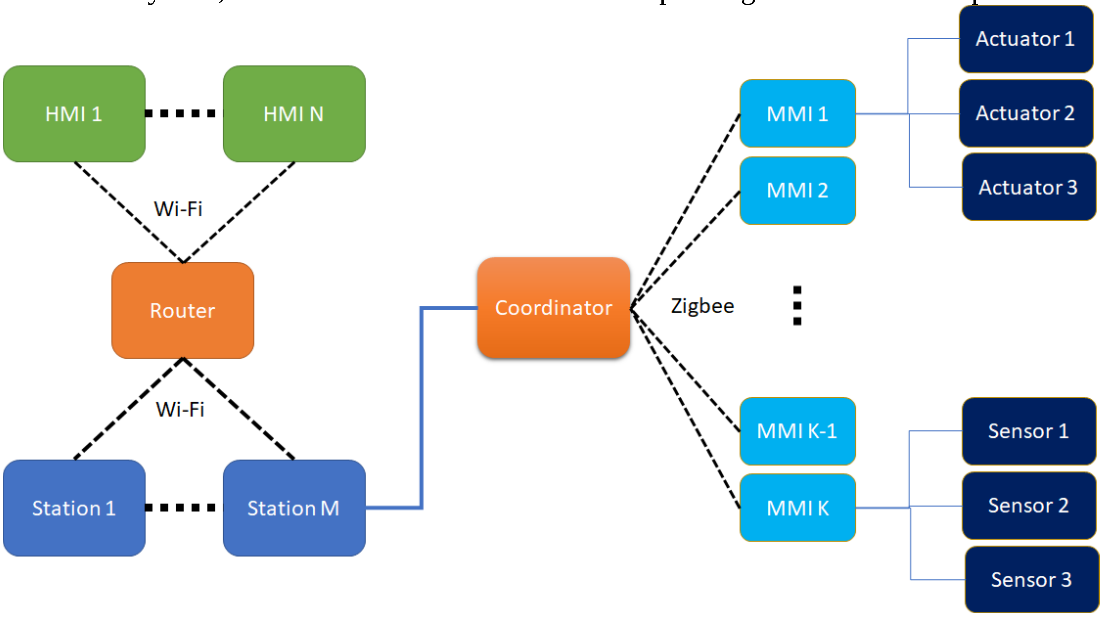
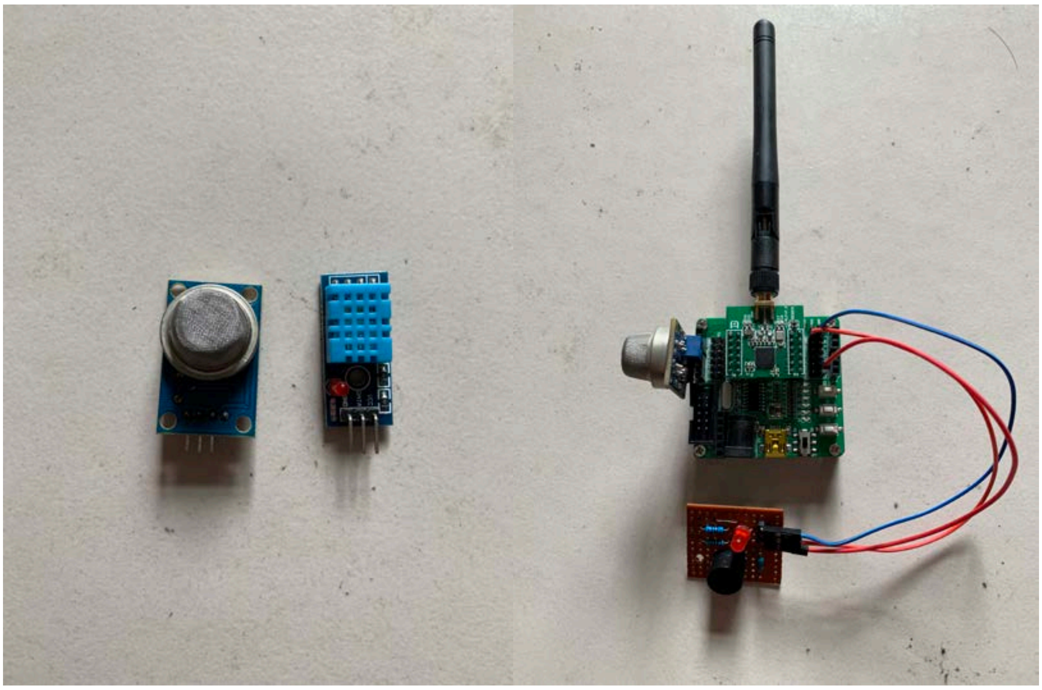


Fig. 1. System Architecture

The sensor node in this system is mainly responsible for the collection and transmission of environmental information, and the coordinator node is mainly responsible for network establishment, terminal node management, data processing, and data communication to the station. In practice, the sensor nodes can be increased or decreased according to the size of the building and living environment and the content to be monitored, and only a few changes are required. When the monitoring area is large, the method of adding sensor nodes can be used to ensure the connectivity of the network. On the contrary, when the area is small, the settings of router nodes can be reduced according to the situation to save system resources and reduce costs. What is noticeable, routers, coordinators, stations, HMI and MMI nodes in the system are agents to perform the computational data processing for smart building monitoring and operations.

4.2 Sensing Subsystem

For temperature and humidity sensing, we adopted a DHT11 digital temperature and humidity sensor (as shown in **Fig. 2(a)**). The sensor contains a temperature and humidity composite sensor with calibrated digital signal output. It uses special comb module acquisition technology and temperature and humidity sensor technology to ensure that the product has extremely high reliability and excellent long-term stability. The sensor includes a resistive humidity sensing element and an NTC temperature measuring element and is connected with a high-performance 8-bit microcontroller. The system uses a single-wire serial interface, which makes system integration simple and quick. Small size and low power consumption, the signal transmission distance can reach more than 20 meters.



(a) **(b)**

Fig. 2. Sensing Subsystems (a) DHT11 Kit and (b) MQ-2 Kit

The smoke sensor uses the MQ-2 gas sensor (as shown in **Fig. 2(b)**). It uses gas-sensitive material tin dioxide (SnO2), which has low conductivity in clean air. When there is combustible gas in the environment where the sensor is located, the conductivity of the sensor increases with the increase of the combustible gas concentration in the air. Using a simple circuit, the change in conductivity can be converted into an output signal corresponding to the gas concentration. Two voltages need to be applied to the sensor in the circuit system, which are heater voltage (VH) and test voltage (VC). VH is used to provide a specific working temperature for the sensor. VC is used to measure the voltage (VRL) on the load resistance (RL) connected in series with the sensor. This kind of sensor has a slight polarity, and VC requires a DC power supply. On the premise of meeting the electrical performance requirements of the sensor, VC and VH can share the same power source. In order to better utilize the performance of the sensor, it is necessary to adjust the appropriate RL value of the adjustable resistance Rp. When smoke is detected, the controller outputs an alarm.

4.3 Communication Subsystem

The system development is mainly based on the hardware of the home environment monitoring system based on CC2530 - Zigbee as the communication module used in the system. By combining circuit design and embedded software flow, the integration of monitoring and control of physical quantities of the building environment is realized. In terms of physical parameters, we used the monitoring of temperature, humidity, smoke, and other environmental parameters in the experiment to perform virtual data perception of the environment. The information volume of temperature and humidity is processed by the single-chip microcomputer and sent to the upper computer through the router. In terms of communication, the system adopts communication protocols such as Zigbee and Wi-Fi to transmit and control system parameters, which can communicate and monitor in real-time with the host computer.

In the realization of the communication part, we adopt the Zigbee protocol and realize the system-on-chip (SoC) solution based on 2.4-GHz IEEE802.15.4, Zigbee, and RF4CE applications based on the CC2530 module (as shown in **Fig. 3**). This module can build a powerful network node with a low-cost integration. The built-in RF transceiver and enhanced microcontroller 8051CPU endow the module subsystem with flash programming, 8-KB RAM, and many other powerful functions. **Fig. 3** shows the minimum system, mainly including reset and oscillation circuit system.

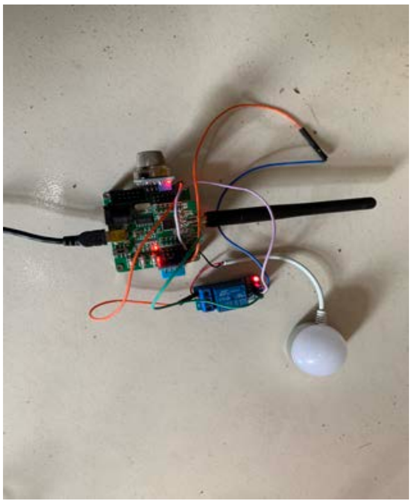


Fig. 3. Minimum System for Zigbee Communicaiton Module (CC2530 Zigbee)

4.4 Communication Protocol

According to the characteristics of the above scenes, in the system we designed, a cluster sensing design is adopted. Coordinate and realize communication among thousands of tiny sensors. These sensors use short-distance communication to achieve wireless network connection so that only a small amount of energy is required to transmit data from one sensor to another through radio waves in a relay manner. For this reason, we use the IEEE 802.15.4-based Zigbee protocol in the wireless sensor network protocol in the system to achieve low-consumption and high-error-tolerant short-range communication between wireless sensors.

Based on the protocol interaction between the underlying MAC layer and the physical layer of IEEE802.15.4, we combined the Zigbee network layer protocol and standardized API to design the corresponding data acquisition and transmission communication module embedded software system, responsible for the node to conduct mutual data exchange (Including MAC layer, routing, network layer, application layer, etc.). In the process of network communication, each Zigbee network data transmission module works like a base station, and they can communicate with each other within the entire network. The standard range of communication distance between each network node is set to 75 meters. At the same time, we also program the extended mode, which can extend the system to be able to communicate in several hundred meters or even several kilometers when needed.

We also considered the application scenarios of local positioning. Taking into account the situation of indoor deployment, the effect of GPS may be poor. Based on Zigbee, we have also developed an application for local area target positioning, but this part of content will not appear in this article, which will be included in another article of our paper.

5. Agent-Based Subsystem

5.1 Overview

The introduction of agents gives the system more flexible computing capabilities. The intelligent system in this system is mainly responsible for the three main functions of communication, information processing, and situation estimation. In terms of communication, it mainly receives and transmits data from the communication module and the sensor subsystem. In terms of information processing, the received information is encoded and sent, which mainly includes the information of the communication module itself and the information collected by the physical environment.

5.2 Agent States

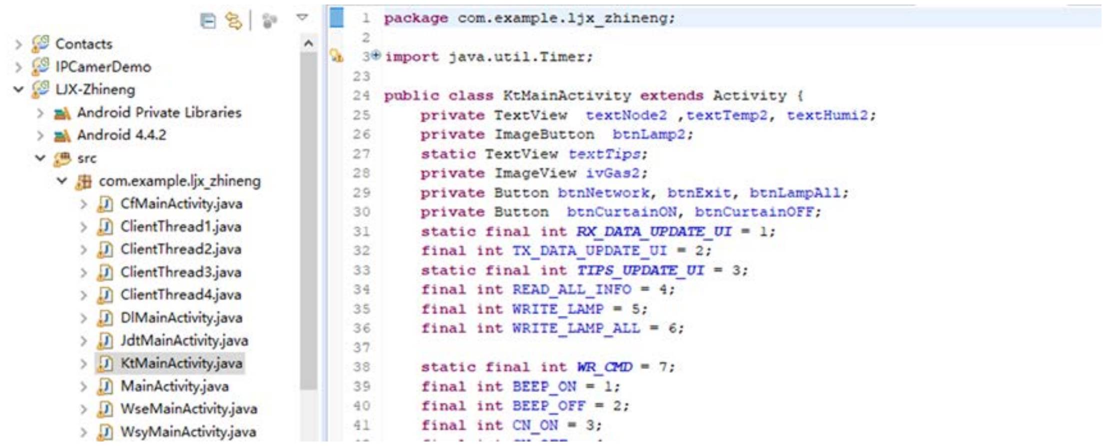
Each agent is implemented with a embedded system and it logically serves as a set of states during the computations. The states reflect the state of the situation as a part of the virtual model of the cyber-physical data logic for interactions. The sense of the cyber-physical model is built by these pieces of the elementary states. Each area is monitored by a group of agent nodes, and the situation is modeled according to the scatters of the data from the nodes.

The state of the nodes are modeled as $N_i=[V_i \mid D_i]$, where $V_{\mathrm{i}}=[v_0,v_1,v_2,v_3]=\{v_j\}_4$, and $D_{\mathrm{i}}=[d_0,d_1,d_2,d_3]=\{d_j\}_4$, where i refers to the ID number of the node (a circuit system in the physical world integrated with a sensor or acuator), and refers to the data source where $j = 0$ represents the data received as temperature physics, 1 for humidity, 2 for smoke detection, and 3 for light switch (as an acuator); V refers to the sign vector that represents the state of detection and D refers to the data vector that represent of the physics captured by the node.

Table 1. The States of the sign vector V

V_i=[v_0,v_1,v_2,v_3]	**Temperature (T)**	**Humidity (H)**	**Smoke (S)**	**Light (L)**	**State Explanation**
1110				off	light off
1111				on	All Good State (AGS)
1100			✓	off	deviant S, light off
1101			✓	on	deviant S, light on
1010		✓		off	deviant H, light off
1011		✓		on	deviant H, light on
1000		✓	✓	off	deviant H & S, light off
1001	✓	✓	✓	on	deviant H & S, light on
0110	✓			off	deviant T, light off
0111	✓			on	deviant T, light on
0100	✓		✓	off	deviant T & S, light off
0101	✓		✓	on	deviant T & S, light on
0010	✓	✓		off	deviant T & H, light off
0011	✓	✓		on	deviant T & H, light on
0000	✓	✓	✓	off	All Alarm State (AAS)
0001	✓	✓	✓	on	deviant T & H & S, light off

5.3 MMI and HMI Agent



```
package com.example.ljx_zhineng;

import java.util.Timer;

public class KtMainActivity extends Activity {
    private TextView  textNode2 ,textTemp2, textHumi2;
    private ImageButton  btnLamp2;
    static TextView textTips;
    private ImageView ivGas2;
    private Button btnNetwork, btnExit, btnLampAll;
    private Button  btnCurtainON, btnCurtainOFF;
    static final int RX_DATA_UPDATE_UI = 1;
    final int TX_DATA_UPDATE_UI = 2;
    static final int TIPS_UPDATE_UI = 3;
    final int READ_ALL_INFO = 4;
    final int WRITE_LAMP = 5;
    final int WRITE_LAMP_ALL = 6;

    static final int WR_CMD = 7;
    final int BEEP_ON = 1;
    final int BEEP_OFF = 2;
    final int CN_ON = 3;
```

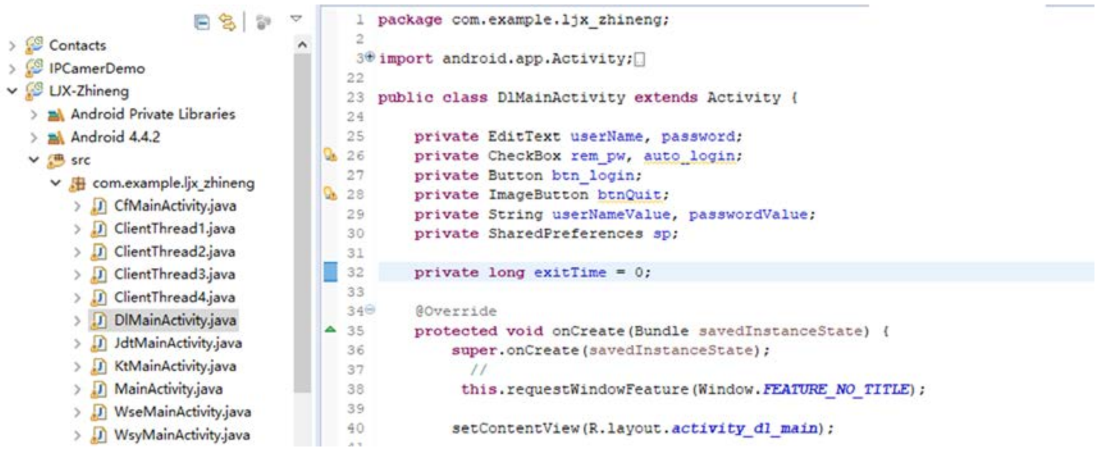
Fig. 4. An Embedded Software Module of Agents (KtMainActivity)

The data structure in the system of the node vector is modeled as *NodeData*[*i*][*j*] where i refers to the ID of the node, and *j* refers to the ID of sensors and/or actuators. When *NodeData*[*i*][*j*] is 1, the sign of specific detection is activated as true. For example, *NodeData*[*i=0*][*j=0*], *NodeData*[*i=0*][*j=1*], the first *i*=0 represents the temperature and humidity of the sensor board ID i = 0, and j=0 represents the temperature, and *j*=1 humidity. While, *NodeData*[*i=0*][*j=2*] == 0 means node ID=0 has detected smoke/gas, and the output 0 is alarming for high level to show that the gas is detected as is deviant. Another state is light on state. For example, *NodeData*[*i=0*][*j=3*] == 0 means sensor board ID *i*=0, and 3 means light control. *NodeData*==0 means the light is off and it will be on when the turn on condition is detected.

Based on the data interaction, MMI Agents realize the network connection with the module KtMainActivity (shown in **Fig. 4**), then control the corresponding equipment to trigger the actuators (switching equipment). KtMainActivity mainly realizes network connection first. Then the agent can control the corresponding device to perform tasks, such as switching all equipment, controlling the switch of the door, and so on.

There is another module WsyMainActivity interacts with the sensors and actuators. WsyMainActivity is mainly used to control initialization and coordinate the field data collection and information feedback of hardware nodes, including textNode1 for text information interaction, textTemp1 for temperature interaction, textHumi1 for humidity interaction, ivGas1 for gas interaction, and btnLamp1 for lighting Interaction: It is used for several main interactive functions of btnNetwork network communication.

HMI is designed similar with the MMI with same structure of sign and data vector. The main difference lies in that HMI has a portal interface for human interactions. **Fig. 5** and **Fig. 6** shows the the modules and main function of DlMainActivity is the navigation bar of the software. The main function of DlMainActivity is the interactive function and navigation of the HMI agent. It interacts with the MMI through the other two agent modules (WsyMainActivity, KtMainActivity) to complete the overall HMI-MMI interaction.



```
package com.example.ljx_zhineng;

import android.app.Activity;

public class DlMainActivity extends Activity {

    private EditText userName, password;
    private CheckBox rem_pw, auto_login;
    private Button btn_login;
    private ImageButton btnQuit;
    private String userNameValue, passwordValue;
    private SharedPreferences sp;

    private long exitTime = 0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
          //
         this.requestWindowFeature(Window.FEATURE_NO_TITLE);

        setContentView(R.layout.activity_dl_main);
```

Fig. 5. DlMainActivity Module of Agents

5.4 Wireless Communication of Agents

The agents communicate within a wireless network environment, and the routing protocol between agents uses the Open Shortest Path First (OSPF) protocol for routing communication. The Agent network can perform routing scheduling through the link-state, and all routers in each routing sub-area maintain the same link-state database (LSDB). Areas are divided into backbone areas (the backbone area must be numbered 0) and non-backbone areas (non-zero-numbered areas). If an Agent network running OSPF has only a single area, the area can be a backbone area or a non-backbone area. A backbone area is set up in a network with multiple areas, and all non-backbone areas are directly connected to the backbone area. The agent will use the maintained link-state database to calculate the routing table through the shortest path first algorithm (SPF algorithm), and then perform routine communication.

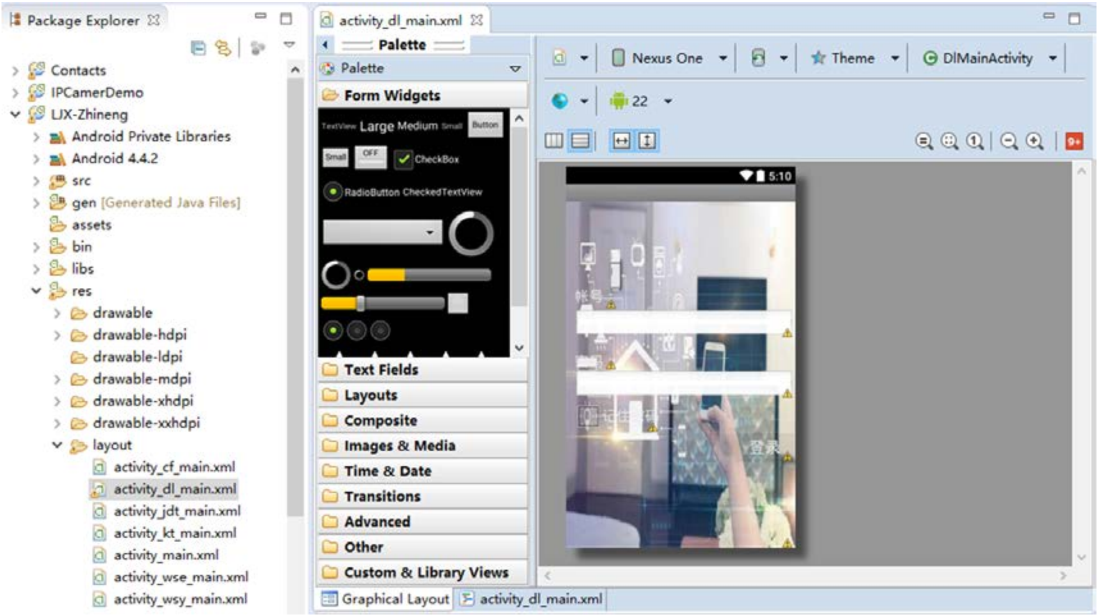


Fig. 6. Interface of the HMI Agent Portal

5.5 Station Agent

Station agents are the agents with powerful computational ability. They serve as the main backend of the processing and monitoring. Due to the necessity of interacting with operators, similar to HMI agents, station agents also provide user-interface for humans to check the situation directly and send the commend through the operation platform. The back-end system platform is connected with the front-end field agents through wireless communication of the Internet of Things. The data displayed is collected and transmitted by on-site MMI agents.

As shown in **Fig. 7**, the interface mainly include the six operations, including initialization, fault scanning, self-healing, monitoring, help and exit (left side of **Fig. 7**). Operators send the corresponding command through coordinator to communicate with on site MMI angents (shown in **Fig. 1**) to trigger the operations. The data captured on site by agents will be transmitted back via the coordinator as well and the on site situation will be built and visualized on the right hand side screen, thus form an informative close-loop for interactions.

```
================================================================
IoT Monitoring
================================================================
This version provides a Internet of Things solution with wireless
sensor network systems to a smart building monitoring system.
The version mainly demonstrates the functions of the monitoring
station, including the processing of initialization, scanning,
fault detection and self-healing, monitoring data,
and data analysis.
----------------------------------------------------------------
0. Initialization
1. Fault Scanning
2. Self-Healing
3. Monitoring
4. Readme and Help
5. Exit
================================================================
Enter number to operate or enter the Station By Y or Exit with N:
Please Enter:
```

Fig. 7. User Interface of the Station Backend (left: operation interface, right: visualization screen)

6. Agent-Based Collaborative Information Fusion

Taking into account the possible noise and local uncertainty in the acquisition process, a set of filtering strategies are introduced to reduce the interference. The agents in the network collect the measures of same physics but from different sensors. The strategies we use in the system main include Kalman filtering (KF [31]), median filtering [35], sliding average filtering [36][37], Butterworth filtering [38][39], Weiner filtering [40][41], and wavelet filtering [42][43]. Each strategy is performed by an agent. Therefore, each measurement has six types of filtering results after the processing of the agents. Then the system will fuse the six results into one as the final output, which performs the collaborative information fusion.

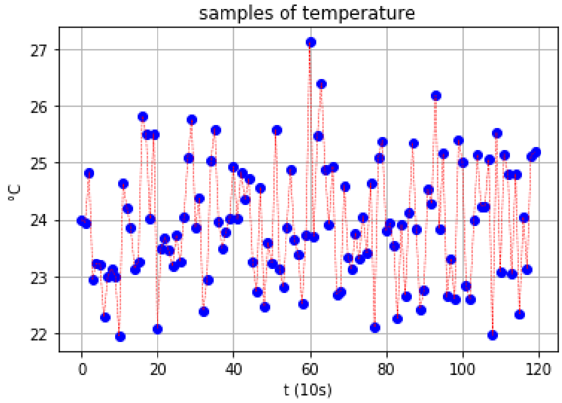


Fig. 8. Samples of Physical Measurements (temperature in celsius degree by minute)

Fig. 8 is a sample diagram of temperature physical quantity collection. The collected temperature is in degrees Celsius. The sensor collects the temperature every 10 seconds and records it and uploads it to the network. It can be seen that due to the limitation on the accuracy of the sensor and thermal noise, the collection results are not stable, and there are many jitters in the data. These jitters will bring troubles to the management and control of intelligent buildings. Unconventional responses may occur in the system under unstable excitation, which may cause oscillations in system response actions. This kind of oscillation will make the system unable to respond in time or make wrong judgments. In severe cases, the shaking will also cause the system's mechanical parts to malfunction or control to collapse.

For this reason, we apply 6 agents with different filtering strategies to estimate the data respectively. **Fig. 9** presents the results given by 6 agents. It can be seen that different agents have different estimation results. To this end, we further information fusion of 6 different results to obtain a result and record it as a system (as shown in **Fig. 10** with mean fusion strategy, i.e., calculating the average value of the 6 estimations). It can be seen that after the fusion filtering, the system obtains a stable result, and can respond with a smooth control accordingly.

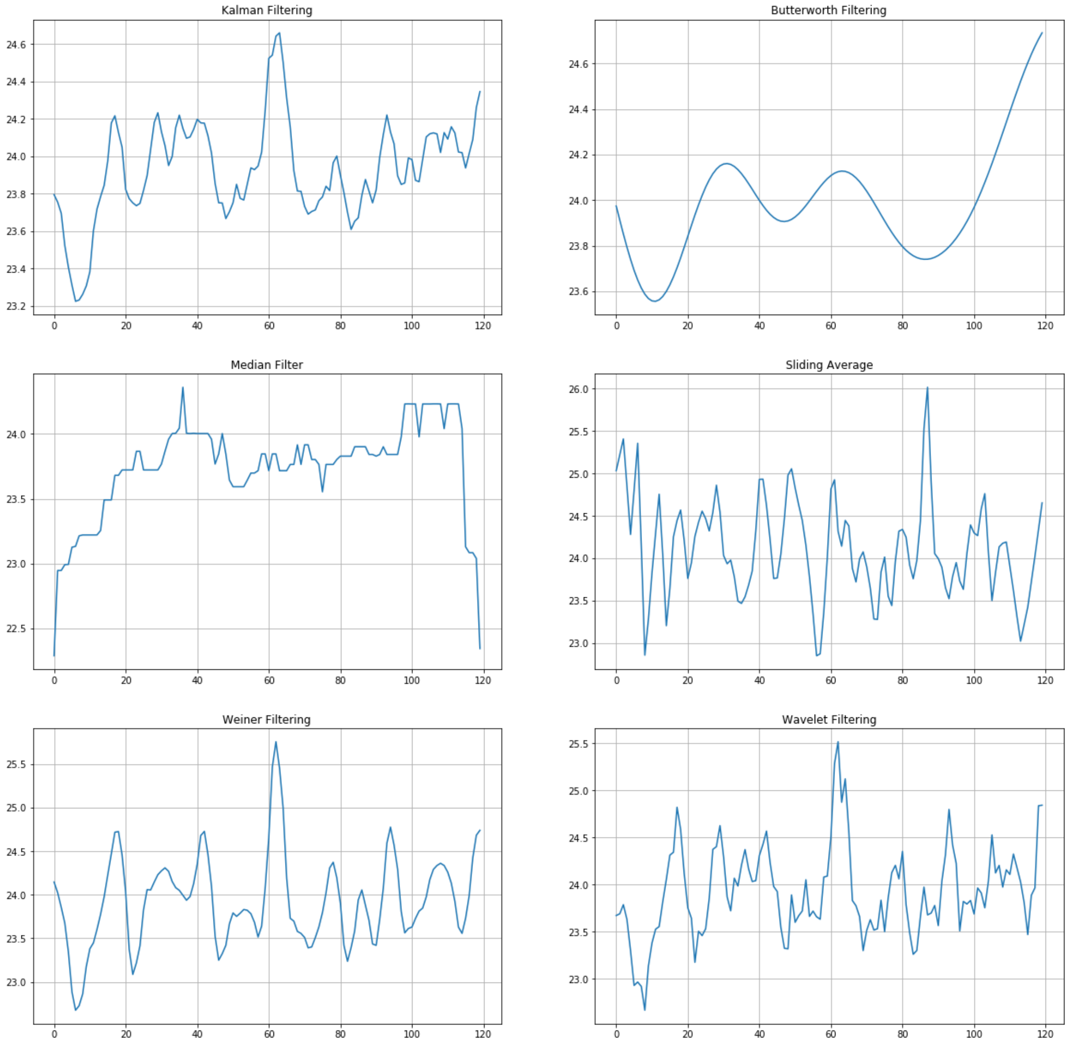


Fig. 9. Physic Estimation Based on the Samples in **Fig. 8** by Agents with Different Strategies

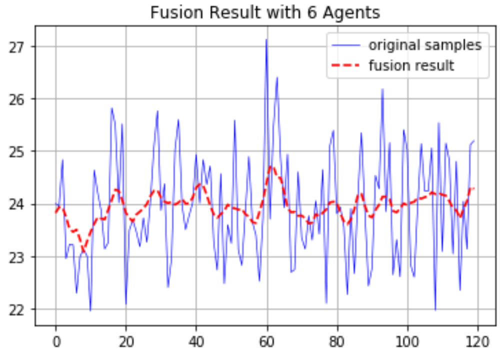


Fig. 10. Fusion Result (temperature in celsius degree by minute)

Similarly, the collaborative fusion has been applied to humid measurements and can be extended to quantitative physical quantity sensing applications (e.g., carbon dioxide concentration sensing and illuminance sensing). Because of the advantages of different filtering. the fusion can introduce a stabilized estimation by synthesizing various results.

It can be seen in the fusion result that after collaborative processing of the collected data through multi-agents, the control action of the system can not only better capture the time changes of the physical quantity on the spot, but also will not cause frequent switching of control actuators due to data jitter, resulting in actuators The instability and wear. Due to different strategies, each agent can play the role of a sub-expert system. With the cooperation of multiple sub-expert systems, the accuracy of the data reaches the optimal balance between accuracy and stability.

In this paper, we apply a mean fusion strategy to perform this fusion. However, more complicated fusion strategies can be applied to achieve the consensus. Due to page limitations, this part of the content will not be discussed in this article. Interested readers can refer to related methods (e.g., [28][29][30][31][32][33]).

7. Use Case

7.1 Building Monitoring

7.1.1 Scenario

The use case of the scenario of the CPS system is a multi-story building, each building has 8 main observation points (as shown in **Fig. 11**). The observed data are mainly environmental physical quantities, including temperature, humidity, and smoke. The monitoring front end of the CPS system is a series of MMI agents equipped with sensors, actuators, and communication modules. Each agent uses battery power. The system builds a CPS environment model by collecting the data from the front-end site and displays the visualized results through the user interface. The visualization results require that the status of each observation point can be fully displayed and the location of abnormal events can be reflected, and the time should be recorded. The presented results are dynamically displayed through graphics, and the corresponding results are recorded in the database.

The tested building scene is a four-story building. The monitored area is eight monitoring points in the layer. Eight agents were set up to monitor the environmental conditions on-site. The monitored environment will be measured by the three main physical quantities of temperature, humidity, and smoke, and returned to the station. The station will visualize the data and show the situation of the scene on the user interface for the operator to use as a reference for interaction with the scene.

7.1.2 Initialization

The system's initialization is carried out when the CPS information system is running for the first time. It is mainly used to set up the database, set up communication connections, and check and compatible with a series of operations such as existing data records. System initialization will have an important impact on the subsequent operation of the system, so the system initialization work will be performed automatically every time an abnormality or restart occurs.

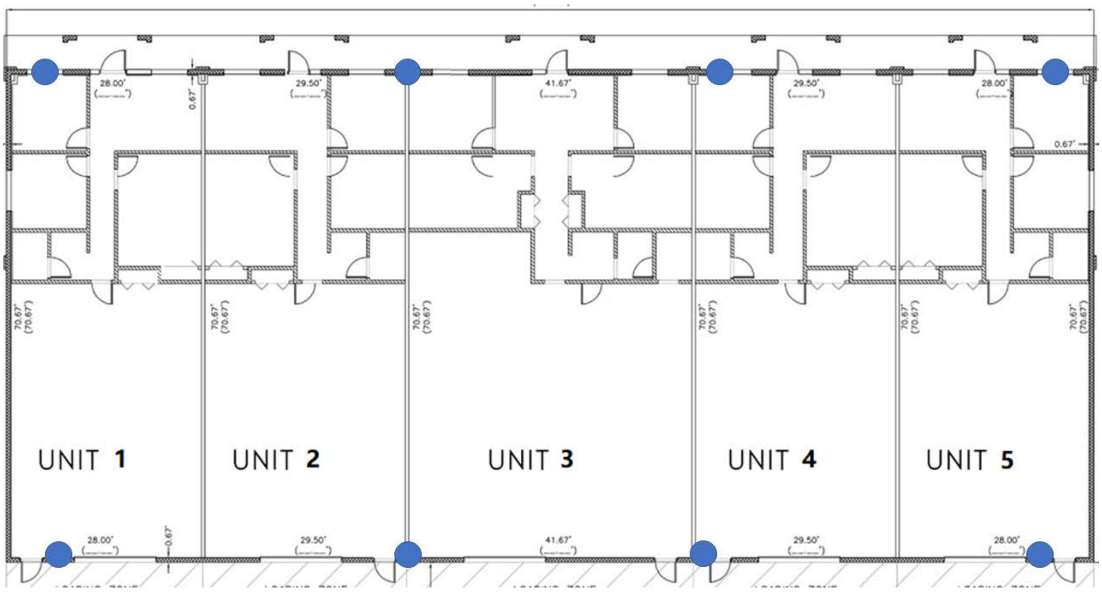
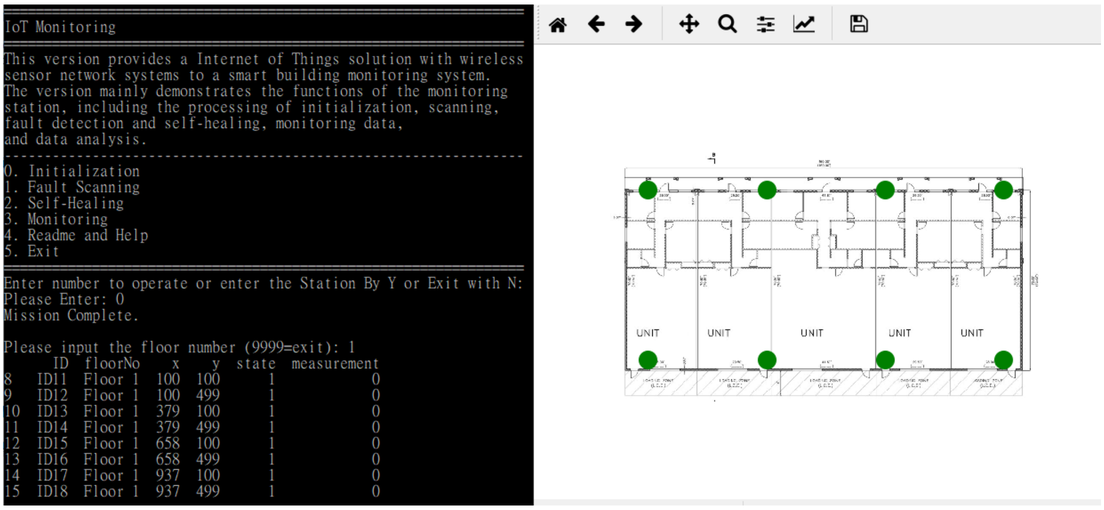


Fig. 11. Test Scenario, the indoor plat plot of each floor



```
IoT Monitoring

This version provides a Internet of Things solution with wireless
sensor network systems to a smart building monitoring system.
The version mainly demonstrates the functions of the monitoring
station, including the processing of initialization, scanning,
fault detection and self-healing, monitoring data,
and data analysis.
-----------------------------------------------------------------
0. Initialization
1. Fault Scanning
2. Self-Healing
3. Monitoring
4. Readme and Help
5. Exit

Enter number to operate or enter the Station By Y or Exit with N:
Please Enter: 0
Mission Complete.

Please input the floor number (9999=exit): 1
      ID  floorNo    x    y  state  measurement
8    ID11  Floor 1  100  100     1            0
9    ID12  Floor 1  100  499     1            0
10   ID13  Floor 1  379  100     1            0
11   ID14  Floor 1  379  499     1            0
12   ID15  Floor 1  658  100     1            0
13   ID16  Floor 1  658  499     1            0
14   ID17  Floor 1  937  100     1            0
15   ID18  Floor 1  937  499     1            0
```

Fig. 12. Demonstration of the Station Backend (*V* state AGS: alright working state)

Fig. 12 shows the state of each agent after initialization. It can be seen from the figure that there are a total of 8 nodes on this floor. The green dot indicates that the monitoring system in this area is working normally and no deviant states are found in this area.

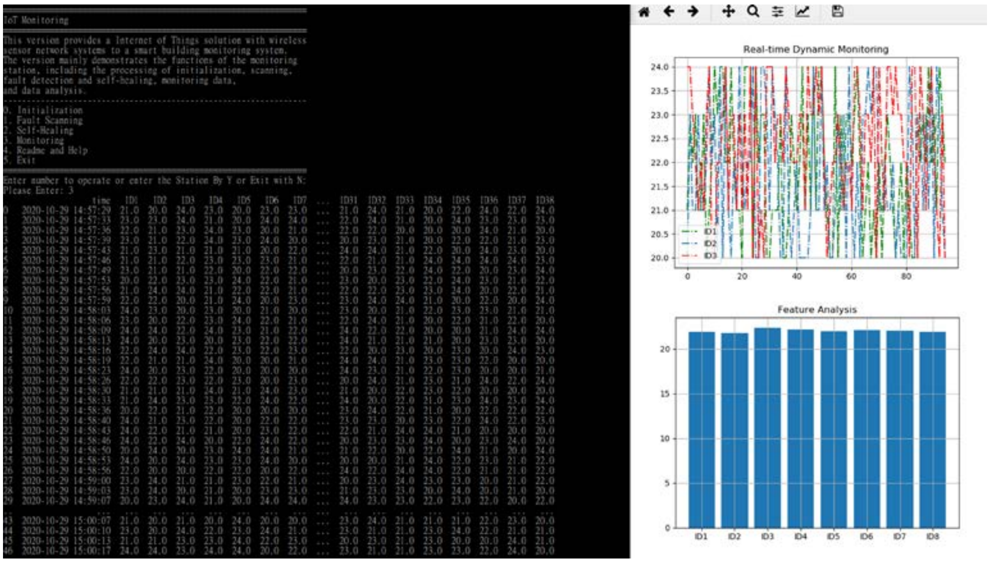
During the initialization process, we design a compatible process, so the initialization will check the previous records and save them. Afterward, the restored system will check the original system history and configuration, and perform data integration according to the corresponding time and address, so as to be compatible and integrated with the current system operation and records.

7.1.3 Data Monitoring

Data monitoring is one of the main functions of CPS. It starts the system monitoring function and accepts data from the front-end MMI agents. These data will be transmitted to the back end through the coordinator and recorded in the corresponding database. This part of the data reflects the D_i in $N_i[V_i \mid D_i]$ in the calculation logic of the agent.

As shown in **Fig. 13**, the display in the interface on the left is real-time on-site collection data, and the display on the right is a graphical visualization display. In the visualization interface on the right, the top is real-time monitoring live dynamic data, and the bottom is the feature displayed after feature analysis. In this test, the numerical expectation is used as a statistical feature to characterize the situation on the spot. When the application scene is clear, other features and customized features can also be introduced to visualize the pattern of the building.

What is noticeable that after the station receives the data, it will perform feature analysis and other processing, which will give the situation of the scene within the scope of quantification. In other words, the on-site situation can be characterized and quantitatively modeled through feature analysis, thereby providing a reference for situation assessment and emergency measures.

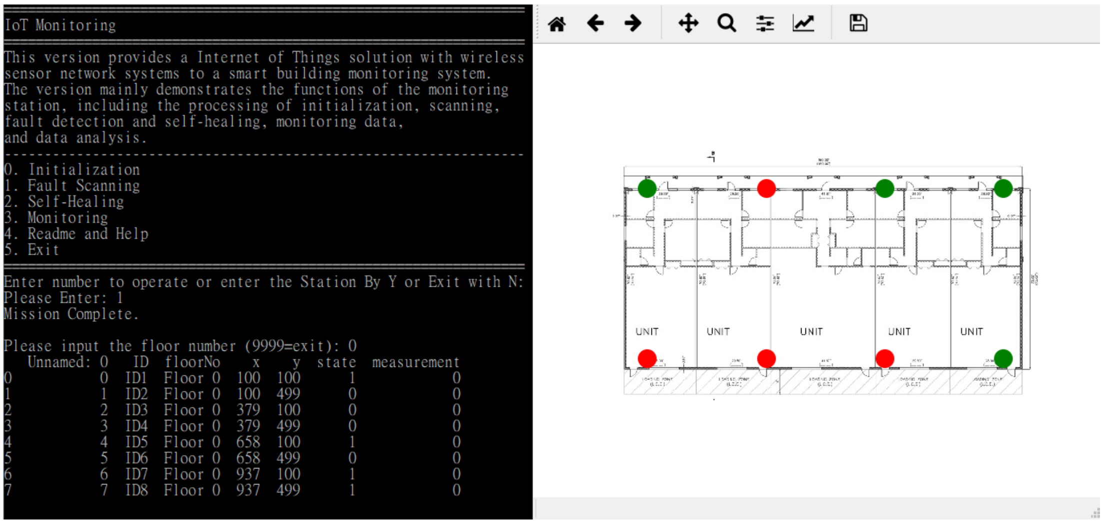


```
IoT Monitoring
This version provides a Internet of Things solution with wireless
sensor network systems to a smart building monitoring system.
The version mainly demonstrates the functions of the monitoring
station, including the processing of initialization, scanning,
fault detection and self-healing, monitoring data,
and data analysis.
-------------------------------------------------------------------
0. Initialization
1. Fault Scanning
2. Self-Healing
3. Monitoring
4. Readme and Help
5. Exit
Enter number to operate or enter the Station By Y or Exit with N:
Please Enter: 3
                   time   ID1   ID2   ID3   ID4   ID5   ID6   ID7  ...  ID31  ID32  ID33  ID34  ID35  ID36  ID37  ID38
0   2020-10-29 14:57:29  21.0  20.0  24.0  23.0  20.0  23.0  23.0  ...  21.0  24.0  21.0  20.0  22.0  24.0  22.0  24.0
1   2020-10-29 14:57:33  23.0  23.0  24.0  21.0  20.0  24.0  24.0  ...  22.0  24.0  21.0  20.0  24.0  23.0  23.0  23.0
2   2020-10-29 14:57:36  22.0  21.0  23.0  24.0  23.0  20.0  21.0  ...  22.0  22.0  20.0  20.0  20.0  24.0  21.0  20.0
3   2020-10-29 14:57:39  23.0  21.0  22.0  24.0  23.0  24.0  20.0  ...  20.0  23.0  21.0  20.0  22.0  22.0  21.0  23.0
4   2020-10-29 14:57:43  21.0  23.0  21.0  21.0  21.0  20.0  22.0  ...  24.0  24.0  21.0  22.0  20.0  24.0  23.0  20.0
5   2020-10-29 14:57:46  21.0  21.0  22.0  23.0  23.0  23.0  21.0  ...  22.0  21.0  21.0  24.0  24.0  24.0  24.0  23.0
6   2020-10-29 14:57:49  23.0  21.0  21.0  22.0  20.0  22.0  22.0  ...  20.0  23.0  22.0  24.0  22.0  20.0  23.0  24.0
7   2020-10-29 14:57:53  20.0  22.0  23.0  23.0  24.0  22.0  21.0  ...  23.0  20.0  23.0  22.0  24.0  23.0  21.0  22.0
8   2020-10-29 14:57:56  21.0  24.0  24.0  21.0  22.0  22.0  21.0  ...  22.0  22.0  23.0  23.0  24.0  20.0  22.0  21.0
9   2020-10-29 14:57:59  22.0  22.0  20.0  21.0  24.0  20.0  23.0  ...  23.0  24.0  24.0  21.0  20.0  22.0  20.0  24.0
10  2020-10-29 14:58:03  24.0  23.0  20.0  23.0  20.0  21.0  20.0  ...  23.0  20.0  21.0  22.0  23.0  23.0  21.0  21.0
11  2020-10-29 14:58:06  23.0  20.0  22.0  23.0  24.0  22.0  21.0  ...  22.0  24.0  21.0  20.0  22.0  21.0  22.0  20.0
12  2020-10-29 14:58:09  24.0  24.0  22.0  24.0  23.0  21.0  22.0  ...  24.0  22.0  22.0  20.0  20.0  21.0  24.0  24.0
13  2020-10-29 14:58:13  24.0  20.0  23.0  20.0  23.0  22.0  22.0  ...  24.0  21.0  21.0  21.0  20.0  23.0  23.0  20.0
14  2020-10-29 14:58:16  22.0  24.0  24.0  22.0  23.0  22.0  23.0  ...  22.0  20.0  23.0  20.0  23.0  20.0  24.0  23.0
15  2020-10-29 14:58:19  22.0  21.0  21.0  24.0  20.0  20.0  21.0  ...  24.0  24.0  21.0  23.0  23.0  22.0  20.0  20.0
16  2020-10-29 14:58:23  24.0  20.0  23.0  22.0  20.0  20.0  20.0  ...  24.0  23.0  21.0  22.0  23.0  20.0  20.0  21.0
17  2020-10-29 14:58:26  22.0  22.0  23.0  22.0  23.0  20.0  23.0  ...  20.0  24.0  21.0  23.0  21.0  24.0  22.0  24.0
18  2020-10-29 14:58:30  21.0  21.0  21.0  24.0  21.0  24.0  23.0  ...  21.0  20.0  22.0  23.0  22.0  20.0  20.0  20.0
19  2020-10-29 14:58:33  21.0  24.0  23.0  23.0  22.0  24.0  22.0  ...  24.0  20.0  22.0  21.0  23.0  24.0  23.0  24.0
20  2020-10-29 14:58:36  20.0  22.0  21.0  22.0  20.0  20.0  20.0  ...  23.0  24.0  22.0  21.0  20.0  22.0  22.0  22.0
21  2020-10-29 14:58:40  24.0  21.0  23.0  22.0  20.0  22.0  22.0  ...  23.0  23.0  20.0  23.0  22.0  24.0  22.0  23.0
22  2020-10-29 14:58:43  24.0  22.0  21.0  21.0  20.0  23.0  22.0  ...  22.0  21.0  24.0  24.0  21.0  20.0  20.0  20.0
23  2020-10-29 14:58:46  24.0  22.0  24.0  20.0  22.0  24.0  22.0  ...  20.0  23.0  23.0  24.0  20.0  23.0  24.0  20.0
24  2020-10-29 14:58:50  20.0  24.0  20.0  23.0  24.0  24.0  21.0  ...  21.0  22.0  20.0  22.0  24.0  21.0  20.0  24.0
25  2020-10-29 14:58:53  24.0  20.0  24.0  23.0  23.0  24.0  20.0  ...  20.0  20.0  21.0  24.0  22.0  23.0  21.0  22.0
26  2020-10-29 14:58:56  22.0  20.0  20.0  22.0  22.0  20.0  22.0  ...  24.0  22.0  24.0  24.0  22.0  21.0  21.0  22.0
27  2020-10-29 14:59:00  23.0  24.0  21.0  21.0  23.0  22.0  21.0  ...  20.0  23.0  24.0  23.0  23.0  20.0  21.0  22.0
28  2020-10-29 14:59:03  23.0  24.0  20.0  21.0  20.0  23.0  23.0  ...  21.0  23.0  23.0  20.0  24.0  20.0  21.0  20.0
29  2020-10-29 14:59:07  20.0  23.0  24.0  21.0  20.0  24.0  24.0  ...  24.0  23.0  23.0  22.0  23.0  22.0  20.0  22.0
..                  ...   ...   ...   ...   ...   ...   ...   ...  ...   ...   ...   ...   ...   ...   ...   ...   ...
43  2020-10-29 15:00:07  21.0  20.0  21.0  20.0  24.0  20.0  20.0  ...  23.0  24.0  21.0  21.0  21.0  22.0  23.0  20.0
44  2020-10-29 15:00:10  23.0  20.0  24.0  22.0  23.0  24.0  21.0  ...  23.0  21.0  21.0  23.0  24.0  22.0  21.0  21.0
45  2020-10-29 15:00:13  21.0  21.0  23.0  23.0  24.0  22.0  23.0  ...  20.0  23.0  21.0  23.0  20.0  20.0  24.0  21.0
46  2020-10-29 15:00:17  24.0  24.0  23.0  24.0  24.0  20.0  22.0  ...  23.0  21.0  21.0  23.0  23.0  22.0  24.0  20.0
```

Fig. 13. Demonstration the Station Backend (Data Monitoring for D_1-D_8)

7.1.4 Fault Detection

The failure defined in this CPS test is the failure time on site. In this test, we set smoke as the key physical quantity as the fault quantity for testing. The actual scene corresponding to the smoke fault is a fire event. By monitoring the physical quantity of smoke at the scene to warn of fire, so as to buy time for follow-up firefighting measures.



```
================================================================
IoT Monitoring
================================================================
This version provides a Internet of Things solution with wireless
sensor network systems to a smart building monitoring system.
The version mainly demonstrates the functions of the monitoring
station, including the processing of initialization, scanning,
fault detection and self-healing, monitoring data,
and data analysis.
-----------------------------------------------------------------
0. Initialization
1. Fault Scanning
2. Self-Healing
3. Monitoring
4. Readme and Help
5. Exit
================================================================
Enter number to operate or enter the Station By Y or Exit with N:
Please Enter: 1
Mission Complete.

Please input the floor number (9999=exit): 0
   Unnamed: 0   ID  floorNo    x    y  state  measurement
0           0  ID1  Floor 0  100  100      1            0
1           1  ID2  Floor 0  100  499      0            0
2           2  ID3  Floor 0  379  100      0            0
3           3  ID4  Floor 0  379  499      0            0
4           4  ID5  Floor 0  658  100      1            0
5           5  ID6  Floor 0  658  499      0            0
6           6  ID7  Floor 0  937  100      1            0
7           7  ID8  Floor 0  937  499      1            0
```

Fig. 14. Demonstration of the Station Backend (Agent ID 2, 3, 4 and 6 detect deviant *V* states)

In order to simulate such a scene, we selected Unit 2 for a smoke test. Tests are performed on the spot by releasing smoke. In the back-end monitoring station, as shown in the **Fig. 14**, the observation points 2, 3, 4, and 6 adjacent to Unit No. 2 all have red dots, indicating that the CPS system can not only detect the smoke triggers but also evaluate the fire based on the smoke. The situation of the situation is spreading, which provides situation assessment and prediction for evacuation and rescue.

7.2 Power Consumption

In order to further evaluate the effect of the system, we compared the data on electricity and water use after the system was used and before it was used. This data screen excludes the impact of the closedown from January to March and October to December caused by the epidemic. **Table 2** shows the power consumption of buildings B8495 to B8983 before the system is deployed, and **Table 3** shows the power consumption after the system is deployed.

In order to further show the efficiency, we have made statistics on the electricity consumption of each building, and the results are shown in **Fig. 15**. It can be seen from the statistics that after the deployment of the system, the power consumption of each building has been improved, in which B8495 has reduced power consumption by 30.03%, B6999 reduced by 8.35%, B4922 been reduced by 21.01%, and B8525 reduced by 25.05%, B8020 reduced by 31.59%, B8983 reduced by 4.12%, and B2745 is reduced by 29.47%.

Table 2. Power Stats of Buildings before Applying the System

Building No.	April	May	June	July	August	September
B8495	47175	49395	38085	37050	44430	45645
B6999	27147	41626	41135	43999	46953	33088
B4922	20297	29352	24451	21323	21908	21140
B8525	25854	31936	21762	17037	22830	24799
B8020	304500	457440	350520	94440	402120	363420
B8033	6251	23140	21496	17084	19443	10315
B8983	20187	21892	21428	23749	25312	20755
B2745	48101	67444	38720	5664	57557	69559

Table 3. Power Stats of Buildings after Applying the System

Building No.	April	May	June	July	August	September
B8495	20865	36705	36630	34755	5505	48720
B6999	17828	32409	31381	36568	64643	31594
B4922	8740	16084	17547	18436	28768	19799
B8525	9064	18377	17773	15422	26346	21106
B8020	68520	338400	288300	82920	303120	268080
B8033	3693	19915	17379	14913	39625	9594
B8983	18495	22036	19212	22405	23411	22266
B2745	8987	33950	30845	6988	61662	60010

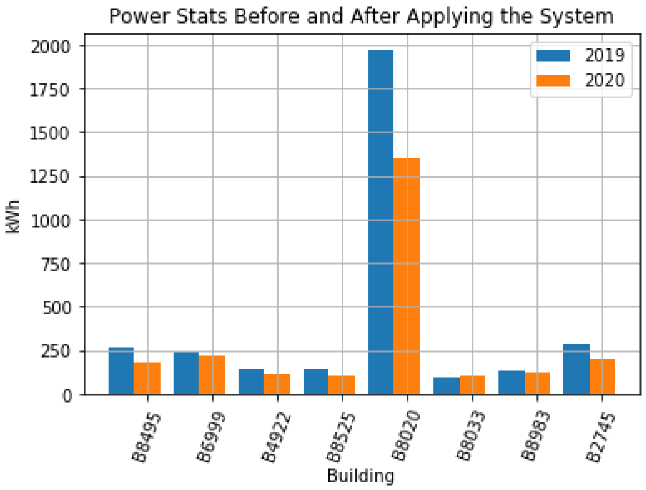


Fig. 15. Building-Wise Power Consumption Stats before (2019) and after (2020) Applying the System

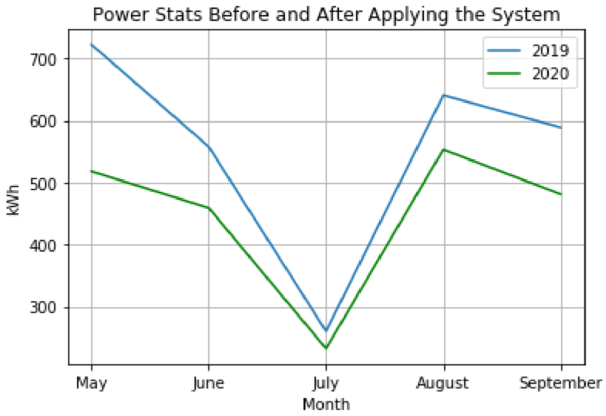


Fig. 16. Power Consumption Trend Stats before (2019) and after (2020) Applying the System

In terms of time, **Fig. 16** shows the trend of power consumption. Although the overall trend has the same appearance, the overall power consumption after using the system is lower than before. Although B8033 consumes 7.56% more energy than last year, the overall power is still lower than the previous average. From the statistical point of view of the same period, the total power consumption before system deployment is about 3270.97kWh, and the average power consumption is about 408.62kWh; the total power consumption after system deployment is 2401.81kWh, and the average power consumption is 299.97kWh.

It can be seen that after the deployment of the system, the overall power consumption has been saved by 26.57%, while the average power consumption has been saved by 26.59%.

8. Discussion

It can be seen from the actual test application case results, the system deployment makes the building supervision more intelligent. First of all, in terms of monitoring, due to the design of access to the Internet of Things, it reduces the workload of normal employees for equipment inspection and maintenance, which saves a lot of space and time in human resources, improves economic benefits, and also improve the work experience. The working status of each component in the floor is also connected to the network in real-time, and the overall status and data can be monitored. In this way, the actual on-site evaluation can be carried out on the two dimensions of data and data collection equipment, to obtain a more comprehensive on-site status.

Secondly, concerning the indicators of power-saving and emission reduction, it can be seen that after the system is applied, the power consumption is greatly reduced than in the past, while the efficiency is greatly improved. Through the collection of information in the dimensions of light, temperature, and time in the floor, it is possible to comprehensively determine the status of lighting and heating, ventilation, and air conditioning in the building, to control the length and intensity of the lighting system and the air conditioning system at the necessary time. From the result, compared with simple timing control, the control after information fusion reduces energy consumption, which means that collaborative information fusion can provide a more effective trigger mechanism for control actions.

9. Conclusion

This article mainly presents a CPS information system for smart buildings and introduces in detail the architecture and composition of the system and the method of implementation. In order to improve the intelligence of the system, we have adopted an intelligent body for design and integration. In the design of the front-end sensing agent, we adopted the system integration idea of IoT to design the front-end sensing subsystem and combined the HMI and MMI agents to design the processing terminal of the overall system. Since the use of the agent is very flexible, the function of the CPS can be further upgraded through the upgrade of each agent to achieve an overall upgrade, thereby improving the scalability of the system, so that the system can pass partial intelligence without the need for overall reconstruction upgrade to gradually emerge the overall intelligent changes.

Acknowledgments

This work was supported by Major Natural Science Research Projects of Colleges and Universities in Anhui Province Grants (KJ2019ZD74) and School Leadership Project (ZD2019009).

References

[1] R. Apanaviciene, A. Vanagas, and P. A. Fokaides, "Smart Building Integration into a Smart City (SBISC): Development of a New Evaluation Framework," *Energies*, vol. 13, no. 9, p. 2190, 2020. Article (CrossRef Link)

[2] B. Dong, V. Prakash, F. Feng, O. N. Zheng, "A review of smart building sensing system for better indoor environment control," *Energy and Buildings*, vol. 199, pp. 29-46, 2019. Article (CrossRef Link)

[3] J. G. Bhatt, O. K. Jani, and C. B. Bhatt, "Automation Based Smart Environment Resource Management in Smart Building of Smart City," *Smart Environment for Smart Cities*, Singapore, Springer, pp. 93-107, 05 April 2019. Article (CrossRef Link)

[4] C. Iddianozie and P. Palmes, "Towards smart sustainable cities: Addressing semantic heterogeneity in Building Management Systems using discriminative models," *Sustainable Cities and Society*, vol. 62, pp. 102367, 2020. Article (CrossRef Link)

[5] J. Harvey, M. Poorrezaei, T. Woodall, G. Nica-Avram, G. Smith, T. Ajiboye, K. Kholodova, and K. Zhu, "The Smart Home: How Consumers Craft New Service Networks by Combining Heterogeneous Smart Domestic Products," *Journal of Service Research*, vol. 23, no. 4, pp. 504-526, 2020. Article (CrossRef Link)

[6] B. N. Silva, M. Khan, and K. Han, "Integration of Big Data analytics embedded smart city architecture with RESTful web of things for efficient service provision and energy management," *Future generation computer systems*, vol. 107, pp. 975-987, 2020. Article (CrossRef Link)

[7] H. Dibowski, and F. M. Gray, "Applying Knowledge Graphs as Integrated Semantic Information Model for the Computerized Engineering of Building Automation Systems," in *Proc. of European Semantic Web Conference*, Cham, Springer, pp. 616-631, 2020. Article (CrossRef Link)

[8] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2nd Edition, Hoboken, New Jersey, United States: John Wiley & Sons, Oct. 2012. Article (CrossRef Link)

[9] M. L. Carlos, C. Miguel, M. S. José, A. G. Luis, "A game engine to make games as multi-agent systems," *Advances in Engineering Software*, vol. 140, p.102732, 2020. Article (CrossRef Link)

[10] F. Castiglione, "Agent-Based Modeling and Simulation, Introduction to," *Complex Social and Behavioral Systems*, pp. 661-665, 2020. Article (CrossRef Link)

[11] E. A. Lee, "The past, present and future of cyber-physical systems: A focus on models," *Sensors*, vol. 15, no. 3, pp. 4837-4869, 2015. Article (CrossRef Link)

[12] G. V. Bonin, “Cybernetics or control and communication in the animal and the machine,” *Bulletin of Mathematical Biophysics*, vol. 11, pp. 145–147, 1949. Article (CrossRef Link)

[13] T. J. Smith and K. U. Smith, "The cybernetic basis of human behavior and performance," *Continuing the Conversation*, vol. 2, pp. 1, 1988. Article (CrossRef Link)

[14] T. J. Smith, R. A. Henning, and K. U. Smith, "Performance of hybrid automated systems—A social cybernetic analysis," *International journal of human factors in manufacturing*, vol. 5, no.1, pp. 29-51, 1995. Article (CrossRef Link)

[15] S. Kajita, K. Kaneko, F. Kaneiro, et al, "Cybernetic human hrp-4c: A humanoid robot with human-like proportions," *Robotics Research*, pp. 301-314, 2011. Article (CrossRef Link)

[16] E. A. Lee and S. A. Seshia, *Introduction to embedded systems: A cyber-physical systems approach*, Cambridge, Massachusetts, United States: MIT Press, 2016. Article (CrossRef Link)

[17] M. O. Stehr, M. Kim, C. Talcott, "Toward distributed declarative control of networked cyber-physical systems," in *Proc. of International Conference on Ubiquitous Intelligence and Computing*, Springer, Berlin, Heidelberg, pp. 397-413, Oct 26, 2010. Article (CrossRef Link)

[18] I. Kabashkin and J. Kundler, "Reliability of sensor nodes in wireless sensor networks of cyber physical systems," *Procedia Computer Science*, vol. 104, pp. 380-384, 2017. Article (CrossRef Link)

[19] B. Borja, A. Ramón, R. Tomás, Martína. Diego, "Cyber–physical systems: Extending pervasive sensing from control theory to the Internet of Things," *Pervasive and mobile computing*, vol. 40, pp. 156-184, 2017. Article (CrossRef Link)

[20] M. Wolf, *High-Performance Embedded Computing: Applications in Cyber-Physical Systems and Mobile Computing*, 2nd Edition, Amsterdam, Netherlands, Elsevier, April 2014. Article (CrossRef Link)

[21] F. J. Wu, Y. F. Kao, and Y. C. Tseng, "From wireless sensor networks towards cyber physical systems," *Pervasive and Mobile computing*, vol. 7, no.4, pp. 397-413, 2011. Article (CrossRef Link)

[22] C. L. Chen, J. Yan, N. Lu, Y. Y. Wang, X. Yang and X. P. Guan, "Ubiquitous monitoring for industrial cyber-physical systems over relay-assisted wireless sensor networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 3, pp. 352-362, 2015. Article (CrossRef Link)

[23] C. Tercero, H. Kodama, H. C. Shi, K. Ooe, S. Ikeda, T. Fukuda, F. Arai, M. Negoro, G. Kwon, Z. Najdovski, "Technical skills measurement based on a cyber-physical system for endovascular surgery simulation," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 9, no. 3, pp. e25-e33, 2013. Article (CrossRef Link)

[24] H. S Li, L. f. Lai, and H. V. Poor, "Multicast routing for decentralized control of cyber physical systems with an application in smart grid," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1097-1107, 2012. Article (CrossRef Link)

[25] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Computers in industry*, vol. 81, pp. 11-25, 2016. Article (CrossRef Link)

[26] H. Y. Liu, L. H. Wang, "Remote human–robot collaboration: A cyber–physical system application for hazard manufacturing environment," *Journal of Manufacturing Systems*, vol. 54 pp. 24-34, 2020. Article (CrossRef Link)

[27] P. Fantini, M. Pinzone and M. Taisch, "Placing the operator at the centre of Industry 4.0 design: Modelling and assessing human activities within cyber-physical systems," *Computers & Industrial Engineering*, vol. 139, pp. 105058, 2020. Article (CrossRef Link)

[28] B. Kogut, U. Zander, "What firms do? Coordination, identity, and learning," *Organ. Sci.*, vol. 7, no. 5, pp. 469-592, Sep.-Oct. 1996. Article (CrossRef Link)

[29] N. P. Nguyen, L. V. Ngo, T. Bucic, and N. D. Phong, “Cross-functional knowledge sharing, coordination and firm performance: The role of cross-functional competition,” *Ind. Market. Manag.*, vol. 71, pp. 123-134, May 2018. Article (CrossRef Link)

[30] T. Dingsøyr, N. B. Moe, and E. A. Seim, “Coordinating knowledge work in multiteam programs: findings from a large-scale agile development program,” *Proj. Manag. J.*, vol. 49, no. 6, pp. 64-77, Oct. 2018. Article (CrossRef Link)

[31] F. Smarandache, J. E. Ricardo, E. G. Caballero, M. Y. L. Vázquez, and N. B. Hernández, “Delphi method for evaluating scientific research proposals in a neutrosophic environment,” *Infinite Study*, vol. 34, pp. 204-213, 2020. Article (CrossRef Link)

[32] N. Dalkey, O. Helmer, "An experimental application of the Delphi method to the use of experts," *Management science*, vol. 9, no. 3, pp. 351-515, 1963. Article (CrossRef Link)

[33] A. Ishikawa, M. Amagasa, T. Shiga, G. Tomizawa, R. Tatsuta, and H. Mieno, "The max-min Delphi method and fuzzy Delphi method via fuzzy integration," *Fuzzy sets and systems*, vol. 55 no. 3, pp. 241-253, 1993. Article (CrossRef Link)

[34] D. Simon, “Kalman filtering,” *Embedded systems programming*, vol.14, no. 6, pp. 72-79, 2001. Article (CrossRef Link)

[35] G. Arce, M. McLoughlin, “Theoretical analysis of the max/median filter,” *IEEE transactions on acoustics, speech, and signal processing*, vol. 35, no. 1, pp. 60-69, 1987. Article (CrossRef Link)

[36] A. Levant, X.H Yu, "Sliding-mode-based differentiation and filtering," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 3061-3067, 2018. Article (CrossRef Link)

[37] A. R. Jose, R. Eduardo and C. E. Juan, "Detrending fluctuation analysis based on moving average filtering," *Physica A: statistical mechanics and its applications*, vol. 354, pp.199-219, 2005. Article (CrossRef Link)

[38] B. Ehard, B. Kaifler, N. Kaifler, and M. Rapp, “Evaluation of methods for gravity wave extraction from middle-atmospheric lidar temperature measurements,” *Atmospheric Measurement Techniques*, vol. 8, no. 11, pp. 4645-4655, 2015. Article (CrossRef Link)

[39] A. V. Gribok, M. J. Buller, R. W. Hoyt, and J. Reifman, “A real-time algorithm for predicting core temperature in humans,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 4, pp. 1039-1045, 2010. Article (CrossRef Link)

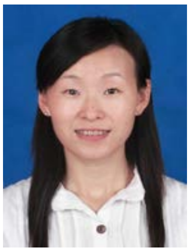
[40] F. Elsner, B. D. Wandelt, "Efficient Wiener filtering without preconditioning," *Astronomy & Astrophysics*, vol. 549, pp. A111, 2013. Article (CrossRef Link)

[41] S. Rainieri, F. Bozzoli, and G. Pagliarini, "Wiener filtering technique applied to thermographic data reduction intended for the estimation of plate fins performance," *Experimental Thermal and Fluid Science*, vol. 28, no. 2-3, pp. 179-183, 2004. Article (CrossRef Link)

[42] E. A. Steel, A. L. Ian, "Using wavelet analysis to detect changes in water temperature regimes at multiple scales: Effects of multi-purpose dams in the Willamette River basin," *River Research and Applications*, vol. 23, no.4, pp. 351-359, 2007. Article (CrossRef Link)

[43] S. Podtaev, M. Morozov, and P. Frick, "Wavelet-based correlations of skin temperature and blood flow oscillations," *Cardiovascular Engineering*, vol. 8, no. 3, pp. 185-189, 2008. Article (CrossRef Link)

Qing Liu received his M.S. degree in computer from the Anhui University，He is currently a visiting scholar of Nanjing University of Aeronautics and Astronautics.He is an associate professor in Chuzhou Polytechnic.His research interests include Wireless Sensor Networks and The Internet of Things..



Lanlan Li is an associate professor in Chuzhou Polytechnic, received a B.S. degree from Jilin University and a M.S. degree from Nanjing Normal University. Her research direction is intelligent control and Internet of Things applications.