



Asian Research Association



IPL: An Intelligent, Predefined, and Lightweight Recovery Scheme for Node Failures in Topology-Aware Software-Defined Wireless Sensor Networks

S. Sathish ^a, J. Poongodi ^{b,*}, G. Sudha ^c, K. Satheshkumar ^d

^a School of Computing, SRM Institute of Science and Technology, Tiruchirappalli-621105, Tamil Nadu, India

^b Department of Computer Science and Engineering, Dr.N.G.P. Institute of Technology, Coimbatore-641 048, Tamil Nadu, India.

^c Department of Computer Science and Engineering, Sona College of Technology, Salem-636 005, Tamil Nadu, India

^d Department of Artificial Intelligence and Machine Learning, Bannari Amman Institute of Technology, Sathyamangalam-638 401, Tamil Nadu, India

* Corresponding Author Email: poongodiapcse@gmail.com

DOI: <https://doi.org/10.54392/irjmt2646>

Received: 22-08-2025; Revised: 13-06-2026; Accepted: 25-06-2026; Published: 04-07-2026



Abstract: Software-defined wireless sensor networks (SDWSNs) improve network programmability and centralized control, but maintaining connectivity under node and link failures remains difficult because of node mobility, limited energy, and dynamic topology changes. This study proposes IPL, an Intelligent, Predefined, and Lightweight recovery framework for topology-aware SDWSNs. The framework integrates three coordinated mechanisms: predictive link-lifetime estimation using energy and mobility parameters, energy-aware target positioning through a weighted midpoint strategy, and ring-based coordination among mobile IPL relay nodes for deterministic and low-overhead recovery. IPL is designed to handle both isolated and multiple concurrent failures while reducing controller burden and avoiding expensive global recomputation. The method was evaluated in a Mininet/Floodlight-based SDWSN environment with 150 nodes under identical settings against four benchmark schemes: IFT, Fed-TSN, P4Neighbor, and LCD. Across varying failure conditions, IPL consistently achieved faster recovery and better communication reliability. Relative to the baselines, the proposed method reduced recovery time by up to 26%, lowered latency by up to 27%, decreased energy consumption by up to 18%, improved packet delivery ratio by up to 19%, increased recovery success rate by up to 17%, and extended network lifetime by up to 19%. These gains arise from proactive link monitoring, rapid relay repositioning, and structured recovery coordination. Overall, IPL offers an efficient and scalable recovery solution for dynamic SDWSNs, particularly in environments with moderate failure rates, while highlighting opportunities for future enhancement through adaptive relay allocation and improved mobility-aware prediction.

Keywords: Software-Defined Wireless Sensor Networks, Network Connectivity, Node Failure, Recovery, IPL, Energy Consumption.

1. Introduction

Since the beginning of the industrial revolution, wireless sensor networks (WSNs) have also gained significant attention of the industry and scientists. The possible implementation of WSNs in various industry and real-time applications are stated in [1, 2]. Jamshed et al. [3], enumerated the numerous challenges faced by WSNs owing to energy limitation, computation speed as well as bandwidth. A software-defined network (SDN) as stated by [4, 5] is a programmable network, which adds valuable solutions to the problems of WSNs through the separation of the of the control plane and the data plane. Articles [5, 6] explained how software-defined network idea has been integrated with WSNs and gave rise to a

new model called software-defined wireless sensor networks (SDWSNs), which offer solutions to issues facing both WSNs and SDNs. SDWSNs enhance the management of the network by providing centralized control utilizing software and overall SDNs and WSNs performance. [7, 8] emphasized the characteristics of integrating SDNs and WSNs.

SDWSNs have a data plane [4], which is composed of a large number of wireless nodes, and ensuring that such nodes remain connected is very important. Node failures can take place because of limited energy, mobility of the node or because of external factors or hardware limitations. These failures may divide the network into unconnected segments

resulting in a loss of data, a greater delay, and consumption of more energy and poor Quality of Service (QoS), and eventually the overall network performance. The recovery of the node failures should be done effectively so that the network can be resilient and the QoS demands in the various applications would not be compromised.

Although much has been achieved in SDWSN and SDN-based recovery mechanisms, the relevant approaches [9-12] have failed to provide the opportunity to deal with combined issues of recovery of multi-link failures, scalability, energy consumption, and dynamism in changing networks. The existing solutions are either fast localized recovery with hardware limitations, based on computationally heavy parameterization models, or learning-based strategies with no deterministic guarantees but with extra overhead. Besides, the majority of the available methods consider resource efficiency, recovery latency, and topology dynamics to be independent issues, which leads to suboptimal performance in large-scale and highly dynamic network environments.

In order to overcome these constraints, this paper suggests a proposed Intelligent Predefined Lightweight (IPL) topology-based recovery scheme which will permit the effective management of various concurrent failures whilst being maintained as scalable and energy-aware. The proposed method combines predictive link stability estimation with structured relay coordination to provide adaptive and low-overhead protection without involving the use of complex recomputation and models that are based on training. This single design offers a feasible and strong solution to maintain network connectivity and performance in SDWSN dynamic environments.

Continuing this design, the peculiarities of the offered approach are the following. In contrast to the current failure recovery schemes in SDWSNs, based on reactive rerouting, fixed relays choice, or controller-based recovery schemes, IPL framework proposes a predictive recovery design. It combines link lifetime estimation, energy-aware relay selection and coordinated topology management to allow distributed recovery that is controlled. The design minimises control overhead, eliminates global recomputation and enhances performance in the presence of multiple failures. Thus, the novelty of the work is in the combined design of a predictive, structured and lightweight recovery model that can be used to fill the gap of reactive SDN rerouting versus uncoordinated recovery mechanisms of relays in dynamic SDWSN environments.

The following sections of the paper are structured in the following way: Section II is dedicated to the detailed literature review. Section III explains the problem statement and model of the system. Section IV provides the proposed IPL-based topology control

mechanism of recovery of multiple nodes failure in SDWSNs. The fourth part of the simulation defines the parameters and metrics of the simulation and compares the results with some of the benchmarked methods. Section V is the conclusion of this article with the prospect of improvement.

2. Related Work

The development of failure recovery of the software defined wireless sensor networks has moved away traditional mechanisms of reactive recovery [13], [14] to the paradigm of optimization based, intelligent, and data planes-based recovery [15]. These strategies are inherently different with regard to achieving a balance between recovery latency, resource use and scalability during conditions of multi-link failure. The research study hypothesizes this part of the article and assumes an extrapolation of the well-known contributions in the recovery of nodes after a failure in the context of SDN as well as SDWSNs.

2.1 Programmable Data Plane and Hardware-Centric Recovery

The use of programmable data planes (PDPs) has made sub-millisecond failure recovery possible as control logic moves onto forwarding devices instead of the control plane. Franco *et al.* [16] showed this potential using the vFFR architecture on Intel Tofino hardware where recovery times less than 500 ms, with insignificant frame loss, are obtained by integrating Loss of Signal (LOS) detection with precomputed alternate paths. In line with this method, Xu *et al.* [10] solved TCAM scalability limitations by proposing P4Neighbor that exploits register-based local state maintenance and header-encoded backup path representation as the central tool to cut the flow entry requirements by a very high factor. Although such ways reduce the dependence of controllers, they are mainly local optimization techniques that do not consider network-wide coordination.

In order to improve the robustness of control-planes, Wong & Lee [17] presented the bootstrapping mechanism that relies on prepared Minimum Recovery Trees (MRT) so that faster restoring of controller-switch connectivity within in-band architectures can be achieved. Expanding recovery to multilayer configurations, Sgambelluri *et al.* [18] examined the topic of MANTRA-based resilience in hybrid optical-packet systems and determined that coordinated recovery enhances reliability and changes resources distribution trade-offs between layers. Even though PDP-based designs are effective in minimizing recovery latency, the state management complexity and hardware limitations continue to be limiting factors, as well as high TCAM and register usage, which add up to impediments in the scalability of large-scale deployment.

2.2 Optimization-Driven and Structural Topology Models

Recovery mechanisms based on optimization create the resilience as a constraint path computation problem, which tries to optimize the protection coverage and minimize resource utilization. Qiu *et al.* [19] enhanced the speed of computation using pruned search algorithm to select shortest recovery path, and the convergence is faster than the conventional methodology. These structural finding are further developed by Hieu & Kitsuan [20], who employed the use of the Integer Linear Programming (ILP) to build arc-Disjoint directed spanning trees that greatly minimized flow table overheads and ensured structural resiliency in case of several failures. In the multicast case, Jue *Chen et al.* [21] added iterative calculating the backup paths coupled with optimization of space compression using VLANs to enhance the efficiency of recovery.

These methods are however to a large extent based on simplified failure models. As Vuppapapati *et al.* [22] emphasized, the available solutions are limited to single-link failure and suggested the selection of the candidate algorithms based on non-shortest path algorithms to support a connectedness in the occurrence of multi-link failures. On the same note, Daha *et al.* [23] presented community detection-based recovery in an attempt to localize handling of failures but the computing overhead grows in tandem with the network size. Therefore, though optimization-based approaches offer powerful theoretical guarantees, their use of static assumptions as well as their use of computationally expensive formulations restrict their use in dynamic and large-scale contexts.

2.3 Intelligent and Learning-Based Paradigms

Data-driven decision-making is used to assist the learning-based method in handling uncertainty in network behavior in the recovery of failure. Balasubramanian *et al.* [11] suggested a federated learning-based Time-Sensitive Networks that reduces the joint failure probability using predictive migration strategies thus enhancing bandwidth utilization. On the same note, Kaur *et al.* [9] integrated reinforcement learning with metaheuristic optimization to improve the IIoT routing efficiency and recovery performance. Ali *et al.* [24] Applied blockchain with neural networks within a security-aware context where both fault tolerance and resilience to distributed denial-of-service attacks were achieved. Huang & Yu [12] also suggested lightweight control model that would add an additional control forwarding layer so as to reduce interference in the wireless sensor networks. Regardless of such improvements, learning-based methods have a structural drawback of applicability and generalizability. Their use of training information limits flexibility to a heterogeneous network environment, and the absence of deterministic guarantees creates difficulties with their

deployment in a latency sensitive network. Moreover, model inference and update overhead further add extra delays which are unfavorable in a situation where fast or simultaneous failures are likely to occur.

2.4 Congestion-Awareness and TCAM Optimization

To stabilize the economy in the aftermath of recovery, there must be mechanisms in place that place congestion and resource use in explicit consideration. Isyaku *et al.* [25] resolved this by coming up with a Bayesian model of congestion-awareness that approximates the path-criticality, and it is founded on patterns of link utilization thus avoiding routes subject to bottlenecks. Jiang *et al.* [26] developed this notion by introducing the K-backup model that then distributes large-volume flows through multiple backup tracks in order to reduce the maximum utilization of links and enhance load balancing. This mitigates shortcomings that have been previously witnessed in the prior one on single-backup system like Petale & Thangaraj [27] and Nurwarsito *et al.* [28], where absence of load distribution results into post-recovery hotspots.

The novel learning of switch memory has also been considered by Wang *et al.* [29] who presented methods of various forms of flow classification and resource sharing rings to minimize TCAM usage without compromising recovery efficiency. Parallel to it, Hsieh *et al.* [30], have proposed results on control-plane resiliency by identifying the Byzantine Agreement problem in multi-controller SDN systems and obtaining conclusions on the complexity limits of messages necessary to reach a consensus under adversarial conditions.

In spite of these developments, the available solutions to this problem do not pay much attention to the fact that congestion control, TCAM optimization, and control-plane reliability are separate issues. This localized view makes them less effective in the presence of multi-link failure, where these aspects are inseparably coupled and have to be optimized together. The summary of the literature review is presented in the table 1. The problem is formally stated in the following section based on the identified limitations.

3. Problem Statement and System Model

The literature survey demonstrates that a number of techniques have been proposed to support fault tolerance and failure recovery in SDN and related network paradigms. However, significant challenges remain in handling multi-link failures under dynamic network conditions. Current optimization-based methods, including those proposed by Qiu *et al.* [19] and Hieu & Kitsuan [20], are not sufficiently adaptable for large-scale settings because they rely on static formulations.

Table 1. A summary of the reviewed literature

References	Methodology Used	Control/Data Plane Recovery	Single or Multi Link Failure	Energy Consumption	QoS	Topology	Delay
[9]	RLWOA (Reinforcement Learning)	Data Plane (Intelligent)	Single/Node Failure	Yes	Yes	Yes	Yes
[10]	P4Neighbor (Header encapsulation)	Data Plane (Proactive)	Multi-Link	No	No	Yes	Yes
[11]	Fed-TSN (Federated Learning)	Control/Data Plane	Multi-Link	No	Yes	Yes	Yes
[12]	LCD (MIS-Tree Algorithm)	Data Plane	Single/Multi	Yes	No	Yes	Yes
[16]	vFFR (Flow/Link/Switch Centric)	Data Plane	Single Link	No	Yes	Yes	Yes
[17]	P4IBN (Minimum Recovery Tree)	Control Plane (In-Band)	Single/Multi	No	Yes	Yes	Yes
[18]	MANTRA (Multi-layer coordination)	Control Plane	Single Link	No	Yes	Yes	Yes
[19]	Pruned searching algorithm	Control Plane	Multi-Link	No	Yes	Yes	Yes
[20]	Swift Recovery (ILP + Heuristic)	Control Plane (Hybrid)	Multi-Link	No	No	Yes	Yes
[21]	Multicast Tree Reconstruction	Data Plane	Multi-Component	No	No	Yes	Yes
[22]	CSA (Candidate Selection)	Control Plane	Multi-Link	No	No	Yes	No
[23]	CDRA (Community Detection)	Control Plane (Reactive)	Single/Multi	No	No	Yes	Yes
[24]	TOPSIS + Blockchain + ANN	Control Plane (Reactive)	Single Link	No	Yes	Yes	Yes
[25]	RFRC (Bayesian + Criticality)	Control Plane	Single Link	No	Yes	Yes	Yes
[26]	K-Backup (Congestion-aware)	Data Plane (Proactive)	Multi-Link	Yes	Yes	Yes	Yes
[27]	FF Group Table Integration	Data Plane	Single Link	No	No	Yes	Yes
[28]	VLAN ID + Fast Failover	Data Plane	Single Link	No	No	Yes	Yes
[29]	RSFR (Ring-based selection)	Data Plane (Proactive)	Single Link	No	No	Yes	Yes
[30]	Byzantine Agreement Consensus	Control Plane	Malicious Failure	No	No	Yes	Yes

Similarly, the candidate-selection and community-based approaches proposed by Vuppalapati *et al.* [22] and Daha *et al.* [23] also incur additional computational cost as network size increases.

Learning-based approaches, such as Fed-TSN proposed by Balasubramanian *et al.* [11] and reinforcement learning frameworks proposed by Kaur *et al.* [9], attempt to address the lack of adaptability but suffer from scalability and generalization limitations. In addition, lightweight control mechanisms such as LCD,

proposed by Huang & Yu [12], do not adapt well to rapidly changing topologies. Recovery latency has been reduced in programmable data-plane designs, including vFFR proposed by Franco *et al.* [16] and P4Neighbor proposed by Xu *et al.* [10]. However, these designs face limitations related to TCAM usage and hardware constraints, which may restrict scalability. Moreover, congestion-sensitive models such as K-Backup proposed by Jiang *et al.* [26] and Bayesian models proposed by Isyaku *et al.* [25] focus primarily on load balancing and do not fully address interactions caused by multi-link failures.

These constraints show that there is no single unified recovery mechanism that can be used to tackle multi-link failure scalability, dynamic topology adaptation and resource efficiency simultaneously. Thus, a lightweight and scalable recovery framework that would help to sustain network stability and performance in large-scale, dynamic SDN environments is needed.

3.1 System Model and Assumptions

This sub-part gives the abstract system model and operation assumptions of the proposed IPL-based recovery framework. The parameters of the detail simulation and experimental settings are given later in the section 5. The system model is characterized by the following elements

3.1.1 Network Model

The Software-Defined Wireless Sensor Network (SDWSN) is represented as a dynamic graph, $G(V, E)$ in which $V = \{V, w\}$ is a set of wireless nodes (w -nodes), and V IPL is a set of IPL nodes. There is a communication connection between two nodes in case they are both in the range of transmission and have adequate residual energy. The topology of the network changes as the mobility of the node's changes with time.

3.1.2 B. Node Capabilities

All nodes are supposed to be location-sensitive and able to report their states periodically. The nodes are wireless and carry out sensing, forwarding of data and monitoring links, whereas IPL also carry out mobility-assisted recovery and monitoring relay positioning. The IPL nodes are involved in the organized recovery coordinating mechanism.

3.1.3 Controller Model

An SDN controller is logically centralized and provides a global view of a network topology in terms of periodic state updates. The controller will verify failures, classify (node vs. link failure) and prioritize recovery requests. Implementation of recovery is assigned to the

IPL network in order to minimize overhead by the controller.

3.1.4 Failure Model

This system takes into account node failures, link failures as well as simultaneous numerous failures. A node failure is associated with loss of communication with the node whilst a link failure refers to a disconnection between two active nodes. Multiple failures are defined as failures which are in short time frame and are managed by prioritized recovery.

3.1.5 Timing Assumptions

Periodic heartbeat messages are used to detect failure and recovery is initiated after verification of the controller. The assumption of the system is that detection and recovery delays are bound to guarantee a timely re-establishment of connectivity.

3.1.6 Communication Model

The communication is also assumed to be range based model of communication, with the only links being within a designated range of transmission. The mobility of nodes and remaining energy determines the link stability.

This abstraction defines the operational framework of the proposed method, while Section 5 provides the specific parameter settings and experimental conditions used for evaluation

4. Proposed Methodology

This section presents the proposed Intelligent Predefined Lightweight (IPL) topology-based failure recovery scheme of Software-Defined Wireless Sensor Networks (SDWSN).

4.1. System overview and Data Plane Construction

This section gives a coherent summary of the suggested IPL-based recovery system, stating how data plane, control plane, and IPL network interrelate. The aim is to introduce an illustrative end-to-end perspective of the process of failure detection, processing, and resolution in the SDWSN. Figure 1 elaborates on how the data plane in the proposed approach is structured.

The architecture is characterized by three key elements; (i) wireless nodes (w -nodes) which constitute data plane, (ii) a logically centralized SDN controller which constitutes control plane, and (iii) a ring topology of mobile IPL nodes.

Table 2. Notation and Symbol Definitions

Symbol	Description
$G = (V, E)$	Network graph with node set V and link set E
(V_w)	Set of wireless nodes (w-nodes)
$(V_{\{IPL\}})$ </td <td>Set of IPL nodes</td>	Set of IPL nodes
i, j	Communication link between nodes i and j
E_i, E_j	Residual energy of node i and j
d_{ij}	Euclidean distance between nodes i and j
v_{rel}	Relative velocity between nodes i and j
R	Transmission range
α	Smoothing factor
Δt	Heartbeat interval
ΔT	Controller topology update interval
θ	Link stability threshold
LL_{ij}	Link lifetime between nodes i and j
v_i	Speed of node i
θ_i, θ_j	Movement direction of node i and j
Req	Recovery request
$dist$	Distance between IPL node and target position
N	IPL ring size

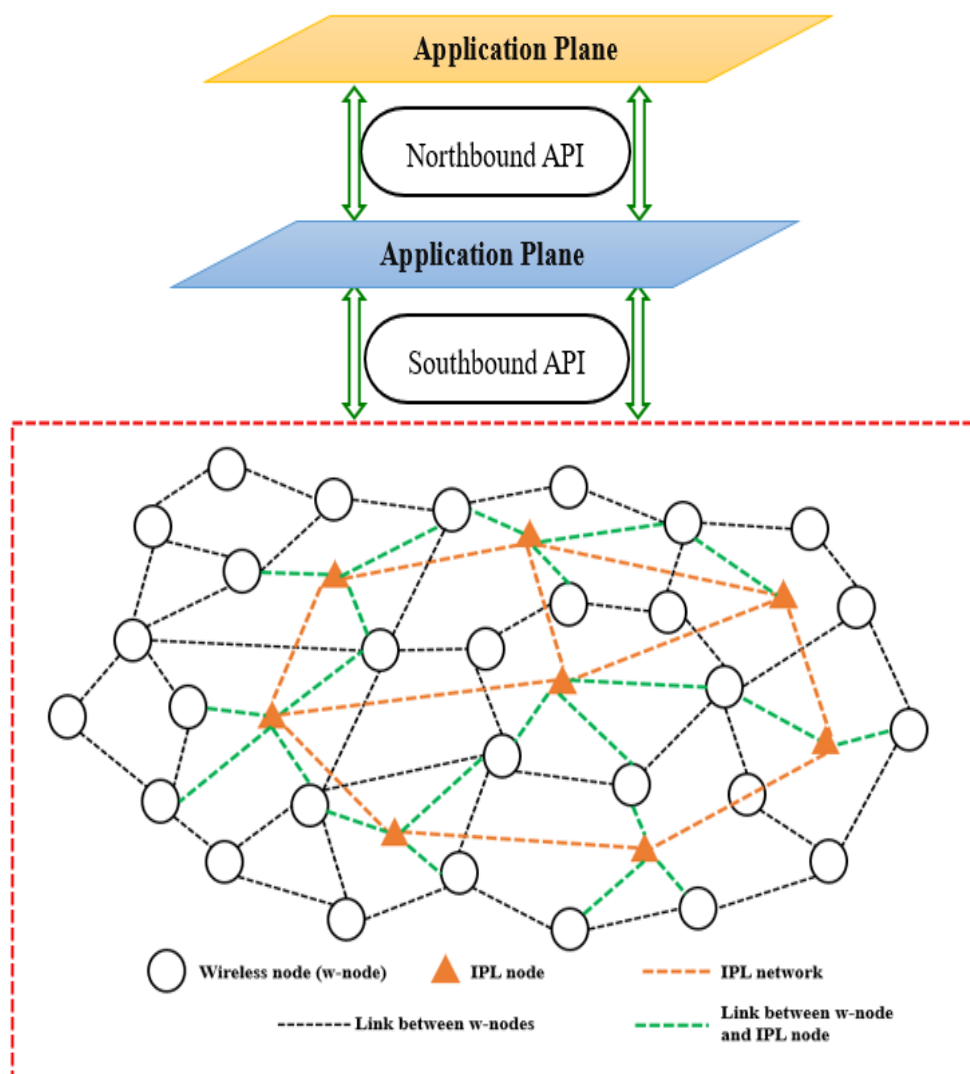


Figure 1. Proposed IPL-enabled SDWSN architecture with wireless nodes, SDN controller, and mobile IPL relay ring.

The wireless nodes are meant to sense, forward data, and monitor the links. SDN controller has the global topology view and does the verification and prioritization of failures. The nodes of IPL are mobile relay agents which can be used to restore connectivity by repositioning themselves between nodes that are disconnected.

4.2 Failure Detection

W- Nodes trigger the data transfer process between the source and the destination after the completion of the construction of a network. To ensure that they are alive periodically, nodes send a heartbeat message to their one-hop neighbors at a constant time interval of $\Delta t = 1 s$ to spread any alteration of their energy state or position.

4.2.1 Failure Declaration Rules

Nodes send periodic heartbeat messages to their one-hop neighbors in order to inform them of their operational state and to make available updates in the residual energy, position, speed and direction. As a monitoring interval, $\Delta t = 1 s$, the heartbeat interval is equal.

A node is considered failed when three consecutive heartbeat messages are not obtained within the expected time. Maximum silence period $3\Delta t = 3 s$ because interval between heartbeats is $\Delta t = 1 s$. Each of the nodes keeps a record of the last time it has received a heartbeat among their neighbors. In case time taken is greater than $3\Delta t$, failure report is created and sent to SDN controller.

SDN controller has a global topology view that is updated periodically $\Delta t = 1 s$ with periodic state reports. When a failure report is received, the controller checks and confirms that it does not receive any heartbeat updates in its topology record in the same $3\Delta t$ window. In case of confirmation, the node is declared as failed and a recovery message is sent to the IPL network.

Proactive recovery reaction is also caused by the predicted link lifetime measure on top of the reactive detection. When the estimated lifetime of link between two nodes n_1 and n_2 is less than the set threshold $\theta = 0.3$ which is the least acceptable normalized link stability to allow reliable communication, a recovery request is initiated.

4.2.2 Distinguishing Node Failure from Link Failure

In this work, the failures are limited to node failures or link failures. A node failure is a total failure of the functionality of a node and a link failure is a lack of connection between two operational nodes. Path-level disruptions are addressed as the result of these major

types of failures. In the suggested solution the controller distinguishes node failures and link failures by the pattern of heartbeat reception

4.2.2.1 Node Failure

A failure of a node is announced when none of its one-hop neighbors receive any heartbeat messages within the $3\Delta t$. This means that the node has gone off line. The controller eliminates the node and all links connected with it out of the topology graph.

4.2.2.2 Link Failure

A link failure is pronounced when a node keeps on sending heartbeat messages to its neighbors except one particular neighbor. This implies that the node is active but the connection between two nodes has been weakened to a level that is below the communication threshold. In case the failed node is a leaf node no recovery action is initiated. In case the failure interferes with the connectivity between the active segments of the network or interrupts the further transmission of data, the controller sends a recovery request to the IPL network.

4.2.3 Handling Multiple Simultaneous Failures

There are several failures which are defined as failures in a $3\Delta t$ based interval which is within the same monitoring window Δt based. In cases where the multiple failures are identified in this period, the requests of recovery are added to a priority queue by the network criticality. The score of criticalities is calculated using: The number of data paths that were disrupted, and the number of nodes out of network. Greater impact failures are worked on first. IPL network handles requests of recovery in sequence. When the IPL node is already in bridging mode it sends the request that arrives to the next IPL node in the ring. This ensures that: Each node of the IPL does not support several recovery operations at the same time. The processing of recovery requests is non-deadlocked. Deterministically, failures that occur near each other are simply taken into account.

4.3 Link Lifetime Model

The functional feature of a link lifetime between two wireless nodes n_1 and n_2 is a factor of the residual energy, distance between nodes and relative movement. The formulation proposed will make the links having more energy, shorter distance, and low divergence speed more preferred.

The link lifetime between nodes i and j , denoted as LL_{ij} , is defined as:

$$LL_{ij} = \frac{\min(E_i, E_j)}{d_{ij} \cdot (1 + v_{rel})} \quad (1)$$

Where E_i, E_j are the residual energy of nodes i and j , d_{ij} denotes the Euclidean distance between the nodes, and v_{rel} represents the relative velocity of the node.

The distance is computed as:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2)$$

Where x_i, y_i and x_j, y_j represents the coordinates of nodes i and j respectively.

The relative velocity is computed as:

$$v_{rel} = \sqrt{v_i^2 + v_j^2 - 2v_i v_j \cos(\theta_i - \theta_j)} \quad (3)$$

Where v_i and v_j represent node speeds, and θ_i and θ_j denotes movement directions.

A link is considered valid only if:

$LL_{ij} > \theta$ AND $d_{ij} \leq R$, if either condition fails, the link is considered unstable and a recovery request is triggered.

4.3.1 Runtime Parameter Acquisition:

At runtime, all the parameters needed to estimate a link lifetime are acquired with periodic heartbeat messages sent at time interval $\Delta t = 1$ s

Residual Energy (E): Every node has an internal energy counter with an initial value of 2 J. Energy is reduced in response to both transmission and receipt events. The present energy level is added to each message of heartbeat.

Node Position (x, y): Nodes are location-aware and obtain their coordinates using a lightweight localization mechanism. The coordinates are included in heartbeat messages.

Distance (d_{ij}): Computed as the Euclidean distance between coordinates of nodes n_1 and n_2 .

Mobility Parameters (v, θ): Each node will indicate its speed and direction of movement in every message of heartbeat.

Sampling Frequency: Each parameter is computed at each monitoring interval $\Delta t = 1$ s, which provides real-time assessment of links.

The link between the wireless nodes and sensors is monitored using algorithm 1. It mostly guarantees that the connection life of every connection is preserved over a specific threshold θ . and when a connection is on the point of failure, it implements remedial measures to deal with the failure by enforcing IPL nodes. This algorithm is periodically implemented every timer event interval Δt to maintain constant link monitoring between the nodes. The variables applied in

the algorithms are in line with the notation specified in Table 2.

4.4 Link Prediction

The proposed method employs an Exponentially Weighted Moving Average (EWMA) in order to predict future link conditions:

Algorithm 1. IPL Link Monitoring and Recovery Trigger

Input:

- Set of active links L ;
- Threshold $\theta = 0.3$;
- Monitoring interval $\Delta t = 1$ s;
- Transmission range $R = 40$ m

Output:

Recovery request forwarded to the IPL network

Runs at:

Wireless nodes (monitoring), SDN Controller (verification)

Procedure

1. Initialize monitoring loop
2. Repeat every interval Δt :
3. For each link $l_{ij} \in L$:

Retrieve $E_i, E_j, (x_i, y_i), (x_j, y_j), v_i, v_j, \theta_i, \theta_j$

4. Compute:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

5. Compute:

$$v_{rel} = \sqrt{v_i^2 + v_j^2 - 2v_i v_j \cos(\theta_i - \theta_j)}$$

6. Compute:

$$LL_{ij} = \frac{\min(E_i, E_j)}{d_{ij}(1 + v_{rel})}$$

7. If $LL_{ij} < \theta$ OR $d_{ij} > R$ then

Generate recovery request Req

Send Req to SDN Controller

Controller verifies failure and forwards to IPL network

10. End If
11. End For
12. End Repeat

$$\hat{x}(t+1) = \alpha x(t) + (1-\alpha)\hat{x}(t) \quad (4)$$

In which $\alpha = 0.6$ is a smoothing factor, $x(t)$ is a current observation, and $\hat{x}(t)$ is a previous estimate.

Dead-reckoning is used to estimate the position of the node in the future:

$$x' = x + v \cdot \cos(\theta) \cdot \Delta t \tag{5}$$

$$y' = y + v \cdot \sin(\theta) \cdot \Delta t \tag{6}$$

The distance between the nodes is predicted and then calculated:

$$d'_{ij} = \sqrt{(x'_i - x'_j)^2 + (y'_i - y'_j)^2} \tag{7}$$

If $d'_{ij} \leq R$, then a connection is expected to be maintained but when not, proactive recovery is considered. The future link life workflow is represented in Algorithm 2.

Algorithm 2. Calculate Future Link Life

Input:

- Node states
- EWMA factor $\alpha = 0.6$
- Monitoring interval $\Delta t = 1$ s
- Transmission range $R = 40$ m

Output:

Predicted link status (1 = reachable, 0 = failure)

Runs at: Each wireless node

Procedure

1. Predict speed:

$$\hat{v}_i = \alpha v_i + (1 - \alpha) \hat{v}_i^{prev}$$
2. Predict direction:

$$\hat{\theta}_i = \alpha \theta_i + (1 - \alpha) \hat{\theta}_i^{prev}$$
3. Predict position:

$$x'_i = x_i + \hat{v}_i \cos(\hat{\theta}_i) \Delta t, y'_i = y_i + \hat{v}_i \sin(\hat{\theta}_i) \Delta t$$
4. Compute predicted distance:

$$d'_{ij} = \sqrt{(x'_i - x'_j)^2 + (y'_i - y'_j)^2}$$

5. If $d'_{ij} \leq R$ AND energy > 0 then
Return 1
6. Else
7. Return 0

4.5. Intelligent, Pre-Defined, Lightweight (IPL) Recovery Mechanism

The proposed method uses W-nodes and IPL nodes which perform different functions to help in managing topology. Table 3 summarizes the functions that are executed in the controller and w- nodes and IPL nodes, and the state that each node should be in.

When a recovery request (created by Algorithm 1) is received by the IPL network, the network will start a coordinated recovery process. The recovery request will be in the form of:

Req
 $= \{n_i, n_j, (x_i, y_i), (x_j, y_j), E_i, E_j, best_node, best_distance\}$,
 Where, n_i and n_j denote the affected nodes, (x_i, y_i) and (x_j, y_j) represent their positions, and E_i, E_j denote their residual energy.

A weighted midpoint of nodes n_i and n_j calculated by the energies, is used to compute the target recovery location:

$$x_t = \frac{E_i x_i + E_j x_j}{E_i + E_j}, y_t = \frac{E_i y_i + E_j y_j}{E_i + E_j} \tag{8}$$

This formulation guarantees that the target position is biased with the node having higher residual energy and this improves stability of the links and increases network lifetime.

The IPL nodes compare themselves to their suitability by calculating the distance between themselves and the target location:

$$dist_1 = \sqrt{(x_s - x_t)^2 + (y_s - y_t)^2} \tag{9}$$

Where (x_s, y_s) is the current IPL node position.

The distance between the currently selected best IPL nodes is maintained as:

Table 3. IPL Components

Component	Functions Executed	State Maintained
SDN Controller	Topology view maintenance; failure confirmation; recovery request prioritization	Global topology map; active link list; failure queue
W-node	Heartbeat broadcasting; Algorithm 1 (link monitoring); Algorithm 2 (future link life); recovery request generation	Own location, speed, direction, energy; neighbour heartbeat timestamps; link life history
IPL Node	Algorithm 3 (request routing and bridging); distance computation; bridging mode management	Own location, energy, bridging status; ring neighbour IDs; current request R

$$dist_2 = \sqrt{(x_p - x_t)^2 + (y_p - y_t)^2} \quad (10)$$

Where (x_p, y_p) is the location of the previously selected candidate IPL node.

If $dist_1 < dist_2$, the IPL node then adjusts the recovery request to be the best candidate.

This procedure makes sure that the node in the IPL with the nearest distance to the target location is chosen to connect the links between nodes n_i and n_j .

Algorithm 3. Functions of IPL Network

Input:

Recovery request *Req*, IPL ring size *N*

Output:

Selected IPL node or NACK

Runs at:

Each IPL node

Data Structure

Req
= {*n1, n2, target_location, best_node, best_distance*}

Procedure

1. If IPL node is in bridging mode, then

Forward request to next IPL node

Return

2. Compute target location (x_t, y_t) using Eq. (8)

3. Compute:

$$dist_1 = \sqrt{(x_s - x_t)^2 + (y_s - y_t)^2}$$

4. If $dist_1 < Req.best_distance$, then

Update

$Req.best_distance = \text{Current IPL node}$

$$Req.best_distance = dist_1$$

5. Forward request to next IPL node

6. If the request returns to the originating IPL node, then

7. If $Req.best_node \neq NULL$, then

Selected IPL node moves to (x_t, y_t)

Establish link between n_i and n_j

Return

8. Else

Return NACK

This algorithm ceases once the ring of the IPL has been traversed. The number of IPL nodes visited is limited by the size of the ring *N*, so naturally terminates. This process maintains a seamless recovery in the system in the event of node and link failures by identifying an IPL node with the shortest distance to the

optimal recovery point, and by consistency with link monitoring (Algorithm 1) and prediction model (Algorithm 2).

4.6 Design Rationale

The following subsection explains the justification of the main design decisions of the proposed IPL-based recovery framework. The IPL nodes are arranged in logical ring topology to provide structured and deterministic coordination in the recovery. The ring also guarantees that every IPL node is contacted at most once in the process of relay selection, which makes traversal complexity bounded ($O(N\{IPL\})$), coordination overhead minimal, and redundant message transmissions and contention are avoided. Yet this design brings about dependency on IPL node connectivity, and in case of the break of the ring through IPL node failure, traversal can be altered. IPL nodes in this work are considered to be reliable and dealing with failures of the same will be addressed in future research.

The proposed method will choose the closest IPL node to the target recovery point so that there is minimization of relocation distance which directly translates to a reduction in recovery time and movement energy. This strategy is effective but can cause unequal use of IPL nodes in situations with frequent failures and does not directly discuss load balancing or congestion. This is partly alleviated by the ring-based traversal which makes sure that every IPL node is considered in the process of selection. Also, mobility-assisted placement of relay allows IPL nodes to dynamically respond to topology changes, which is better placed to restore connectivity than its stationary relay counterparts. It is further balanced by using an energy-weighted midpoint to give stability and energy efficiency since the placement of the relay is biased towards the nodes with more residual energy. In general, its design is focused on rapid recovery and minimal coordination cost, but recognizes that performance can be poor in the event of parallel failures or prediction errors. This combined scheme is a trade-off between recovery speed, energy efficiency as well as scalability in dynamic SDWSN environments.

5. Results and Discussion

This section discusses the proposed approach under various criteria of performance, which include node recovery time, node recovery success rate, packet delivery ratio (PDR), latency, energy consumption and lifespan of the SDWSN. All these parameters are evaluated with respect to a different number of nodes. They are compared against the existing benchmarked methods including IFT, Fed-TSN, P4Neighbor and LCD. All benchmarked methods (IFT, Fed-TSN, P4Neighbor and LCD) have been applied to the same experimental conditions in order to provide a fair comparison. Both

methods follow identical network topology, packeting size, speed of traffic generation, controller settings, and failure injection methodology. This is to ascertain that variations in performance that are being realized in the experiments are as a result of the recovery mechanisms and not variations in the experiment setup. Besides comparison to baseline procedures, ablation study is performed to isolate the contribution of individual elements of IPL approach proposed. The set-up of the experiment below realizes the system model in section 3.1 with the particular parameter settings.

5.1 Experiment Design

Mininet 2.2 is used alongside Ubuntu 22.04, the SDN controller is Floodlight, and the control-plane communication is through OpenFlow. All the experiments are carried out on a simulated size of 200 m x 200 m. The subsequent sub-sections describe the topology generation process, mobility model, radio model, and failure injection process.

5.1.1 Topology Generation and Node Placement

The network has a total of 150 nodes comprising of 120 wireless nodes (w-nodes) and 30 IPL nodes. The forwarding feature is designed as a part of the w-nodes themselves, and no forwarding equipment is utilized. Each node is involved in forwarding packets and IPL nodes also offer recovery. In order to obtain a valid initial topology, node positions are repeated until the network created is a connected graph within the transmission range of 40 m. This prevents biasness because of initially unrelated deployments.

5.1.2 Mobility Model

W-nodes are based on the Random Waypoint (RWP) mobility model whose parameters are as follows: minimum speed = 0.5 m/s, maximum speed = 2.0 m/s, and pause time = 5 seconds. A node pauses 5 seconds to choose another random destination within the simulation boundary at each waypoint. IPL nodes are mobile on demand: they do not move until they are deployed to a bridging location, and then travel at a constant speed of 2.0 m/s towards the target destination.

5.1.3 Radio and Channel Model

There is a transmission range of 40 m in each node, as indicated in Table 2. The links are assumed to be binary: two nodes can only communicate when they are at a Euclidean distance that is below 40 m and when residual energy is positive on both nodes. The channel capacity is 10 Mb/s per link, propagation delay is equally distributed in the range [1, 100] ms as described in Table 2. No further fading or path-loss model is employed; reproducibility is obtained with the binary disk model and

evaluation is concentrated on recovery mechanism as opposed to physical layer.

5.1.4 Failure Injection

Mininet is used to inject node failures by disabling a network interface of a node randomly with time values generated by a Poisson process with the average inter-failure time of 250 seconds. The number of failures per experiment is an average of 10 during the period of 2500 seconds of the simulation. In the case of multiple-failures, a maximum of 3 parallel failures are induced in the same time span of 3 seconds to examine the recovery process in parallel as it was outlined in Section 4.2.3. The failed node fails permanently in the course of the simulation run; all the recovery is only done through the IPL mechanism. Only wireless nodes (w-nodes) are introduced failures to make sure that the recovery mechanism can be tested without being subject to the failures of relay nodes.

5.1.5 Traffic Model

Constant bit rate (CBR) flows between randomly chosen source-destination pairs are used to obtain traffic. At a rate of 35 packets per second, each node produces packets of 128 bytes. To be fair, traffic patterns are the same in all the experimental runs and in all compared methods.

5.1.6 Statistical Repeatability

The experiments were repeated 10 times using 10 independent simulation runs with different random seeds for node placement, mobility generation, and failure injection. The values shown in all figures (Figures 3–8) represent the mean values obtained from these 10 runs. Error bars indicate one standard deviation ($\pm 1\sigma$) from the mean. The seed values are fixed and will be made available together with the simulation scripts to ensure reproducibility.

5.2 Baseline Implementation and Experimental Fairness

Each of the four baseline techniques [IFT, Fed-TSN, P4Neighbor, and LCD] was re-executed in the identical environment of Mininet 2.2/Floodlight/OpenFlow as that here with the IPL approach. All methods were applied uniformly to the same simulation area, nodes placement, mobility model (RWP, with identical speed and pause parameter), radio model, traffic parameters (128-byte packets, 35 packets/s) and failure injection schedule. Each of the methods did not obtain a special arrangement.

- **IFT [9]:** IFT scheme was developed in WSN-IIoT. It was redefined on the IFT [9]: SDWSN context by overlaying its node-failure and link-failure events onto the event-processing model

of Floodlight controller. The threshold parameters have been initialized to the values that are reported in [9] and confirmed to be active under SDWSN simulation parameters. The identical 2 J base energy and the same transmission range (40 m) like in the proposed method was used.

- **Fed-TSN [11]:** Fed-TSN utilizes federated learning to forecast link failures on a time-sensitive network. The federated learning model was modified to execute on Floodlight controller. The model aggregation had been done at 10-second intervals and reported in [11]. Local model updates are performed in all nodes, and the global model is aligned at the controller. The mobility and failure injection processes were the same.
- **P4Neighbor [10]:** P4Neighbor is a programmable switch based on recovering link failures fast. Floodlight flow rule pre-installation mechanism was used to simulate P4 switch behaviour in the Mininet environment. Precomputation of backup paths was done in [10]. The traffic matrix (35 packets/s) was used along with the same packet size (128 bytes).
- **LCD [12]:** LCD defines an independent layer of control forwarding. This was done through the assignment of a group of w-nodes as control-forwarding nodes in Floodlight as are in the LCD architecture. The control nodes were proportionately assigned depending on the size of the network as proposed in [12]. There were no other benefits in the computation of the path given to LCD compared to the other techniques.

Table 4. Simulation Metrics

Attributes	Specifications
Network entities	150
Data channels	172
Forwarding devices	52
Channel capacity	10 MB
Rate	20 bps – 250 kbps
Delay	1-100 (in ms)
Base energy level	2J
Packet size	128 Bytes
Packet range	35 packets / s
Sensing coverage	20 m
Transmission range	40 m
Simulation duration	2500 (in seconds)

Table 4 identifies the typical simulation parameters that are used in all methods. Each of the baseline methods was re-run in the same SDWSN

simulation environment to remove differences that might occur due to dissimilar initial baseline evaluation configurations. Thus, every reported outcome can be directly compared in the same conditions, and the performance differences are indicative only of the efficiency of the corresponding recovery processes.

5.3 Result Analysis

This section presents a detailed analysis of the performance of the proposed IPL framework using key metrics, including node recovery time, packet delivery ratio, latency, energy consumption under varying network conditions

Node recovery time: This is a metric used to indicate the time of recovery of the connectivity between two nodes following the occurrence of failure. Figure 2 depicts the node recovery time of the proposed approach as compared to the growing number of nodes. The different number of nodes is plotted in X-axis and node recovery time was plotted in the Y-axis. The node failure recovery mechanism, which proposes IPL topology, is used to implement the IPL nodes, which are used to restore the connection with the node failure recovery mechanism through the selection of the IPL node based on energy and distance. Subsequently, the proposed solution saves 8%, 16%, 21%, and 26% of node recovery time than IFT, Fed-TSN, P4Neighbor and LCD solutions, respectively.

Node recovery success rate: Node recovery success rate gives the rate of node recovery over all the failed nodes. Figure 3 illustrates the success rate of node recovery that is followed by the proposed method. The suggested IPL-based topology model assumes continuous monitoring of the condition of the link life between the two nodes depending on their energy content, speed, distance, and their past records. The proactive and predictive abilities of the proposed solution improve the node recovery success rate by over 6%, 8%, 11%, and 17% of the versions of benchmarked solutions IFT, Fed-TSN, P4Neighbor, and LCD, respectively.

The proposed IPL strategy decreases node recovery time since the predictive monitoring in Algorithm 2 causes IPL repositioning prior to the complete collapse of the link reduction the reactive detection delay that causes the recovery time in all four baselines.

IFT has comparatively better recovery time among the baselines because of its node level tracking of failures; it fails to proactively position its relays and this contributes to extra routing overhead which limits its performance as compared to the proposed IPL approach.

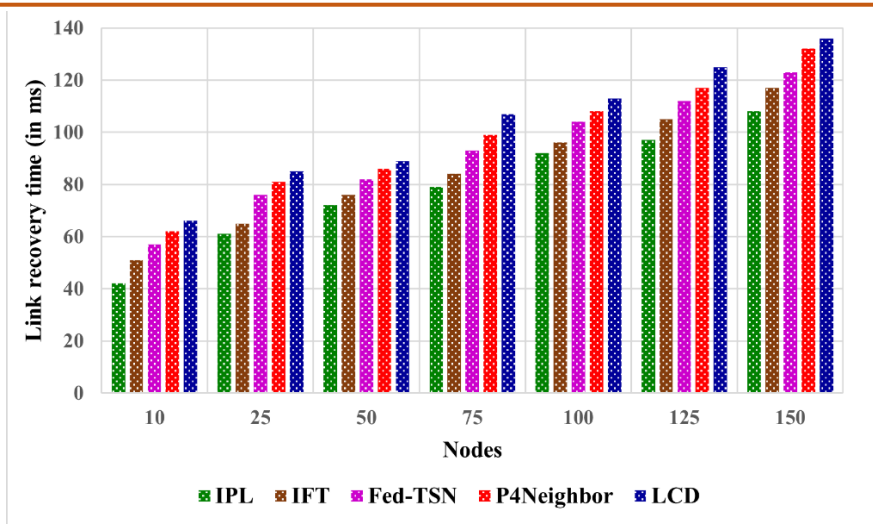


Figure 2. Node recovery time versus number of nodes for IPL and benchmark methods

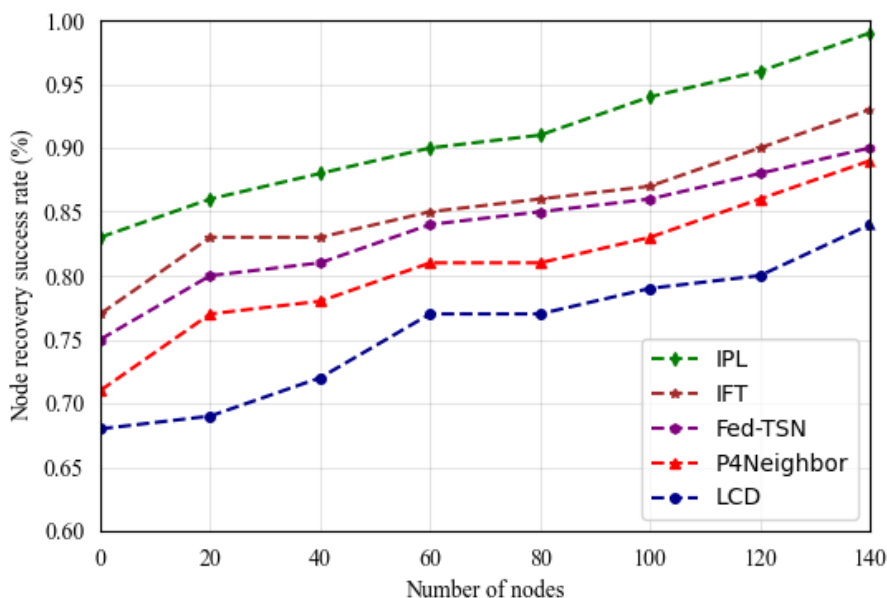


Figure 3. Node recovery success rate of IPL and benchmark methods under varying network sizes.

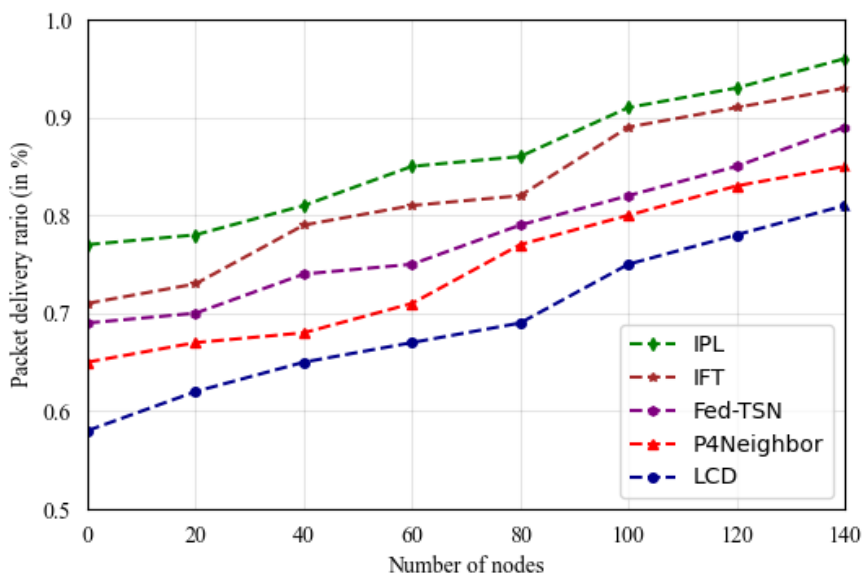


Figure 4. Packet delivery ratio (PDR) comparison for IPL and existing recovery schemes.

In their time-sensitive network environment, Fed-TSN [11] indicated a prediction accuracy of around 91% in link failures, but it has a federated network model update rate of 10 seconds as a structural latency floor, which is not sufficiently fast to recover faster in dynamic SDWSN systems. P4Neighbor [10] and LCD [12] are simply reactive and hence have the longest recovery times.

The IPL strategy has a performance limit where all IPL nodes are in bridging mode at the same time; in this case, the NACK response (Algorithm 3, Step 2.1) will be generated and recovery will be reverted to a reactive mechanism.

Packet delivery ratio: This metric measures the number of packets that were delivered successfully as compared to the total packet delivery after the node recovered after being affected by a network failure. This makes the proposed method reliable even when there is a failure of the node. The suggested method employed proactive and predictive based IPL nodes so as to maintain a stable and reliable communication among nodes. The active control of link life and quick moving of IPL nodes following the failure, aids in decreasing the loss in packets resulting in the failure. Figure 4 is a representation of the ratio of the packets delivery that the IPL-based topology method has achieved. The proposed approach outperforms the benchmarked approaches namely IFT, Fed-TSN, P4Neighbor, and LCD by 4%, 9%, 13%, and 19% respectively, with respect to packet delivery ratio.

IPL enhances PDR by ensuring proactive link continuity Proactive recovery by positioning an IPL bridge node ahead of the break in the link prevents the loss of packets that form a packet loss window during reactive recovery in the baseline methods. IFT [9] has a

reported PDR of almost 4% higher than that of Fed-TSN in their own reported environment [9]; in our re-implementation with SDWSN, the difference between IFT and IPL is even greater since the mobile SDWSN environment exhibits frequent link failures as compared to the IIoT environment, which IFT was intended to operate with. LCD [12] reports the lowest PDR in our analysis since the performance of LCD is sensitive to the density of the control nodes, in our 150-node network with 30 IPL nodes operating independently, the control forwarding layer in LCD is under-provided, resulting in control bottlenecks in case of multiple independent failures.

Latency: This is a number that is used to determine the time that is mandatory when a packet travels between its sources to the destination so as to determine the efficiency of the network in the process of recovering the node. Figure 5 provides the latency performance of the offered approach. The approach proposed has a 9%, 16%, 22%, and 27% lower latency than the current benchmarked approaches IFT, Fed-TSN, P4Neighbor, and LCD respectively. The major technique of IPL is that it decreases the time spent by packets being re-transmitted over the endpoint: since the connections are already being preemptively kept alive, packets do not sit in the transmission queue until a failed link is recovered. In reactive methods like P4Neighbor [10] and LCD [12], recovery window (between failure detection and path restoration) causes buffered packets to be resent and ensures queuing delay which continues even after recovery. The IPL method has been shown to be most efficient in the case when the latency is to be minimized, and the failure rates are moderate (about one failure per 250 seconds in this case study).

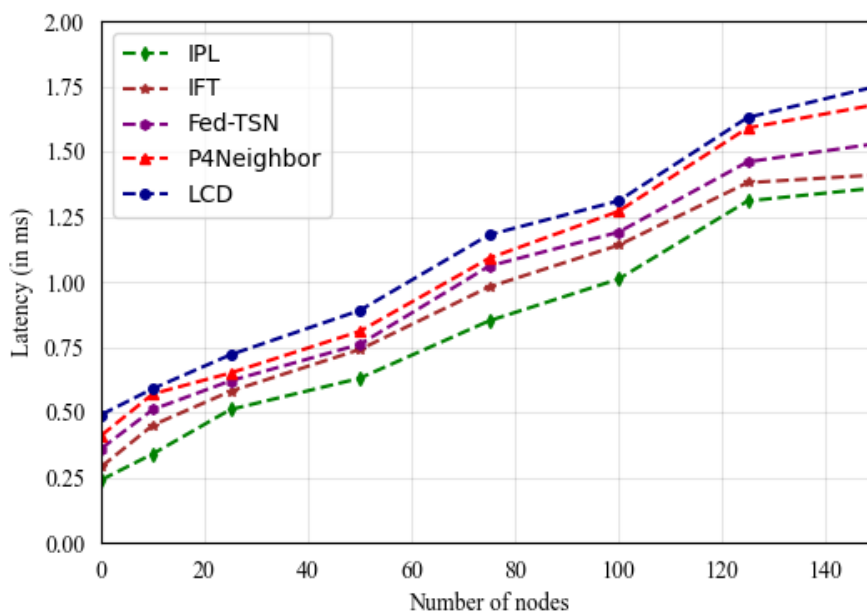


Figure 5. End-to-end latency comparison of IPL with benchmark recovery methods

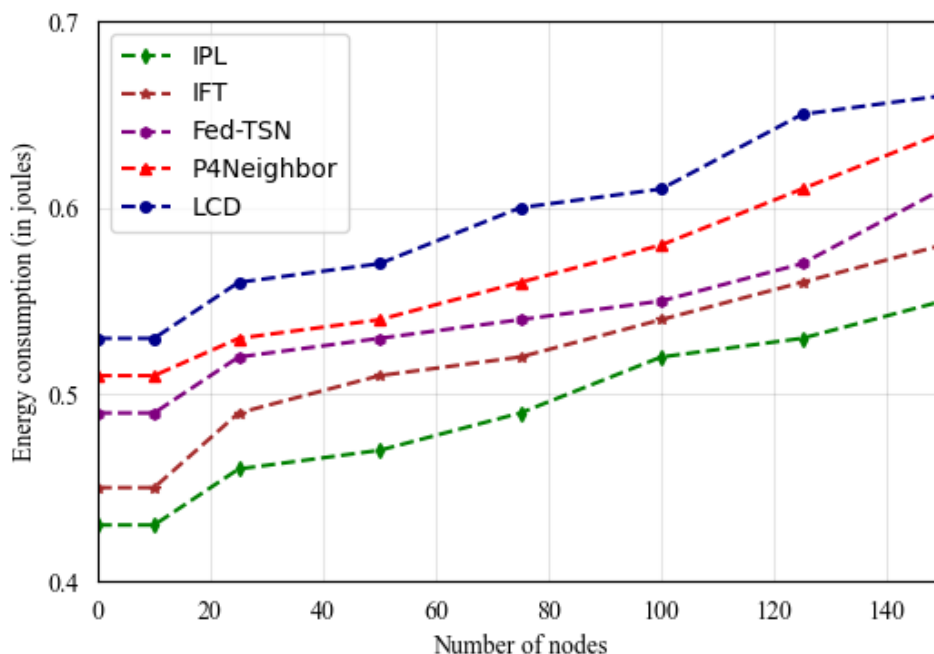


Figure 6. Energy consumption of IPL and benchmark methods under different node densities.

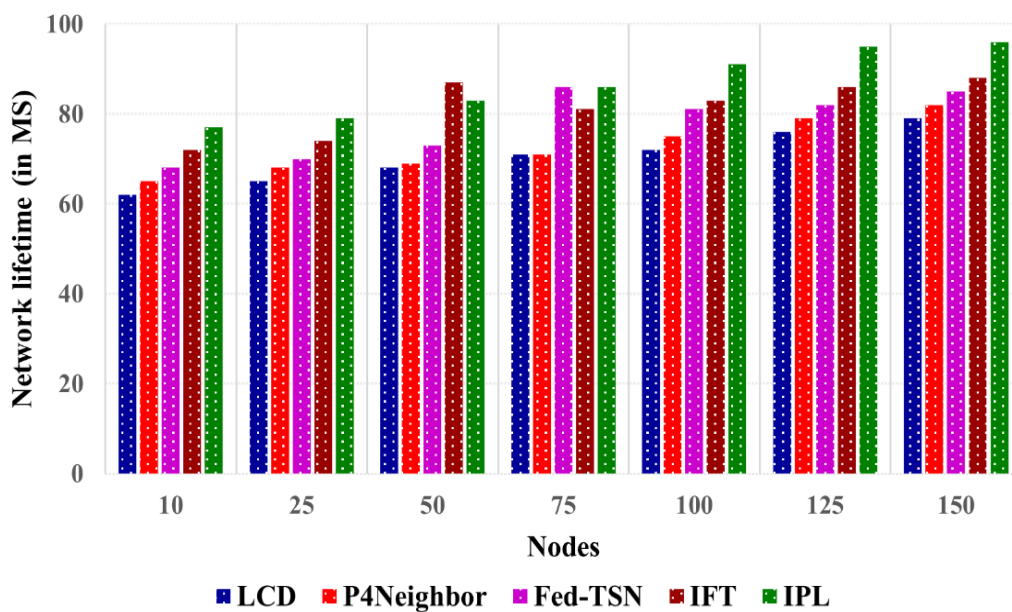


Figure 7. Network lifetime comparison between IPL and existing recovery schemes.

At extremely high failure rates, when the IPL ring is flooded with bridging tasks, the latency saving is reduced since not all recovery requests can be immediately served, and some have to queue at the controller.

Energy consumption: The energy deployed by the node is recorded in crossing preceding the failure recovery and crossing following the failure recovery in the movement of the packet onward towards the destination. The energy usage of IPL-based approach is highlighted by Figure 6. In the IPL-based topology approach, IPL nodes are made lightweight, owing to the predictive nature of IPL nodes, they deal with node failure before the node is dead or inactive and hence the

energy use is minimized. The proposed solution enhances a consistent interrelationship among nodes that greatly help to minimize the occurrence of node failure. This helps avoid the continuous retransmission of packets over a long distance. The proposed approach reduces energy consumption by approximately 5%, 10%, 14%, and 18% compared to well-known benchmarked approaches IFT, Fed-TSN, P4Neighbor, and LCD, respectively.

The IPL method helps to save energy as it prevents node failure before the node is totally inactive. This eliminates the energy-consuming cost of retransmission of packets in long alternative paths constantly incurred by other methods in case of failure.

Kaur and Chanak reported that the IFT scheme [9] has a greater energy consumption because of its dual-phase detection scheme [9]; which is justified in our findings. .

The main factors that make IPL have an energy advantage are its lightweight heartbeat (one message per second per node) and the fact that the complete topology is not precomputed after each failure. Boundary-condition Large-scale deployments (greater than about 150 nodes) in which the frequency of IPL node repositioning grows and where the retransmission savings are starting to be offset by the cost of movement energy is the boundary condition of energy performance.

Network lifetime: Figure 7 illustrates the network lifetime that has been achieved through the IPL-based method. The collaboration offered in the IPL topology model has the effect of enhancing the network lifetime by reducing the number of node failure incidences and actively addressing the node failures. Also, IPL nodes balance the energy levels throughout the network, this causes the prevention of premature reduction of single nodes, and this prolongs the life span of the networks. The IPL topology proposed roughly extends the network lifetime by 7%, 12%, 16%, and 19% with reference to benchmarked methods IFT, Fed-TSN, P4Neighbor and LCD respectively.

5.3.1 Discussion

The observed increase in performance of the proposed IPL approach is mainly because of its proactive and systematic recovery mechanism in case of varied conditions of failure. When the scenario of single link degradation is to be considered, it is possible to make predictions regarding the lifetime of the links which allows the possibility to reposition the IPL nodes prior to the disconnection. This lowers the window of packet loss related to reactive approaches and enhances the ratio of the packet delivery (PDR). The recovery queue performed in a priority order along with ring-based coordination is also useful in the case of multiple failures, where coordinating recovery is performed to ensure high impact failures are addressed first before the network is disconnected.

Latency reduction is attained through elimination of retransmission delay. Reactive methods like P4Neighbor [10], LCD [12] buffer or drop packets

whilst alternative paths are found. On the other hand, IPL approach has connectivity by pro-actively bridging, thus minimizing queuing delay and retransmission overhead. Likewise, much less energy would be consumed by not using long multi-hop rerouting and packet recovery on failure after failure and mobile IPL nodes can allow local and efficient link recovery.

The success of the suggested strategy, however, hinges on the conditions of the network and system limitations. The availability of IPL nodes is a bottleneck under several concurrent failures, since they are served in a sequence in the ring, which may add to the recovery latency. Also, nearest-node selection strategy, despite reducing the movement cost, can result in disproportionate use of IPL nodes and does not directly consider load balancing. Suboptimal placement of relays can be caused due to inaccuracy of prediction in highly dynamic mobility situations. These observations demonstrate that though the proposed IPL framework is effective when the failure rates are moderate and the mobility is controlled, the framework can be inefficient when the failure density is large or the topology changes rapidly. This underscores the efficiency of proactive recovery and the availability of the resources in the dynamic SDWSN environments.

5.4 Ablation Study

In order to decompose the contribution of specific elements of the proposed IPL approach, ablation study is designed to selectively disable some of the key mechanisms and maintain all other parameters as the same as during the initial set-up. The variants to be considered are as follows:

- IPL-Full: Full suggested approach (ring structure and prediction and closest selection)
- IPL-NoPred: This is where the prediction element is eliminated; the recovery is only initiated once failure detection has been detected (reactive mode).
- IPL-Random: IPL selection is disabled; IPL node is chosen at random in the ring.
- IPL-NoRing: Ring structure eliminated; IPL nodes are uncoordinatedly traversed.

Table 5. Ablation study results

Method	Recovery Time ↓	PDR ↑	Latency ↓	Energy ↓
IPL-Full	92	0.88	0.81	1.01
IPL-NoPred	96 (+4.3%)	0.83 (-5.7%)	0.84 (+3.7%)	1.14 (+12.9%)
IPL-Random	104 (+13.0%)	0.81 (-8.0%)	0.88 (+8.6%)	1.19 (+17.8%)
IPL-NoRing	113 (+22.8%)	0.72 (-18.2%)	0.97 (+19.8%)	1.31 (+29.7%)

Variants are all run in a common simulation environment with the same topology, traffic model, mobility and failure injection process to achieve fair comparison.

Table 5 present an ablation study result and values in parentheses indicate percentage change relative to IPL-Full. The ablation variants are approximated using corresponding baseline methods to reflect the impact of removing specific components of the IPL framework. The outcomes of ablation prove that all the elements of the proposed IPL approach are important contributors to the performance in general:

Prediction Component (IPL-NoPred): The elimination of prediction makes recovery time and latency to be higher in reactive behavior. This proves that the link lifetime estimation should be proactive in minimizing disruption time.

Nearest IPL Selection (IPL-Random): Random selection of IPL nodes results in suboptimal placement of relays, which means more energy is spent and there is more recovery delay. This confirms the significance of optimal selection which is based on distance.

Ring Structure (IPL-NoRing): With the removal of the ring coordination mechanism, the worst performance is achieved because the request propagation is not structured and IPL is not well used. This brings out the importance of the ring topology in providing systematic and strife free recovery.

On the whole, the IPL-Full setup is always more effective than any ablated one, which proves that the improvements in performance are determined not by the fact that more relay options are available but by the synergistic effect of predictive monitoring, smart IPL choice and organized ring coordination.

The given improvements in the performance of the suggested IPL approach can be even better explained by comparing its design principles with the current ones. In contrast to IFT [9] which uses reactive rerouting by detecting link failures, IPL incorporates proactive link lifetime prediction so that the recovery actions may precede the occurrence of link disconnection. Equally, Fed-TSN [11] also uses predictive modeling, but with longer update rates because of federated learning aggregation, which restricts its responsiveness in very dynamic environments. P4Neighbor [10] and LCD [12] concentrate on rapid recovery based on precomputed paths or control-layer forwarding, but they lack mobility-aware prediction as well as dynamic relay repositioning, so they do not perform so well in the such conditions of high topology change frequency.

Conversely, the IPL approach proposed integrates three complementary mechanisms, which are predictive link monitoring, optimal relay selection based

on distance and energy, and coordination structure through a ring topology. This combined design gives IPL an opportunity to reduce recovery delay, decrease packet loss, as well as establish steady connectivity in the dynamic conditions. Thus, the performance improvements in the outcomes can be attributed to the inherent restrictions of the current strategies to be found in the literature and confirm the efficiency of the suggested methodology.

6. Conclusion

In this paper, an Intelligent Predefined Lightweight (IPL) topology-based recovery system of managing various node failures in software-defined wireless sensor networks (SDWSNs) was introduced. The three key elements incorporated in the proposed framework are (i) predictive link lifetime estimation depending on the energy and mobility parameters, (ii) distance aware optimal selection of relays by mobile IPL nodes, and (iii) a ring-based coordination mechanism that can be determined to recover in a deterministic and conflict free manner. The results of the simulation show that the IPL approach is always better than the baseline strategies that include IFT, Fed-TSN, P4Neighbor, and LCD when all of them operate under the same experimental conditions. The enhancements, up to 26% recovery time savings, 19 % improvement of the packet delivery ratio and 19 % longer network lifetime are mainly due to the proactive link monitoring, minimized retransmission overhead, and localized recovery using relays.

The findings also suggest that the suggested technique is especially useful in the dynamic setting that has moderate failure rates, where predictive monitoring and coordinated relay position can ensure constant connectivity. But high failure density or performance is poor when it comes to high IPL availability or when recovering requests can be delayed and inaccuracies in prediction can influence the placement of the relay. The future work will be aimed at adaptive resource allocation of IPL, load based relay selection, and better mobility aware prediction models to achieve better performance in extreme conditions of the network. Moreover, the testbed validation will be performed in the real world to determine the strength of the proposed framework in real SDWSN applications.

References

- [1] M. Trigka, E. Dritsas, *Wireless Sensor Networks: From Fundamentals and Applications to Innovations and Future Trends*. IEEE Access, IEEE, 13, (2025) 96365–96399. <https://doi.org/10.1109/ACCESS.2025.3572328>
- [2] M. Ayaz, M. Ammad-Uddin, I. Baig, E.H.M. Aggoune, *Wireless Sensor's Civil Applications*,

- Prototypes, and Future Integration Possibilities: A review. *IEEE Sensors Journal*, IEEE, 18(1), (2018) 4–30.
<https://doi.org/10.1109/JSEN.2017.2766364>
- [3] M.A. Jamshed, K. Ali, Q.H. Abbasi, M.A. Imran, M. Ur-Rehman, Challenges, Applications, and Future of Wireless Sensors in Internet of Things: A Review. *IEEE Sensors Journal*, 22(6), (2022) 5482–5494.
<https://doi.org/10.1109/JSEN.2022.3148128>
- [4] D. Kreutz, F.M.V. Ramos, P.E. Verissimo, C.E. Rothenberg, S. Azodolmolky, S. Uhlig, Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, IEEE, 103(1), (2015) 14–76.
<https://doi.org/10.1109/JPROC.2014.2371999>
- [5] A. Narwaria, A.P. Mazumdar, Software-Defined Wireless Sensor Network: A Comprehensive Survey. *Journal of Network and Computer Applications*, 215, (2023) 103636.
<https://doi.org/10.1016/j.jnca.2023.103636>
- [6] J.A.P. Fernández, L.J.G. Villalba, T.H. Kim, Software Defined Networks in Wireless Sensor Architectures. *Entropy*, 20(4), (2018) 225.
<https://doi.org/10.3390/e20040225>
- [7] G.A. Abed, Enhancing Wireless Sensor Networks Features Using Software-Defined Networking Techniques and ACO Algorithms. *Iraqi Journal for Computer Science and Mathematics*, 5(3), (2024) 489–500.
<https://doi.org/10.52866/ijcsm.2024.05.03.030>
- [8] B.B. Letswamotse, R. Malekian, C.-Y. Chen, K.M. Modieginyane, Software Defined Wireless Sensor Networks (SDWSN): A Review on Efficient Resources, Applications and Technologies. *Journal of Internet Technology*, 19(5), (2018) 1303–1313.
<https://doi.org/10.3966/160792642018091905003>
- [9] G. Kaur, P. Chanak, An Intelligent Fault Tolerant Data Routing Scheme for Wireless Sensor Network-Assisted Industrial Internet of Things. *IEEE Transactions on Industrial Informatics*, IEEE, 19(4), (2023) 5543–5553.
<https://doi.org/10.1109/TII.2022.3204560>
- [10] J. Xu, S. Xie, J. Zhao, P4Neighbor: Efficient Link Failure Recovery with Programmable Switches. *IEEE Transactions on Network and Service Management*, 18(1), (2021) 388–401.
<https://doi.org/10.1109/TNSM.2021.3050478>
- [11] V. Balasubramanian, M. Aloqaily, M. Reisslein, Fed-TSN: Joint Failure Probability-Based Federated Learning for Fault-Tolerant Time-Sensitive Networks. *IEEE Transactions on Network and Service Management*, IEEE, 20(2), (2023) 1470–1486.
<https://doi.org/10.1109/TNSM.2023.3273396>
- [12] M. Huang, B. Yu, LCD: Light-Weight Control Model for Data Plane in Software-Defined Wireless Sensor networks. *Transactions on Emerging Telecommunications Technologies*, 30(6), (2019) e3557.
<https://doi.org/10.1002/ett.3557>
- [13] A. Menaceur, H. Drid, M. Rahouti, Fault Tolerance and Failure Recovery Techniques in Software-Defined Networking: A Comprehensive Approach. *Journal of Network and Systems Management*, 31(4), (2023) 83.
<https://doi.org/10.1007/s10922-023-09772-x>
- [14] T. Semong, T. Maupong, A.M. Zungeru, O. Tabona, S. Dimakatso, G. Boipelo, M. Phuthogo, A review on Software Defined Networking as a Solution to Link Failures. *Scientific African*, 21, (2023) e01865.
<https://doi.org/10.1016/j.sciaf.2023.e01865>
- [15] S. Alhiyari, S.H.A.B. Hamid, N.N. Daud, A Survey of Link Failure Detection and Recovery in Software-Defined Networks. *Computers, Materials & Continua*, 82(1), (2025) 103–137.
<https://doi.org/10.32604/cmc.2024.059050>
- [16] D. Franco, M. Higuero, A. Sanz, J. Unzilla, M. Huarte, vFFR: A Very Fast Failure Recovery Strategy Implemented in Devices with Programmable Data Plane. *IEEE Open Journal of the Communications Society*, IEEE, 5, (2024) 7121–7146.
<https://doi.org/10.1109/OJCOMS.2024.3493417>
- [17] T.S. Wong, S.S.W. Lee, Design of an In-Band Control Plane for Automatic Bootstrapping and Fast Failure Recovery in P4 Networks. *IEEE Transactions on Network and Service Management*, IEEE, 20(3), (2023) 3612–3629.
<https://doi.org/10.1109/TNSM.2023.3242222>
- [18] A. Sgambelluri, D. Scano, R. Morro, F. Cugini, J. Ortiz, J.M. Martinez, E. Riccardi, P. Castoldi, P. Pavon, A. Giorgetti, Failure Recovery in the MANTRA Architecture with an IPoWDM SONiC Node and 400ZR/ZR+ Pluggables. *Journal of Optical Communications and Networking*, 16(5), (2024) B26–B34.
<https://doi.org/10.1364/JOCN.514179>
- [19] K. Qiu, J. Zhao, X. Wang, X. Fu, S. Secci, Efficient Recovery Path Computation for Fast Reroute in Large-Scale Software-Defined Networks. *IEEE Journal on Selected Areas in Communications*, IEEE, 37(8), (2019) 1755–1768.
<https://doi.org/10.1109/JSAC.2019.2927098>

- [20] T.T. Hieu, N. Kitsuan, Swift Recovery: An Innovative Routing Approach for Multi-Failure Protection in Software-Defined Networks. *IEEE Access*, IEEE, 12, (2024) 158685–158702. <https://doi.org/10.1109/ACCESS.2024.3487211>
- [21] J. Chen, F. Yan, D. Li, S. Chen, X. Qiu, Recovery and Reconstruction of Multicast Tree in Software-Defined Network: High Speed and Low Cost. *IEEE Access*, IEEE, 8, (2020) 27188–27201. <https://doi.org/10.1109/ACCESS.2020.2970275>
- [22] N. Vuppapapati, T.G. Venkatesh, Candidate Selection Algorithms for Hybrid IP/SDN Networks with Multi-Link Failures. *IEEE Transactions on Network and Service Management*, IEEE, 22(2), (2025) 1219–1231. <https://doi.org/10.1109/TNSM.2024.3504534>
- [23] M.Y. Daha, M.S.M. Zahid, B. Isyaku, A.A. Alashhab, CDRA: A Community Detection based Routing Algorithm for Link Failure Recovery in Software Defined Networks. *International Journal of Advanced Computer Science and Applications*, 12(11), (2021) 712-722. <https://doi.org/10.14569/IJACSA.2021.0121181>
- [24] J. Ali, G. Shan, N. Gul, B.H. Roh, An Intelligent Blockchain-based Secure Link Failure Recovery Framework for Software-defined Internet-of-Things. *Journal of Grid Computing*, 21(4), (2023) 57. <https://doi.org/10.1007/s10723-023-09693-8>
- [25] B. Isyaku, K.B.A. Bakar, W. Nagmeldin, A. Abdelmaboud, F. Saeed, F.A. Ghaleb, Reliable Failure Restoration with Bayesian Congestion Aware for Software Defined Networks. *Computer Systems Science and Engineering*, 46(3), (2023) 3729–3748. <https://doi.org/10.32604/csse.2023.034509>
- [26] S. Wang, Z. Jiang, X. Li, J. Ma, K-Backup: Load- and TCAM-Aware Multi-Backup Fast Failure Recovery in SDNs. *IEEE/ACM Transactions on Networking*, 32(4), (2024) 3347–3360. <https://doi.org/10.1109/TNET.2024.3386091>
- [27] S. Petale, J. Thangaraj, Link Failure Recovery Mechanism in Software Defined Networks. *IEEE Journal on Selected Areas in Communications*, 38(7), (2020) 1285–1292. <https://doi.org/10.1109/JSAC.2020.2986668>
- [28] H. Nurwarsito, G. Prasetyo, Implementation Failure Recovery Mechanism using VLAN ID in Software Defined Networks. *International Journal of Advanced Computer Science and Applications*, 14(1), (2023) 709-714. <https://dx.doi.org/10.14569/IJACSA.2023.0140178>
- [29] Y. Wang, S. Feng, H. Guo, X. Qiu, H. An, A Single-Link Failure Recovery Approach Based on Resource Sharing and Performance Prediction in SDN. *IEEE Access*, IEEE, 7, (2019) 174750–174763. <https://doi.org/10.1109/ACCESS.2019.2957141>
- [30] H.C. Hsieh, M.L. Chiang, T.Y. Chang, Improving the fault-tolerance of software-defined networks with dynamic overlay agreement. *Cluster Computing*, 24(3), (2021) 2597–2614. <https://doi.org/10.1007/s10586-020-03224-w>

Author Contributions Statement

S. Sathish: Conceptualization, Methodology, Software, Writing - Original Draft. J. Poongodi: Data Curation, Investigation, Writing - Review & Editing. G. Sudha: Formal Analysis, Visualization, Supervision. K. Satheshkumar: Resources Validation, Writing - Review & Editing. All the authors have read and agreed to the published version of the manuscript.

Funding

The authors declare that no funds, grants or any other support were received during the preparation of this manuscript.

Competing Interests

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Data Availability

The data supporting the findings of this study can be obtained from the corresponding author upon reasonable request.

Has this article screened for similarity?

Yes

About the License

© The Author(s) 2026. The text of this article is open access and licensed under a Creative Commons Attribution 4.0 International License.