

Retraction

Retracted: An IoT-Based Water Level Detection System Enabling Fuzzy Logic Control and Optical Fiber Sensor

Security and Communication Networks

Received 8 January 2024; Accepted 8 January 2024; Published 9 January 2024

Copyright © 2024 Security and Communication Networks. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

- [1] Y. Zheng, G. Dhiman, A. Sharma, A. Sharma, and M. A. Shah, "An IoT-Based Water Level Detection System Enabling Fuzzy Logic Control and Optical Fiber Sensor," *Security and Communication Networks*, vol. 2021, Article ID 4229013, 11 pages, 2021.

Research Article

An IoT-Based Water Level Detection System Enabling Fuzzy Logic Control and Optical Fiber Sensor

Yani Zheng ¹, Gaurav Dhiman ², Ashutosh Sharma ³, Amit Sharma,⁴
and Mohd Asif Shah ⁵

¹College of Mathematics and Information Science, Xianyang Normal University, Xianyang, Shaanxi 712000, China

²Government Bikram College of Commerce, Patiala, Punjab, India

³Institute of Computer Technology and Information Security, Southern Federal University, Rostov-on-Don, Russia

⁴Chitkara University, Chandigarh, India

⁵Bakhtar University, Kabul, Afghanistan

Correspondence should be addressed to Yani Zheng; yanizheng31@outlook.com and Mohd Asif Shah; ohaasif@bakhtar.edu.af

Received 8 June 2021; Revised 28 July 2021; Accepted 10 August 2021; Published 27 August 2021

Academic Editor: Qaisar Khan

Copyright © 2021 Yani Zheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The usage of wireless sensors has become widespread for the collection of data for various Internet of Things (IoT) products. Specific wireless sensors use optical fiber technology as transmission media and lightwave signals as carriers, showing the advantages of antielectromagnetic interference, high sensitivity, and strong reliability. Hence, their application in IoT systems becomes a research hotspot. In this article, multiple optical fiber sensors are constructed as an IoT detection system, and a Transmission Control Protocol (TCP)/Internet Protocol (IP) communication stack is used for the sensor module. Furthermore, design of gateway module, data server, and monitoring module is established in order to run the data server in the Windows system and communicate across the network segments. Furthermore, the optical fiber sensor is connected to the development board with WiFi, meanwhile considering the optical fiber wireless network's congestion problem. The fuzzy logic concept is introduced from the perspective of cache occupancy, and a fiber sensor's network congestion control algorithm is proposed. In the experiment, the IoT detection system with multiple optical fiber sensors is used for water level detection, and the sensor's real-time data detected by the User Interface (UI) are consistent with the feedback results. The proposed method is also compared with the SenTCP algorithm and the CODA algorithm, and it was observed that the proposed network congestion control algorithm based on the fuzzy logic can improve network throughput and reduce the network data packet loss.

1. Introduction

With the development of information technologies such as personal computer (PC), Internet, and mobile Internet in the current information technology era, a new information technology era represented by the Internet of Things (IoT) [1–3] comes. The IoT [4, 5] concept was proposed by the International Telecommunication Union (ITU) in 2005. This technology regards mobile devices such as mobile phones, pads, or smartwatches as networks to control them remotely in smart homes. A typical IoT-based system using the fuzzy approach is depicted in Figure 1.

High-performance sensors are required for signal sending and receiving to realize the interconnection of everything and the links among devices. The sensors need to have strong anti-interference and remote-control supporting abilities. An optical fiber sensor [6–8] is a high-performance sensor contributing to the network connection among devices. The optical fiber as its medium allows this sensor to form a new optical technology with lasers and semiconductor detectors. As society develops, optical fiber sensing technology with optical fiber as the transmission medium has been affirmed and welcomed in many fields [9–11]. The optical fiber sensor emits light from the light source into the optical fiber, and the optical signals are

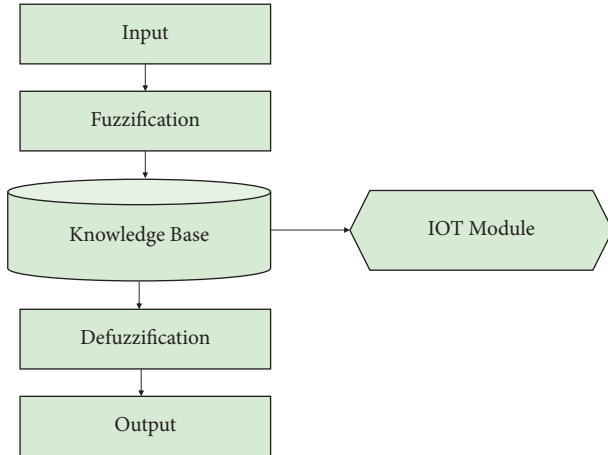


FIGURE 1: A typical IoT-based system using the fuzzy approach.

transmitted to the modulator in the optical fiber. The modulated lightwave and the parameter to be measured interact to change the signals' optical properties. The modulated lightwave signals are detected by a photoelectric detector and uploaded to the demodulation processing module, realizing remote detection.

Optical fiber sensors are categorized into functional optical fiber sensors (FFSs) and nonfunctional optical fiber sensors (NFFSs) according to whether the optical fiber is a sensitive element for signal sensing [12–14]. The FFSs' optical fiber can sense the measured signals and transmit optical signals [15–17]. The optical fibers in FFSs are continuous, and their characteristics make FFSs sensitive elements. With optical fibers as their transmission medium, NFFSs transmit signals in remote or harsh environments. NFFSs' output and incident fibers are discontinuous, and the modulators among them are unique sensitive elements. The more light is received, the more it is conducive to the terminal signal processing. More light coupled devices are required [18–20] because optical fibers are primarily used in optical signal transmission. Therefore, communication fibers or multimode fibers are appropriate for NFFSs.

There are various articles which utilize different wireless sensor-based methodologies along with fuzzy logics for mitigating the risk of detection in various scenarios. Ali et al. [21] presented an assessment model for decision making using the fuzzy logical approach to meet the software testing criteria for the cloud-based approach. The authors in [21] presented a wireless sensor-based methodology for false alarm warning reduction in case of smart home application system. Saeed et al. [22] proposed a multisensor-based mechanism for data fusion enabling artificial intelligence (AI) and wearable sensor network for controlling the home appliances working and positioning. A microcontroller based fuzzy logic technology is introduced by Chou et al. [23] for the detection of fire in case of automobiles. This technology utilizes the temperature, smoke, and flame sensors. Olivares-Mercado et al. [24] utilize an early detection methodology using the smoke sensor for the detection of fire events. Park et al. [25] used the multifunctionality of AI framework along with adaptive

fuzzy-based methodology to detect the early fire events. Further various other present techniques use the fuzzy-based methods for the navigation of robots [26], monitoring of healthcare [27], controlling the air conditions [28], identification of flood susceptibility [29], and many more. Recently, the success rate of IoT for smart systems has equipped its usage along with fuzzy logic for achieving better accuracy and high efficiency rates.

This article contributes in building a multiple optical fiber sensor-based IoT detection system for water level detection. It uses a Transmission Control Protocol (TCP)/Internet Protocol (IP) communication protocol stack for sensor module along with gateway module, data server, and monitoring module design for establishing the data server in the Windows system to communicate across the network segments [30, 31]. The data server end is connected to the Ethernet with the gateway program implementing protocol conversion, and the optical fiber sensor is connected to the development board with WiFi, considering the optical fiber wireless network's congestion problem. The fuzzy logic concept is introduced in this article from the perspective of cache occupancy and a fiber sensor's network congestion control algorithm is proposed. In the experiment, a monitoring terminal analyzes the optical fiber sensor's data collection, and a network congestion algorithm is introduced to test the algorithm's network performance in cache occupancy and cache changes.

The rest of this article is organized as follows: Section 2 presents the overall design of the IoT detection system followed by the design and implementation of the detection system in Section 3. Section 4 depicts the fuzzy logic control method for network congestion, and the test results analysis is presented in Section 5. Section 6 reveals the concluding remarks along with the future research directions.

2. Overall Design of the IoT Detection System

This section provides the basics of the fuzzy set along with the system architecture of the IoT detection system. The detection system's architecture is introduced, and then each module is elaborated, including the sensor, gateway, data server, and monitoring terminal. The sensor module includes a WeMos D1 circuit board and an optical fiber sensor.

2.1. Basics of Fuzzy Sets. The fuzzy sets are considered as the extension of the classical set, and this can be better interpreted using the varying degrees of membership function. The fuzzy logic builds a connection between the qualitative and quantitative modelling using input-output mapping. The fuzzy logic control system should have at least one component with fuzzy representation of knowledge. The fuzzy sets used to represent this information are the noncrisp sets with a specific membership function. Its representation is provided in the following equation:

$$\tilde{X} = \{(y, \mu_{\tilde{X}}(y)) | y \in U\}, \quad (1)$$

where the fuzzy set is represented by \tilde{X} indicating the information U which can be arranged as a fuzzy pair set with

$\mu_{\tilde{X}}(y)$ indicating the degree of membership of y which ranges from 0 to 1.

For the discrete information representation, the fuzzy set is indicated in equation (2), whereas the continued representation provides the fuzzy illustration in equation (3).

$$\tilde{X} = \left\{ \frac{(\mu_{\tilde{X}}(y_1))}{y_1} + \frac{(\mu_{\tilde{X}}(y_2))}{y_2} + \frac{(\mu_{\tilde{X}}(y_3))}{y_3} + \dots \right\}, \quad (2)$$

$$\tilde{X} = \left\{ \int \frac{(\mu_{\tilde{X}}(y))}{y} \right\}. \quad (3)$$

These basic assumptions are considered for the experimentation in this article.

2.2. System Architecture. The detection system includes the sensor, gateway, a data server, and a monitoring terminal. The sensor includes an optical fiber sensor and a WeMos D1 development board. Figure 2 shows the system architecture.

In Figure 2, a data server connects multiple monitoring terminals and gateways, a gateway connects multiple WeMos D1 development boards, and a WeMos D1 development board connects an optical fiber sensor. The monitoring terminal and data server run in the Windows system, and the gateway runs in the Debian-Raspbian system. The monitoring terminal, data server, and gateway are connected through the network for Socket communication. The WeMos D1 development board is connected to the network through WiFi and to the gateway through socket communication.

2.3. Other Modules in the Detection System. The sensor is composed of a WeMos D1 development board and optical fiber sensor. Figure 3 shows the multiple leads between the sensor and the development board. When the optical fiber sensor is energized, it can obtain the surrounding environment's real-time data based on its principles. The leads between the sensor and the development boards include a power line, a ground line, and multiple data lines.

The data server connects the monitoring terminal and the gateway. Assuming that the monitoring terminal and the gateway are in the same local network, the monitoring terminal is connected to the gateway through the gateway IP address and port number for communication. Otherwise, the monitoring terminal cannot access through the internal network address and port number where the gateway is located. The PC in the local network cannot communicate directly through the internal network IP address. Figure 4 shows that a data server can forward the messages for cross-network segment communication between the PC and the internal network IP address.

Figure 4 includes the monitoring module, the monitoring-terminal message processing module, and a gateway message processing module. The monitoring module monitors data transceiving of the gateway and the monitoring terminal. The data server filters the received messages and starts the corresponding message processing thread. The

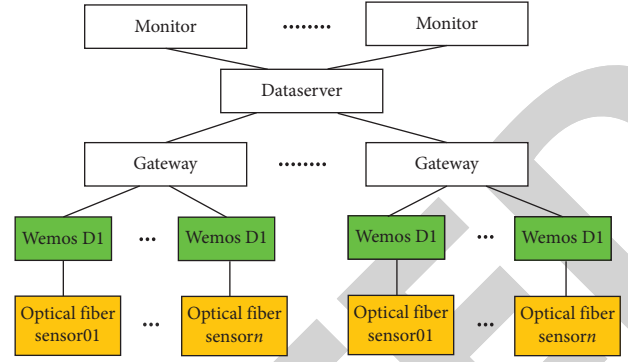


FIGURE 2: System architecture.

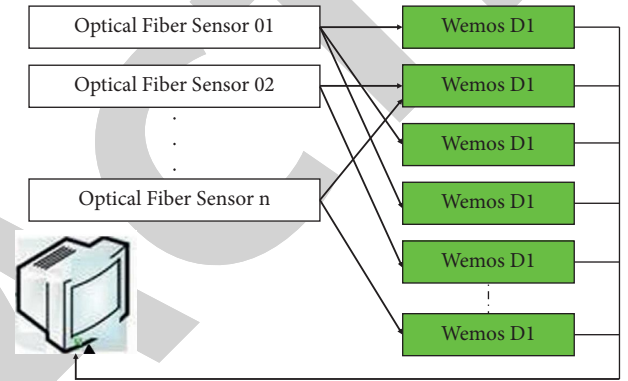


FIGURE 3: Sensor data paths.

monitoring message processing module receives messages sent by the monitoring terminals and sends corresponding processing commands. It takes out the messages from the message queue and analyzes and processes them. The gateway message processing module arranges the gateway's messages into message queues and starts the corresponding threads for analysis and processing.

3. Detailed Design and Implementation of the Detection System

The detection system designed herein includes the optical fiber sensor and the gateway. The sensor includes the WeMos development board and the optical fiber sensor module. The gateway includes the program's design and implementation.

3.1. Design and Implementation of the Optical Fiber Sensor.

The optical fiber sensor's design enables the WeMos D1 development board to collect data from the optical fiber sensor and transmit the acquired data to the gateway via WiFi. The hardware includes the sensor and WeMos D1 development board, and the software includes the Arduino program. Figure 5 shows the partial circuit of the WeMos D1 development board.

The development board includes six power pins, one analog input pin, and eleven digital input/output pins with the analog input pin number represented by A0 and the

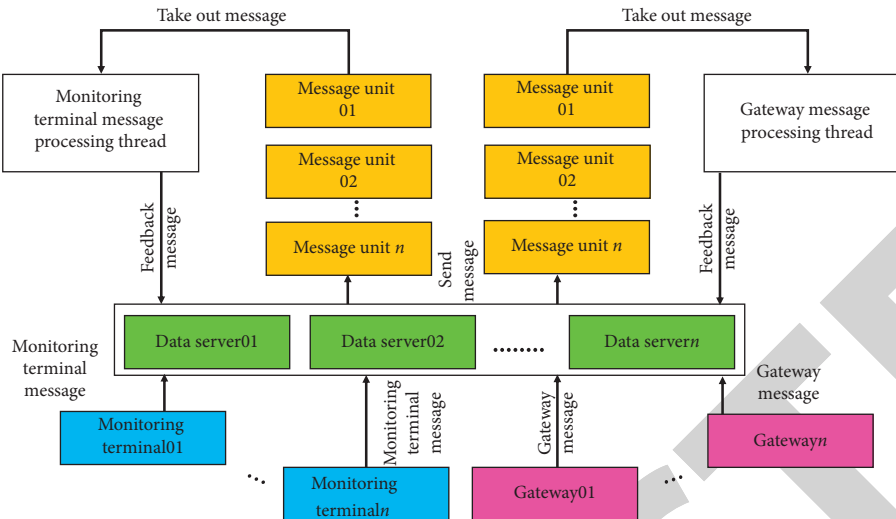


FIGURE 4: The data server's processing.

digital input and output number D0~D10. The development board's digital signal pin D0 represents the serial signal pin RX, and the digital signal pin D1 represents the serial signal pin TX. The development board's digital signal pin D2 can be connected to the LED to display the analysis results.

The fiber optic sensor module adopted herein includes laser, fiber coupler, optical circulator, fiber optic cable, and photodetector. The laser wavelength is 1550 nm, the working current is 40 mA, the output power is 50 mW, and the photodetector's wavelength range is 800 nm~1700 nm with a bandwidth of 3 dB. The optical fiber coupler can divide the optical signal from one optical fiber to multiple optical components for optical signal power distribution or combination. An optical fiber coupler of 1550 nm, 3 dB, and 1×2 is adopted. An optical circulator makes the optical signal transmitted in the ports' order. Figure 6 shows an optical circulator with 1550 nm, 0.6 dB insertion loss, and 46 dB optical isolation bandwidth. The optical fiber sensor module designed is used for water level measurement, adjusting the circuit length according to the water depth and the measurement's analog quantity.

In a software development environment of Arduino IDE v1.8.5, the WeMos D1 development board adopts the ESP8266EX chip. The native Arduino IDE does not support development on the ESP8266, so an open-source library that supports ESP8266 in the Arduino IDE is necessary. After the program is written, the development board's ESP8266EX chip can be connected to WiFi and the target host port by a socket for network communication.

A TCP connection between the ESP8266EX chip and the gateway is established on the development board to communicate between the client and the gateway. Figure 7 shows the design of the corresponding program development process based on the class and interface functions.

After the Arduino program is completed, the sensor and WeMos D1 development board are connected. When the development board is energized, the optical fiber sensor is energized by the power and ground pins and transmits data to the WeMos D1 development board, which runs in Flash

programs and sends the collected optical fiber signals to the gateway.

3.2. Design and Implementation of the Gateway Programs. The sensor has a unique ID in the data architecture herein, and sensor information is represented by a structure that includes a type field, an ID field, a sensor data field, a sensor connection status field, and a sensor communication sentence field. Different sensors have different IP addresses. A static IP address is configured on the router. After the gateway accepts the development board's connection, the development board can be distinguished based on the IP address.

With the gateway's connection to multiple optical fiber sensors, the socket is set in a nonblocking state, connecting to the sensor efficiently. During program development, with the local monitoring, the gateway cannot establish the monitoring if the corresponding network address is configured on the Raspberry Pi, so the monitoring address needs to be modified. The wired or the wireless network addresses' modification realizes the connection between the gateway and the optical fiber sensor.

The gateway program's parameters can be set in the form of IP address + port number; that is, after the target host runs the data server program, the gateway connects the data server. When the gateway sends messages to the data server continuously, the transmission efficiency is significantly reduced with the gateway's socket in the blocking mode. Therefore, the gateway's socket needs to be set in the nonblocking mode.

The data processed by the gateway come from the data server and the optical fiber sensor. The interactive messages of the gateway and the data server include the gateway information response messages, the polling feedback messages, and the sensor data feedback messages. In message processing, message types need to be distinguished. The message types include the gateway request messages sent by the monitoring terminal to the data server, the gateway

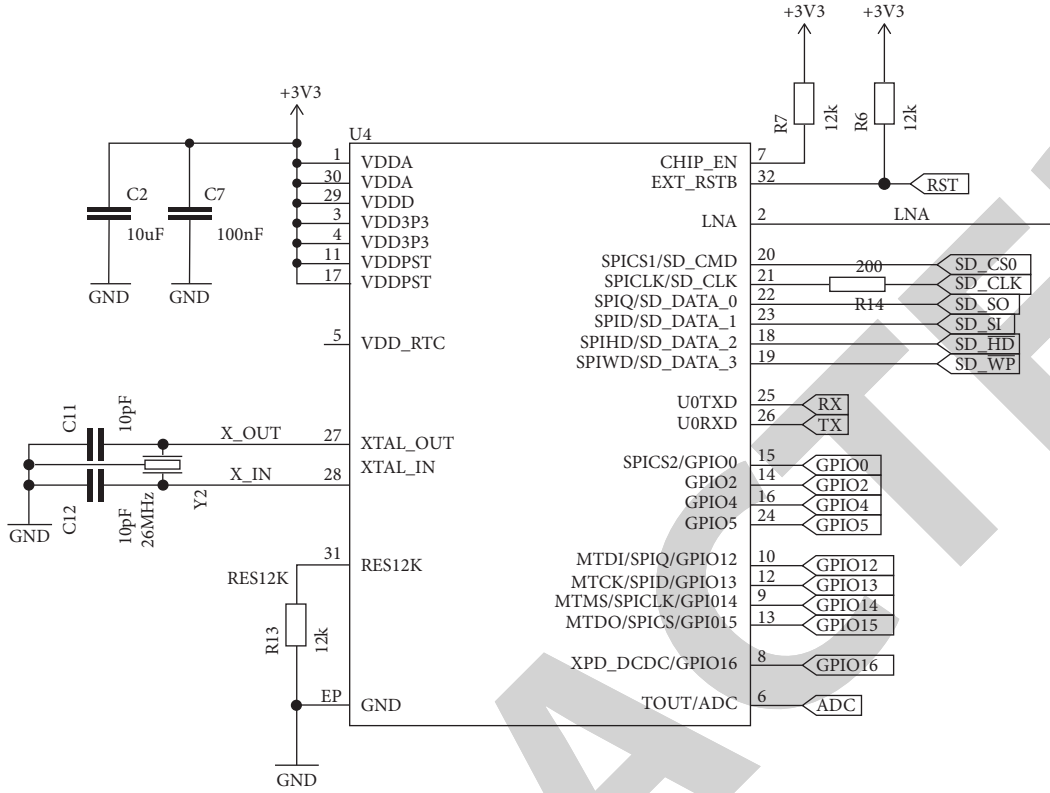


FIGURE 5: The partial circuit in the WeMos D1 development board.

response messages sent by the gateway to the data server, the sensor request messages sent by the monitoring terminal to the gateway, the sensor response messages sent by the gateway to the monitoring terminal, the polling messages sent by the gateway to the monitoring terminal, and the sensor data messages sent by the sensor to the gateway.

4. Fuzzy Logic Control Method for Network Congestion

When detecting network congestion, the fuzzy logic reasoning process [32, 33] is adopted. The network's cache occupancy state and cache usage state are analyzed in the IoT detection system based on fiber optic sensors. Cache usage state indicates the congestion variation trend and the degree. This parameter is defined as the ratio of the local data service time to the adjacent data packets' time interval. When cache occupancy is used for reasoning, the congestion usage state parameters [34, 35] are also used, and the network congestion is obtained by combining the two indicators.

Fuzzy logic is used for congestion detection in the IoT network with optical fiber sensors, and hop-by-hop realizes the traversal process. Figure 8 shows the specific algorithm implementation's three parts.

In Figure 9, when network congestion occurs at node B, the fuzzy logic-based congestion control algorithm is executed to increase the data transmission rate and reduce the possible signal access in the contention window, thus

reducing other nodes' possible signal occupation. Meanwhile, the nodes near B process the received data control frame.

The fuzzy logic process includes data fuzzing and data logical derivation. The data fuzzing adopts the Mamdani model, a dual-input and single-output model, which supports two variables as input, namely, cache occupancy rate and cache usage rate. The output parameter is the network congestion degree (FCD).

Fuzzy Inference System (FIS) calculates a value when the cache occupancy rate and cache queue changes are input to the FIS. The first input parameter involves the current buffer size and total buffer size, and the second local data's average packet sending time (T_{local}) and the average time interval between two adjacent data packets ($T_{adjacent}$). The following equation shows the second input parameter's calculation:

$$CT = \min \left\{ \frac{T_{local}}{T_{adjacent}} \right\}. \quad (4)$$

In equation (4), CT represents the cache queue's variation trend, and $CT < 6$. The corresponding parameters T_{local} and $T_{adjacent}$ are obtained by optical fiber sensors' numerical analysis in the network. When N data packets arrive, $T_{adjacent}$ is updated periodically. The following equation shows the parameter's calculation:

$$T_{adjacent} = 1 - W_{adjacent} * T_{adjacent} + W_{adjacent} * \frac{T_N}{N}. \quad (5)$$

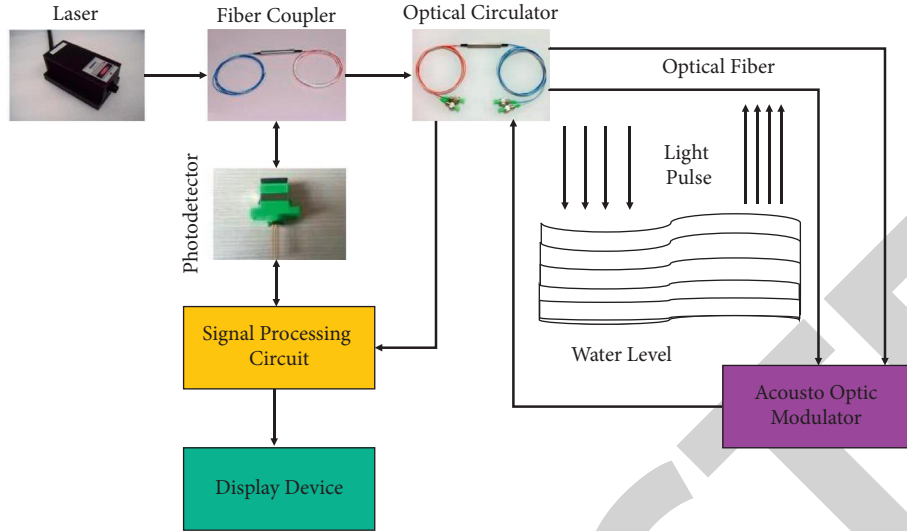


FIGURE 6: The architecture of the fiber optic sensor module.

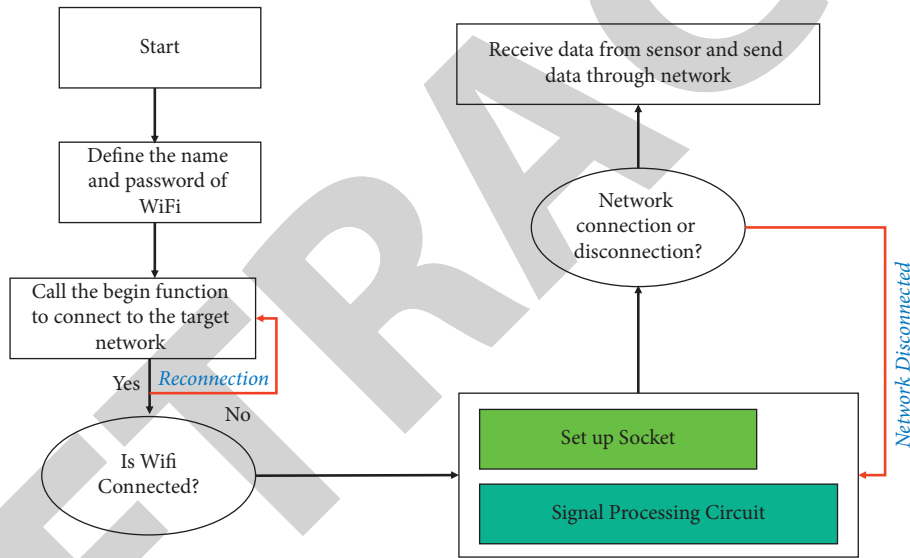


FIGURE 7: The development board's program development process.

In equation (5), W_{adjacent} represents a variable between 0 and 1, and T_N represents the transmitted data packet's serial number. A membership function (MF) is introduced to express the congestion status. The higher the cache occupancy rate, the greater the congestion degree among the nodes. Figure 10 shows the MFs of the cache occupancy rate and the cache queue's variation trend.

In Figure 10(a), {MS, S, K, H, MH} represent very small, small, medium, high, and very high cache occupancy, respectively. In Figure 10(b), {HL, ML, LL, VHL} represent high-low, medium-low, low-low, and very high-low cache usage trends, respectively. FCD's membership degree output is categorized into minimum, maximum, or average value according to membership rules. Equation (6) shows the congestion referring to the average congestion value in multiple possible conditions.

$$C(i) = \frac{\sum_{u=1}^e X_u \cdot h(X_u)}{\sum_{u=1}^e h(X_u)}. \quad (6)$$

In equation (6), X_u represents the occurrence probability of the u th congestion value with the value range in [0, 1]. The congestion detection mechanism comprehensively considers the cache occupancy rate and cache variation trends. Table 1 shows the congestion rules based on fuzzy logic.

5. Test Results and Analysis

An IoT detection system with multiple optical fiber sensors includes the sensor, gateway, data server, and monitoring terminal. The sensor includes the WeMos D1 development board and optical fiber sensor module. The PC adopts Arduino IDE v1.8.5 for program development. The sensor

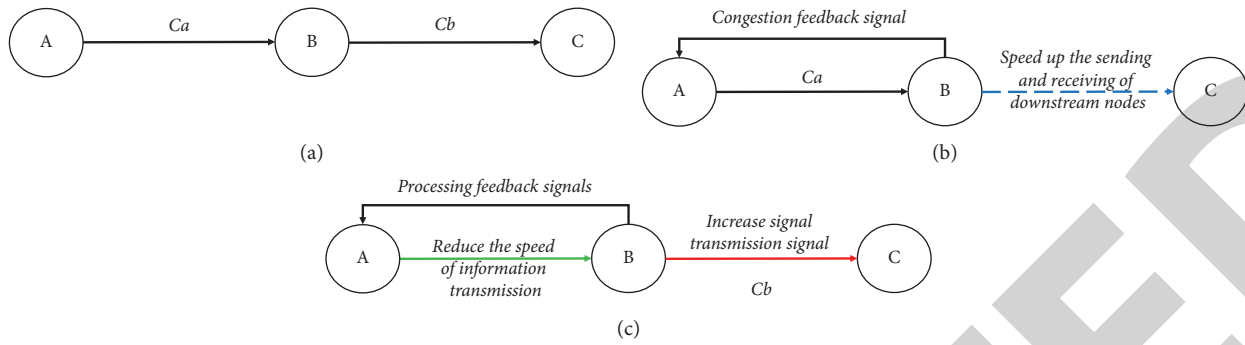


FIGURE 8: The congestion control method based on fuzzy logic. (a) Congestion detection, (b) feedback signal transmission and local congestion processing, and (c) feedback signal processing.

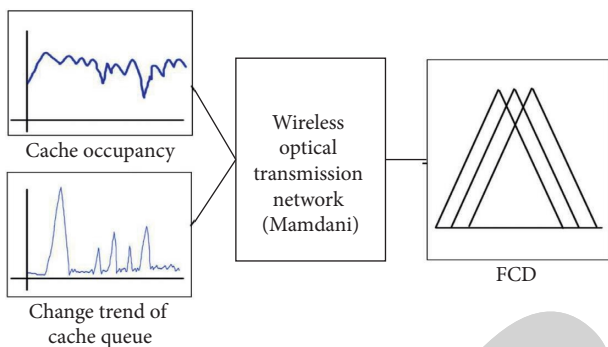


FIGURE 9: Fuzzy logic's derivation.

programs are developed by VC++ and MFC framework. The gateway runs on the Raspberry 3b+'s Raspbian system, and the programs are developed by C++, compiled by the makefile. The data server and monitoring terminal programs are developed by VS2015 and MFC framework.

In the system running, the data server programs are executed on the host with an address of 10.192.168.96 in the first step. Figure 11(a) shows the operation result. The data server starts monitoring the connection requests from the gateway and the monitoring terminal. The executable files are obtained by compiling the Make commands on the above host, namely, gateway. The default data server's connection port is set to run the command/gateway 10.192.168.96 and change the gateway-data server's connection port. Meanwhile, Figure 11(b) shows that the gateway information can be seen by checking the data server program's real-time status.

With the optical fiber sensor connected to the WeMos D1 development board and Arduino program compiled by the PC uploaded to the development board, the development board sends the sensor data through WiFi when the board is energized. Targeted at water level measurement with analog data in the experiment, the sensor output line is connected to the development board's A0 pin. Figure 12 shows part of the gateway's running after the development board's energization and connection to the gateway.

In Figure 12, the gateway is connected to the host at 10.192.168.96. The sensor starts to analyze the water level data and actively informs the gateway of its location

information. Here, the optical fiber sensor's default type is 1, meaning water level measurement usage. The sensor's ID is 100 with the location name "Eric Johnson 501." Data represent real-time data, that is, the current data value measured by the sensor.

In Figure 13, the monitoring terminal feeds back the current sensor status and data collected by the monitoring terminal and gateway in real time. Multiple optional items are set, including data server, gateway, alarm, and early warning. According to different water level values, different warning levels are displayed. With any option clicked, the corresponding information is displayed in the middle of the displayed page.

In water level detection depicted in Figure 14, the Sensor Operation subwindow displays the water level's real-time curve values, including the solid-line section and dashed-line section representing the water level's dynamic and stable changing sections, respectively.

When the gateway is connected to multiple optical fiber sensors, each sensor is polled to determine an early warning. A WeMos D1 development board is adopted in the system design, so a virtual sensor connection program is needed to achieve multiple connections. Figure 15 shows that ten fiber optic sensors are connected with IDs from 101 to 110. Multiple connection threads are started here for multiple fiber optic sensors' data transmission to the gateway with connection time limits because only the fiber optic sensor with the ID of 100 keeps a constant connection to the gateway.

The network topology in Figure 16 is adopted in IoT's congestion control with multiple optical fiber sensors. The topology involves 15 nodes, and the transmission distance between nodes is limited to 200 m. With a detection radius of 650 m, each hop-by-hop data packet's length is set to 1500 bits.

The SenTCP algorithm [36, 37] and CODA algorithm [38, 39] are introduced for network congestion performance comparison. In Figures 17 and 18, the network packet loss and network throughput are compared. When the network data packets are sent and received in the proposed IoT with optical fiber sensors, the packet loss is reduced, with an average packet loss of 40 packets/s. However, those of the SenTCP algorithm and the CODA algorithm are over 100 packets/s. The proposed congestion control algorithm achieves a throughput of 180 packets/s, significantly better

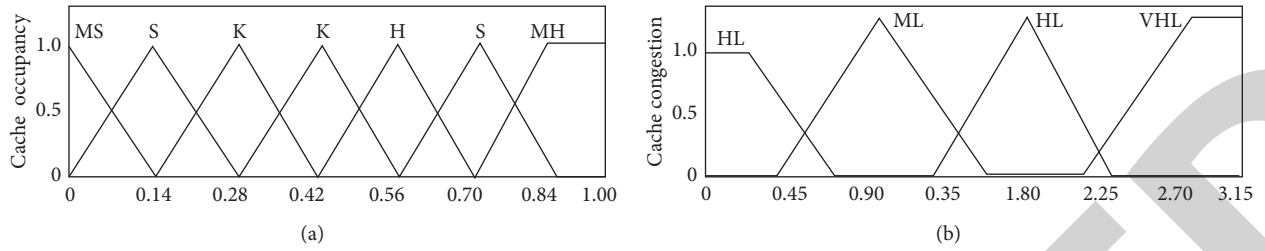


FIGURE 10: (a) Cache occupancy MF. (b) Cache congestion MF.

TABLE 1: Congestion rules based on fuzzy logic.

| Cache occupancy rate | Cache variation trend | Congestion |
|----------------------|-----------------------|------------|
| MS | VL | LL |
| MS | L | LL |
| S | VL | LL |
| S | M | LM |
| K | M | MM |
| K | H | MH |
| H | M | MH |
| H | H | HH |
| MH | L | MH |
| MH | M | HH |

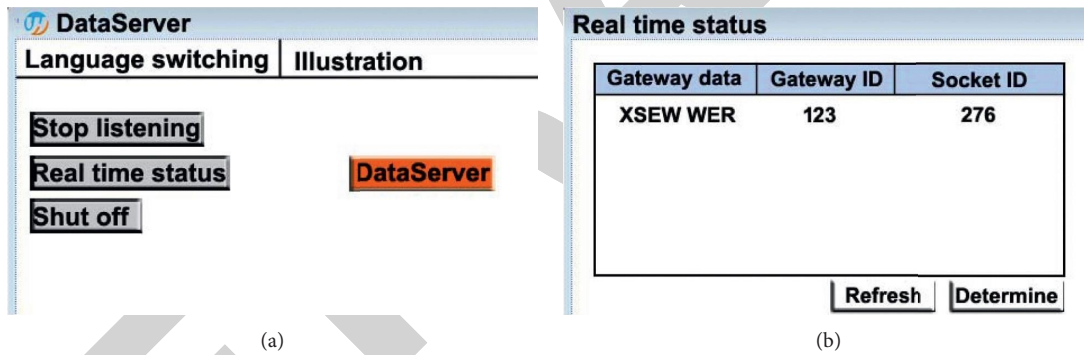


FIGURE 11: (a) The data server's operation interface. (b) The data server's gateway information.

```

pi@raspberrypi:~/test/180250/epoll_gateway_gh180242 $ ^C
pi@raspberrypi:~/test/180250/epoll_gateway_gh180242 $ ./gateway 10.192.168.96
epollEngine startup
connected with ip: 10.192.168.96 and port: 10563
Type:1 ID: 100 addr: eric johnson 501 Data: 30
Type:1 ID: 100 addr: eric johnson 501 Data: 30
Type:1 ID: 100 addr: eric johnson 501 Data: 30
Type:1 ID: 100 addr: eric johnson 501 Data: 30
Type:1 ID: 100 addr: eric johnson 501 Data: 8
Type:1 ID: 100 addr: eric johnson 501 Data: 32
Type:1 ID: 100 addr: eric johnson 501 Data: 28
Type:1 ID: 100 addr: eric johnson 501 Data: 28
Type:1 ID: 100 addr: eric johnson 501 Data: 30
Type:1 ID: 100 addr: eric johnson 501 Data: 12
Type:1 ID: 100 addr: eric johnson 501 Data: 32
Type:1 ID: 100 addr: eric johnson 501 Data: 32
Type:1 ID: 100 addr: eric johnson 501 Data: 12
Type:1 ID: 100 addr: eric johnson 501 Data: 12
    
```

FIGURE 12: Gateway-fiber sensor information.

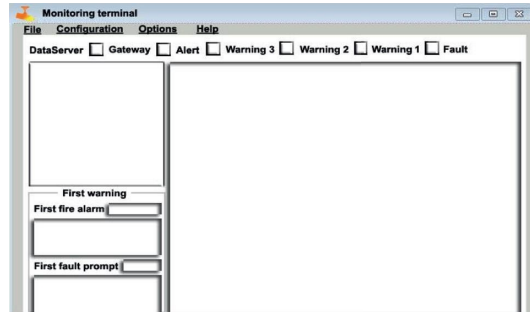


FIGURE 13: The monitoring terminal's main program interface.

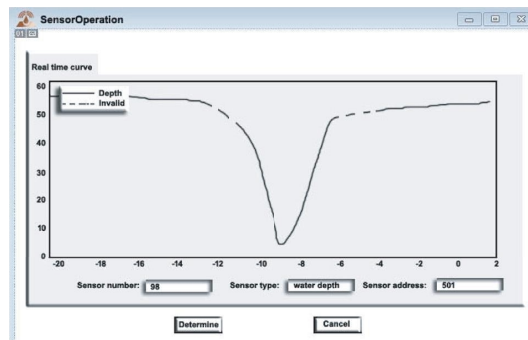


FIGURE 14: The real-time monitoring of water level measurement based on IoT detection.

```

Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 100 addr: eric johnson 501 Data: 3
Type:1 ID: 101 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 103 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 104 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 105 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 106 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 107 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 110 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 102 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 109 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 108 addr: Wei Zerd 1335 Data: 53
Type:1 ID: 100 addr: eric johnson 501 Data: 3
    
```

FIGURE 15: The gateway's connection to 13 optical fiber sensors.

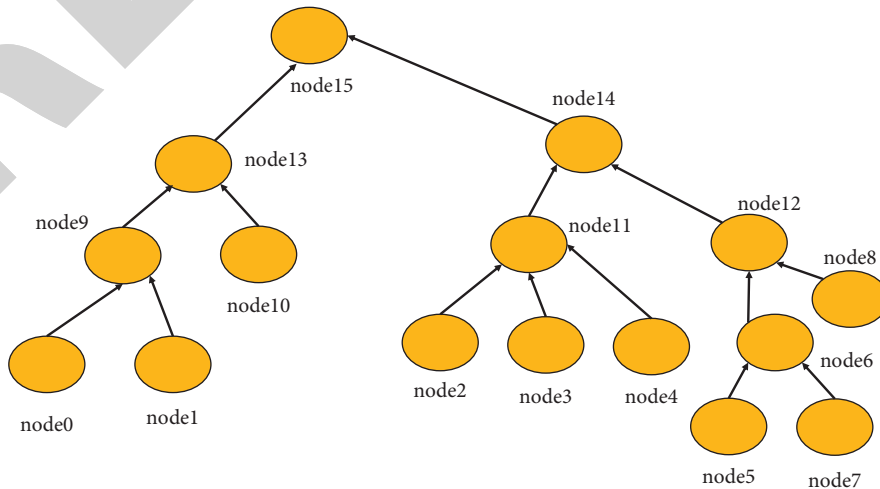


FIGURE 16: The IoT network topology.

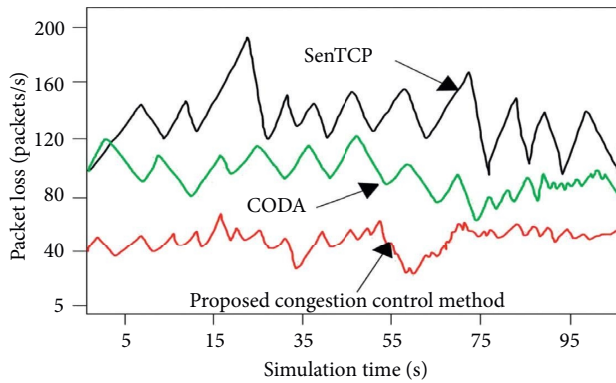


FIGURE 17: Network packet losses of different congestion control algorithms.

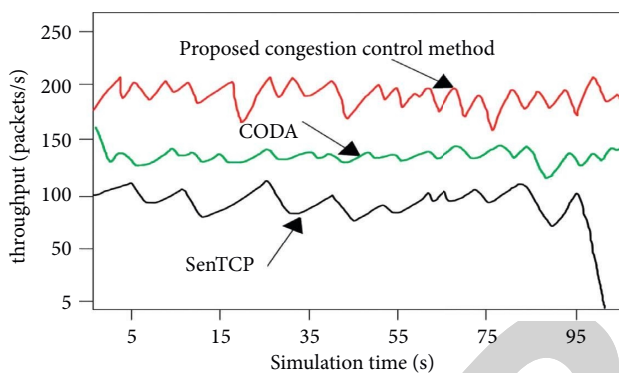


FIGURE 18: Network throughputs of different congestion control algorithms.

than introducing the SenTCP algorithm and CODA algorithm.

6. Conclusion

This article designed a multiple optical fiber sensor-based IoT detection system for water level detection based on optical fiber sensors' characteristics. A network congestion control method is proposed based on the fuzzy logic principles. The Transmission Control Protocol (TCP)/Internet Protocol (IP) communication protocol stack is used for sensor module, gateway module, data server, and monitoring module design for establishing the data server in the Windows system for communicate across the network segments. The experimental analysis opens the data server for detection first and after the gateway program's running, the WeMos D1 development board is connected to the fiber sensor. The regular operation of the curve drawing, water level alarm, and multisensor connection indicates that the proposed IoT detection system is effective. It is revealed that the fuzzy logic-based network congestion control method performs better than other traditional network congestion control methods, such as SenTCP and CODA, in network operation. Further, the future research directions indicates that the combination of fuzzy, multiple optical fiber sensor, and IoT detection methodology can be applied for the detection in various other domains of detection and transmission.

Data Availability

All data are included within the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was supported by the research program of the Education Department of Shaanxi Province (Grant no. 15JK1829) and the 13th Five-Year Plan Project of Education Science in Shaanxi Province (Grant no. SGH18H375).

References

- [1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [2] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
- [3] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Sensing as a service model for smart cities supported by internet of things," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 1, pp. 81–93, 2014.
- [4] V. Pande, C. Marlecha, and S. Kayte, "A review-fog computing and its role in the internet of things," *International Journal of Engineering Research in Africa*, vol. 6, no. 10, pp. 2248–96227, 2016.
- [5] L. Yancong, Y. Jiachen, L. Zhihan, W. Wei, and S. Houbing, "A self-assessment stereo capture model applicable to the internet of things," *Sensors*, vol. 15, no. 8, pp. 20925–20944, 2015.
- [6] M. A. Soto and L. Thévenaz, "Modeling and evaluating the performance of brillouin distributed optical fiber sensors," *Optics Express*, vol. 21, no. 25, pp. 31347–31366, 2013.
- [7] C. Elosúa, I. Vidondo, F. J. Arregui et al., "Lossy mode resonance optical fiber sensor to detect organic vapors," *Sensors and Actuators B: Chemical*, vol. 187, pp. 65–71, 2013.
- [8] W. C. Wong, C. C. Chan, P. Hu et al., "Miniature ph optical fiber sensor based on waist-enlarged bitaper and mode excitation," *Sensors and Actuators B: Chemical*, vol. 191, pp. 579–585, 2014.
- [9] A. G. Mignani, I. R. Matias, and R. O. Claus, "Special issue on optical fiber sensors," *Journal of Display Technology*, vol. 41, no. 2, pp. 111–116, 2003.
- [10] Q. Li, G. Li, G. Wang, F. Ansari, and Q. Liu, "Elasto-plastic bonding of embedded optical fiber sensors in concrete," *Journal of Engineering Mechanics*, vol. 128, no. 4, pp. 471–478, 2015.
- [11] I. García, "Optical fiber sensors for aircraft structural health monitoring," *Sensors*, vol. 15, no. 7, pp. 15494–15519, 2015.
- [12] B. Bao, L. Melo, B. Davies, H. Fadaei, D. Sinton, and P. Wild, "Detecting supercritical CO₂ in brine at sequestration pressure with an optical fiber sensor," *Environmental Science & Technology*, vol. 47, no. 1, pp. 306–313, 2013.
- [13] L. Mescia and F. Prudeniano, "Advances on optical fiber sensors," *Fibers*, vol. 2, no. 1, pp. 1–23, 2013.

- [14] T. Geernaert, "Temperature and strain measurements in concrete using micro-structure optical fiber sensors," *Pure and Applied Chemistry*, vol. 56, no. 2, pp. 215–230, 2015.
- [15] A. Masoudi, M. Belal, and T. P. Newson, "Distributed dynamic large strain optical fiber sensor based on the detection of spontaneous Brillouin scattering," *Optics Letters*, vol. 38, no. 17, pp. 3312–3315, 2013.
- [16] A. Urrutia, J. Goicoechea, P. J. Rivero, I. R. Matias, and F. J. Arregui, "Electrospun nanofiber mats for evanescent optical fiber sensors," *Sensors and Actuators B: Chemical*, vol. 176, pp. 569–576, 2013.
- [17] M. Boerkamp, D. W. Lamb, and P. G. Lye, "An intrinsic exposed core optical fiber sensor as a quantitative surface crystallization monitoring sensor," *Sensors and Actuators B: Chemical*, vol. 177, pp. 964–969, 2013.
- [18] C. Chen, Y. S. Yu, R. Yang et al., "Reflective optical fiber sensors based on tilted fiber Bragg gratings fabricated with femtosecond laser," *Journal of Lightwave Technology*, vol. 31, no. 3, pp. 455–460, 2013.
- [19] H. Zhang, S. Wang, Y. Gong et al., "A quantitative robustness evaluation model for optical fiber sensor networks," *Journal of Lightwave Technology*, vol. 31, no. 8, pp. 1240–1246, 2013.
- [20] H. Tu, T. Sun, and K. T. Grattan, "Spr-based optical fiber sensors using gold–silver alloy particles as the active sensing material," *IEEE Sensors Journal*, vol. 13, no. 6, pp. 2192–2199, 2013.
- [21] S. Ali and H. Li, "Moving software testing to the cloud: an adoption assessment model based on fuzzy multi-attribute decision making algorithm," in *Proceedings of the 2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA)*, pp. 382–386, IEEE, Tokyo, Japan, April 2019.
- [22] F. Saeed, A. Paul, A. Rehman, W. H. Hong, and H. Seo, "IoT-based intelligent modeling of smart home environment for fire prevention and safety," *Journal of Sensor and Actuator Networks*, vol. 7, no. 1, p. 11, 2018.
- [23] P. H. Chou, Y. L. Hsu, W. L. Lee et al., "Development of a smart home system based on multi-sensor data fusion technology," in *Proceedings of the 2017 International Conference on Applied System Innovation (ICASI)*, pp. 690–693, IEEE, Sapporo, Japan, May 2017.
- [24] J. Olivares-Mercado, K. Toscano-Medina, G. Sánchez-Perez et al., "Early fire detection on video using LBP and spread ascending of smoke," *Sustainability*, vol. 11, no. 12, p. 3261, 2019.
- [25] J. H. Park, S. Lee, S. Yun, H. Kim, and W. T. Kim, "Dependable fire detection system with multifunctional artificial intelligence framework," *Sensors*, vol. 19, no. 9, p. 2025, 2019.
- [26] B. Sarwar, I. S. Bajwa, S. Ramzan, B. Ramzan, and M. Kausar, "Design and application of fuzzy logic based fire monitoring and warning systems for smart buildings," *Symmetry*, vol. 10, no. 11, p. 615, 2018.
- [27] S. Y. Chiang, Y. C. Kan, Y. S. Chen, Y. C. Tu, and H. C. Lin, "Fuzzy computing model of activity recognition on WSN movement data for ubiquitous healthcare measurement," *Sensors*, vol. 16, no. 12, p. 2053, 2016.
- [28] D. Tien Bui, K. Khosravi, S. Li et al., "New hybrids of anfis with several optimization algorithms for flood susceptibility modeling," *Water*, vol. 10, no. 9, p. 1210, 2018.
- [29] Y. Bao, Y. Huang, M. S. Hoehler, and G. Chen, "Review of fiber optic sensors for structural fire engineering," *Sensors*, vol. 19, no. 4, p. 877, 2019.
- [30] S. Ali, N. Ullah, M. F. Abrar, Z. Yang, and J. Huang, "Fuzzy multicriteria decision-making approach for measuring the possibility of cloud adoption for software testing," *Scientific Programming*, vol. 2020, Article ID 6597316, 24 pages, 2020.
- [31] S. Ali, H. Li, S. U. Khan, Y. Zhao, and L. Li, "Fuzzy multi attribute assessment model for software outsourcing partnership formation," *IEEE Access*, vol. 6, pp. 55431–55461, 2018.
- [32] F. L. Lewis, W. K. Tim, L. Z. Wang, and Z. X. Li, "Deadzone compensation in motion control systems using adaptive fuzzy logic control," *Control Systems Technology IEEE Transactions on*, vol. 7, no. 6, pp. 731–742, 2015.
- [33] B. D. Baets, F. Esteva, J. Fodor, and L. Godo, "Systems of ordinal fuzzy logic with application to preference modelling," *Fuzzy Sets and Systems*, vol. 124, no. 3, pp. 353–359, 2015.
- [34] B. Lee, J. Yen, L. Yang, and J. C. Liao, "Incorporating qualitative knowledge in enzyme kinetic models using fuzzy logic," *Biotechnology and Bioengineering*, vol. 62, no. 6, pp. 722–729, 2015.
- [35] S. Saad and L. Zellouma, "Fuzzy logic controller for three-level shunt active filter compensating harmonics and reactive power," *Electric Power Systems Research*, vol. 79, no. 10, pp. 1337–1341, 2016.
- [36] H. Wu, Z. Feng, C. Guo, and Y. Zhang, "ICTCP: incast congestion control for TCP in data-center networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 2, pp. 345–358, 2013.
- [37] R. Mittal, V. T. Lam, N. Dukkupati et al., "Timely: RTT-based congestion control for the datacenter," *Computer communication review*, vol. 45, no. 4, pp. 537–550, 2015.
- [38] M. A. Jan, P. Nanda, X. He, and R. P. Liu, "PASCCC: priority-based application-specific congestion control clustering protocol," *Computer Networks*, vol. 74, pp. 92–102, 2014.
- [39] R. Manoharan and J. M. Gnanasekar, "Congestion control in heterogeneous wireless ad hoc network using FRCC," *Australian Journal of Basic and Applied Sciences*, vol. 9, no. 7, pp. 698–702, 2015.