Journal of King Saud University – Computer and Information Sciences 35 (2023) 357–367

HOSTED BY

ELSEVIER

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

An energy fault and consumption optimization strategy in wireless sensor networks with edge computing

Guozhi Li [a], Yan Tong [a], Ge Zhang [a,*], Yue Zeng [b]

[a] School of Information Technology, Henan University of Chinese Medicine, Zhengzhou 450008, PR China
[b] Department of Computer Science and Engineering, Nanjing University, Nanjing 210046, PR China

A R T I C L E   I N F O

Article history:
Received 12 July 2022
Revised 16 November 2022
Accepted 6 December 2022
Available online 13 December 2022

Keywords:
Artificial intelligence
Causality reasoning graph
Energy hole and consumption balance (EHCB) problem
NS-2

A B S T R A C T

With the operation of wireless sensor networks (WSNs), energy-constrained sensor nodes and inadequate energy constraint methods are becoming obsolete, as they reduce the efficiency of data transmission. The appearance of edge computing (EC) and causality graph provide a new opportunity for energy fault analysis and assessment in WSNs. As a result, by selecting a reliable data transmission path and averaging the energy consumption, we present an energy fault and consumption optimization (EFCO) algorithm for solving the energy hole and consumption balance (EHCB) problem in wireless sensor networks with edge computing (ECWSNs). Specifically, we first build a novel four-layer network architecture by using edge computing technology and causality graph theory. Then, the energy fault cost (EFC) in ECWSNs is formulated as an optimization problem that is constrained by the energy allocation of relay nodes. Furthermore, we propose a causal reasoning algorithm to deduce the single-value fault status probability of relay nodes. Finally, we utilize version 2 of a network simulator (NS-2) to evaluate the fault derivation and energy allocation efficiency of the EFCO algorithm in ECWSNs.

1. Introduction

Wireless sensor networks (WSNs) have been integrated into every corner of modern society, for example, in deep forest fire warning, water pollution monitoring, vehicle speeding supervision, and wildlife detection systems (Li et al., 2018). Building a high-performance network is the main factor for ensuring the completion of sensor tasks. The main challenges of effective wireless sensor networks are the fast energy consumption caused by the large amount of transmitted data and the inability to detect faulty sensor nodes in complex environments (Zhou et al., 2022). When a faulty wireless sensor node is used as an effective relay node to transmit data, there is a large amount of data packet loss and multiple retransmission problems, especially when the faulty node is located in the path near the target node, which leads to network energy consumption. Therefore, a reasonable energy consumption strategy for faulty node diagnosis would prolong the life cycle of wireless sensor networks, thereby improving the efficiency and robustness of the network detection environment.

A causality graph is a method of obtaining causality expressions and reasoning under dynamic uncertainty based on directed graphs (Liu et al., 2018). The characteristics of information processing are uniquely advantageous in the fault diagnosis of WSN processes with complex structures (Bruni et al., 2002). A causality graph expresses the causal mechanism with virtual independent random events, quantifies it as the probability of random events and the probability of causal relationships, and performs makes logical reasoning to qualitatively determine the probability of faults(Zhang, 2012). Edge computing (EC) gives edge entities more powerful information processing abilities and content delivery capabilities by pushing computation and storage from the central cloud to the edge cloud (Guo et al., 2019). EC also provides an efficient and low-latency support platform for service implementation (Ma et al., 2021). We integrate EC technology into WSNs to propose a novel network architecture called an *edge computing wireless sensor network* (ECWSN), which can highlight the advantages of EC technology in intelligent reasoning networks.

* Corresponding author.
E-mail address: zhangge@hactcm.edu.cn (G. Zhang).

Peer review under responsibility of King Saud University.
This work was supported by the National Natural Science Foundation of China under Grant 81703946 and Henan Province Key Scientific Research Projects of Colleges and Universities of China under Grant 22A520032.

ELSEVIER Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2022.12.005
1319-1578/© 2022 The Author(s). Published by Elsevier B.V. on behalf of King Saud University.

Compared to the traditional WSN architecture, the ECWSN architecture can better use powerful AI perception nodes to achieve network status awareness and perform node fault reasoning, data analysis and processing, and other functions (Jain et al., 2022). We call this kind of AI perception node an *edge computing server* (ECS). The scheduling of energy consumption in ECWSNs is the root cause of node faults (Kamal and Adouane, 2018). An ECS can deduce the energy fault probability of ECWSNs based on the real-time node operating status, which determines a reliable transmission link (Li and Xu, 2019). On the other hand, an ECS can select the relay nodes with the most appropriate energy allocation through the perception of the global network status to avoid the energy hole problem to the greatest extent possible (Li and Xu, 2019).

In this paper, we consider the energy holes and consumption balance (EHCB) problem in ECWSNs. Our goal is to find reliable relay nodes and identify the optimal energy allocation strategy by using the established energy fault cost function. Furthermore, the proposed scheduling strategy can avoid faulty nodes and balance energy allocation based on causal reasoning and EC technologies. Specifically, we first build a four-tier architecture for ECWSNs to depict the EHCB problem. Second, we analyse the fault probability of each node based on a causality graph constructed according to ECWSN characteristics. Afterwards, the energy fault and consumption optimization (EFCO) algorithm is proposed according to the causality reasoning result of the available relay nodes. Finally, we analyse the reasons for choosing version 2 of the network simulator (NS-2) as simulation software, and the experimental results prove that the EFCO algorithm has a lower energy fault detection efficiency and a higher reliable transmission ratio.

The contributions of this paper are summarized as follows.

- We formulate an energy fault cost model that characterizes the multivalued failure probability states of perceptual nodes in ECWSNs.
- We adopt a causal reasoning algorithm to convert the probabilities of the multivalued fault status of a sensor node into a single-valued fault status. Then, the Lagrangian duality mechanism is used to characterize the energy fault cost optimization model. The optimal value is used as a scale to solve the energy hole problem.
- We propose an EFCO algorithm that minimizes the fault probability by selecting the optimal EC system and data transmission path. Our EFCO algorithm not only guarantees the reliable transmission of data but also balances the energy consumption of sensing nodes in ECWSNs.

The remainder of this paper is organized as follows. Section 2 introduces the related research results. Section 3 provides the network and system model. Section 4 formalizes the EHCB problem. In Section 5, the EFCO algorithm is presented. Section 6 evaluates the performance of the EFCO algorithm. Finally, the conclusion is summarized in Section 7.

2. Related work

However, the existence of energy failure nodes in traditional WSNs invalidates the routing algorithm. The research results in Dong et al. (2019) and Zhu et al. (2019) address this problem. The fairness cooperation algorithm (FCA) presented in Dong et al. (2019) defines fairness as the proportion of shared resources occupying the remaining resources, and it is used as the weight in the optimization function to determine the cooperation strategy (Zeng et al., 2019). The FCA transmits the node status to the ECS in the current domain, which establishes a transmission path to satisfy the constraints of resource cooperation. However, this process is based on the premise of a healthy network. This scheduling strategy fails if energy hole phenomena occur in ECWSNs. The adaptive multiservice network selection scheme (MSNS) in Zhu et al. (2019) is designed to solve the heterogeneous network selection problem for multiservice users with different node statuses. It combines the dynamic adaptability of fuzzy logic with near-optimal multiattribute decision making. The MSNS algorithm infers the transmission status of ECWSNs based on fuzzy logic theory. However, this strategy does not consider the global optimal network status, so the found path cannot select the appropriate relay nodes, which results in an inefficient energy consumption balance and reduces the life cycle of ECWSNs.

Although these routing algorithms can exploit the new idea of using edge computing to improve the performance of data transmission, these scheduling strategies are based on fault-free perception devices and average energy consumption. Therefore, these algorithms cannot fundamentally solve the problems of energy holes or the energy consumption balance. The above two strategies inspire our pursuit of a reliable data transmission path with a minimum fault probability and reasonable energy allocation.

3. Network and system model

3.1. Network model

In a WSN, we deploy an ECS on a sensor with powerful AI computing, storage capacity, and the largest remaining energy. The ECS undertakes network fault analysis and the effective scheduling of data transmission. This type of WSN is called an ECWSN. To consume less energy by shortening the transmission distance of the sensors in ECWSNs, we generally deploy the ECS in the centre of the perceived node region. The region managed by the ECS is called the edge computing system.

We divide the ECWSN architecture into four decoupled layers: the transmission layer, control layer, application layer, and perception layer. Fig. 1 illustrates this layered structure. We use layer transmission and perception for effective data transmission. Fault analysis and data scheduling strategy formulation are performed in the control layer. The application layer determines the specific user requirements. The four-layer ECWSN architecture is described in detail below.

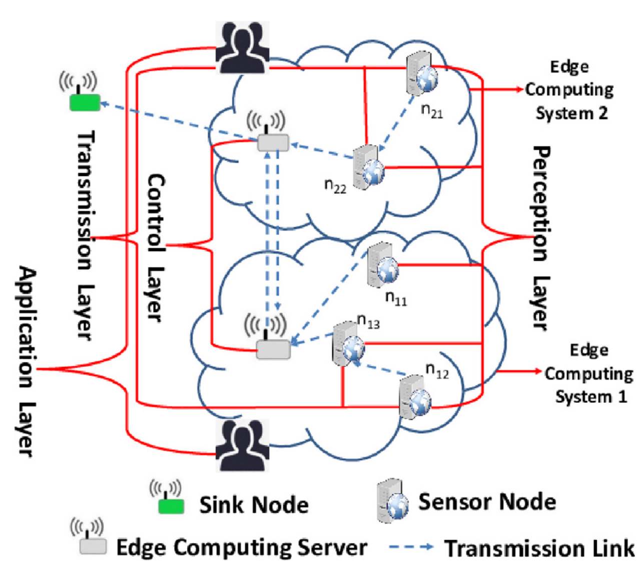


Fig. 1. Edge computing wireless sensor networks (ECWSNs).

(1) Transmission Layer: Sensors with a larger energy fault may form an energy hole, which causes data transmission failure in ECWSNs. In addition, a higher level of area malfunction caused by a faulty node can result in local network faults and even network crashes (Zhang, 2018). Thus, the design of a transmission architecture and the optimization of resource allocation among the sensors in view of fault prediction and data scheduling are critical to network performance in data transmission.

(2) Control Layer: The sensors first send the transmission requirements to the ECS, and then, the ECS develops the scheduling strategy based on the node fault statuses. The data transmission requirement is generally forwarded by the relay nodes, and the control commands sent by the ECS are also transmitted by the relay nodes. Therefore, a sudden node fault invalidates the entire scheduling task in the ECWSN. The control architecture and fault diagnosis and prediction among the sensors are constructed by the ECSs.

(3) Application Layer: The ECS needs to satisfy the various demands of network applications, and the ECS adopts the corresponding scheduling strategy according to different applications. These applications mainly include the perceived environment, data transmission, fault diagnosis and prediction, and network status monitoring. Thus, the design of the architecture in the application layer is critical in speeding up the deployment of new applications.

(4) Perception Layer: We need to consider various types of perceived data in various complex environments. Since adjacent sensor nodes have similar perceptual environments, these nodes collect similar or identical redundant data, so eliminating redundant data is an important process for saving network resources. The design for storing unstructured data in the memory of sensors is critical to the efficient management of the collected data.

3.2. System model

We consider an edge computing system with j sensors, represented by a set $J = \{1, 2, \cdots, j\}$, each of which has a sensing module, an energy module, a computing and storage module, and a radio frequency (RF) communication module. The ECWSNs have i edge computing systems. When the data perceived by the sensing module of the sensor need to be transmitted, the RF communication module of the sensor sends a transmission requirement ω to the ECS of the current region. Then, the ECS selects the appropriate relay nodes according to the real-time state of the ECWSNs and sends the scheduling strategy information to the corresponding devices.

There are four types of faults in sensors: sensing faults, energy faults, computing and storage faults, and RF communication faults. If the fault probability of the energy module is not within the effective value range, the sensor cannot perform the data transmission tasks, which leads to energy hole phenomena. Therefore, we need to choose a reasonable method to judge the mechanism for avoiding the energy hole problem. A causality graph is a probability-based knowledge representation method developed based on a reliability network. Many examples have proven that a causality graph can be effective at system fault diagnosis (Volpe et al., 2012). The system model for node fault reasoning considered in this paper is illustrated in Fig. 2.

We define the symbol for each element in the system model as follows.

◆ **Circle Node:** x denotes a node event or a node event variable. A node event variable takes an element of a set of mutually exclusive node events. These mutually exclusive node events form the complete sample space of node events. For a valid value range of a node event variable, its fault probability does not exceed 30%.

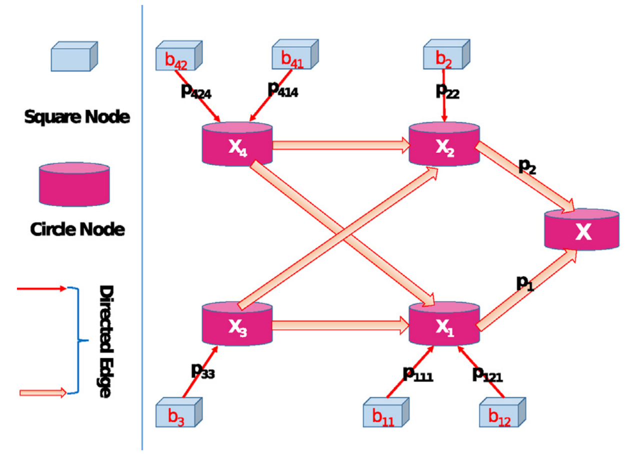


Fig. 2. Causality reasoning graph.

◆ **Square Node:** b represents a basic event or a basic event variable. It is an independent way of connecting events or connecting event variables. For a valid value range of a node event variable, its fault probability does not exceed 50%.

◆ **Directed Edge:** p represents a connected event or connected event probability. It allows an input event to cause or not cause a corresponding output event. The valid value of a directed edge is often obtained by expert subjective reliability or statistics.

In these definitions, the term **variable** represents a set of mutually exclusive events or an event vector or matrix, each of whose member events is called an instance or a value of the corresponding event variable. If the fault probability of a variable exceeds its range of valid values, we consider this sensor the faulty node. The fault probability of each module is adjusted to within the valid value range. Table 1 lists some descriptions and valid value ranges of nodes and basic events.

A basic event has no input, but it has at least one output for a node event. A node event has at least one input and any number of outputs. A directed edge can start with a basic event or a node event, but it always points to a node event.

According to our definitions, x_1 represents an RF communication fault in an ECWSN, so we can obtain

$$x_1 = p_{111}b_{11} \bigcup p_{121}b_{12} \bigcup p_{31}x_3 \bigcup p_{41}x_4. \tag{1}$$

In the same way, sensing faults, energy faults, and computing and storage faults are represented as x_2, x_3, and x_4, respectively. Thus, the node events are expressed as follows:

Table 1
Event description.

Notation	Description	Valid value range
X	Node fault probability	0–20%
x_1	Fault probability of the RF communication module	0–30%
x_2	Fault probability of the sensing module	0–30%
x_3	Fault probability of the energy module	0–20%
x_4	Fault probability of the computing and storage module	0–30%
b_{11}	Fault probability of sending data	0–30%
b_{12}	Fault probability of receiving data	0–20%
b_2	Error probability of perceptual data	0–50%
b_3	Probability that the voltage is less than 1.9 V	0–10%
b_{41}	Probability that the temperature is greater than 70 degrees Celsius	0–20%
b_{42}	Degradation probability of the calculation speed	0–50%

$$x_2 = p_{22}b_2 \bigcup p_{32}x_3 \bigcup p_{42}x_4 \tag{2}$$

$$x_3 = p_{33}b_3 \tag{3}$$

$$x_4 = p_{414}b_{41} \bigcup p_{424}b_{42} \bigcup p_{34}x_3. \tag{4}$$

A sensor fault is a fault in one or more modules. Each node event is composed of a sum of one or more basic events; a basic event contributes to node event generation, and the connection probability p determines the probability of the node event occurring. Therefore, we can obtain the fault of the i-th sensor X_i.

$$X_i = p_1x_1 \bigcup p_2x_2 \tag{5}$$

Combining formulas (1), (2), (3), and (4) with (5), we can obtain the following causality reasoning model for node faults:

$$\begin{aligned} X_i = \; & p_1p_{111}b_{11} \bigcup p_1p_{121}b_{12} \bigcup p_2p_{22}b_2 \bigcup((p_1+p_2)p_{34}p_{42} \\ & +(p_1p_{31}+p_2p_{32}))p_{33}b_3 \bigcup(p_1p_{41}+p_2p_{42})p_{414}b_{41} \\ & \bigcup(p_1+p_2)p_{42}p_{424}b_{42}. \end{aligned} \tag{6}$$

Assuming that one data transmission task requires m relay nodes for completion, these sensor nodes should have a relatively low fault probability to ensure data transmission. Therefore, we must choose a reliable link for data transmission through an effective scheduling strategy.

Note that any ECS can communicate freely with other ECSs, and the energy fault reasoning of sensors does not affect the analysis or processing of the perceived data in ECWSNs. In addition, at least one valid link can be found for each data transmission.

4. Problem formulation

To ensure reliable data transmission, we should avoid passing through the energy hole area on the transmission path. Therefore, the fault probability of the selected transmission path must be the lowest. Energy hole phenomena occur due to faulty nodes and uneven energy consumption in ECWSNs. We call this difficulty the EHCB problem. ECWSNs consist of four layers, and the control layer is responsible for node fault analysis and scheduling of perceived data. The transmission layer transmits the data of the perception layer to the destination node based on the different user requirements proposed by the application layer and a strategy formulated by the control layer.

According to the requirements of different applications, data acquisition requires energy, sensing, and computing and storage modules, and the network functions of fault diagnosis and data transmission are implemented by energy, computing and storage, and transmission modules. An energy module fault is a key factor in evaluating node faults, and energy damage can cause the nodes to lose all their functions. Therefore, Theorem 1 proves the relationship between the energy depletion probability $p_i^{(e)}$ and the random fault probability $p_i^{(r)}$ of node i in an ECWSN:

Theorem 1. *In energy fault analysis in an ECWSN, the energy depletion probability $p_i^{(e)}$ of the i-th node is greater than the random fault probability $p_i^{(r)}$ of node i, i.e., $p_i^{(e)} > p_i^{(r)}$.*

Proof. See the proof in the appendix. □

The EHCB problem formulates the function of the energy fault cost. Then, the proposed scheduling strategy ensures the reliable transmission of data flows and maximizes energy savings in the ECWSN. The optimization function of the energy fault cost can be given by

$$OPT-1 \quad \min \sum_{k\in K}\sum_{j\in J}\rho_{kj}F(\frac{X_{kj}}{\tilde{X}}), \tag{7}$$

subject to the requirements such that $\forall k \in K$ and $\forall j \in J$,

$$\begin{aligned} C1: \quad & \sum_{k\in K}\sum_{j\in J}p_{kj}^{(e)} > |KJ|p_{\max}^{(r)}, \\ C2: \quad & \sum_{k\in K}\sum_{j\in J}\tilde{E}(X_{kj}) \le |KJ|E_{\min}, \\ C3: \quad & \rho_{kj} \in \{0,1\}, \end{aligned} \tag{8}$$

where X_{kj} represents the fault event of the j-th relay node in the k-th edge computing system of the ECWSN and $\tilde{X}$ is the relay node that contains the most basic events. $\tilde{E}(X_{kj})$ and $E_{\min}$ are the energy consumption of the kj-th node and the minimum remaining energy of a node in the available transmission path, respectively. kJ denotes the set of relay nodes in a transmission path. The energy fault cost function can be described as follows:

$$F(\tilde{x}) = p(\tilde{x})^{1+\varepsilon}, 0 < \varepsilon < 1, \tag{9}$$

where $\tilde{x}$ is an expression consisting of basic events and connection events in the ECWSN. $\varepsilon = \frac{\tilde{E}(X_{kj})}{E_{\min}}$ indicates the impact factor of the node energy fault.

To convert the energy fault cost F to the fault probability function p, we need to transform expression x into the final cut set of node events. The function of the energy fault cost F increases in terms of X_{kj}, and the OPT-1 function is strictly convex.

When the energy depletion probability $p^{(e)}$ is less than the maximum value of the random fault probability $p_{\max}^{(r)}$ on a data transmission path, one or more relay nodes are at risk of faults in an ECWSN. Therefore, constraint $C1$ describes the fault probability condition. All modules must have sufficient energy to ensure normal operation; hence, constraint $C2$ indicates that the total energy consumption of an effective transmission link cannot exceed the minimum energy $E_{\min}$ of $|KJ|$ times. The selection scale of the effective relay nodes is expressed in constraint $C3$, and the selected relay node has the lowest fault probability and the appropriate energy consumption.

The optimization function (OPT-1) minimizes the energy fault probability in an ECWSN by comparing the energy depletion probabilities $p^{(e)}$s and by selecting the appropriate energy consumption, thereby ensuring that all application needs of the users are met. Constraint $C3$ is a discrete integer variable, and we reformulate this constraint for the fault probability of node events as follows:

$$\sum_{k\in K}\sum_{j\in J}\rho_{kj}p_{kj} \le |KJ|p_{\max}, \tag{10}$$

where $p_{\max}$ is the maximum fault probability, which a valid data transmission path does not exceed.

The traditional optimization solution repeatedly calculates $\frac{X_{kj}}{\tilde{X}}$. $\tilde{X}$ is a complex causal reasoning expression. This situation increases the time complexity of our proposed EFCO algorithm. Therefore, we set $y_{kj} = \frac{X_{kj}}{\tilde{X}}$ and rewrite the function OPT-1 to OPT-2 as follows:

$$OPT-2 \quad \min \sum_{k\in K}\sum_{j\in J}\rho_{kj}F(y_{kj}) \tag{11}$$

subject to $\forall k \in K, \forall j \in J$,

$$\sum_{k\in K}\sum_{j\in J}p_{kj}^{(e)} > |KJ|p_{\max}^{(r)} \tag{12}$$

$$\sum_{k\in K}\sum_{j\in J}\tilde{E}(y_{kj}) \le |KJ|E_{\min} \tag{13}$$

$$\sum_{k\in K}\sum_{j\in J}\rho_{kj}p_{kj}\leq |KJ|p_{\max}. \tag{14}$$

We adopt the Lagrangian dual method (Sl et al., 2021) to solve the function OPT-2, and the optimization variable y^* is determined. The minimization function value $F(y^*)$ is a metric for selecting a reliable transmission path with average energy consumption in an ECWSN.

5. Energy fault and consumption optimization algorithm

In this section, we first propose the causality reasoning algorithm to deduce the energy fault probability p_{kj} of all nodes in the current edge computing system. Afterwards, the Karush–Kuhn–Tucker (KKT) condition (Liu et al., 2019) of the minimization function (OPT-2) is solved by the Lagrangian dual method. Finally, we propose a reliable routing and energy allocation strategy to address the EHCB problem.

5.1. Causality reasoning algorithm

To solve the optimization problem (OPT-2), we need to transform the multivalue fault status of each module into the fault probability of the relay node. In the node causality graph, each status of the basic event variable b_{ij} has a probability value. We first define the occurrence probabilities of the different states of the event variable, and then, the fault probability of the relay node on the data transmission path is derived by Algorithm 1.

Definition 1 (*Event status probability*). Let event variable V_{ij} (x_{ij} or b_{ij}) consist of k' mutually exclusive states; then, the probability of the k'-th state $V_{ij}^{k'}$ occurring is $\psi(V_{ij}^{k'})$, with $0<\psi(V_{ij}^{k'})<1$. The probability of the k'-th state of event b_{ij} is $\psi(b_{ij}^{k'})$, with $\psi(b_{ij}^{k'})=P(b_{ij}^{k'})$.

Each node event has more than two fault state probabilities, which makes it difficult to describe the fault condition of a single-node event. For any multivalued state of the event variable, there is always a corresponding single-valued fault probability, which is obtained by considering all states of the basic event or node event variable (Zhang, 2015). The reason for transforming the multivalued states into a single-valued fault probability is to obtain the probability of a connected event. This transformation method is expressed in Assumption 1:

Assumption 1. Let each fault probability of the event variables (x_{ij} or b_{ij}) be proportional to its occurrence probability.

$$(V_{ij})=\begin{pmatrix}V_{ij}^{1}\\ \vdots\\ V_{ij}^{k'}\end{pmatrix}\vdots\begin{pmatrix}p\{V_{ij}^{1}\}\\ \vdots\\ p\{V_{ij}^{k'}\}\end{pmatrix}\vdots\begin{pmatrix}\psi\{V_{ij}^{1}\}\\ \vdots\\ \psi\{V_{ij}^{k'}\}\end{pmatrix} \tag{15}$$

We set

$$\varphi(V_{ij}^{k'})=\frac{\max\limits_{k'}\psi(V_{ij}^{k'})}{\sum\limits_{k'=1}^{K'}\psi(V_{ij}^{k'})}, \tag{16}$$

where $\varphi(V_{i}^{k'})$ denotes the probability distribution factor; then,

$$\varphi(V_{ij}^{k'})=\frac{\max\limits_{k'}p(V_{ij}^{k'})}{\sum\limits_{k'=1}^{K'}p(V_{ij}^{k'})} \tag{17}$$

where K' indicates the set of node fault states.

Each ECS is responsible for calculating the node fault probability in the local region and storing the result in the storage space of the ECS. The node fault probability serves as a basic metric for the selection of a reliable transmission path.

If the fault probability of a node event is deduced by the use of the traditional disjoint logical calculation, it greatly increases the number of calculations because this derivation process is very complicated, making it unsuitable given the limited resources in ECWSNs. Therefore, we adopt Assumption 2 to simplify this calculation process. Moreover, Assumption 1 does not have a significant impact on the calculation result of event fault probabilities in practical applications (Zhang et al., 2014).

Assumption 2. Assume that node event x_i has one or more basic events and that the relationship among the basic events is *OR*. If a basic event can provide probability $\psi(x_i)$ for the occurrence of node event x_i, then the value of $\psi(x_i)$ is equal to the individual probability $p(x_i)$ of node event x_i based on the hypothesis stating that other basic events do not occur. Otherwise, the incidence probability of node event x_i is equal to the fault probability sum for all linked basic events.

See (Zhang, 2015) for the solution to the final cut set expression in Algorithm 1. B is the set of basic events b_{ij}.

Algorithm 1. Causality reasoning algorithm

1: **Require**: Basic events b_{ij}, connection probabilities p_{ij}.
2: **while** $\{B\}\neq\varnothing$ **do**
3: Send the statuses of b_{ij} and p_{ij} for satisfying the user requirement to the i-th ECS.
4: Transform the multivalued probabilities of basic events into the single-valued fault probabilities $p(b_{ij})s$ via Assumption (1);
5: **if** $\tilde{E}_i\geq E_\omega$ **then**
6: Calculate the first-order cut set of the node event variables by solving Eqs. ()()()()(1)–(4);
7: Calculate the final cut set expression of the node event variables;
8: Extend the final cut set expressions of the node event variables to the state matrix forms;
9: Calculate the fault probability of the node events based on Assumption (2);
10: **else**
11: $i=i+1$;
12: **end if**
13: Save the node fault probabilities $p(X_i)s$ in the i-th ECS;
14: **end while**.

5.2. Lagrangian dual approach

We first use the Lagrangian dual decomposition approach (Chiang et al., 2007) to convert the minimization function (OPT-2) into an unconstrained optimization function (Tubishat et al., 2021). Afterwards, the Lagrangian multipliers α^*, β^* and γ^* are derived by the steps below.

To convert the objective function (OPT-2) into an unconstrained optimization expression, we introduce Lagrangian multipliers α, β and γ to relax constraints ()()()(12)–(14), so we can obtain

$$\begin{aligned} L(y, \alpha, \beta, \gamma) &= \sum_{k\in K}\sum_{j\in J}\rho_{kj}F(y_{kj}) \\ &+\alpha_{kj}(\sum_{k\in K}\sum_{j\in J}p_{kj}^{(e)} - |KJ|p_{\max}^{(r)}) \\ &+\beta_{kj}(|KJ|E_{\min} - \sum_{k\in K}\sum_{j\in J}\tilde{E}(y_{kj})) \\ &+\gamma_{kj}(|KJ|p_{\max} - \sum_{k\in K}\sum_{j\in J}\rho_{kj}p_{kj}). \end{aligned} \tag{18}$$

Eq. (18) can be solved by the following step:

$$\max_{\alpha\geq 0,\beta\geq 0,\gamma\geq 0} d(\alpha, \beta, \gamma). \tag{19}$$

The dual function (18) can be abbreviated as follows:

$$d(\alpha, \beta, \gamma) \equiv \min L(y, \alpha, \beta, \gamma). \tag{20}$$

Replacing Eqs. (18) and (20) yields the following Eq. (21):

$$d(\alpha, \beta, \gamma) = \min[d_0 + d_{kj}(y_{kj}, \alpha, \beta, \gamma)], \tag{21}$$

where

$$d_0 = \alpha_{kj}\sum_{k\in K}\sum_{j\in J}p_{kj}^{(e)} + |KJ|\beta_{kj}E_{\min} + |KJ|\gamma_{kj}p_{\max} \tag{22}$$

$$\begin{aligned} d_{kj}(y_{kj}, \alpha, \beta, \gamma) &= \sum_{k\in K}\sum_{j\in J}\rho_{kj}F(y_{kj}) - |KJ|\alpha_{kj}p_{\max}^{(r)} \\ &-\beta_{kj}\sum_{k\in K}\sum_{j\in J}\tilde{E}(y_{kj}) \\ &-\gamma_{kj}\sum_{k\in K}\sum_{j\in J}\rho_{kj}p_{kj}. \end{aligned} \tag{23}$$

We adopt the derivation method for Eq. (24) to solve Eq. (23).

$$\frac{\partial(d_{kj}(y_{kj}, \alpha, \beta, \gamma))}{\partial(p_{kj}, \tilde{E}_{kj})} = 0 \tag{24}$$

The optimal energy fault and allocation solution is derived by using Eqs. (23) and (24).

$$y_{kj}^* = \sum_{k\in K}\sum_{j\in J}\rho_{kj}(p_{kj}^{\varepsilon} - \gamma^*) - |KJ|\alpha^* - \beta^* \mid I \mid \tag{25}$$

We update the recursive form of the optimal Lagrangian multiplier iteratively as follows:

$$\alpha(t+1) = [\alpha(t) - s_1(\sum_{k\in K}\sum_{j\in J}p_{kj}^{(e)} - |KJ|p_{\max}^{(r)})]^+ \tag{26}$$

$$\beta(t+1) = [\beta(t) - s_2(|KJ|E_{\min} - \sum_{k\in K}\sum_{j\in J}\tilde{E}(y_{kj}))]^+ \tag{27}$$

$$\gamma(t+1) = [\gamma(t) - s_3(|KJ|p_{\max} - \sum_{k\in K}\sum_{j\in J}\rho_{kj}p_{kj})]^+, \tag{28}$$

where t is the number of iterations and s_1, s_2 and s_3 are the iteration step sizes. The optimal Lagrange multipliers ($\alpha^*, \beta^*, and \gamma^*$) are calculated by repeating the iterative process described above (Jalil Piran et al., 2020). y^* is the optimal solution for the energy fault and allocation in the ECWSN.

5.3. Energy fault and consumption optimization algorithm

To ensure reliable data transmission from the perception layer to the cloud centre, we need to select the transmission link with the lowest fault probability. Therefore, the EFCO algorithm is proposed in view of the fault reasoning results of Algorithm 1 in ECWSNs. The EFCO algorithm considers two aspects: the energy status and fault probability of the relay node.

In the first step, each ECS calculates the node fault probability of the current edge computing system based on Algorithm 1. To establish the four-tier architecture of ECWSNs and save valuable resources in the ECS, the optimal numbers of edge computing systems are calculated based on Theorem 2. Afterwards, the number of perceptual nodes belonging to each ECS does not exceed $\lfloor\frac{n}{\mathcal{O}_{opt}}\rfloor$, and $\tilde{E}_\omega$ is the energy consumption for the transmission of data flow requirement ω. In this step, each relay node sends the node status to the ECS in the current system, and T_i is the set of node fault probabilities. It is assumed that each data transmission has sufficient energy to satisfy its requirements in an ECWSN.

Theorem 2. *In an ECWSN, the optimal number of edge computing systems* $\mathcal{O}_{opt}$ *is equal to* $\sqrt{\frac{n\tau}{2\pi\varphi}}\frac{D}{d^2}$.

Proof. See the proof in the appendix. □

The second step is to solve the energy fault cost (OPT-2) problem in ECWSNs. We first derive the optimal value y^* and the dual vectors $\alpha^*, \beta^* and \gamma^*$ based on the Lagrangian dual method; then, the ECS sends these values to other ECSs to determine the energy fault cost $F(y^*)$. The optimal function $F(y^*)$ is the criterion for the selected path with the lowest fault probability and average energy consumption.

The optimal path with the minimum energy fault cost $F(y^*)$ is selected by the set $\mathcal{T}_j$ in the third step. The smaller the value of the energy fault cost $F(y^*)$ is, the more reliable the selected path and the more efficiently the perceptual data can be transferred to the destination node. y^* is the minimal energy fault probability for satisfying the constraints in the ECWSNs, and the energy fault cost $F(y^*)$ and constraints concentrate data transmission in nodes with smaller energy fault probabilities. $\mathcal{L}$ is the number of reliable transmission paths. Finally, the network status in the ECWSNs is updated.

We summarize the process of the EFCO algorithm in Algorithm 2.

Algorithm 2. Energy fault and consumption optimization (EFCO) algorithm

Input: Data transmission requirement ω, energy status E_{kj} in the kj-th sensor, network real-time status.
Output: The path with the minimal energy fault probability.
/*Step 1: Determine the energy fault probability for each node in the edge computing system */
1: Calculate $\mathcal{O}_{opt}$ by Theorem (2);
2: **for** $k = 1$ *to* $\mathcal{O}_{opt}$ **do**
3: **for** $j = 1$ *to* $\lfloor\frac{n}{\mathcal{O}_{opt}}\rfloor$ **do**
4: Select the node with an AI chip and the highest energy $E_{\max}$ as the ECS;
5: **if** $E_{ij} \geq \tilde{E}_\omega$ **then**
6: Calculate the node fault probability p_{kj} by using Algorithm 1;
7: Place the node fault probability p_{kj} into set T_k;
8: **end if**

```
        j = j + 1;
    end for
    Save set T_i in the k-th ECS;
    k = k + 1;
end for
  /*Step 2: Solve the optimization function (OPT-2) */
for j = 1 to ⌊n/𝒪_opt⌋ do
    Obtain the optimal solution y* and the dual variables
  α*, β*, andγ* by solving Eqs. ()()()()(25)–(28), respectively;
    Broadcast y*, α*, β*, γ* to other ECSs;
    j = j + 1;
    Obtain the optimal objective function (OPT-2) through
  the dual expression (18);
end for
  /*Step 3: Select the path with the minimal fault
  probability*/
for k = 1 to 𝒪_opt do
    for j = 1 to ⌊n/𝒪_opt⌋ do
        Find all the nodes in T_k to satisfy data transmission
  requirement ω;
        if ℒ ≥ 2 then
            j = j + 1;
            Select the path with the minimum function of the
  energy fault cost F(y*);
        else
            Deliver the path to set 𝒯_j;
        end if
    end for
    k = k + 1;
end for
Send the control commands for data transmission to the
  corresponding devices.
```

5.4. Time complexity analysis

To prove the efficiency of the EFCO algorithm, we analyse the time complexity of the EFCO algorithm below.

First, the EFCO algorithm calculates the node fault probability in accordance with the energy consumption requirements, and the procedure takes $O(\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor)\lg n$ time, where $\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor\lg n \leq n$. Second, it takes $O(\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor)$ time to derive the optimal energy fault cost $F(y^*)$. Finally, the EFCO algorithm needs to select the optimal transmission path with the minimum energy fault cost function $F(y^*)$. The transmission path needs to compare $F(y^*)$ reliable paths in every edge computing system, and there are $\mathscr{O}_{opt}$ edge computing systems in the ECWSNs; this process takes $O(\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor)\lg\mathscr{L}$ time. Then, the EFCO algorithm sends the operation results and requested packets by employing $O(\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor)$ time, and this step takes $O(\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor\lg\mathscr{L}+\mathscr{O}_{opt}\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor)$ time. Therefore, the total time complexity of the EFCO algorithm is as follows:

$$\begin{aligned} O \quad &(\mathscr{O}_{opt}\lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor + \lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor\lg n + \mathscr{O}_{opt}\lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor\lg\mathscr{L} \\ &+\mathscr{O}_{opt}\lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor) \\ &\leq O(3\mathscr{O}_{opt}\lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor\lg n + \lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor) \\ &\leq O(3n + \lfloor\tfrac{n}{\mathscr{O}_{opt}}\rfloor). \end{aligned} \tag{29}$$

Note that the complexity of the EFCO algorithm is linear in the number of perceptual nodes n and the number of edge computing systems $\mathscr{O}_{opt}$; the time complexity is much lower if the values of n and $\mathscr{O}_{opt}$ are much smaller. Therefore, the EFCO algorithm is time-efficient owing to the lower time complexity in ECWSNs.

6. Performance evaluation

In this section, we first describe the choice of network simulation software and the setting of simulation parameters. Then, we test the degree of fit between the simulation results and theoretical results of our algorithm. Finally, we compare the performance of our algorithm and with the performances of several benchmark algorithms in terms of the fault distribution ratio (FDR) and reliable transmission ratio (RTR).

6.1. Simulation environment and setting

To better verify the performance of the proposed EFCO algorithm, we must select appropriate simulation software to deploy the network and effectively predict the failure of wireless sensor nodes in the ECWSNs. NS-2 uses the split object model mechanism to separate the simulation script from the protocol implementation. The architecture is clear. The simulation script is written by Tcl, and the protocol is implemented by C++. Therefore, NS-2 has the advantages of a high C++ running speed and no compilation required for Tcl interpretation and execution(Hussain et al., 2020). Although version 3 of the network simulator (NS-3) has rich simulation resources and updates quickly in the physical layer, compared to 5G, millimetre wave technology, the Internet of Vehicles and other fields, it is easier for NS-2 to simulate the data transmission process in terms of the design of network construction and routing scheduling algorithms in ECWSNs. Therefore, we implement the EFCO algorithm by using NS-2 to evaluate its performance (Phillips and du Plessis, 2021).

Small-scale and large-scale simulation monitoring areas are set to 800 m by 800 m and 3000 m by 3000 m with 50 to 500 perceptual nodes, respectively. An ECS is responsible for fault analysis and energy allocation of $\lfloor\frac{n}{\mathscr{O}_{opt}}\rfloor$ nodes. The radius of each edge computing system is r. Assume that the remaining energy of the perceptual nodes is expressed by three statuses: 100%, 75%, and 50%. Table 2 lists the experimental parameter settings in the experiment.

6.2. Small-scale experiments

In this subsection, we use small-scale experiments to verify the network survival ratio (NSR) and fault reasoning error ratio (FRER) based on the different ECSs. In addition, the number of ECSs $\mathscr{O}_{opt}$ is determined by the network parameter settings and Theorem (2). The network distribution topology is shown in Fig. 3.

It is assumed that the ECWSNs generate energy holes as the experimental time increases. Energy holes are caused by dead sensor nodes that are exhausted. Fig. 4 shows that the node survival ratio in the simulation results essentially matches the theoretical results. This experiment proves that our EFCO algorithm is consistent with the actual network operation status.

To verify the correctness of Theorem (2) in deriving the number of ECSs, we use the simulation in Fig. 5 to prove the range of the FRER on the edge computing server ratio (ECSR) in ECWSNs. The FRER is defined as follows:

Table 2
Parameter settings.

Parameter	Description
Packet Transmission Rate	250 kbit/s
Timeout	864 μs
System Radius r	50 m
Communication Distance	30 m
τ	10 $pJ/bit/m^2$
φ	0.0013 $pJ/bit/m^4$
δ	1.6
ξ	1.5
Simulation Time	20 min

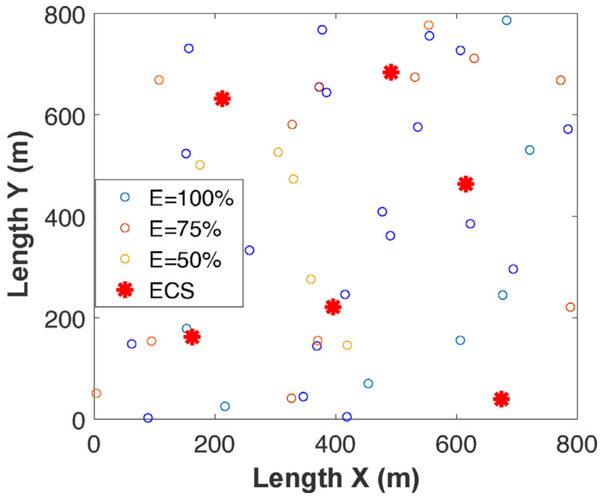


Fig. 3. Small-scale topology of ECWSNs.

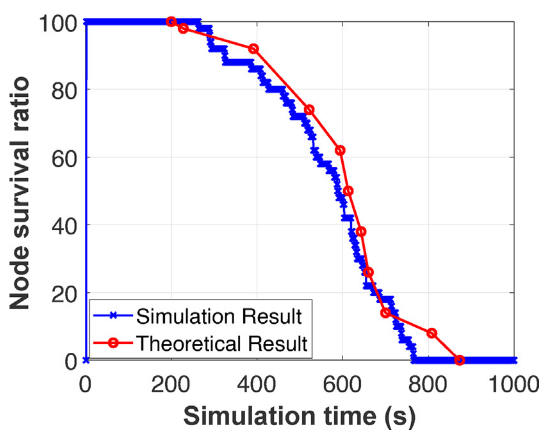


Fig. 4. Comparison of the node survival ratios.

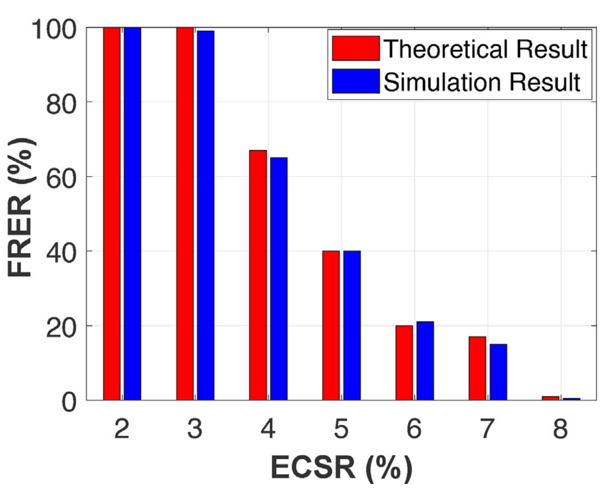


Fig. 5. Comparison of the fault reasoning error ratios.

Definition 2 (*Fault Reasoning Error Ratio*). The fault reasoning error ratio *FRER* is defined as the proportion of erroneous reasoning results to all results of the EFCO algorithm in the process of fault reasoning in ECWSNs.

The FRER reflects the efficiency of energy fault reasoning in ECWSNs. The smaller the FRER is, the lower the level of energy fault reasoning, and vice versa.

Employing an appropriate number of ECSs saves network resources and reduces the fault ratio of data transmission. Fig. 5 shows that the FRER decreases rapidly when the ECSR increases, and the FRER decreases to 0 when the ECSR increases to 8%. The reason is that this ECSR is sufficient to support data transmission in ECWSNs; a higher ECSR increases the network device costs, and a lower ECSR reduces the data transmission efficiency in ECWSNs. This ECSR is consistent with Theorem (2). Moreover, the simulation results are consistent with our theoretical results.

6.3. Comparison of the fault distribution ratio

In this subsection, we first define the FDR, and then, we compare the FDRs of our EFCO algorithm, the FCA (Dong et al., 2019) and the MSNS algorithm (Zhu et al., 2019) for the fault node ratio.

Definition 3 (*Fault Distribution Ratio*). The fault distribution ratio *FDR* is defined as the proportion of the sum of the absolute values of the probability distribution factor differences in different nodes to the total fault probability in the ECWSNs, i.e.,

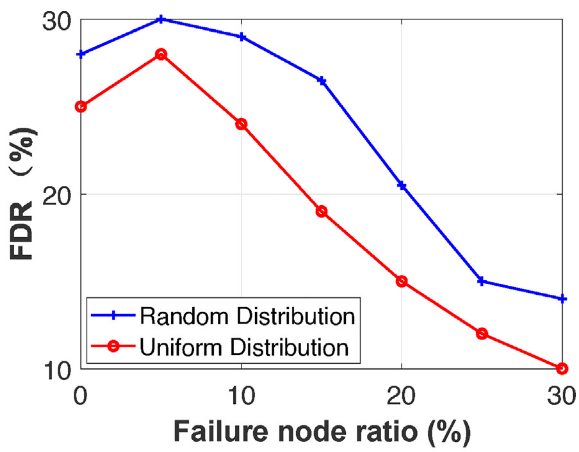
$$FDR = \frac{\sum_{i=1}^{\mathcal{N}_{opt}} \sum_{j=1}^{\lfloor \frac{n}{\mathcal{N}_{opt}} \rfloor} |\varphi(V_{ij}) - \varphi(V_{i(j+1)})|}{\sum_{i=1}^{n} \varphi(V_i)}. \tag{30}$$

$\lfloor \frac{n}{\mathcal{N}_{opt}} \rfloor$ denotes the optimal number of edge computing systems, and $\varphi(V_{ij})$ is the probability distribution factor of the i-th perception node in the j-th edge computing system.

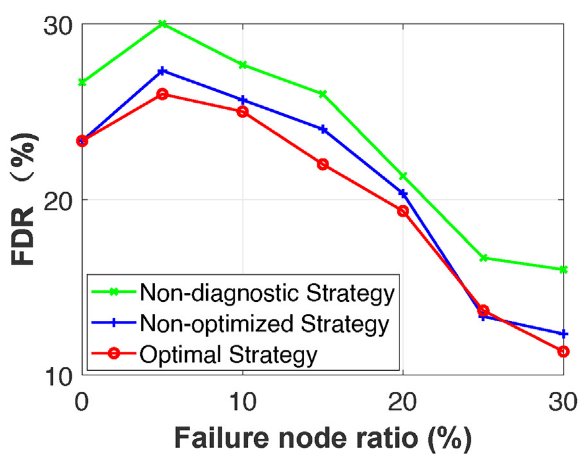
The FDR reflects the level of node fault detection in ECWSNs. The larger the FDR is, the smaller the level of node troubleshooting, and vice versa.

We divide the perception node topology into the two statuses of uniform distribution and random distribution and distribute the fault nodes with different proportions to the ECWSNs. A uniform distribution means that the perception nodes are evenly placed in the ECWSNs, and a random distribution has unevenness in node placement. Fig. 6(a) shows that the FDR of a random distribution is much higher than the FDR of a uniform distribution. The reason is that the uniformly distributed energy consumption is relatively balanced during the operation of the EFCO algorithm in ECWSNs, which makes the difference between the derivation results of the fault probability factor relatively small, and the randomly distributed perception nodes inevitably experience energy hole phenomena, which causes a large gap with respect to the energy consumption of perception nodes in ECWSNs. Therefore, this explains the relatively large difference in the fault probability factor.

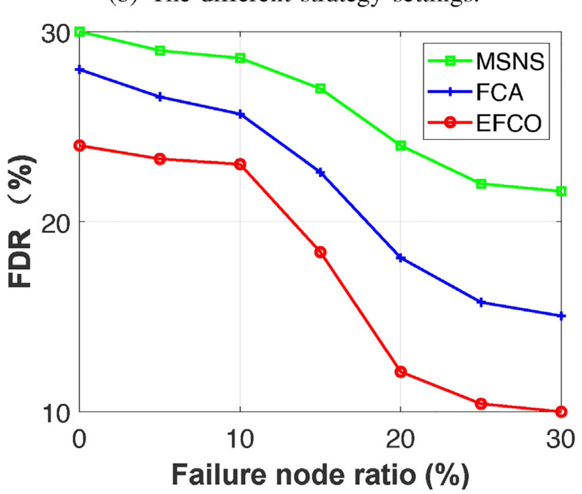
We define the EFCO algorithm that adopts Algorithm 1 and the energy fault cost minimization method as the *nondiagnostic strategy* and *nondeoptimized strategy*. Accordingly, the EFCO algorithm with the fault reasoning algorithm and energy fault cost function is the *optimal strategy*. Fig. 6(b) shows that the FDR of the optimal strategy becomes smaller than the FDR of the nondiagnostic and nonoptimized strategies as the fault node ratio increases. For this reason, the EFCO algorithm comprehensively considers the fault nodes existing in ECWSNs and the optimal fault cost. However, the nondiagnostic strategy may select the fault nodes as the relay nodes. Therefore, the transmission path becomes an invalid link. The nonoptimized strategy cannot select the optimal transmission path, which results in uneven energy consumption.



(a) The different node distribution statuses.



(b) The different strategy settings.



(c) The different routing algorithms.

Fig. 6. The fault distribution ratio (FDR) comparison.

Fig. 6(c) shows a comparison of the FDRs of the EFCO algorithm, FCA, and MSNS algorithm. Our EFCO algorithm has a lower FDR than the other two routing algorithms. This is because the EFCO algorithm can determine the faulty nodes and select the relay nodes with the optimal energy consumption. The FCA does not judge the faulty nodes in the ECWSNs, which causes the ECS to incorrectly select the relay nodes. Although the MSNS algorithm can use the ECS to select the data transmission link, it does not comprehensively consider the global energy optimization objective and the energy hole problem caused by the faulty perception nodes, which results in lower energy allocation efficiency in ECWSNs.

6.4. Comparison of the reliable transmission ratio

In this subsection, we adopt the RTR to evaluate the data scheduling efficiency of the EFCO algorithm in ECWSNs. The RTRs are compared by different distribution mechanisms and different strategies as the simulation time increases. Furthermore, we compare the RTRs of the EFCO algorithm and other baseline algorithms.

Definition 4 (*Reliable Transmission Ratio*). The reliable transmission ratio *RTR* is defined as the proportion of the successfully transmitted packets to the sum of the received packets in the ECWSNs, i.e.,

$$RTR = \frac{\sum_{i=1}^{\mathcal{N}_{opt}} \left(\mathcal{Y}_i + \sum_{j=1}^{\lfloor \frac{n}{\mathcal{N}_{opt}} \rfloor} \left(\mathcal{Y}_{ij} - \mathcal{Y}_{ij_{(i+1)}}\right)\right)}{\sum_{i=1}^{n} \mathcal{Y}_i}, \tag{31}$$

where $\mathcal{Y}_i$ indicates the forwarded packets in the i-th perception node.

We take the RTR as the metric for the transmission efficiency of packets in ECWSNs. The larger the RTR is, the higher the transmission efficiency, and vice versa.

Fig. 7(a) shows the RTR comparison between a uniform distribution and random distribution as the simulation time increases. The RTR of the uniform distribution is higher and more stable than the RTR of the random distribution. The more even the node distribution is, the more balanced the energy consumption in the ECWSNs, and the data transmission efficiency is higher given the lower probability of node faults. With the continuous operation of the EFCO algorithm in ECWSNs, energy holes appear in the random distribution method, which causes a large number of packet loss and retransmission problems during data processing. Therefore, the RTR of the random distribution exhibits jitter in the beginning. Through the node fault analysis of the EFCO algorithm, the data transmission efficiency is gradually improved and tends to stabilize with increasing simulation time.

In addition, we contrast the RTRs of the *nondiagnostic strategy*, *nonoptimized strategy*, and *optimal strategy* in ECWSNs with increasing simulation runtime. Fig. 7(b) shows that the optimal strategy has a higher RTR than the other strategies, and the RTR results of all strategies decrease at the beginning of the simulation. The reason is that the EFCO algorithm needs to analyse the faulty node and select a reliable transmission path according to the real-time status of the ECWSNs, which affects effective data transmission to a certain extent. In addition, the nondiagnostic strategy cannot effectively judge the fault status of the perception node, and the transmission path may be incorrectly selected. Therefore, the data transmission performance of the nondiagnostic strategy is worse than that of the nonoptimized strategy.

Fig. 7(c) compares the RTRs of the EFCO algorithm, FCA, and MSNS algorithm. The proposed EFCO algorithm has the best RTR results in this simulation. This is because the EFCO algorithm can accurately determine the faulty nodes through the strong computing power of the ECS in ECWSNs, and the EFCO algorithm can select the optimal transmission path according to the energy fault cost. The MSNS algorithm does not select the optimal transmission link

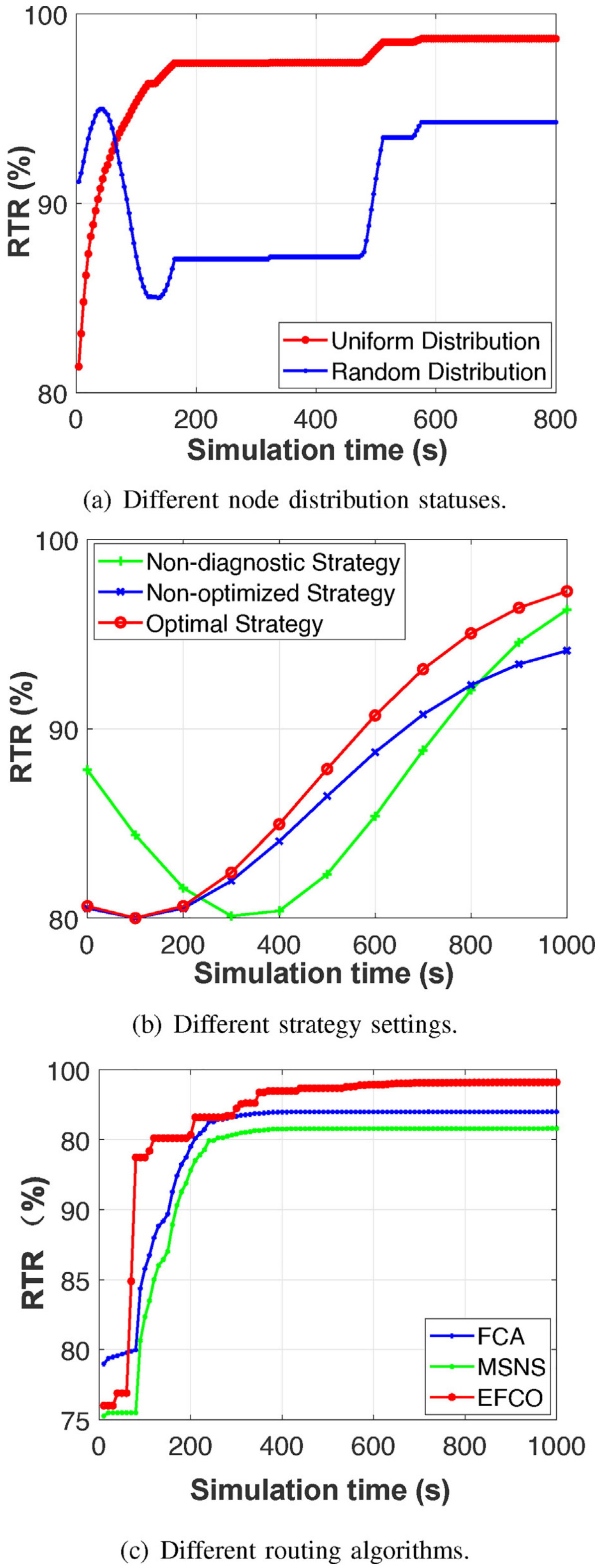


Fig. 7. Reliable transmission ratio (RTR) comparison.

according to the energy consumption status of the sensors, which results in uneven energy consumption by the ECWSNs and reduces the reliable data transmission efficiency. The FCA cannot accurately calculate the statuses of the faulty nodes, which blocks the FCA scheduling strategy.

7. Conclusion and future work

In wireless sensor networks, the main challenge of the EHCB problem is to select a reliable data transmission path based on analysing failed nodes and averaging the energy consumption of relay nodes. Therefore, we first build a novel four-tier architecture for ECWSNs by merging edge computing technology and the concept of causality graphs. Second, we propose the minimization function of the energy fault cost by using the formulated EHCB problem, which is constrained by the average energy allocation in ECWSNs. Then, the minimized value of the energy fault cost is determined by utilizing the Lagrangian dual decomposition approach. Furthermore, we propose an EFCO algorithm to solve the EHCB problem in ECWSNs. Finally, we implement the EFCO algorithm in NS-2 and evaluate its performance in terms of the fault distribution rate and reliable transmission ratio.

For future work, the plan is to find the optimal critical value of the predicted frequency of faulty nodes in the ECWSNs. If the predicted frequency of the fault node is greater than the critical value, the network status will be monitored in real time; otherwise, the fault wireless sensor node will not be effectively judged, and data packet loss or network congestion may occur. In addition, we seek and judge the effectiveness of the fault node judgement strategy of data transmission equalization, further optimize the energy consumption and prolong the life cycle in ECWSNs.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proof of theorem 1

The $p_i^{(e)}$ values hold by the proof in Sediyono et al. (2022):

$$p_i^{(e)} = 1 - e^{-\frac{1}{E_i^{(0)}/E_i^{(c)}} t_i}, \tag{32}$$

where $E_i^{(0)}$ and $E_i^{(c)}$ represent the initial energy and energy consumption of node i, respectively. t_i is the running time of node i.

We use the energy consumption model of wireless communication provided in Hu et al. (2016). The energy consumed by a node over a distance d for transmitting l bits of data is the sum of the energy consumed E_{tx} by the signal transmitting circuit and the signal amplifying circuit.

$$E_{tx} = E_{elec}.l + \varepsilon_{amp}.l.d^2 \tag{33}$$

The consumption energy E_{rx} of the node for receiving l bits of data is calculated by Eq. (34).

$$E_{rx} = E_{elec}.l \tag{34}$$

Then, the total energy consumption $E_i^{(c)}$ of node i can be expressed in the following form:

$$E_i^{(c)} = E_{tx} + E_{rx} = 2E_{elec}l + \varepsilon_{amp}ld^2. \tag{35}$$

Assuming that n nodes are evenly dispersed in the 2-dimensional bounded monitoring area G (area A), the probability that a node falls within the circular domain D with the communication distance d as the radius is as follows:

$$P = \int\int_D f(x,y)dxdy = \frac{\pi d^2}{A}. \tag{36}$$

In addition, the relationship between the node transmission distance d and the node degree $\tilde{k}$ is as follows:

$$k^{\cdot} = n.P = n.\pi d^2/A. \tag{37}$$

We can obtain the relationship between the energy value $E_i^{(c)}$ of node i and its degree $\tilde{k}$ by substituting Eq. (37) into Eq. (36) and then substituting Eq. (32) to obtain the energy depletion of probability node i described by node degree $\tilde{k}$ and running time t.

$$p_i^{(e)} = 1 - e^{-(a+b\tilde{k})t_i}, \tag{38}$$

where $a = \frac{2E_{elec}l}{E_i^{(0)}}$ and $b = \frac{\varepsilon_{amp}lA}{n\pi E_i^{(0)}}$.

Because the random fault of node $p_i^{(r)}$ is caused by environmental damage from the sparse network, the node is characterized by being isolated and lost in the operating environment of ECWSNs, and its probability has an exponential relationship with the node degree $\tilde{k}$ (Chu et al., 2019).

$$p_i^{(r)} = e^{-\tilde{k}_i} \tag{39}$$

We define the difference $\tilde{p}$ between the energy depletion probability $p_i^{(e)}$ and the random fault $p_i^{(r)}$ in the i-th node. Then, we calculate the first-order derivative of $\tilde{p}$ with respect to the node degree $\tilde{k}_i$.

$$\frac{\partial \tilde{p}}{\partial \tilde{k}_i} = (abt_i + \tilde{k}_i b^2 t_i^2)e^{-(a+b\tilde{k})t_i} + e^{-\tilde{k}_i} > 0 \tag{40}$$

Therefore, $p_i^{(e)}$ is a monotonically increasing function for $\tilde{k}_i$, i.e., $p_i^{(e)} > p_i^{(r)}$.p

This completes the proof.

Appendix B. Proof of theorem 2

The radius of the edge computing system is $r = \frac{D}{\sqrt{\pi \mathcal{O}}}$, with $\rho = \frac{1}{(\frac{D^2}{\mathcal{O}})}$ (Nakagaki et al., 2008); then, the energy consumption requirement for data transmission with transmission distance d is

$$\begin{aligned} E(d^2) &= \rho \int_{\theta=0}^{2\pi} \int_{r=0}^{\frac{D}{\sqrt{\pi\mathcal{O}}}} r^2 dr d\theta \\ &= \frac{\rho D^4}{2\pi \mathcal{O}^2} \\ &= \frac{D^2}{2\pi \mathcal{O}}. \end{aligned} \tag{41}$$

Thus, we can obtain Eq. (42) from Eq. (35) in Theorem (2):

$$E_i^{(c)} = l\bar{e} + l\tau e(d^2). \tag{42}$$

Then, the energy consumed by the ECWSNs is as follows:

$$\begin{aligned} E_{all} &= \mathcal{O}E \\ &= l(nE + n\varphi d^4 + nE + n\tau \frac{D^2}{2\pi\mathcal{O}}). \end{aligned} \tag{43}$$

We find partial derivatives for $\mathcal{O}$ and obtain the optimal number of edge computing systems.

$$\mathcal{O}_{opt} = \sqrt{\frac{n\tau}{2\pi\varphi}\frac{D}{d^2}} \tag{44}$$

This completes the proof.

References

Bruni, R., Gadducci, F., Montanari, U., 2002. Normal forms for algebras of connections. Theoret. Comput. Sci. 286 (2), 247–292.

Chiang, M., Low, S.H., Calderbank, A.R., Doyle, J.C., 2007. Layering as optimization decomposition: A mathematical theory of network architectures. Proc. IEEE 95 (1), 255–312.

Chu, Y., Champagne, B., Zhu, W.P., 2019. Noma-based cooperative relaying for secondary transmission in cognitive radio networks. IET Commun. 13 (12), 1840–1851.

Dong, Y., Guo, S., Liu, J., Yang, Y., 2019. Energy-efficient fair cooperation fog computing in mobile edge networks for smart city. IEEE Internet Things J. 6 (5), 7543–7554.

Guo, S., Liu, J., Yang, Y., Xiao, B., Li, Z., 2019. Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing. IEEE Trans. Mobile Comput. 18 (2), 319–333.

Hu, C., Li, H., Huo, Y., Xiang, T., Liao, X., 2016. Secure and efficient data communication protocol for wireless body area networks. IEEE Trans. Multi-Scale Comput. Syst. 2 (2), 94–107.

Hussain, M., Ren, J., Akram, A., 2020. Classification of dos attacks in wireless sensor network with artificial neural network. Int. J. Network Sec. 22 (3), 542–549.

Jain, V., Kumar, B., Gupta, A., 2022. Cybertwin-driven resource allocation using deep reinforcement learning in 6g-enabled edge environment. J. King Saud Univ.-Comput. Informat. Sci. 34 (8, Part B), 5708–5720.

Jalil Piran, M., Pham, Q.-V., Islam, S.R., Cho, S., Bae, B., Suh, D.Y., Han, Z., 2020. Multimedia communication over cognitive radio networks from qos/qoe perspective: A comprehensive survey. J. Network Comput. Appl. 172, 1–44.

Kamal, E., Adouane, L., 2018. Hierarchical energy optimization strategy and its integrated reliable battery fault management for hybrid hydraulic-electric vehicle. IEEE Trans. Vehicular Technol. 67 (5), 3740–3754.

Li, G., Xu, Y., 2019. Energy consumption averaging and minimization for the software defined wireless sensor networks with edge computing. IEEE Access 7 (2), 173086–173097.

Li, G., Guo, S., Yang, Y., Yang, Y., 2018. Traffic load minimization in software defined wireless sensor networks. IEEE Internet Things J. 5 (3), 1370–1378.

Liu, G., Guo, S., Zhao, Q., Yang, Y., 2018. Tomogravity space based traffic matrix estimation in data center networks. Future Generat. Comput. Syst. 86, 39–50.

Liu, A., Lau, V.K.N., Kananian, B., 2019. Stochastic successive convex approximation for non-convex constrained stochastic optimization. IEEE Trans. Signal Process. 67 (16), 4189–4203.

Ma, X., Wang, S., Zhang, S., Yang, P., Lin, C., Shen, X., 2021. Cost-efficient resource provisioning for dynamic requests in cloud assisted mobile edge computing. IEEE Trans. Cloud Comput. 9 (3), 968–980.

Nakagaki, T., Tero, A., Kobayashi, R., Onishi, I., Miyaji, T., 2008. Computational ability of cells based on cell dynamics and adaptability. New Generat. Comput. 27 (1), 57–81.

Phillips, A., du Plessis, M.C., 2021. Evolutionary robotics controllers with proprioception facilitated by neural-network-based simulators. Artif. Life 26 (4), 455–483.

Sediyono, E., Wibowo, F.W., Purnomo, H.D., 2022. Leader election of dynamic wireless intelligent control machine in sensor network distributed processing. J. King Saud Univ.- Comput. Informat. Sci. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157822003202.

Sl, A., Cc, A., Wq, A., Jm, A., Li, Y.A., Zla, B., 2021. Bi-layer optimal planning of rural distribution network based on kkt condition and big-m method. Energy Reports 7, 637–644.

Tubishat, M., Idris, N., Abushariah, M., 2021. Explicit aspects extraction in sentiment analysis using optimal rules combination. Future Generat. Comput. Syst. 114, 448–480.

Volpe, L., Ramis, R., Baton, S., Nicolai, P., Perez, F., Santos, J.J., Vauzour, B., Batani, D., Koenig, M., 2012. Three-dimensional simulations of cylindrical target implosion imaging using laser-driven proton source. IEEE Trans. Plasma Sci. 40 (4), 1131–1133.

Zeng, Y., Guo, S., Liu, G., 2019. Comprehensive link sharing avoidance and switch aggregation for software-defined data center networks. Future Generat. Comput. Syst. 91, 25–36.

Zhang, Q., 2012. Dynamic uncertain causality graph for knowledge representation and reasoning: Discrete dag cases. J. Comput. Sci. Technol. 27 (1), 1–23.

Zhang, Q., 2015. Dynamic uncertain causality graph for knowledge representation and probabilistic reasoning: Directed cyclic graph and joint probability distribution. IEEE Trans. Neural Networks Learn. Syst. 26 (7), 1503–1517.

Zhang, W., 2018. A distribution network fault training evaluation method based on isolation information matrix similarity. Power Syst. Protect. Control 46 (19), 158–164.

Zhang, Q., Dong, C., Cui, Y., Yang, Z., 2014. Dynamic uncertain causality graph for knowledge representation and probabilistic reasoning: Statistics base, matrix, and application. IEEE Trans. Neural Networks Learn. Syst. 25 (4), 645–663.

Zhou, X., Yang, X., Ma, J., Wang, K.I.-K., 2022. Energy-efficient smart routing based on link correlation mining for wireless edge computing in iot. IEEE Internet Things J. 9 (16), 14988–14997.

Zhu, A., Guo, S., Liu, B., Ma, M., Yao, J., Su, X., 2019. Adaptive multiservice heterogeneous network selection scheme in mobile edge computing. IEEE Internet Things J. 6 (4), 6862–6875.