

SDN-COR: An Efficient Network Coding Opportunistic Routing Method for Software-Defined Wireless Sensor Networks

Yifan Hu^{1,2*}, Xiqiang Hou¹, and Fuqiang Wang²

¹ School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China

² Shandong Computer Science Center and Shandong Provincial Key Laboratory of Computer Networks, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China
[e-mail: huyif12@163.com]

*Corresponding author: Yifan Hu

*Received December 10, 2023; revised April 4, 2024; accepted July 2, 2024;
published July 31, 2024*

Abstract

A Software-Defined Wireless Sensor Networks (SDWSNs) architecture is firstly proposed to address the issues of inflexible architecture strategies and low scalability of traditional WSNs in this article. The SDWSNs architecture involves the design of a software-defined sensor network model and a customized controller architecture, along with an analysis of the functionalities of each management module within the controller architecture. Secondly, to tackle limited energy problem of sensor nodes, a network coding opportunistic routing method (SDN-COR) is presented based on SDWSNs. This method incorporates considerations of coding opportunities, vertical distance, and remaining energy of nodes to design a metric for encoding opportunistic routing. By combining opportunistic forwarding mechanisms, candidate forwarding sets are selected and sorted based on priority to prioritize data transmission by higher-priority nodes. Simulation results indicate that, comparing with conventional methods, this approach achieves reduction in energy consumption by an average of 21.5%, improves network throughput by 24%, and extends network lifetime by 20%.

Keywords: Software-defined networking; Network coding; Opportunistic routing; Wireless sensor networks

1. Introduction

In recent years, wireless sensor networks (WSNs) have attracted considerable attention due to their self-organizing capability, low cost, and integration of communication, computation, as well as sensing capabilities within sensor nodes. WSNs have been extensively utilized in numerous applications, including natural disaster monitoring, rescue operations, smart agriculture, and marine ranching [1]. Wireless sensor networks play a crucial role in many application areas. They are utilized in fields such as environmental monitoring, agricultural engineering surveillance, Internet of Things [2], fault diagnosis [3], and smart grids. Typical sensor nodes comprise ordinary nodes, aggregation nodes, and management nodes. Ordinary nodes, with their limited battery, storage, and processing capabilities, necessitate coordination among nodes for the collective execution of information processing and storage tasks. Aggregation nodes, possessing superior storage and processing capabilities compared to other nodes, mainly connect to gateways to facilitate protocol interactions. Management nodes are tasked with overseeing sensor nodes and acquiring network information through them.

Nevertheless, the widespread deployment of WSNs has revealed several challenges:

In the conventional architecture, each node handles all functions ranging from the physical layer to the application layer, encompassing network control and data forwarding. This autonomous architecture becomes increasingly challenging to manage as the network expands and node distribution grows. This issue is especially pronounced in WSN applications necessitating low energy consumption and long-range communication. Consequently, architectural innovation for WSNs has emerged as a crucial research focus. Software-defined networking (SDN) technology separates the network architecture into the application layer, control layer, and underlying forwarding layer, thereby achieving decoupling of control and forwarding [4]-[5]. SDN offers a programmable northbound interface, enabling customization of network functions according to application requirements. Integrating this SDN architecture into WSNs can improve network resource utilization efficiency and facilitate centralized management of a substantial number of nodes. Software-defined sensor networks can diminish the hardware complexity of sensor nodes, streamline node designs, and reduce node costs, thereby proving highly advantageous for the development of large-scale, low-power WSNs.

The limited energy of sensor nodes, once depleted, is difficult to replenish and may dictate the lifetime of the WSNs. Implementing network coding techniques in WSNs has the potential to decrease node energy consumption. By merging data packets into coded packets and simultaneously forwarding multiple data streams, the number of node transmissions can be minimized, resulting in reduced energy consumption and an extended network lifetime. This aspect is especially crucial for energy-efficient WSNs. Moreover, network coding bolsters network resilience, facilitating efficient data recovery even amid data loss or node failures. Furthermore, network coding mitigates redundancy in data transmission. In contrast to traditional routing and forwarding mechanisms, which often entail substantial duplicate data forwarding, network coding considerably diminishes redundancy by encoding and forwarding data flows, thereby enhancing network capacity [6]-[7]. S. Katti *et al* [8] introduced a new architecture called COPE, which is a coding opportunity entity scheme specifically designed for Mesh networks. COPE marks the first introduction of network coding technology into wireless networks, aiming to improve the network's throughput via this technology. Despite this, the design principle of COPE is mainly based on the passive waiting for coding opportunities, which limits the full potential of network coding.

The defining feature of SDN is the separation of network control functions from forwarding devices. The forwarding devices in the data plane are relieved from control-related computations, handling only simple operations such as packet forwarding. SDN provides programmable interfaces to the application layer, enabling customized network services to satisfy distinct business requirements. By separating control from forwarding, SDN gains a holistic view of the network. Integrating SDN with network coding technology simplifies the identification process of network coding and introduces additional functionalities, making it one of the recent focal points of research.

In traditional wireless sensor networks, an innovative network coding-aware routing protocol designed by Parisa *et al.* [9], known as FNCR based on fuzzy logic, introduced a dimension to routing. This approach redefined routing metrics, enhanced the identification of coding opportunities, and refined the original coding conditions. However, the prioritization of nodes within the candidate forwarding set was achieved through self-organization, with each node broadcasting its packet information and listening to cached packet information from others. This necessitated the calculation of their own coding opportunities, thereby increasing computational and caching overheads and thus energy consumption of nodes. To avoid it, we utilize the strategy of separating control and forwarding in software-defined sensor networks. In a software-defined sensor network, the algorithm computation, priority determination, and execution instructions are centralized and executed by the controller. Nodes no longer compute their own metric values or calculate other nodes' metric values for priority ordering. Instead, they concentrate these tasks within the controller for centralized priority ordering, minimizing the exchange of data packet information between nodes.

The Hybrid Coding-Aware Routing protocol (HCAR) proposed by Su *et al.* [10] was akin to existing studies, utilizing coding opportunities solely as a routing metric to enhance the likelihood of data passing through encoded nodes. However, this results in an increased load on coding nodes, prematurely depleting their energy [11]. Therefore, in energy-constrained networks, we take into account not only the coding opportunity value of nodes as a routing metric but also the coding opportunistic routing metric based on the coding opportunities, vertical distance, and residual energy of forwarding candidates. The proposed method, named **SDN-COR** (Coded Opportunistic Routing based on Software Defined Networking), reduces data packet transmissions and precludes the selection of distant nodes with low residual energy as forwarding candidates.

This paper aims to provide integration of WSNs and SDN, and explores the application of network coding technology in this context. The main contributions are as follows:

- 1) Providing an optimized software-defined sensor network architecture. This architecture, by introducing a virtual layer and expanding the functionality modules of controllers, allows multiple controllers to work collaboratively. It provides a customized architecture for software-defined wireless sensor network controllers, effectively processing and transmitting data.

- 2) Proposing a novel network coding opportunistic routing method (SDN-COR). It aims to reduce the number of transmitted data packets and avoid choosing nodes with low energy or long distance. This method combines network coding routing metrics with optimized software-defined sensor network architecture, enhancing route path selection based on three key attributes: coding opportunity, vertical distance, and remaining energy of the nodes, thereby improving link quality.

- 3) Performing comprehensive simulation validations. The results demonstrate that the proposed method significantly improves network throughput and data transmission efficiency, and simultaneously extends the network lifetime while reducing energy consumption.

This paper is structured as follows: Section 2 reviews research related to SDWSNs and coding opportunity routing methods, providing a theoretical foundation and background for this study. Section 3 details the system architecture design and optimization strategies for the controller architecture. Section 4 introduces the proposed system model, covering the network model and energy consumption model, laying the foundation for understanding the proposed method. Section 5 elucidates the proposed SDN-COR method. Section 6 evaluates the performance of the proposed scheme through experiments. In conclusion, Section 7 summarizes the research findings and proposes future research direction.

2. Related Work

2.1 Software Defined Sensor Networks

Given the continuous growth in scale and complexity of WSNs, traditional WSNs are no longer able to achieve real-time control and manual maintenance. SDN addresses this limitation by facilitating centralized programmable control of the network, significantly simplifying network management. Furthermore, the open and programmable nature of SDN promotes enhanced integration with emerging communication technologies. Therefore, SDN represents an inevitable trajectory in the evolution of wireless sensor networks, as it enables their development toward larger scales, improved performance, and simplified management [12]-[13].

The application of SDN principles to structured hierarchical WSN has addressed several fundamental challenges. Anadiotis *et al.* [14] proposed a SDWSN architecture, termed SD-WISE, that utilizes the existing wireless sensor nodes' operating systems to virtualize network functions. This capability allows for the deployment of diverse network functions. Furthermore, the architecture ensures close interaction between the trusted hardware and software, ensuring sensor nodes operate in accordance with the contextual requirements and comply with authoritative remote authentication directives. Nguyen *et al.* [15] have developed SeArch, a novel security architecture tailored for SDN-based cloud IoT networks. This architecture integrates an array of intelligent network intrusion detection systems (NIDS) with collaborative capabilities. The architecture is characterized by a three-tiered deployment of IDS nodes, comprising Edge-IDS at the edge layer, Fog-IDS within the fog computing layer, and Cloud-IDS across the cloud layer. The objective is to facilitate effective collaboration among nodes across these disparate levels.

To streamline network management, Duan *et al.* [16] presented an advanced SDWSNs framework to bolster network reliability by segregating the control plane and data plane, utilizing an external controller to assign flow rules, which minimizes network coupling. However, this strategy exhibited limitations as it failed to incorporate any energy-aware algorithms. Azizi *et al.* [17] proposed a protocol enabling certain sensor nodes to function as auxiliary controllers for the entire network, thereby conserving network energy and extending its lifespan. Younus *et al.* [18] unveiled an energy-conscious multi-hop routing protocol for SDWSNs, optimizing routing by considering residual energy and distance. Cicioğlu *et al.* [19] utilized residual energy and hop count as metrics and applied the fuzzy Dijkstra algorithm within the controller to determine the optimal routing strategy, taking into account network QoS parameters like end-to-end delay and throughput, yet they did not tackle the optimization of network clustering. Kumar *et al.* [20] introduced an energy-optimized, multi-constrained sustainable routing model for SDWSNs, formulating a mixed-integer linear programming (MILP) model to optimize energy utilization based on network resources. Hajian *et al.* [21] introduced an innovative SDN architecture aimed at reducing load distribution and extending network longevity, consisting of components

including topology, base stations (BS), controller discovery, links, and virtual routing. They also proposed a new mechanism for load-balanced routing via SDN and virtualization.

The existing body of literature has established the feasibility of SDWSNs and proposed network architectures comprising the application layer, control layer, and data plane layer. Nonetheless, deploying a single controller in such architectures presents challenges with respect to network stability and performance assurance. To mitigate these issues, this study introduces a virtual layer into the prevailing SDWSNs architecture to surmount challenges associated with the concurrent issuance of instructions by multiple controllers, a singular controller making multiple decisions, and nodes receiving duplicate information. In light of the execution of coding-aware routing by controllers, this paper proposes a tailored SDWSNs controller architecture by expanding the controller's functional modules, thereby resolving the aforementioned challenges. Please refer to Section 3 for detailed explanations.

2.2 Encoding aware routing algorithm

The COPE coding scheme developed by Katti *et al.* [8] represents a significant milestone in the application of network coding techniques within WSNs. This approach fully exploits the broadcast nature of wireless networks, performing coding operations at the network's relay nodes, which effectively reduces the number of transmissions and thereby enhances the data throughput capacity of the entire network. Nevertheless, the COPE scheme has a clear limitation in that the routing metric it employs does not take into account the relevance of coding. To address this issue, He *et al.* [22] designed CARTA, a coding-aware routing scheme tailored for wireless sensor networks. CARTA evaluates coding conditions using a cost-effective method and utilizes a streamlined tree-based addressing structure to dynamically capture coding opportunities, thereby enhancing network performance.

Zhao *et al.* [23] tackled the rate-matching problem by assessing the output queue length and introduced the Constrained Coding-Aware Routing (CCAR) scheme, which enhances the probability of decoding by imposing constraints on coding conditions. Peng *et al.* [24] developed a novel coding-aware routing metric that proactively generates coding opportunities within the network, culminating in an augmented coding quantity, enhanced throughput, and diminished latency.

As expounded by Hai *et al.* [25], the assumption that paths with a higher number of coding opportunities are invariably optimal for data transmission is not always accurate. Guo *et al.* [26] introduced a novel routing metric along with a set of generalized coding conditions (GCC). Kok *et al.* [27] refined the coding conditions, termed as Improved GCC (IGCC), and established the Network Coding-Aware Routing (NCRT) protocol. This protocol integrates IGCC with a novel coding-aware routing metric that takes into account the node workload. NCRT demonstrates superior performance compared to the previous methods. Renugadevi [28] introduced a routing metric that contemplates both coding opportunities and network load, albeit without considering node energy depletion. Zhao *et al.* [29], to improve the performance of sparse underwater wireless sensor networks with respect to the efficiency and reliability of data transmission, proposed a Coding-Aware Opportunistic Routing Scheme (CORS). This scheme adaptively uses network topology information for selecting candidate nodes and introduces a forwarding strategy that cleverly integrates network coding and opportunistic routing. In the field of Cognitive Radio Social Internet of Things (CR-SIoT), Zhong *et al.* [30], have introduced an innovative energy-aware opportunistic routing algorithm with network coding aimed at enhancing overall network performance. The algorithm integrates energy conservation seamlessly with the characteristics of social networks, employing network coding, a novel routing metric, and an auction-based model to select the most appropriate data-

forwarding nodes. Utilizing this approach, it addresses the challenges associated with the dynamic utilization of spectrum and mobility of devices.

Current coding-aware routing approaches predominantly emphasize enhancing coding efficiency, often neglecting path optimization and node energy equilibrium. The pursuit of maximal coding opportunities does not ensure optimal routing performance, as energy depletion can result in transmission disruptions. Consequently, developing efficient network coding routing necessitates a holistic examination of various factors, including node energy consumption, transmission distance, and load balancing to circumvent bottleneck nodes. In conventional WSN coding-aware opportunistic routing, each node must compute its coding opportunity value and consistently broadcast its information and received packets to the central node. This distributed computation approach escalates node processing and storage overhead, expending substantial node energy. Thus, this paper introduces a novel network coding opportunistic routing method, termed SDN-COR, which seamlessly integrates with the SDN architecture. It capitalizes on the centralized control benefits of SDN, offloads complex coding calculations from nodes to controllers, and accomplishes task segregation. This method not only diminishes node processing and storage overhead but also optimizes data processing and transmission processes, surpassing the constraints of prior solutions.

3. Preliminaries

3.1 Software-Defined Sensor Network Architecture.

Traditional wireless sensor networks are beset by complexities in configuration, challenges in maintenance, and limited scalability. The SDN architecture, however, decouples the control plane from the data plane, facilitating easier management and dynamic optimization through self-adjusting capabilities. This evolution signifies a progressive direction for the next-generation of sensor networks. As depicted in [Fig. 1](#), the proposed SDN architecture for wireless sensor networks in this paper comprises three principal components: the control node, the central node, and the ordinary node. The control node, possessing a global view, exerts logical control across the entire network, managing network resources, coordinating sensor nodes, and scheduling data flows. Central nodes primarily perform flow table-based forwarding within the data plane, executing routing and forwarding decisions in accordance with the controller's instructions and policies. Ordinary nodes are responsible for receiving data and executing actions. Each node in the data plane maintains an abstract flow table model. The flow tables are configured with match and action rules derived from the control plane's commands. Upon receiving a data packet, a node consults its flow table to sequentially match the rules. Through the configuration and orchestration of flow tables, the specific networking behaviors of nodes are defined.

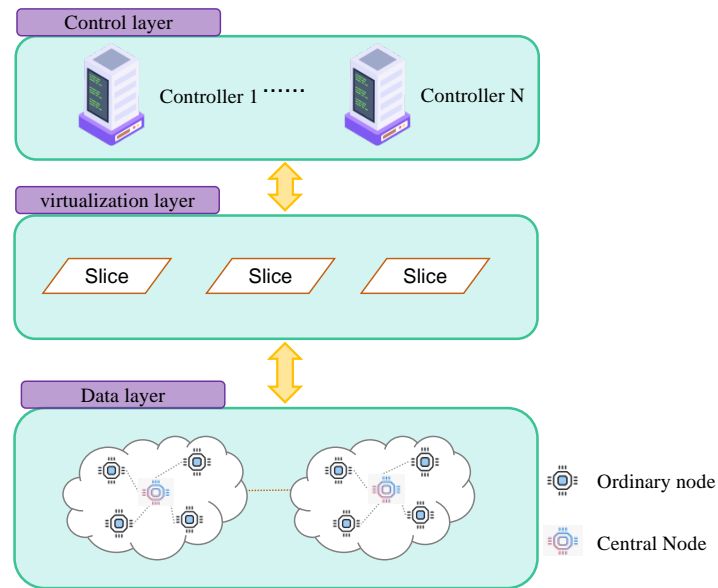


Fig. 1. SDWSNs Architecture

To mitigate the risk of a single point of failure, multiple controllers may be deployed within the control layer to manage the data plane in a distributed fashion, thus providing controller redundancy. Should individual controllers fail, other controllers can swiftly assume control of their respective zones, thereby ensuring network reliability and resilience against crashes stemming from single-point failures, consequently enhancing the high availability and stability of SDN-based sensor networks. However, the deployment of multiple simultaneous controllers may also precipitate command conflicts and decision ambiguities. To tackle this issue, we propose the introduction of a Flow Visor between the control and data layers, as illustrated in [Fig. 1](#). The virtualization layer employs multiple independent logical slices to prevent interference among the various controllers. Each logical slice operates independently, with a single controller potentially corresponding to multiple applications, and multiple controllers being capable of managing different switches. This architecture prevents controller conflicts and also addresses issues related to nodes receiving duplicate neighbor information.

3.2 Software-Defined Sensor Network Controller Architecture.

Within the architecture of SDWSNs, the controller is the pivotal component that coordinates operations across the entire sensor network through resource, network, and security management. The controller schedules sensor node resources, manages the quality and reliability of sensory data, ensures the confidentiality and integrity of sensory data, and supports higher-level applications. Customized and configurable network management is achievable by designing application-specific modular functions within the control layer. Considering the necessity for the SDWSNs controller to execute coding-aware routing algorithms, this study extends the controller's functional modules and proposes a tailored SDWSNs controller architecture, as illustrated in [Fig. 2](#). This architecture integrates a network coding management module within the controller to facilitate support for network coding algorithms. Such modular design endows the controller with the capability to orchestrate network coding techniques. It intelligently schedules routing based on network conditions, achieving efficient and reliable data transmission while maximizing network lifetime. The controller, endowed with built-in coding capabilities, offers a flexible and adaptable solution

for managing quality of service in SDWSNs.

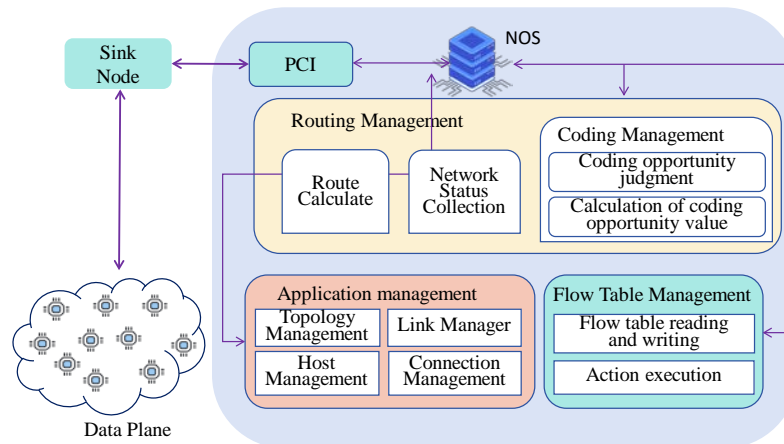


Fig. 2. SDWSNs Controller Architecture

The Network Operating System (NOS) performs network service functions, encompassing routing management, application management, flow table management, and PCI. The PCI application manages the communication between receivers and the controller.

The application management module, a critical component, comprises functions such as topology, host, link, and connection management. In this study, we primarily employ the topology management function to ascertain and uphold the connectivity of network links, that is, the connections between nodes and between nodes and the controller.

Within the routing management module, we utilize the network status collection, coding management, and routing calculation function modules. This module gathers network status information, including the link quality and node status, and forwards the gathered parameters to both the coding management and routing calculation modules. The routing calculation module interacts with the network topology, coding management, network status collection, and flow table management modules. Specifically, the network topology module identifies the global network topology and transmits the topology information to the routing calculation module. The network status collection module conveys link and node information to both the coding management and routing calculation modules. The coding management module transmits node coding status to the routing calculation module. Within the network coding management module, the coding opportunities for nodes are evaluated, and the corresponding coding opportunity values are computed.

The flow table constitutes the central unit that governs actions pertaining to network data flows. To facilitate network coding capabilities, we have extended the structure of the flow table as depicted in [Fig. 3](#), specifically by incorporating action items for encoding and decoding to implement network coding techniques within SDWSNs. The flow table management module configures the nodes in accordance with the flow table entries prescribed by the controller and carries out the corresponding actions for data flows.

packet header	Import interface	Ethernet source address	Ethernet Destination Address	Vlan ID	Encoding queue	Package length
Counter						
Action	Forward	Discard	Join team	Code	Decode	

Fig. 3. Expanded Flow Table Structure

4. System Models

4.1 Network Model

The system model used in this paper is a multi-hop wireless network composed of multiple nodes, where the source nodes are denoted as $S = \{S_1, S_2, \dots, S_n\}$ and the destination nodes as $D = \{D_1, D_2, \dots, D_m\}$. Apart from the source and destination nodes, there are intermediate/relay nodes denoted as $R = \{R_t \mid t \in \{1, 2, 3, \dots, w\}\}$, as shown in Fig. 4.

The source node sends original data packets $P = \{P_1, P_2, \dots, P_n\}$ to the destination nodes. There exist multiple data flows $f = \{f_i \mid i \in \{1, 2, 3, \dots, n\}\}$ in the network. Cross nodes, such as node R2, perform network coding on the original packets from different data flows and then broadcast the generated coded packets to downstream nodes. The downstream nodes decode the coded packets using their corresponding buffered packets and obtain the desired original packets. As shown in Fig. 4, let's consider two data flows: $f_1: S_1 \rightarrow R_1 \rightarrow R_2 \rightarrow D_1$ and $f_2: S_2 \rightarrow R_8 \rightarrow R_7 \rightarrow R_2 \rightarrow D_2$. The source nodes S_1 and S_2 send original packets P_1 and P_2 , respectively. The intermediate node R_2 receives both original packets and encodes them into coded packet $P_1 \oplus P_2$, which is then broadcasted to the downstream nodes $\{D_1, D_2, R_3\}$. The downstream nodes utilize their buffered packets to assist in decoding and retrieve the desired original packets.

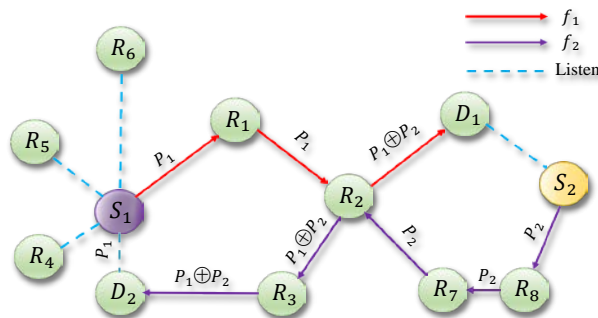


Fig. 4. Network Model

4.2 Energy Consumption Model

In this study, we adopted a simplified energy model to calculate the energy consumption of packet transmission and reception [31]. The model considers both free-space transmission (ϵ_{fs}) and multi-path fading channel (ϵ_{mp}). The distance is determined by calculating the distance between the sender and receiver using the distance formula (4). The transmission energy required to transmit a k-bit data packet over distance d can be calculated using Eq. (1).

$$\begin{aligned}
E_{Tx}(k, d) &= E_{Tx-elec}(k) + E_{Tx-amp}(k, d) \\
&= \begin{cases} k \cdot E_{elec} + k \cdot \varepsilon_{fs} d^2, & d < d_* \\ k \cdot E_{elec} + k \cdot \varepsilon_{mp} d^4, & d \geq d_* \end{cases} \quad (1)
\end{aligned}$$

where d_* represents the distance threshold.

The reception energy required to receive a k-bit data packet over distance d can be calculated using Eq. (2).

$$\begin{aligned}
E_{Rx}(k) &= E_{Rx-elec}(k) \\
&= k \cdot E_{elec} \quad (2)
\end{aligned}$$

where E_{elec} represents the unit energy consumed for operating the transmitter or receiver circuitry. Parameter ε_{mp} and ε_{fs} represent model of amplifier used for transmission. d_* is transmission distance threshold calculated using Eq. (3), k denotes the length of the data packet.

$$d_* = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \quad (3)$$

The Euclidean distance Eq. (4) is used to calculate distance of each node to the central node.

$$d_{i,j} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4)$$

5. Coding Opportunistic Routing Method Based on SDWSNs

The opportunistic coding-aware routing strategy presented in this study integrates network coding (NC) with opportunistic routing (OR), capitalizing on the inherent broadcast capability of wireless networks and designing a coding mechanism to facilitate seamless integration of NC and OR. To reduce the number of packet transmissions and enhance link quality, a novel routing metric has been established to improve transmission efficiency. The SDN-COR method comprises four main components: evaluating encoding opportunities, constructing the forwarding candidate set, establishing priority ordering, and configuring the forwarding countdown.

5.1 Coding Opportunity Determination

To minimize energy consumption, this study introduces a method that regulates the coding opportunities for network coding nodes to reduce data volume. The term "coding opportunity value" denotes the quantity of original data packets encompassed within a coded packet. Determining a node's coding opportunity necessitates verification of the node's capacity to execute coding. A node's coding capacity hinges on the ability of its next-hop neighbor to cache the coded data packet for decoding. To facilitate this, identifiers of the sending node and the next-hop neighbors are incorporated into the header of each data packet.

For instance, take the network depicted in Fig. 4. When node S transmits data packet P, its header will contain the identifiers of node S and the next-hop neighbors R, R1, R5, R6, D2, as illustrated in Fig. 5. Likewise, when S2 dispatches data packet P2, the header of P2 will comprise the identifiers of node S2 and the next-hop neighbors D1, R3. When P1 and P2 converge at node R2, the packet containing P1 includes the destination node D2 of P2, signifying that D2 has stored the requisite information to decode the P data packet. Upon receiving $(P1 \oplus P2)$, node D2 can retrieve data packet P2 by decoding $P1 \oplus (P1 \oplus P2)$. In a parallel manner, node D1 can retrieve data packet P1. Consequently, node R2 is able to encode both P1 and P2. By incorporating identifiers in the data packet headers, it becomes feasible to ascertain if the destination node has already stored the required information for decoding prior to coding, thereby establishing a node's capacity to execute coding.

Data packet	Packet holders	Data packet	Packet holders
P ₁	S ₁ , R ₁ , R ₄ , R ₅ , R ₆ , D ₂	P ₂	S ₂ , R ₈ , D ₁

Fig. 5. Additional ID Information for Packets P1 and P2

5.2 Constructing a Set of Forwarding Nodes

Table 1. Algorithm for determining candidate node sets

Input: $N(s)$, n // The set of neighboring nodes and the number of candidate nodes specified.

Output: The candidate node set C of node s .

```

C = [] // Initialize the candidate node set C as an empty set.
while (|N(s)| > 0 && |C| < n) {
    v = selectMinETXNode(Ns)
    if (ETX(v, d) < ETX(s, d) && for all r in C, ETX(v, d) < ETX(r, d)) {
        Ns = Ns - {v} // Remove the node v from the set of neighboring nodes Ns.
        C = C ∪ {v} // Add the node v to the candidate node set C.
    } else {
        break
    }
}
return C

```

During the network initialization process, each node broadcasts probe packets to obtain a set of neighboring nodes and gather topology information (such as link quality, signal strength, etc.) among the neighboring nodes. Using the Floyd algorithm, each node calculates the expected number of transmissions (ETX) to reach other nodes in the network, thus completing the network initialization. Before sending data, the sender node s selects multiple nodes from its set of neighbors to form a candidate node set C . The addresses of these candidate nodes are added to the header of the data packet, which is then broadcasted. In this protocol, when selecting a node v as a candidate node from its neighbor set $N(s)$, node s must satisfy the following conditions:

- (1) Node v is closer to the destination node d than node s , i.e., $ETX(v, d) < ETX(s, d)$.
- (2) Node v is able to listen to other nodes in the candidate forwarding set.
- (3) The number of candidate nodes should not exceed n , i.e., $|C| < n$.

The algorithm for constructing the forwarding node set is shown in **Table 1**.

5.3 Encoding-Aware Routing Metric

During each data transmission, nodes may transmit data to a designated convergence point via various potential routes. Priority rules established in the flow tables direct the specific routing strategies for data streams. The opportunistic routing method presented in this study, considers three key factors: node residual energy, the vertical distance from the convergence point, and coding opportunities, all with the aim of enhancing the communication link quality.

5.3.1 Encoding Opportunity

In conventional WSNs, it is required for each node in the coding opportunistic routing to compute its own coding opportunities and to broadcast its information periodically to the respective sink node. This approach not only escalates the computational and storage overhead of nodes but also imposes an additional burden on the sink node to derive opportunity values for

all network nodes. This distributed computing model results in excessive energy consumption by nodes. In SDWSNs, the computation of coding opportunities is centralized at the controller. The controller computes the values of coding opportunities across the network based on the data packets received and information reported by the sink nodes. This centralized computation methodology reduces the burden on nodes and conserves their energy. **Table 2** delineates the process of computing the value of coding opportunities for nodes. In this study, we initially assume that the forwarding set comprises two nodes, x and y . Upon receiving a data packet P within the network, the process to compute the value of coding opportunities for a node, as represented in pseudo code, is as follows:

Table 2. Algorithm for finding the encoding opportunity value

Input: p , report of node x ;	//Control node receives packet p and node x 's information
Output: count (x);	//The encoding opportunity value of node x
Procedure;	
count = 1; // Initialize encoding opportunity value	
While(node x output queue is not empty)	
{ for ($i=1$; $i \leq$ size of node x output queue; $i++$) // Traverse the output queue of node x	
{ // Determine coding opportunities	
if(destination ID of packet P is in packet holders of p_i && destination ID of p_i is in packet holders of P)	
{ Pcode = $P(p, p_i)$; // Encode to obtain the encoding package Pcode	
count++; // Update encoding opportunity value	
destination ID of Pcode = (destination ID of P) \cup (destination ID of p_i);	
packet holders of Pcode = (packet holders of P) \cap (packet holders of p_i);	
Store Pcode to buffer; // Store encoding package in buffer }	
Remove p_i from output queue; // Remove p_i from the output queue of node x	
$P = Pcode$; // Update data package P to encoding package Pcode }	
Return count(x); // Returns the encoding opportunity value of node x	

5.3.2 Vertical Distance

The existing Geographic Routing protocol leverages the fact that as the node density increases, the shortest path between source and destination nodes becomes closer to a straight line. Based on this observation, it is concluded that nodes with shorter perpendicular distances to this virtual straight line are more suitable to be selected as forwarding nodes. This distribution aims to achieve the shortest routing distance by assigning higher priority to nodes closer to the centerline, as shown in **Fig. 6**.

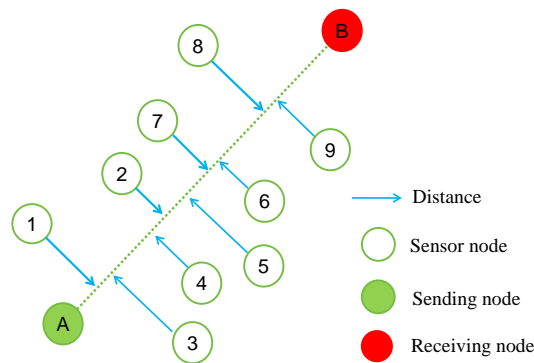


Fig. 6. Vertical Distance Schematic Diagram

The positions of the source node (x_s, y_s) and the destination node (x_b, y_b) are recorded within the data packet's header. Eq. (5) calculates the perpendicular distance ψ_j for a node n_i relative to a virtual straight line $l_{s,b}$. The random variable $\psi_i = (\psi_{i,1}, \psi_{i,2}, \dots, \psi_{i,m})$ quantifies the normalized perpendicular distance of the sending node n_j from its adjacent nodes, where $1 \leq \psi_{i,j} \leq 2$, using Eq. (6). Furthermore, the distribution of perpendicular distances between the sending node and its neighboring nodes is defined by the quality function Eq. (7). In the quality function Eq. (7), the exponential parameter $\gamma_\psi \geq 0$, called the vertical distance distribution control parameter, determines the influence of the perpendicular distance distribution on the forwarding probability.

$$\psi_j = \frac{|x_j(y_b - y_s) - y_j(x_b - x_s) + x_b y_s - y_b x_s|}{\sqrt{(x_b - x_s)^2 + (y_b - y_s)^2}} \quad (5)$$

$$\bar{\psi}_{i,j} = 1 + \left(\frac{\psi_j}{\delta + \psi_i} \right) \forall n_j \in \mathbb{N}_i \quad (6)$$

$$\tilde{\psi}_{i,j} = \frac{e^{-((\bar{\psi}_{i,j})^{\gamma_\psi})}}{\sum_{v=0}^{m_i} e^{-((\bar{\psi}_{i,v})^{\gamma_\psi})}} n_j \in \mathbb{N}_i, \gamma_\psi \geq 0 \quad (7)$$

5.3.3 Remaining Energy

In the design of packet forwarding strategies, priority is given to nodes with greater remaining battery capacity to achieve balanced energy consumption. However, the effectiveness of this approach depends on the nodes' deployment, and, at times, nodes with lower battery levels may be compelled to function as relay nodes. The aim of this strategy is to prioritize the selection of nodes with sufficient power in the course of data transmission.

Eq. (8) defines the energy-normalized random variable $\bar{\Phi}_i$ for the transmitting node n_i , under the assumption that all nodes possess identical initial energy, denoted by e_* . Φ_j represents the remaining energy of the node n_j . The distribution of the remaining energy for the transmitting node is governed by Eq. (9). This distribution aims to prioritize the selection of nodes with higher residual energy. The variable γ_Φ , which is non-negative, influences the selection process; the higher its value, the more likely that nodes with greater residual energy will be selected for transmission.

$$\bar{\Phi}_{i,j} = 1 + \left(\frac{\Phi_j}{e_*} \right), \forall n_j \in \mathbb{N}_i \quad (8)$$

$$\tilde{\Phi}_{i,j} = \frac{e^{(\bar{\Phi}_{i,j})^{\gamma_\Phi}}}{\sum_{v=0}^{m_i} e^{(\bar{\Phi}_{i,v})^{\gamma_\Phi}}}, n_j \in \mathbb{N}_i; \gamma_\Phi \geq 0 \quad (9)$$

5.4 Set Forwarding Countdown

The forwarding delay $t = \Delta t / RM$, where Δt is time coefficient that depends on specific network environment, and RM is the node forwarding priority metric. Thus, nodes with higher forwarding priority will have shorter forwarding delays. Specifically, when a candidate node receives the forwarding data packet from the source node, it sets a forwarding timer based on its own forwarding priority. During the forwarding waiting period, if a packet forwarded by a higher priority node is detected, forwarding is abandoned; if no detection occurs before the deadline, the node forwards the data packet. This ensures nodes strictly forward data packets in order of forwarding priority, and node collaboration avoids transmission failures. Additionally, ACK confirmation information is added to data packets. If source node does not receive an ACK within the time slot, it retransmits the data packet, further improving reliability.

5.5 Summary of This Chapter

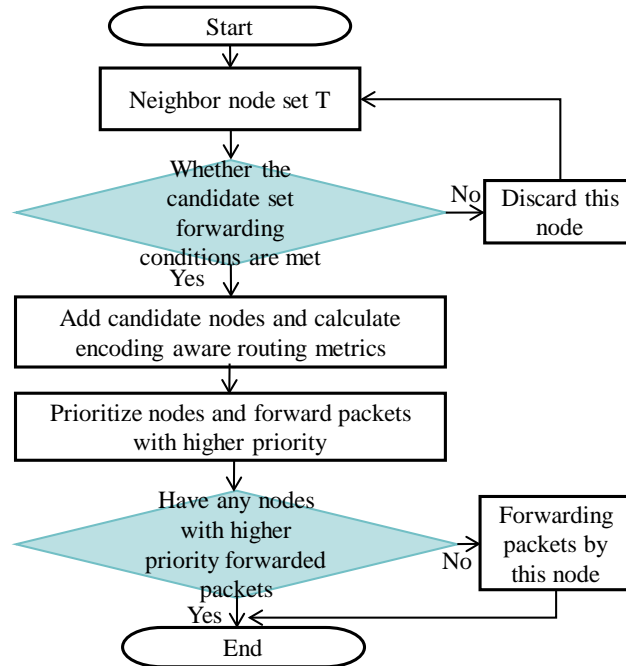


Fig. 7. SDN-COR Process Diagram

In Wireless Sensor Networks (WSNs), nodes possess limited energy. Reducing network energy consumption while maintaining a balance in node energy can be accomplished through appropriate routing strategies. This study examines the challenges associated with conventional coding opportunistic routing in WSNs and proposes a novel method for coding opportunistic routing, taking into account the limited energy of nodes. This algorithm considers factors such as the value of coding opportunities for nodes, vertical distance, and remaining energy. It integrates these factors into the opportunistic forwarding mechanism to identify and prioritize the candidate set of nodes, ensuring higher priority nodes forward data before others. In order to prevent redundant packet transmissions, varying packet forwarding delays are assigned according to the priority of node forwarding. The entire algorithm process is depicted in [Fig. 7](#).

6. Performance Evaluation of the Protocol

6.1 Simulation Scenario

In this research, a simulator developed using Visual Studio 2022 (C# WPF) [32] was utilized to conduct tests and evaluations on the proposed protocol SDN-COR. Specifically, detailed analyses of the protocol's performance were conducted through simulation experiments. To ensure broad applicability of the simulation results, the environment was designed within a square monitoring area, with nodes and convergence points randomly positioned at its center. The setup included one receiver and two controllers. Sensor node quantity ranged from 100 to 300. During the simulation, nodes followed a fixed activity cycle, alternating between 1 s of

activity and a 2 *s* sleeping period. Regarding energy consumption, each node had a battery capacity of 0.5 *J*, consuming energy as the model outlined in Section 4.2. To enhance result reliability and accuracy, 20 independent experiments were conducted for each parameter set, averaging these outcomes as the final result. Detailed parameters of the simulation are presented in **Table 3**.

Table 3. Simulation parameter settings

Parameter	Value
Number of nodes	Varies from 100 to 300
Communication range	Varies from 50 <i>m</i> to 100 <i>m</i>
Active time	1 <i>s</i>
Sleep time	2 <i>s</i>
Packet rate	1/0.1 <i>s</i>
Simulation time	300 <i>s</i>
Packet size	128 bytes

6.2 Evaluation Metrics

A set of evaluation metrics has been defined to assess the performance of the proposed method, and it has been compared with two existing technologies, namely, SRTRBC [33], a machine learning-based wireless sensor network routing approach, and BON [34], an opportunistic network coding algorithm for traditional networks.

Energy Consumption (EC): The energy consumption for data packets sent from source node and successfully transmitted to target node.

Average Redundant Packet Number (AxRP): The average number of candidate nodes that a packet is forwarded to.

Data Delivery Ratio (DDR): The ratio of the number of data packets successfully delivered to the number sent.

Data Loss Ratio (DLR): The ratio of number of packets lost to number of data groups sent.

Throughput (TP): The total amount of data sent through network within given time.

Average Route Distance Efficiency (RDE): The ratio between the actual distance from each source node to the destination and the theoretical shortest distance.

Average Waiting Time (AWT): The average time for a candidate to wake up and receive the data packet.

6.3 Results

The network performance evaluation is simulated in tests 1 to 4.

6.3.1 Test 1- Changing the Communication Range

The simulation duration is 300 seconds. The communication radius is set to range from 60 to 100 meters. The results are presented in **Fig. 8**.

Fig. 8a provides a comparative analysis of the energy consumption between the method proposed in this paper and the existing methods SRTRBC and BON. It can be observed that with the increase in the communication range of nodes, the overall energy consumption decreases for all three methods. The SDN-COR method achieves better results through intelligent routing selection, effective application of network coding technology, centralized control, and adaptive forwarding mechanisms. Compared to SRTRBC and BON, it has respectively improved by 17% and 26%.

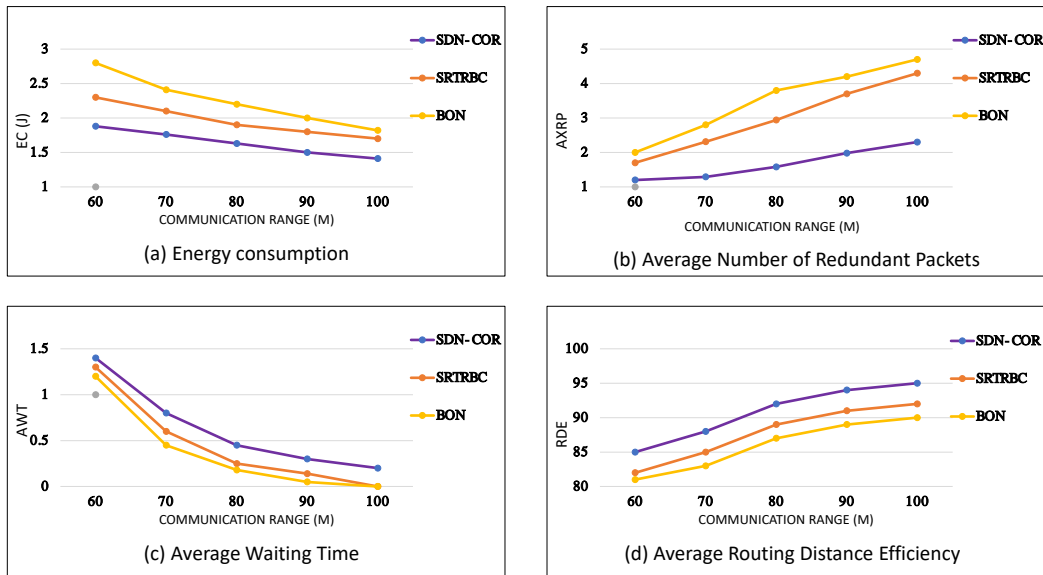


Fig. 8. Performance over Communication Ranges

Fig. 8b shows that as the communication distance increases, the number of redundant data packets in the network increases. When the communication range expands from 60 meters to 100 meters, nodes can choose more neighboring nodes as candidate nodes for data forwarding, resulting in an increase in the number of redundant packets in the entire network. Compared with SRTRBC and BON, AxRP in SDN-COR achieved better results within the communication range because it generated fewer redundant data packets.

Fig. 8c demonstrates that, within the communication range, the Average Waiting Time (AWT) for all three methods achieved satisfactory outcomes, with the SDN-COR method outperforming the others.

Fig. 8d demonstrates the performance of ROE within the communication range. SDN-COR performed significantly better than SRTRBC and BON, as data packets were guided to the sender node through the shortest distance node, which was due to the consideration of vertical distance distribution in the proposed method, while SRTRBC and BON did not consider the transmission distance.

6.3.2 Test 2- Changing the Number of Nodes in the Network

The scenario involved 100 to 300 nodes with a simulation time of 200 seconds and a communication distance set to 100m. The improvement results of various performance indicators are depicted in Figs. 9a-9d.

Fig. 9a illustrates the relationship between the increase in node count and the corresponding rise in energy consumption. As the network scale increases, the complexity of communication between nodes also rises, since each node needs to interact with more candidate nodes. This expansion leads to redundancy in data packet transmission, thereby increasing the overall energy consumption. The experimental results show that the proposed method performs excellently in handling this situation, managing communication and data transfer between nodes more effectively than the benchmark method, thereby reducing energy consumption and enhancing network performance. This indicates that in larger scale networks, adopting this method can better balance the relationship between communication needs among nodes and energy consumption, providing strong support for network sustainable development.

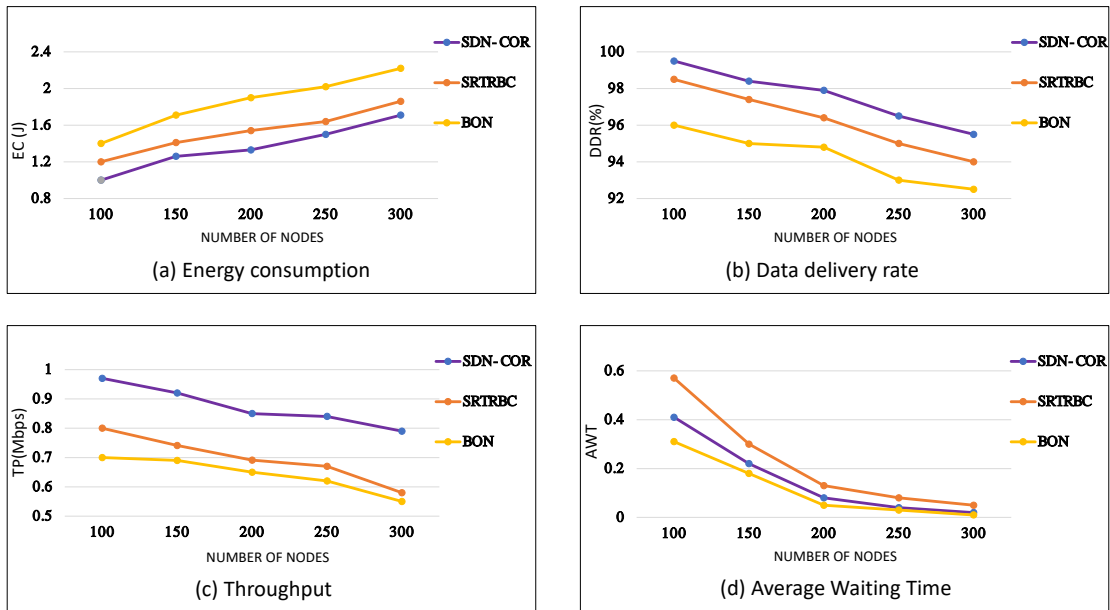


Fig. 9. Performance over number of nodes

Fig. 9b demonstrates a comparative analysis between the proposed DDR method and existing methods SRTRBC and BON in terms of DDR. From the graphical representation, it can be observed that DDR outperforms the other methods.

Fig. 9c displays a comparison analysis of throughput between the proposed method and existing methods SRTRBC and BON. The graphical representation shows that SDN-COR achieves approximately 20% and 28% higher throughput than SRTRBC and BON, respectively.

Fig. 9d illustrates the comparison of Average Waiting Time (AWT) for the three evaluated methods across a range of node counts. As the node count increases, the AWT for each method decreases owing to the increased availability of potential relay nodes for each node. The study reveals that, across all protocols tested, the AWT remains within acceptable limits as the node count increases. In this context, SDN-COR achieves not only lower waiting times but also a reduction in data packet redundancy.

6.3.3 Test 3- Changing the Active Time Interval of Nodes

A test was conducted to assess the impact of duty cycle variations on network performance, with node wake-up intervals ranging from 1 to 5 seconds, and a uniform sleep duration of 2 seconds for each node. The simulation duration was fixed at 480 seconds, involving 100 nodes and a communication range of 75 meters.

Fig. 10a illustrates the relationship between wake-up intervals and energy consumption. The primary reason that extending the wake-up interval increases energy consumption is that nodes remain active for longer periods, necessitating greater energy expenditure to maintain continuous communication and data transmission. Compared to traditional methods, SDN-COR demonstrates superior performance in managing the relationship between wake-up intervals and energy consumption. SDN-COR can more efficiently manage communication needs between nodes and reduce the generation of redundant data packets, thereby lowering energy consumption. By optimizing data transmission and communication processes, SDN-COR can better balance the relationship between node activity and energy utilization, enhancing network performance while reducing overall energy consumption.

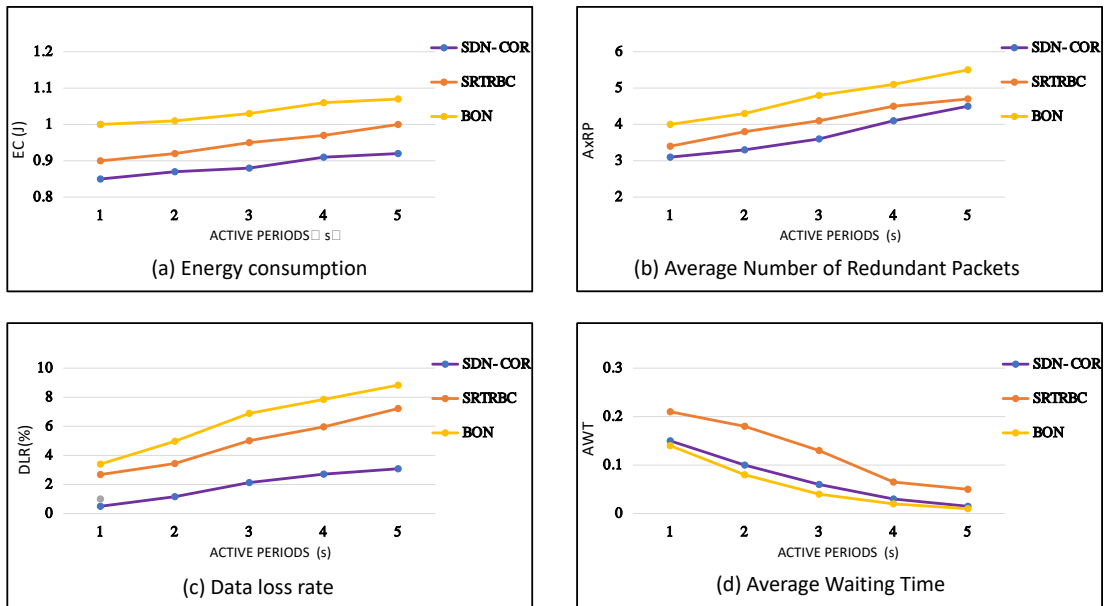


Fig. 10. Performance over wake-up intervals

The performance of AxRP was evaluated in Fig. 10b. The results indicate that AxRP increases with longer wake-up intervals. This is because longer wake-up times increase the probability of multiple receivers being active simultaneously. SDN-COR achieves better results in the same scenario, reducing AxRP by 11% and 27% compared to SRTRBC and BON algorithms, respectively.

Fig. 10c demonstrates that SDN-COR has a lower Delay Limit Ratio (DLR) compared to other methods. AWT evaluation based on wake-up intervals is shown in Fig. 10d, where proposed method achieves satisfactory waiting time.

6.3.4 Test 4- Comparison of Network Lifetime

This test evaluated the performance of network lifetime based on both communication range and wake-up interval, using a network of 100 nodes.

Fig. 11a illustrates the performance of network lifetime as the communication range varies from 50 meters to 100 meters. Traditional methods perform poorly compared to SDN-COR because they lack energy balancing mechanisms and generate excessive redundant packets, consuming a significant amount of energy. SDN-COR incorporates the concept of Software-Defined Networking (SDN) to introduce flexible management into WSNs and utilizes encoding-aware routing metrics to prioritize the selection of nodes with higher remaining energy, shorter transmission distances, and more coding opportunities.

Fig. 11b demonstrates how the wake-up interval impacts network lifetime performance. Based on the results, the following observations can be made: as the wake-up interval increases, network lifetime decreases. Longer active times for the nodes have a negative impact on the probability of multiple receivers being active simultaneously, resulting in the generation of numerous redundant packets.

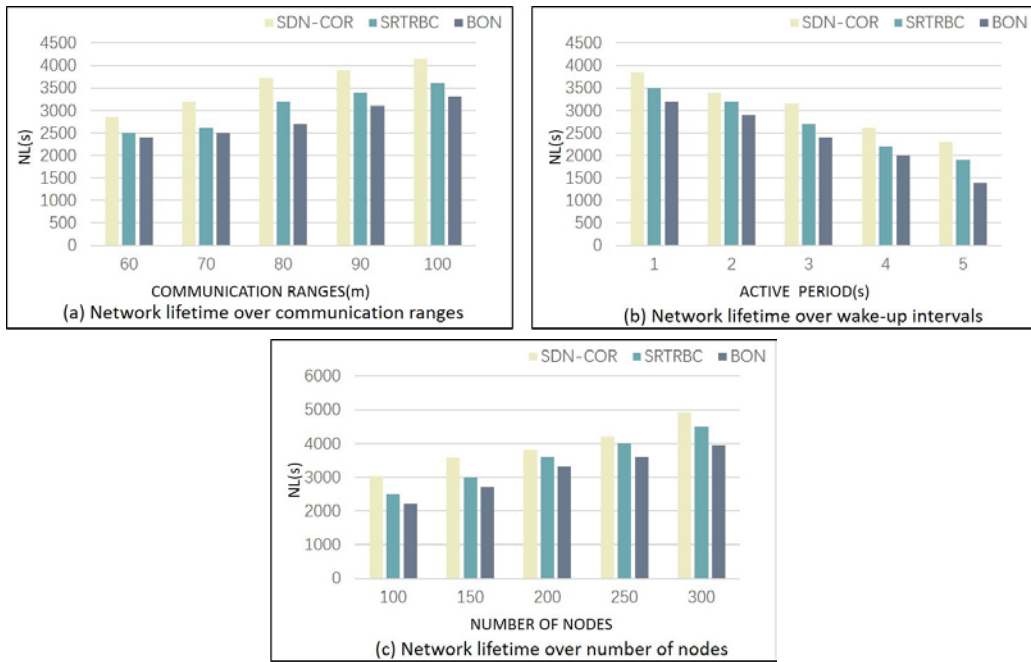


Fig. 11. Performance of network lifetime

The influence of the number of nodes on network lifetime is shown in Fig. 11c, where SDN-COR achieves better results compared to traditional methods.

In summary, the results demonstrate that the proposed SDN-COR method leads to enhanced network lifetime over benchmark routing protocols. Specifically, SDN-COR extends the network lifetime by about 14% compared to SRTRBC, and 26% compared to BON. This significant improvement is attributed to the centralized optimization of coding-aware routing achieved by the multi-controller SDWSNs.

7. Conclusion

This article provides an in-depth and detailed study of SDN-based coding opportunistic routing in WSNs, offering an improved solution for traditional software-defined WSN routing protocols. Addressing the challenges of low scalability, difficulty in policy changes, and low resource utilization in traditional WSN architectures, the article proposes an SDWSNs architecture model and designs a controller architecture by dividing the nodes in the SDWSNs into control nodes, central nodes, and regular sensor nodes. Furthermore, to address the issues of energy constraints in WSNs and coding-aware routing, the article introduces the coding-aware routing method SDN-COR. Simulation results demonstrate that compared with the traditional WSN protocols SRTRBC and BON, this method reduces energy consumption by 17% and 26%, increases network throughput by 20% and 28%, decreases redundant data packets by 11% and 27%, and extends network lifetime by 14% and 26%. Future research work will focus on the reliability of communication between controller and data plane nodes, validating the algorithm in actual WSN systems. Moreover, considering the complexity of large-scale networks and the challenges of routing in dynamic environments, we plan to explore routing strategies in large-scale network environments and study routing optimization methods in dynamic environments.

Acknowledgement

This work was supported by Shandong Key Laboratory Foundation of Computer Networks (No. SDKLCN-2022-02), Shandong Provincial Natural Science Foundation (No. ZR2023MF052).

References

- [1] N. Chauhan, S. Chauhan, "A Novel Area Coverage Technique for Maximizing the Wireless Sensor Network Lifetime," *Arabian Journal for Science and Engineering*, vol.46, pp.3329-3343, 2021. [Article \(CrossRef Link\)](#)
- [2] X. Li, H. Zhao and W. Deng, "BFOD: Blockchain-Based Privacy Protection and Security Sharing Scheme of Flight Operation Data," *IEEE Internet of Things Journal*, vol.11, no.2, pp.3392-3401, 2024. [Article \(CrossRef Link\)](#)
- [3] W. Deng, Z. Li, X. Li, H. Chen and H. Zhao, "Compound Fault Diagnosis Using Optimized MCKD and Sparse Representation for Rolling Bearings," *IEEE Transactions on Instrumentation and Measurement*, vol.71, pp.1-9, 2022. [Article \(CrossRef Link\)](#)
- [4] A. A. Abdul Ghaffar, A. Mahmoud, T. Sheltami, M. Abu-Amara, "A Survey on Software-Defined Networking-Based 5G Mobile Core Architectures," *Arabian Journal for Science and Engineering*, vol.48, pp.2313-2330, 2023. [Article \(CrossRef Link\)](#)
- [5] H. Mudassar, S. Nadir, A. Rashid et al., "Software-Defined Networking: Categories, Analysis, and Future Directions," *Sensors*, vol.22, no.15, 2022. [Article \(CrossRef Link\)](#)
- [6] C. Caucheteux, A. Gramfort, J.R. King, "Evidence of a predictive coding hierarchy in the human brain listening to speech," *Nature Human Behaviour*, vol.7, pp.430-441, 2023. [Article \(CrossRef Link\)](#)
- [7] B. Mao, J. Liu, Y. Wu and N. Kato, "Security and Privacy on 6G Network Edge: A Survey," *IEEE Communications Surveys & Tutorials*, vol.25, no.2, pp.1095-1127, 2023. [Article \(CrossRef Link\)](#)
- [8] S. Katti, H. Rahul, W. Hu et al., "XORs in the air: Practical wireless network coding," *IEEE/ACM Trans. Networking*, vol.16, no.3, pp.497-510, 2008. [Article \(CrossRef Link\)](#)
- [9] R. Parisa, M. Ghasem, "Distributed Network Coding-Aware Routing Protocol Incorporating Fuzzy-Logic-Based Forwarders in Wireless Ad hoc Networks," *Journal of Network and Systems Management*, vol.28, pp.1279-1315, 2020. [Article \(CrossRef Link\)](#)
- [10] Y. Su, Y. Xu, Z. Pang, Y. Kang and R. Fan, "HCAR: A Hybrid-Coding-Aware Routing Protocol for Underwater Acoustic Sensor Networks," *IEEE Internet of Things Journal*, vol.10, no.12, pp.10790-10801, 2023. [Article \(CrossRef Link\)](#)
- [11] S. Amit, N. Aitha, "Heuristic-based opportunistic network coding at potential relays in multi-hop wireless networks," *International Journal of Computers and Applications*, vol.45, no.2, pp.124-135, 2023. [Article \(CrossRef Link\)](#)
- [12] R. Amin, M. Reisslein, N. Shah, "Hybrid SDN Networks: A Survey of Existing Approaches," *IEEE Communications Surveys & Tutorials*, vol.20, no.4, pp.3259-3306, 2018. [Article \(CrossRef Link\)](#)
- [13] Z. Latif, K. Sharif, F. Li et al., "A comprehensive survey of interface protocols for software defined networks," *Journal of Network and Computer Applications*, vol.156, 2020. [Article \(CrossRef Link\)](#)
- [14] A.C. Anadiotis, L. Galluccio, S. Milardo et al., "SD-WISE: A Software-Defined Wireless Sensor Network," *Computer Networks*, vol.159, pp.84-95, 2019. [Article \(CrossRef Link\)](#)
- [15] T.G. Nguyen, T.V. Phan, B.T. Nguyen et al., "SeArch: A Collaborative and Intelligent NIDS Architecture for SDN-Based Cloud IoT Networks," *IEEE Access*, vol.7, pp.107678-107694, 2019. [Article \(CrossRef Link\)](#)

- [16] Y. Duan, W. Li, X. Fu, Y. Luo, L. Yang, "A methodology for reliability of WSN based on software defined network in adaptive industrial environment," *IEEE/CAA Journal of Automatica Sinica*, vol.5, no.1, pp.74-82, 2018. [Article \(CrossRef Link\)](#)
- [17] M.S. AZIZI, M.L. HASNAOUI, "Software Defined Networking for Energy Efficient Wireless Sensor Network," in *Proc. of 2019 International Conference on Advanced Communication Technologies and Networking*, pp.1-7, 2019. [Article \(CrossRef Link\)](#)
- [18] M.U. Younus, S.U. Islam, S.W. Kim, "Proposition and Real-Time Implementation of an Energy-Aware Routing Protocol for a Software Defined Wireless Sensor Network," *Sensors*, vol.19, no.12, 2019. [Article \(CrossRef Link\)](#)
- [19] M. Cicioğlu, A. Çalhan, "Energy-efficient and SDN-enabled routing algorithm for wireless body area networks," *Computer Communications*, vol.160, pp.228-239, 2020. [Article \(CrossRef Link\)](#)
- [20] R. Kumar, U. Venkanna, V. Tiwari, "EOMCSR: An Energy Optimized Multi-Constrained Sustainable Routing Model for SDWSN," *IEEE Transactions on Network and Service Management*, vol.19, no.2, pp.1650-1661, 2022. [Article \(CrossRef Link\)](#)
- [21] E. Hajian, M.R. Khayyambashi, N. Movahhedinia, "A Mechanism for Load Balancing Routing and Virtualization Based on SDWSN for IoT Applications," *IEEE Access*, vol.10, pp.37457-37476, 2022. [Article \(CrossRef Link\)](#)
- [22] X. He, G. Yang, "CARTA: Coding-Aware Routing via Tree-Based Address," *Wireless Communications and Mobile Computing*, vol.2020, pp.1-16, 2020. [Article \(CrossRef Link\)](#)
- [23] Y. Zhao, S. Xiao, H. Gan, L. Li, L. Xiao, "A Constrained Coding-Aware Routing Scheme in Wireless Ad-Hoc Networks," *Sensors*, vol.19, 2019. [Article \(CrossRef Link\)](#)
- [24] Y. Peng, Q. Deng, L. Guo, F. Wang, "A New Network Coding Based Routing Protocol for Enhancing Throughput Capacity in Wireless Mesh Networks," *Chinese Journal of Electronics*, vol.28, no.2, pp.416-422, 2019. [Article \(CrossRef Link\)](#)
- [25] L. Hai, H. Wang, J. Wang, Z. Tang, "HCOR: a high-throughput coding-aware opportunistic routing for inter-flow network coding in wireless mesh networks," *Wireless Communications and Networking*, vol.2014, 2014. [Article \(CrossRef Link\)](#)
- [26] B. Guo, H. Li, C. Zhou, Y. Cheng, "Analysis of General Network Coding Conditions and Design of a Free-Ride-Oriented Routing Metric," *IEEE Transactions on Vehicular Technology*, vol.60, no.4, pp.1714-1727, 2011. [Article \(CrossRef Link\)](#)
- [27] G.X. Kok, C.O. Chow, H. Ishii, "Improving network coding in wireless ad hoc networks," *Ad Hoc Networks*, vol.33, pp.16-34, 2015. [Article \(CrossRef Link\)](#)
- [28] R. Renugadevi, K. Vijayalakshmi, "Modeling a Novel Network Coding Aware Routing Protocol for Enhancement of Network Performance in Wireless Mesh Network," *Wireless Personal Commun.*, vol.107, pp.621-649, 2019. [Article \(CrossRef Link\)](#)
- [29] D. Zhao, G. Lun, R. Xue, "Coding-Aware Opportunistic Routing for Sparse Underwater Wireless Sensor Networks," *IEEE Access*, vol.9, pp.50170-50187, 2021. [Article \(CrossRef Link\)](#)
- [30] X. Zhong, L. Li, S. Zhang, R. Lu, "ECOR: An Energy Aware Coded Opportunistic Routing for Cognitive Radio Social Internet of Things," *Wireless Personal Communications*, vol.110, pp.1-20, 2020. [Article \(CrossRef Link\)](#)
- [31] J. Wang, Y. Gao, W. Liu et al., "An Asynchronous Clustering and Mobile Data Gathering Schema based on Timer Mechanism in Wireless Sensor Networks," *Computers Materials & Continua*, vol.58, pp.711-725, 2019. [Article \(CrossRef Link\)](#)
- [32] A. Hawbani, X. Wang, L. Zhao, A. Al-Dubai, G. Min, O. Busaileh, "Novel Architecture and Heuristic Algorithms for Software-Defined Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol.28, no.6, pp.2809-2822, 2020. [Article \(CrossRef Link\)](#)
- [33] V. Sridhar, K. V. R. Rao, V. V. Kumar, M. Mukred, S. S. Ullah, H. AlSalman, "A Machine Learning-Based Intelligence Approach for Multiple-Input/Multiple-Output Routing in Wireless Sensor Networks," *Mathematical Problems in Engineering*, vol.2022, pp.1-13, 2022. [Article \(CrossRef Link\)](#)
- [34] K. Alic, A. Svirgelj, "A one-hop opportunistic network coding algorithm for wireless mesh networks," *Wireless Networks*, vol.24, pp.1007-1018, 2018. [Article \(CrossRef Link\)](#)



Yifan Hu received his PhD, master and bachelor degrees at the Donghua University, Shanghai, China. He is currently an associate professor in Qingdao University of Technology. His research interests include Internet of Things and Intelligent Edge Computing.



Xiqiang Hou received his bachelor degree at the Bingzhou College, Bingzhou, China. He is currently a graduate student in Qingdao University of Technology. His research interests include Internet of Things and Embedded System.



Fuqiang Wang received his doctor degree at the Shenyang Institute of Automation Chinese Academy of Sciences, China. He is currently an associate researcher in Qilu University of Technology. His research interests include Internet of Things security and information security.