Reliable and Resilient Communication in Duty Cycled Software Defined Wireless Sensor Networks

16 January 2025

Reliable and Resilient Communication in Duty Cycled Software Defined Wireless Sensor Networks

Muhammad Umar Farooq Qaisar[1]*, Weijie Yuan[1]*, Paolo Bellavista[2], Shehzad Ashraf Chaudhry[3], Adeel Ahmed[4],
and Muhammad Imran[5]

[1] Department of Electronic and Electrical Engineering, Southern University of Science and Technology, Shenzhen, China
[2] Department of Computer Science and Engineering, University of Bologna, Bologna, Italy
[3] Department of Computer Science and Information Technology, Abu Dhabi University, Abu Dhabi, UAE
[4] School of Computer Science and Technology, University of Science and Technology of China, Hefei, China
[5] James Watt School of Engineering, University of Glasgow, G12 8QQ Glasgow, United Kingdom
*Corresponding authors: Weijie Yuan and Muhammad Umar Farooq, Email:{yuanwj, muhammad}@sustech.edu.cn

Abstract—**Reliable and resilient network communication with flexible management is one of the significant issues in wireless sensor networks (WSNs). Due to the substantial packet loss, energy usage, and inadequate security of WSNs, the reliable data delivery is necessary when using multi-hop data communication. This paper employs the software-defined networking (SDN) concept to provide flexible and effective management with reliability in network communication. Therefore, it proposes a reliable and resilient communication in duty cycled software defined wireless sensor networks that addresses two parts. First, it considers the four attributes with their probability distributions to provide reliability and resilience in data plane communication. The attributes are direct trust, recommended trust, signal to interference noise ratio, and residual energy. Second, the SDN controller computes those attributes along with the Expected Duty Cycled Wake-ups (EDC) to make it more reliable and assigns communication strategies to each node through the reliable nodes. It also restricts the number of forwarding nodes for each node in order to minimize packet duplication. The simulation results indicate that, when compared to state-of-the-art protocols, the proposed protocol greatly enhances the reliability and resilience of the network.**

Index Terms—**reliable communication, software defined wireless sensor networks, trust aware, duty cycling, resilience.**

I. INTRODUCTION

Wireless sensor networks (WSNs) enable infrastructure-free communications via common wireless mediums, eliminating the necessity of a central access point or permanent infrastructure. Sensor networks, which are one of the most intriguing wireless technologies, are made up of a collection of dynamically interacting nodes. They establish a new wireless transmission paradigm of sending data over multiple hops. WSNs are dynamic that encounter a number of difficulties because of their relatively short transmission range and other constrained resources, such as network administration, security measures, and multi-hop communication etc. [1]. This makes determining the general consensus and applicability of reliable techniques in the network a challenging task because the dispersed management of the network protocol specifies which node may receive or transmit data. For this reason, a lightweight, well-managed, reliable technique with resilience capabilities must be employed. The necessity for flexible management with resilience capabilities has led to the development of Software-Defined Networking (SDN) architecture [2] [3].

A Software Defined Wireless Sensor Network (SDWSN) architectural design makes use of WSNs that include self-configuration, adaptability, auto-surveillance, and customized control capabilities with minimal maintenance and improved reliability. SDWSN uses intelligent Software-Defined sensor nodes to replace conventional sensor nodes. For applications in controlling activities, monitoring, and sensing using programmable interfaces, an SDN node can be configured with various sensing functionalities. SDWSN provides customized capabilities in a virtual environment to manage network resources more efficiently and to improve network resilience and reliability by separating control and data plane operations [4]. SDWSN necessitates duty cycling since energy usage of the sensor node is a critical issue to be solved with this technique when the nodes are not taking measurements. The duty-cycling in WSNs has been recognized as one of the key technologies for conserving energy, in which the radio of the node switches between two modes: active and sleep. Medium access control (MAC) protocols with asynchronous duty-cycling, as opposed to synchronized duty-cycling, have been proven to considerably enhance energy efficiency in various contemporary applications of sensor networks [5] [6].

The aforementioned literature inspired us to incorporate SDN and WSN to provide flexibility in management and reliable communication via a centrally controlled network, as well as to propose a protocol for reliable and resilient communication in duty cycled software defined wireless sensor networks (R^2Com). Here, we define reliability as the likelihood of at least one reliable path between the source nodes and the sink, and resilience as the capacity of a network to operate even in the midst of compromised nodes or mediocre link quality. We use Expected Duty-Cycle Wake-ups (EDC), which is reliable in terms of small delays by selecting paths with low EDC [7]. Hence, we intend to improve it by deeply addressing the reliable and resilience flow computation in the first term of the EDC metric to achieve the following goals. First, the protocol integrates the SDN concept into WSN, which provides reliable network management by assigning flow strategies to each

node rather than each packet in order to reduce information exchange between two planes. Furthermore, it provides the communication strategies from the controller to each node based on reliable path which considers the highly reliable nodes in the path from the controller to the target node. Second, the controller computes reliable communication flows for the data plane based on probabilistic strategies for each node. We consider reliability in link estimation by taking key communication metrics with probability distributions into account, such as direct trust, recommended trust, signal to interference noise ratio (SINR), and residual energy. Third, the controller restricts the number of forwarding nodes for each node to reduce network packet duplication. The goal of this paper is to provide reliable and resilient communication and network management between two planes.

II. RELATED WORK

This paper concentrates on ensuring reliable and resilient communication while minimizing information exchange between the control and data planes. The theories and techniques pertinent to our work are therefore covered in this section.

A Bayesian based trustworthy management scheme (BTMS) is introduced in [8] that employs the direct and recommended trusts to determine the nodes trust in the network. To estimate the direct trust, an improved Bayesian equation with a penalty element is used. The estimation of recommended trust is determined using a third party's recommendations. In [9], a technique is introduced to identify faulty nodes and recognize Sybil attacks. In this technique, the trust values determined by the sensor nodes are used in reliable localization. Similarly, In [10], a reliable trustworthy technique is introduced to mitigate the black hole attacks. According to this technique, the source node chooses the forwarder node as the next hop node with the shortest path to the sink and a trustworthiness greater than the predetermined threshold. In [11], a multi-trust technique is employed to develop an effective energy trustworthy evaluation-based routing scheme (ETERS) that prevents network communication attacks. This method has the benefit of quickly identifying malicious nodes, with the drawback of restricting packet redundancy. In [12] [13], the importance of packet load management and reliability in SDWSN is addressed. In [12], a technique called improved SDWSN is introduced by addressing the reliability issue in WSN. The technique improves network reliability by addressing network coverage and heterogeneous network management issues. In [13], the flow splitting optimization (FSO) method is used to address the traffic load minimization (TLM) concern. The two methodologies are employed to alleviate the packet load problem; the first one involves selecting the most suitable forwarder nodes, and the second one includes transmitting the optimum splitting flow. In [2], an energy aware sustainable communication with multi-constrained technique for SDWSN called EOMCR is introduced, which reduces the over-utilization of nodes by formulating a Mixed Integer Linear Programming (MILP) problem. In light of the multiple network constraints, this methodology facilitates achieving energy efficiency in the net-

work. In [14], a software-defined based energy-efficacy trustworthy routing methodology ETMRM is introduced, which takes the trustworthy and energy metrics into account for reliable communication. This technique involves the cluster head collecting aggregated data from cluster members and then lowering packet size during forwarding to conserve energy resources while ensuring control traffic delivery.

The aforementioned related studies demonstrated the reliability in communication and network management in WSN. However, they lack the combined focus on effective reliability and resilient communication technique with SDWSN. Also, the SDN based WSN studies did not consider the reliable route in each phase when assigning flows to each node in order to avoid inconsistent links in the data plane. The key intention of this work is to determine a reliable and resilient data plane communication that is controlled by the control plane. Hence, it is imperative to consider an effective reliability model when prioritizing the data plane nodes according to probability functions. Therefore, this paper proposes a reliable and resilient communication in duty-cycled software defined wireless sensor networks (R^2Com) to address the above gaps.

III. THE PROPOSED PROTOCOL

We propose reliable and resilient communication in duty cycled software defined wireless sensor networks, which profoundly addresses the reliable data communication flows with EDC in the data plane and the reliable path to assign the communication strategies from controller to target nodes in the flow instantiation.

A. Network Model and Initialization

The network model is based on a software defined wireless sensor network (SDWSN), which is composed of an application layer that handles various network applications, a control layer where an SDN controller manages and controls the overall network features and functionality, and a data layer composed of SDN enabled nodes that communicate with each other by following control layer instructions in the form of communication strategies as shown in Fig. 1. The SDN controller and sink node are assumed to be completely trustworthy. We assume that the malicious nodes are more obviously caused by the compromised nodes. The most prevalent notations in the work are outlined in Table I.
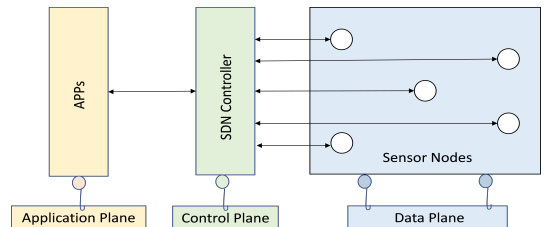

Fig. 1: The network model of SDWSN

Each node in the network initialization phase needs to report the information to the SDN controller, considering that it does not necessarily know anything about the network nodes. The

Notation	Definition
$\mathcal{N}$	$\mathcal{N} = \{Sn_0, Sn_1, Sn_2, ..., Sn_r\}$; Sn_x is a sensor node $\in \mathcal{N}$, and r is its size.
$\mathcal{N}_x$	The Sn_x's neighbor nodes; r_x is the size of $\mathcal{N}_x$.
Sn_b	The sink.
Sn_s	The source node.
Sn_t	The target node.
$\Im_{x,y}$	The SINR of Sn_x's neighbor Sn_y.
$\Upsilon_{x,y}$	The direct trust between Sn_x and Sn_y.
$\aleph_{x,y}$	The recommended trust of Sn_x's neighbor Sn_y
$\xi_{x,y}$	The remaining energy of Sn_x's neighbor Sn_y.
e_*	The initial energy of the nodes.
$\mathcal{R}_c$	The range of communication.
e	*Euler's Constant*. Its approximate value is 2.71828.

nodes send the beacon packet to obtain the topology information and send it to the SDN controller. The controller consists of multiple network control functionalities (e.g., Database, Network Visualization (NV), Topology Constructor (TC) and Flow Engine (FE) etc.), to further process the information. In this section, the greedy approach is employed to initialize the network and obtain topology information, which includes the energy status and location of each node. For each node, the predetermined trust value is set to 1.

B. Reliable Communication Flows

This section describes the data plane reliable communication flows from each node to sink. Data communication reliability is essential to avoid poor link quality or risky nodes in the routing path. Therefore, this work considers the four important attributes to focus on reliability in the communication process. In Eq. (12), the attributes with their probability distributions are defined as the optimized link quality metric that is included in the first term of reliable EDC using Eq. (13). The distributions are comprised of direct trust Eq. (2), recommended trust Eq. (5), signal to interference noise ratio Eq. (8), and residual energy Eq. (10). Fig. 2 depicts precedence of each attribute based on probability distribution functions over their normalized values.

1) Direct Trust Distribution: This distribution's main goal is to provide precedence to nodes that have historically made the most successful direct communications to their neighbors, as seen in Fig. 2a. In the network, each neighbor node of a sender (*i.e.*, $\forall Sn_y \in \mathcal{N}_x$) is described as a vector, where $\Upsilon_x = \{\Upsilon_{x,1}, \Upsilon_{x,2}, ..., \Upsilon_{x,r_x}\}$, and $\Upsilon_{x,y}$ represents the sender Sn_x to neighbor node Sn_y direct trust. Following that, we first normalize the vector Υ_x into $\bar{\Upsilon}_x = \{\bar{\Upsilon}_{x,1}, \bar{\Upsilon}_{x,2}, ..., \bar{\Upsilon}_{x,r_x}\}$ between $[0-1]$ in Eq. (1), and then derived the probability distribution function in Eq. (2) by curve fitting the normalized term, $\tilde{\Upsilon}_x = \{\tilde{\Upsilon}_{x,1}, \tilde{\Upsilon}_{x,2}, ..., \tilde{\Upsilon}_{x,r_x}\}$. Here, $\varsigma_{x,y}$ and $\mu_{x,y}$ represent the successful and unsuccessful direct communication between sender node Sn_x and receiver node Sn_y, respectively. The distribution is controlled by using a variable λ_Υ, where $\lambda_\Upsilon \geq 1$ is set by default. The greater the $\lambda_\Upsilon \geq 1$, the more precedence given to nodes that have historically communicated successfully with nearby nodes.

$$\bar{\Upsilon}_{x,y} = \left(\frac{\varsigma_{x,y}}{\varsigma_{x,y} + \mu_{x,y}}\right) * \left(1 - \frac{1}{\varsigma_{x,y} + 1}\right) \quad (1)$$

$$\tilde{\Upsilon}_{x,y} = \begin{cases} \alpha * \left(1 - e^{\left(-\beta*(\bar{\Upsilon}_{x,y})^{\lambda_\Upsilon}\right)}\right) \\ \alpha = -3.93056; \quad \beta = -0.22905; \\ \lambda_\Upsilon = 1; \end{cases} \quad \forall Sn_y \in \mathcal{N}_x \quad (2)$$

2) Recommended Trust Distribution: This distribution's main goal is to provide precedence to nodes that have most valuable reputation to their neighbors, as seen in Fig. 2b. In the network, each neighbor node of a sender (*i.e.*, $\forall Sn_y \in \mathcal{N}_x$) is described as a vector, where $\aleph_x = \{\aleph_{x,1}, \aleph_{x,2}, ..., \aleph_{x,r_x}\}$, and $\aleph_{x,y}$ represents the sender Sn_x to neighbor node Sn_y recommended trust as obtained in Eq. (3). Following that, we first normalize the vector $\aleph_x$ into $\bar{\aleph}_x = \{\bar{\aleph}_{x,1}, \bar{\aleph}_{x,2}, ..., \bar{\aleph}_{x,r_x}\}$ between $[0-1]$ in Eq. (4), and then derived the probability distribution function in Eq. (5) by curve fitting the normalized term, $\tilde{\aleph}_x = \{\tilde{\aleph}_{x,1}, \tilde{\aleph}_{x,2}, ..., \tilde{\aleph}_{x,r_x}\}$. The distribution is controlled by using a variable $\lambda_\aleph$, where $\lambda_\aleph \geq 1$ is set by default. The greater the $\lambda_\aleph \geq 1$, the more precedence given to nodes that have most valuable reputation with neighboring nodes.

$$\aleph_{x,y} = \sum_{v=1}^{r_x} \bar{\Upsilon}_{x,v} * \bar{\Upsilon}_{v,y} \quad \forall Sn_y \in \mathcal{N}_x \quad (3)$$

$$\bar{\aleph}_{x,y} = \frac{\sum_{v=1}^{r_x} \bar{\Upsilon}_{x,v} * \bar{\Upsilon}_{v,y}}{r_x} \quad \forall Sn_y \in \mathcal{N}_x \quad (4)$$

$$\tilde{\aleph}_{x,y} = \begin{cases} \frac{\alpha}{\left(1+\beta*e^{\left(-\gamma*(\bar{\aleph}_{x,y})^{\lambda_\aleph}\right)}\right)} \\ \alpha = 1.20947; \quad \beta = 16.79708; \\ \gamma = 4.39447; \quad \lambda_\aleph = 1; \end{cases} \quad \forall Sn_y \in \mathcal{N}_x \quad (5)$$

3) Signal to Interference Noise Ratio Distribution: This distribution's main goal is to provide precedence to nodes that have higher signal to interference noise ratio (SINR) among their neighbors, as seen in Fig. 2c. As the matter of fact, the SINR, which is measured as the ratio of the received signal power from the sender of the channel to the received signal power of all nearby channels with noise element, has a strong correlation with link reliability. In the network, each neighbor node of a sender (*i.e.*, $\forall Sn_y \in \mathcal{N}_x$) is described as a vector, where $\Im_x = \{\Im_{x,1}, \Im_{x,2}, ..., \Im_{x,r_x}\}$, and $\Im_{x,y}$ represents the sender Sn_x to neighbor node Sn_y SINR as obtained in Eq. (6). Here, ρ_x symbolizes the transmit power of node Sn_x, $d_{x,y}$ symbolizes the *Euclidean distance* between node Sn_x and node Sn_y, ζ symbolizes the number of interfering nodes, and σ is the ambient noise factor. Following that, we first normalize the vector $\Im_x$ into $\bar{\Im}_x = \{\bar{\Im}_{x,1}, \bar{\Im}_{x,2}, ..., \bar{\Im}_{x,r_x}\}$ between $[0-1]$ in Eq. (7), and then derived the probability distribution function in Eq. (8) by curve fitting the normalized term, $\tilde{\Im}_x = \{\tilde{\Im}_{x,1}, \tilde{\Im}_{x,2}, ..., \tilde{\Im}_{x,r_x}\}$. The distribution is controlled by using a variable $\lambda_\Im$, where $\lambda_\Im \geq 1$ is set by default. The greater the $\lambda_\Im \geq 1$, the more precedence given to nodes that have the higher SINR among the neighboring nodes.

(a) Normalized Dir-Trust   (b) Normalized Rec-Trust   (c) Normalized SINR   (d) Normalized Energy
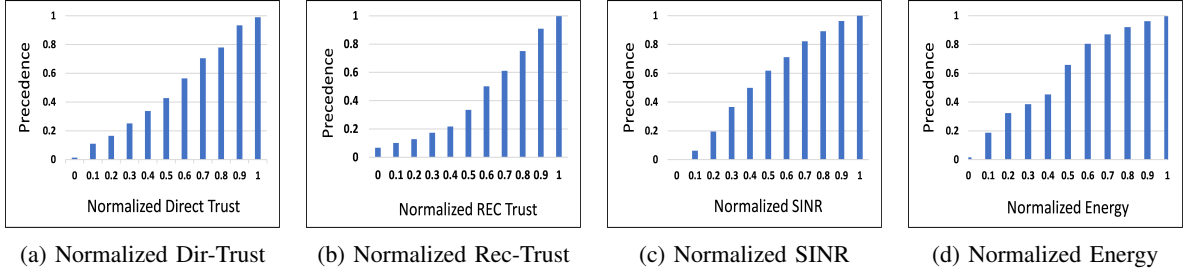
Fig. 2: Distributions precedence graphs

$$\Im_{x,y} = log_2 \left(1 + \frac{\rho_x * d_{x,y}}{\sum_{v \in \zeta} \rho_v * d_{v,y} + \sigma^2} \right) \quad \forall\, Sn_y \in \mathcal{N}_x \quad (6)$$

$$\bar{\Im}_{x,y} = \frac{\Im_{x,y} - \min(\Im_{x,y})}{\max(\Im_{x,y}) - \min(\Im_{x,y})} \quad \forall\, Sn_y \in \mathcal{N}_x \quad (7)$$

$$\tilde{\Im}_{x,y} = \begin{cases} \alpha + \left(\beta * \mathrm{e}^{\left(\gamma * (\bar{\Im}_{x,y})^{\lambda_{\Im}} \right)} \right) \\ \alpha = 2.78513; \quad \beta = -2.7845; \\ \gamma = -0.4451; \quad \lambda_{\Im} = 1; \end{cases} \quad \forall\, Sn_y \in \mathcal{N}_x \quad (8)$$

4) Residual Energy Distribution: This distribution's main goal is to provide precedence to nodes that have the most remaining energy, as seen in Fig. 2d. In the network, each neighbor node of a sender (*i.e.*, $\forall Sn_y \in \mathcal{N}_x$) is described as a vector, where $\xi_x = \{\xi_{x,1}, \xi_{x,2}, ..., \xi_{x,r_x}\}$, and $\xi_{x,y}$ represents the sender Sn_x's neighbor node Sn_y residual energy. Following that, we first normalize the vector ξ_x into $\bar{\xi}_x = \{\bar{\xi}_{x,1}, \bar{\xi}_{x,2}, ..., \bar{\xi}_{x,r_x}\}$ between $[0 - 1]$ in Eq. (9), and then derived the probability distribution function in Eq. (10) by curve fitting the normalized term, $\tilde{\xi}_x = \{\tilde{\xi}_{x,1}, \tilde{\xi}_{x,2}, ..., \tilde{\xi}_{x,r_x}\}$. Here, ξy symbolizes the remaining energy of node Sn_y. The distribution is controlled by using a variable λ_ξ, where $\lambda_\xi \geq 1$ is set by default. The greater the $\lambda_\xi \geq 1$, the more precedence given to nodes with the most remaining energy among neighboring nodes.

$$\bar{\xi}_{x,y} = \frac{\xi y}{e_*} \quad \forall\, Sn_y \in \mathcal{N}_x \quad (9)$$

$$\tilde{\xi}_{x,y} = \begin{cases} \frac{\alpha}{1 + \left(\beta * \mathrm{e}^{\left(\gamma * (-\bar{\xi}_{x,y})^{\lambda_\xi} \right)} \right)} \\ \alpha = 1.109307; \quad \beta = 7.89527; \\ \gamma = 4.26819; \quad \lambda_\xi = 1; \end{cases} \quad \forall\, Sn_y \in \mathcal{N}_x \quad (10)$$

5) Optimized Link Quality Metric: To optimize the link quality, the SDN controller first evaluates the total trust based on direct and recommended trusts, the total trust is obtained by Eq. (11). In the aforementioned trusts; energy, data integrity, and communication link are frequently employed to assess node trust behavior. Because of the noticeable evolving nature of WSNs and the influence of selfish attacks, nodes deliberately avoid engaging in communication to accomplish their own goals while causing network damage. Therefore, the total trust is categorized into three states: trustworthy, unsure, and untrustworthy. To ensure reliability, total trust only considers trustworthy nodes for the reliable communication process.

$$\mathfrak{T}\mathfrak{T}_{x,y} = \sqrt{\tilde{\Upsilon}_{x,y} * \tilde{\aleph}_{x,y}} \quad \forall\, Sn_y \in \mathcal{N}_x$$

$$= \sqrt{\left(\alpha * \left(1 - \mathrm{e}^{\left(-\beta * (\tilde{\Upsilon}_{x,y})^{\lambda_\Upsilon} \right)} \right) \right) * \left(\frac{\alpha}{\left(1 + \beta * \mathrm{e}^{\left(-\gamma * (\tilde{\aleph}_{x,y})^{\lambda_\aleph} \right)} \right)} \right)} \quad (11)$$

$$\begin{cases} Trustworthy & 0.6 \leq \mathfrak{T}\mathfrak{T}_{x,y} \leq 1.0 \\ Unsure & 0.3 \leq \mathfrak{T}\mathfrak{T}_{x,y} < 0.6 \\ Untrustworthy & 0.0 \leq \mathfrak{T}\mathfrak{T}_{x,y} < 0.3 \end{cases}$$

Following that, the SDN controller computes the optimized link quality metric in Eq. (12) that consist of total trust distribution Eq. (11), SINR distribution Eq. (8), and residual energy distribution Eq. (10), where each node in the distribution term is described as vector $\bar{\ell} = (\bar{\ell}_{x,1}, \bar{\ell}_{x,2}, ..., \bar{\ell}_{x,r_x})$ such that $\tilde{\ell}_{x,y} = (\mathfrak{T}\mathfrak{T}_{x,y} * \tilde{\Im}_{x,y} * \tilde{\xi}_{x,y}) / \sum_{v=1}^{r_x} (\mathfrak{T}\mathfrak{T}_{x,v} + \tilde{\Im}_{x,v} + \tilde{\xi}_{x,v})$. To further optimize link quality in terms of reliability, we consider this term in the first term of EDC to propose reliable EDC *(REDC)* in Eq. (13). A more reliable link quality will result in a smaller REDC as depicted in Fig. 3.

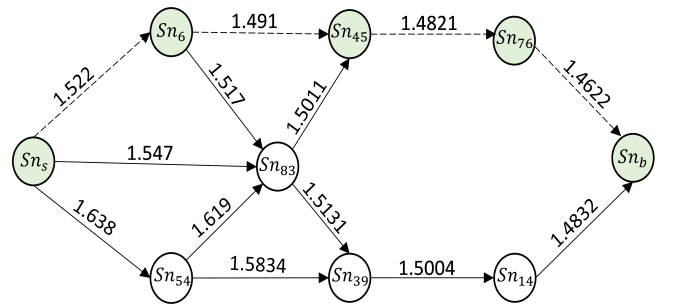Fig. 3: Reliable data communication flow

$$\tilde{\ell}_{x,y} = (\mathfrak{T}\mathfrak{T}_{x,y} * \tilde{\Im}_{x,y} * \tilde{\xi}_{x,y}) / \sum_{v=1}^{r_x} (\mathfrak{T}\mathfrak{T}_{x,v} + \tilde{\Im}_{x,v} + \tilde{\xi}_{x,v}) \quad \mathfrak{T}\mathfrak{T}_{x,y} \geq 0.6 \quad (12)$$

$$REDC_x = \frac{1}{\tilde{\ell}_{x,y} * (Sn_x)} + \frac{\sum_{Sn_y \in \mathcal{N}_x} REDC_y}{Sn_x} \quad (13)$$

C. Reliable Flow Instantiation

The responsibility of the SDN controller is essential because it needs to manage the network intelligence at one centralized point. This section provides the reliable and dependable methodology on the reverse paths from the SDN controller to the end nodes to store the communication strategies in the reliable flow table (Table II) of each node. To avoid any inconsistency with the communication strategies of each node, the methodology only concentrates the (*Trustworthy*) nodes in terms of total trust Eq. (11) greater than or equal to the threshold in the path. This is because the SDN controller necessitates each node to understand how data is handled in terms of reliable communication with dependable management.

TABLE II: Reliable Flow Table

Node ID	REDC Precedence	ACK	Action	Statistics
1	1.522	1	Forward	43
2	1.547	1	Forward	27
3	1.638	1	Drop	11

D. Communication Action Flow

Each sender node has multiple neighbor nodes that act as forwarding nodes and can overhear the same packet during their mutual active period of time to transmit towards the sink node, resulting in data redundancy. To address this issue, the number of forwarding nodes for each node must be controlled using an effective mechanism. This work describes the communication action flow strategy applied to the packet to forward or drop using Eq. (14), where $\mathfrak{F}_x$ is the number of forwarding nodes threshold chosen by the sender node.

$$\overrightarrow{Act}(Sn_x) = \begin{cases} Forward\ \mathfrak{F}_x \leq \lceil 1 + (\ln(r_x)) \rceil;\ REDC_y \leq \frac{\sum_{y=1}^{r_x} REDC_y}{r_x} \\ Drop\ \mathfrak{F}_x > \lceil 1 + (\ln(r_x)) \rceil;\ REDC_y > \frac{\sum_{y=1}^{r_x} REDC_y}{r_x} \end{cases} \quad (14)$$

The SDN controller updates the precedence value based on the residual energy factor. When a node loses 5% of its energy, the SDN controller reevaluates the REDC and updates the nodes' communication strategies in their flow tables.

IV. PERFORMANCE EVALUATION AND DISCUSSION

We used a simulator developed in visual studio 2015 (C# WPF) [5] based on the NS3 models to examine the performance of the proposed protocol R^2Com, by taking various simulation parameters into account listed in Table III. The nodes are deployed at random, and the sink is positioned in the region's center. For simulation convenience, the controller is set to the sink position. Each node employs the BoX-MAC [15] and has the same active (1s) and sleep (2s) durations. The nodes are given 0.5J battery powers and are instructed to use energy in accordance with the energy model described in [5]. To ensure simulation accuracy, we ran the simulations 25 times to obtain the average values for the results.

The proposed protocol R^2Com is evaluated using the following parameters. i) **Average Energy Consumption**: It determines the energy consumed by all nodes during the simulation phase. ii) **Packet Delivery Ratio (PDR)**: It emphasizes the

TABLE III: Simulation Parameters

Parameter	Value
Network nodes	100
Communication range	$60m$
Malicious nodes	Ranges between 10 and 50
Packet rate	$1/0.1s$
Time for simulation	$480s$
Packet size	$128\ bytes$

ratio of packets received at the sink to total packets sent by the source nodes. iii) **Average Latency**: It calculates the arrival latency for each packet during the simulation phase. The latency is measured after the packet has been delivered to the sink. iv) **Network Lifetime**: The moment the first node dies determine how long the network will last.

A. Simulation Results

The proposed protocol R^2Com compares the results of two well-known protocols: ETERS [11] and ETMRM [14].

Fig. 4a depicts the average energy consumption result, demonstrating how the result gradually increases as the number of malicious nodes increases. The proposed protocol R^2Com outperforms the two state-of-the-art protocols for the following reasons. First, it employs an effective reliability approach on the data plane by considering direct and recommended trust, as well as the residual energy of the nodes with their probability distributions, to efficiently prioritize the nodes in the selection of high trustworthy forwarder nodes in each transmission phase. This approach also ensures network resilience. Second, on the control plane, it employs an effective reliable approach in which the controller computes the reliable reverse path to assign communication strategies to each node. ETERS and ETMRM both did not consider effectively distributing node trustworthiness and other parameters to improve the network's reliability and average energy consumption.

Fig. 4b depicts the packet delivery ratio result, demonstrating how the result gradually decreases as the number of malicious nodes increases. The proposed protocol R^2Com outperforms state-of-the-art protocols in terms of reliability and resilience in the selection of forwarder nodes to transmit data using probability distributions that efficiently managed to prioritize the nodes under the malicious nodes. ETERS and ETMRM results fall short in the proper distribution of trust during relay node selection and also in the efficient path selection technique when compared to our proposed protocol.

Fig. 4c depicts the average latency result, demonstrating how the result gradually increases as the number of malicious nodes increases. The proposed protocol R^2Com outperforms the state-of-the-art protocols in terms of average latency results because it uses the direct and recommended trust distributions of the nodes with the EDC metric to provide latency reliability. The data packet is relayed to the sink via highly reliable nodes with shortest paths. ETERS and ETMRM both use the trustworthy routing technique without focusing on latency by shortening the path in the presence of malicious nodes.

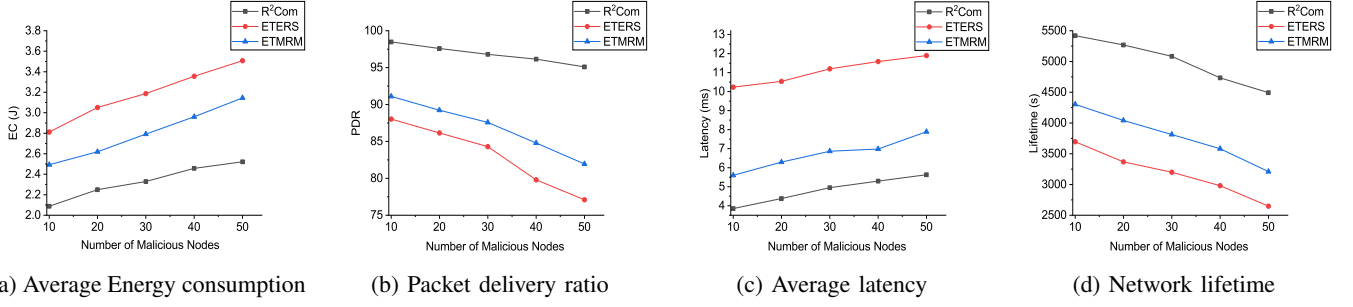| (a) Average Energy consumption | (b) Packet delivery ratio | (c) Average latency | (d) Network lifetime |

Fig. 4: Performance over number of malicious nodes

Fig. 4d depicts the network lifetime result, demonstrating how the result gradually decreases as the number of malicious nodes increases. The proposed protocol R^2Com outperforms state-of-the-art protocols due to its efficient use of the trustworthy, signal-to-interference noise ratio, and residual energy distributions in the reliable communication flow. Furthermore, the SDN controller manages the computation of these distributions with EDC to reduce the resources of data plane nodes while improving network lifetime. Although the presence of malicious nodes in the network consumes a lot of energy, it still provides a more reliable and resilient approach to network functionality. ETERS and ETMRM employ the reliability approach in the communication process, but they lack an energy-efficient and balanced approach to effectively utilizing the trustworthiness of network nodes under malicious nodes.

V. CONCLUSION

A reliable and resilient communication is introduced with effective management in duty cycled software defined wireless sensor networks. It focuses on the reliability of assigning communication strategies to each node through the control plane, as well as the reliability and resilience of data communication between nodes. Therefore, it first considers the four attributes for data plane communication, along with their probability distributions, in order to prioritize the reliable nodes. The attributes are the direct trust, recommended trust, signal-to-interference noise ratio, and residual energy. The attributes are then incorporated into the first term of the EDC metric to make it more reliable in terms of lowering delay in the midst of compromised nodes and mediocre link quality. Second, in the control plane, the controller computes the reliable EDC *(REDC)* along with the probability distributions. The controller then assigns communication strategies in the flow table of each node through the reliable nodes to avoid any inconsistency with the data packets in the data plane. Finally, the controller restricts the forwarder for each node to reduce packet duplication. According to the simulation results, the proposed protocol outperforms the state-of-the-art protocols.

ACKNOWLEDGMENT

REFERENCES

[1] M. U. F. Qaisar, X. Wang, A. Hawbani, A. Khan, A. Ahmed, F. T. Wedaj, and S. Ullah, "Toras: Trustworthy load-balanced opportunistic routing for asynchronous duty-cycled wsns," *IEEE Systems Journal*, 2022.

[2] R. Kumar, U. Venkanna, and V. Tiwari, "Eomcsr: An energy optimized multi-constrained sustainable routing model for sdwsn," *IEEE Transactions on Network and Service Management*, 2021.

[3] A. Fausto, G. Gaggero, F. Patrone, and M. Marchese, "Reduction of the delays within an intrusion detection system (ids) based on software defined networking (sdn)," *IEEE Access*, vol. 10, pp. 109850–109862, 2022.

[4] S. S. G. Shiny, S. S. Priya, and K. Murugan, "Control message quenching-based communication protocol for energy management in sdwsn," *IEEE Transactions on Network and Service Management*, 2022.

[5] M. U. F. Qaisar, X. Wang, A. Hawbani, L. Zhao, A. Y. Al-Dubai, and O. Busaileh, "Sdorp: Sdn based opportunistic routing for asynchronous wireless sensor networks," *IEEE Transactions on Mobile Computing*, 2022.

[6] M. U. F. Qaisar, X. Wang, A. Hawbani, A. Khan, A. Ahmed, and F. T. Wedaj, "Torp: load balanced reliable opportunistic routing for asynchronous wireless sensor networks," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 1384–1389, IEEE, 2020.

[7] E. Ghadimi, O. Landsiedel, P. Soldati, S. Duquennoy, and M. Johansson, "Opportunistic routing in low duty-cycle wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 4, pp. 1–39, 2014.

[8] R. Feng, X. Han, Q. Liu, and N. Yu, "A credible bayesian-based trust management scheme for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, p. 678926, 2015.

[9] P. Li, X. Yu, H. Xu, J. Qian, L. Dong, and H. Nie, "Research on secure localization model based on trust valuation in wireless sensor networks," *Security and Communication Networks*, vol. 2017, 2017.

[10] Y. Liu, M. Dong, K. Ota, and A. Liu, "Activetrust: Secure and trustable routing in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 2013–2027, 2016.

[11] T. Khan, K. Singh, M. H. Hasan, K. Ahmad, G. T. Reddy, S. Mohan, and A. Ahmadian, "Eters: A comprehensive energy aware trust-based efficient routing scheme for adversarial wsns," *Future Generation Computer Systems*, vol. 125, pp. 921–943, 2021.

[12] Y. Duan, W. Li, X. Fu, Y. Luo, and 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, 2017.

[13] G. Li, S. Guo, Y. Yang, and Y. Yang, "Traffic load minimization in software defined wireless sensor networks," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1370–1378, 2018.

[14] R. Wang, Z. Zhang, Z. Zhang, and Z. Jia, "Etmrm: An energy-efficient trust management and routing mechanism for sdwsns," *Computer Networks*, vol. 139, pp. 119–135, 2018.

[15] D. Moss and P. Levis, "Box-macs: Exploiting physical and link layer boundaries in low-power networking," *Computer Systems Laboratory Stanford University*, vol. 64, no. 66, p. 120, 2008.