International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Sovereign Routing: An AI-Driven Framework for Disaster-Resilient Communication Networks

NOEL G, MRUDHUL NR, AKILESHWARAKRISHNAN A
Department of Artificial Intelligence and Data Science, Nehru Institute of Engineering and Technology
Coimbatore, Tamil Nadu 641105, India
Mentor: Kirubhakaran Marisamy, Department of Artificial Intelligence and Data Science

ABSTRACT

Natural disasters such as earthquakes, floods, cyclones, and landslides frequently cause severe damage to conventional communication infrastructure, leading to network outages that disrupt coordination between emergency responders, rescue teams, and affected communities. Maintaining reliable communication during such situations is critical for effective disaster management and timely rescue operations. Traditional routing protocols are typically designed for stable network environments and often fail to adapt when network nodes or communication links become unavailable. Research in disaster communication systems highlights the importance of decentralized, adaptive, and self-organizing network architectures that can sustain connectivity even when existing infrastructure is partially or completely damaged. This paper proposes Sovereign Routing, an intelligent and resilient communication framework designed to maintain connectivity in disaster-affected environments. The proposed system integrates artificial intelligence techniques, including machine learning and reinforcement learning, to monitor network conditions, analyze link reliability, and dynamically determine optimal routing paths for data transmission. By enabling autonomous decision-making within network nodes, the framework reduces dependence on centralized control and enhances the ability of the network to self-organize and recover from failures. The architecture supports distributed communication using lightweight ESP32 hardware nodes which can form temporary mesh networks for emergency deployment. The framework is evaluated using a simulation environment developed with Python, NetworkX, and NS-3 to model network disruptions and node failures realistically. Experimental results indicate that the proposed routing strategy achieves a Packet Delivery Ratio of 93%, reduces average end-to-end latency by 56%, cuts network recovery time by 66%, and nearly triples throughput compared with conventional routing methods. These findings demonstrate that AI-driven routing mechanisms can significantly improve communication resilience during disasters and provide a scalable, cost-effective solution for maintaining connectivity in highly disrupted environments.

Keywords: *Artificial Intelligence, Disaster Communication Networks, Reinforcement Learning, Adaptive Routing, Network Resilience, ESP32, IoT Communication, Emergency Communication Systems*

I. INTRODUCTION

Reliable communication plays a critical role in disaster management and emergency response operations. During natural disasters such as earthquakes, floods, hurricanes, and cyclones, conventional communication infrastructures — including cellular towers, internet services, and wired networks — are often damaged or completely disrupted. The loss of communication severely affects coordination between emergency responders, government agencies, and affected communities, which can delay rescue operations and increase the overall impact of the disaster. Therefore, establishing a resilient and adaptive communication system capable of operating in highly disrupted environments has become an increasingly important research challenge.

The consequences of communication failure during disasters can be catastrophic. Relief coordination breaks down, affected individuals are unable to signal their location to rescue teams, and government agencies lose situational awareness precisely when it is most urgently needed. Despite decades of investment in emergency communication infrastructure, the fundamental vulnerability of centralized, infrastructure-dependent systems remains unresolved. Every major disaster — from the 2011 Tōhoku earthquake and tsunami to the 2015 Nepal earthquake and the 2023 Türkiye–Syria earthquake sequence — demonstrated that even well-resourced nations cannot prevent widespread communication blackouts in the hours immediately following a major event.

A. Importance of Resilient Communication Systems

Traditional communication networks rely heavily on centralized infrastructure and static routing protocols to manage data transmission. These

systems are generally designed to operate under stable network conditions and typically determine communication paths using predefined parameters such as hop count, shortest distance, or fixed routing tables. However, in disaster scenarios where network nodes may fail unexpectedly and communication links become unstable, these conventional routing mechanisms struggle to adapt to the rapidly changing network environment. As a result, network congestion, packet loss, increased latency, and communication failures frequently occur, significantly reducing the reliability and efficiency of information exchange during critical situations.

The requirement for resilient communication is further underscored by the operational realities of disaster response. Rescue teams operate in environments where the network topology is unknown, constantly changing, and partially destroyed. The ability to maintain at minimum a low-bandwidth, high-reliability messaging channel — sufficient to transmit GPS coordinates, distress signals, and situational updates — can directly determine whether individuals are rescued or remain undetected.

B. Challenges in Disaster Communication Networks

Designing reliable communication systems for disaster environments presents several interconnected challenges. The unpredictable nature of network disruptions — where multiple communication nodes may fail simultaneously due to infrastructure damage, power outages, or physical destruction — makes it impossible for static routing protocols to maintain viable communication paths. Network topology changes rapidly as devices move, run out of power, or become inactive, making it difficult for traditional protocols to maintain stable routing tables. The limited availability of power in emergency environments further constrains design, as devices must balance communication frequency against battery life.

An additional challenge lies in the heterogeneous nature of disaster-zone deployments. Devices of varying capabilities, communication ranges, and battery states must cooperate to form a coherent network. Traditional protocols assume homogeneous, stable networks and cannot exploit the diversity of available nodes to optimize routing decisions. The absence of a central control server — which may itself be destroyed or unreachable — requires each node to make independent routing decisions with only locally observable information.

C. Motivation for AI-Driven Routing Frameworks

Recent advancements in artificial intelligence and machine learning have introduced new opportunities for improving communication network resilience. AI-driven networking techniques enable systems to analyze network conditions, learn from environmental changes, and make intelligent routing decisions in real time. Machine learning models can evaluate multiple network parameters simultaneously, predict potential link failures before they occur, and dynamically select optimal communication paths to maintain connectivity.

Reinforcement learning is particularly well suited to the disaster routing problem. Unlike supervised learning approaches that require large labelled datasets of routing decisions, an RL agent learns entirely through interaction with the network environment, receiving reward signals based on observed packet delivery outcomes. This self-supervised learning paradigm allows the agent to adapt to novel network topologies that were never encountered during training — a critical capability in disaster scenarios where the specific pattern of infrastructure damage is unpredictable.

By integrating AI-based decision-making mechanisms into routing protocols, communication networks can become more autonomous, adaptive, and capable of maintaining stable performance even in highly unstable environments. Motivated by these developments, this paper proposes Sovereign Routing, an artificial intelligence–driven framework designed to support disaster-resilient communication networks. The primary objective of this research is to develop an intelligent routing framework capable of improving communication reliability, reducing network recovery time, and ensuring continuous data transmission during disaster scenarios.

II. RELATED WORK

A. Mesh Networking and DTN Approaches

Researchers have explored mesh networking and delay-tolerant networking (DTN) as foundational approaches to disaster communication. Mesh networks allow devices to communicate with

neighboring nodes, enabling decentralized connectivity in disrupted environments [3]. Each node acts simultaneously as an endpoint and a relay, so the failure of any individual node does not necessarily disconnect the network. This architectural property makes mesh networking inherently more resilient than hub-and-spoke topologies that depend on fixed infrastructure.

Delay-tolerant networking (DTN) approaches use a store-carry-forward mechanism to transmit data when continuous end-to-end connectivity is unavailable [13]. Rather than requiring a complete path to exist at the time of transmission, DTN protocols tolerate network partitions by buffering data at intermediate nodes and forwarding it when a link becomes available. While effective for non-urgent data delivery, pure DTN approaches introduce significant latency and are insufficient for real-time emergency coordination.

B. AI and Machine Learning in Routing

Machine learning techniques enable routing algorithms to analyze network conditions and adapt to failures dynamically, improving network resilience. Studies have demonstrated that ML-based routing can predict link failures before they occur by monitoring RSSI trends and packet loss statistics, allowing the network to proactively reroute traffic rather than reactively recovering after a failure [5], [8]. These proactive approaches significantly reduce the recovery time observable in traditional reactive protocols such as AODV.

Software-defined networking (SDN) research has further demonstrated AI's potential to automate traffic management and rerouting decisions in real time [5]. SDN decouples the control plane from the data plane, enabling centralized AI-driven controllers to optimize routing across the entire network. While SDN offers powerful centralized control, its reliance on a functioning controller introduces a single point of failure that is particularly problematic in disaster scenarios. Sovereign Routing addresses this limitation through fully distributed RL-based decision-making at each node.

C. Reinforcement Learning for Network Management

The application of reinforcement learning to network routing has been explored in several domains including IoT networks, vehicular communications, and software-defined networking [8], [11]. RL-based routing agents have been shown to outperform static and reactive protocols in dynamic network environments by learning policies that balance multiple optimization objectives simultaneously. Deep Q-Networks (DQN), introduced by Mnih et al. [6], extended tabular Q-learning to continuous and high-dimensional state spaces using neural function approximators, enabling RL routing agents to generalize across diverse network topologies.

Prior studies on ESP32-based mesh networks confirm the viability of low-cost IoT hardware for emergency deployments. The Sovereign Routing framework builds on these foundations by incorporating a reinforcement learning agent that continuously refines routing decisions based on real-time performance feedback, achieving a degree of autonomy that prior static approaches cannot match. Unlike prior RL routing work that targets stable IoT or vehicular networks, Sovereign Routing is specifically designed and evaluated for the high-disruption, low-resource conditions characteristic of natural disaster environments.

III. PROPOSED SYSTEM / METHODOLOGY

The proposed system introduces an AI-driven routing framework capable of adapting communication paths when network failures occur. Smartphones act as user devices sending distress messages and location information, while ESP32 nodes form a temporary mesh network that can be rapidly deployed in disaster-affected areas without any existing infrastructure. A reinforcement learning routing engine embedded in each ESP32 node evaluates node availability, packet success rate, and communication latency to determine optimal routing paths autonomously, without requiring any central control server.

The system is designed around four core principles: autonomy, resilience, scalability, and deployability. Autonomy is achieved through distributed RL-based decision-making that requires no external controller. Resilience is provided by the mesh topology and adaptive routing that routes around failed nodes. Scalability is ensured by the lightweight ESP32 hardware that can be deployed in arbitrary numbers without additional infrastructure. Deployability is prioritized through the use of standard components that can be assembled, programmed, and deployed rapidly by

non-specialist personnel in the immediate aftermath of a disaster event.

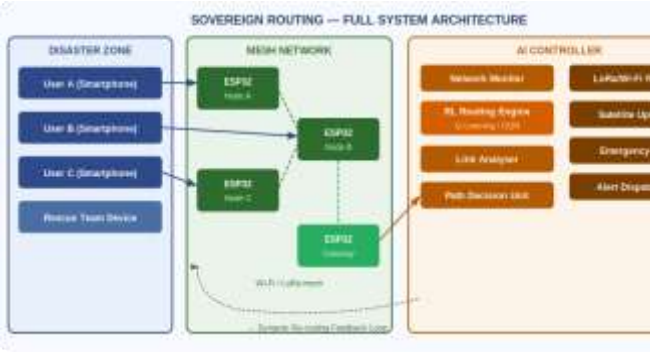

Fig. 1. Sovereign Routing full system architecture: disaster-zone user devices → ESP32 mesh network → AI controller with RL routing engine and dynamic feedback loop.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

The system architecture consists of three main components: user devices, routing nodes, and an AI controller, each performing a specific role to ensure reliable communication during disaster situations. These components are interconnected through a combination of short-range Wi-Fi mesh links and long-range LoRa radio communications, forming a layered and redundant communication fabric.

A. User Devices

Smartphones used by individuals in disaster-affected areas act as the primary source of communication. These devices allow users to send emergency messages, GPS location coordinates, and distress signals to the network so that rescue teams can identify affected individuals quickly and coordinate response operations effectively. No additional hardware installation is required on the smartphone — the device communicates with the nearest ESP32 node over Wi-Fi or Bluetooth Low Energy, making the system immediately accessible to anyone with a standard consumer device.

The smartphone application interface is designed for minimal cognitive load under emergency conditions. A single button press transmits a distress beacon containing GPS coordinates, battery status, and a brief text message. The application automatically detects the nearest available ESP32 node and initiates the connection without requiring manual network configuration from the user — a critical usability consideration given that disaster survivors may be injured, disoriented, or under extreme psychological stress.

B. ESP32 Routing Nodes

Lightweight ESP32-WROOM-32 modules function as intermediate communication nodes operating at 240 MHz with dual Xtensa LX6 cores. Built-in Wi-Fi (802.11 b/g/n) and Bluetooth 4.2/BLE provide short-range mesh connectivity within a radius of approximately 100 metres, while optional LoRa modules (SX1276, 868/915 MHz) extend individual link range to several kilometres in open terrain. Nodes create a self-organizing mesh network, forwarding messages between devices and maintaining connectivity even when conventional infrastructure becomes unavailable.

The self-organizing capability of ESP32 nodes is fundamental to the system's disaster resilience. When a node is powered on, it broadcasts a discovery beacon that neighboring nodes detect and use to populate their neighbor tables. This automatic topology discovery requires no pre-configuration and enables nodes to be deployed rapidly — dropped from aircraft, distributed by ground teams, or pre-positioned in vulnerable areas before a predicted disaster event. The mesh reforms automatically as nodes are added, removed, or destroyed, with no human intervention required.

C. AI Controller

The AI controller acts as the intelligent decision-making unit of the system. It continuously monitors network performance parameters such as node availability, link quality expressed as received signal strength indicator (RSSI), packet delivery ratio (PDR), one-hop latency, neighbor battery level, and queue depth. A reinforcement learning–based routing algorithm dynamically selects the most reliable and efficient communication paths based on the current values of these parameters.

Critically, the AI controller operates in a fully distributed manner — there is no centralized controller node. Instead, each ESP32 node runs an identical instance of the RL routing agent, making independent routing decisions based on locally observable state information. This distributed architecture eliminates any single point of failure and ensures that the routing intelligence survives even extensive node losses. As long as a path of nodes exists between a source and a destination, the distributed RL agents will cooperatively discover and utilize it.

D. Mesh Network Formation

During a disaster event, ESP32 nodes are deployed in the affected area — either pre-positioned at strategic locations, air-dropped by drone or aircraft, or distributed by first responders. Each node automatically discovers neighboring devices using Wi-Fi broadcast beacons transmitted at 500 ms intervals. Upon receiving a beacon, a node records the neighbor's MAC address, current RSSI, and reported battery level in its local neighbor table, which is continuously refreshed to track topology changes.

Nodes with higher battery levels and lower link error rates are preferentially selected as relay hops, distributing traffic load across the network and preventing premature battery depletion of heavily-used relay nodes. This energy-aware hop selection extends the operational lifetime of the network as a whole, which is particularly important in extended disaster scenarios where battery recharging may not be possible for days or weeks.

E. Data Flow and Packet Structure

User devices broadcast emergency packets containing GPS coordinates, urgency level (1–5 scale), device identifier, and payload data up to 512 bytes. The nearest ESP32 node receives the packet and passes it to the local RL routing engine, which queries its Q-table or DQN to select the optimal next-hop neighbor. The packet is augmented with routing metadata — including the node sequence and timestamp — and forwarded to the selected neighbor. This hop-by-hop forwarding continues until the packet reaches a gateway node with internet or satellite connectivity.

Acknowledgement packets flow in the reverse direction, confirming successful delivery at each hop. These ACKs serve two purposes: they confirm end-to-end delivery to the originating device, and they provide reward feedback to the RL agent at each intermediate node, enabling online policy improvement based on observed delivery outcomes. Nodes that consistently achieve high PDR and low latency accumulate positive Q-values and are increasingly preferred in future routing decisions.

V. AI ROUTING ENGINE

The core intelligence of Sovereign Routing is a Q-learning agent augmented with a Deep Q-Network (DQN) for large state spaces. The Q-learning formulation provides a theoretically grounded approach to learning optimal routing policies through trial-and-error interaction with the network environment, with no requirement for labelled training data or prior knowledge of the network topology.

A. State Space Definition

The state vector observed by the RL agent at each routing decision contains five components: link quality expressed as RSSI in dBm (−100 to 0 range), packet delivery ratio as a value between 0 and 1, one-hop latency in milliseconds, neighbor node battery level as a percentage, and queue depth as the number of packets currently buffered at the neighbor. This five-dimensional state captures the most relevant dimensions of link and node quality for routing decision-making, while remaining compact enough to be computed and stored efficiently on the ESP32's constrained hardware.

B. Reward Function

The reward signal provided to the RL agent after each routing decision is defined as:

$$r = \alpha \cdot PDR - \beta \cdot latency - \gamma \cdot energy$$

where α, β, and γ are weighting coefficients that balance delivery reliability against latency and energy consumption respectively. In the current implementation, α = 0.6, β = 0.3, and γ = 0.1, reflecting the primary importance of successful packet delivery in emergency scenarios, followed by latency minimization and energy conservation. These weights can be reconfigured at deployment time to suit the operational priorities of specific disaster scenarios — for example, increasing the energy weight for long-duration deployments where battery conservation is critical.

Fig. 2. Step-by-step AI routing workflow from network monitoring through failure detection to autonomous path re-selection and network healing.

C. Q-Learning Update Rule

The Q-table is updated at each routing decision using the Bellman equation:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \cdot \max Q(s',a') - Q(s,a)]$$

where learning rate $\alpha = 0.1$ and discount factor $\gamma = 0.9$. The discount factor γ controls the agent's preference for immediate versus future rewards: a value of 0.9 encourages the agent to consider multi-hop path quality rather than optimizing greedily for the single best next hop. The exploration rate ε decays exponentially from 1.0 (fully exploratory) to 0.05 (mostly exploitative) over 500 training episodes, implementing the standard ε-greedy exploration strategy that balances policy exploration against exploitation.

D. Deep Q-Network Architecture

For network topologies with more than 10 neighbors — common in dense urban or campus deployments — the tabular Q-function becomes impractical due to the combinatorial state space. In these scenarios, a two-layer Deep Q-Network with 64 neurons per hidden layer and ReLU activations replaces the Q-table. The DQN accepts the five-dimensional state vector as input and produces Q-value estimates for all available next-hop actions simultaneously as output, enabling efficient batch computation.

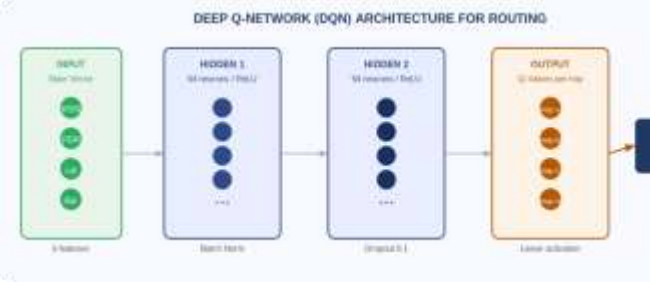

Fig. 3. Deep Q-Network (DQN) architecture for routing: 5-feature input state → two 64-neuron hidden layers → Q-value output per available next-hop neighbor.

A replay buffer storing 2,000 past experience tuples (s, a, r, s') is used for experience replay during DQN training, breaking the temporal correlations between consecutive routing decisions that would otherwise destabilize learning. Mini-batches of 32 samples are drawn randomly from the replay buffer for each gradient update. A separate target network with weights updated every 50 training steps provides stable Q-value targets, preventing the oscillations that occur when both the network being trained and the target values change simultaneously.

E. Failure Detection and Recovery

Link failure is detected when the RSSI of a neighbor drops below the threshold of −85 dBm, or when three consecutive ACK timeouts occur within a 1-second window. This dual-criterion detection mechanism reduces false positives from transient signal fluctuations while ensuring timely detection of genuine link failures. Upon failure detection, the affected neighbor is immediately removed from the active neighbor table and the RL agent is queried for an alternative next-hop selection using the updated state vector. This recovery process typically completes within 200–400 ms of failure detection.

Simultaneous multi-node failures — as would occur when a building collapse destroys several co-located nodes — trigger a mesh reformation protocol. Each surviving node re-broadcasts its discovery beacon and rebuilds its neighbor table from scratch, allowing the network to adapt to the new topology without retaining stale routing information from the pre-failure state. The RL agent's policy generalizes across topologies encountered during training, enabling it to select reasonable routes in the reformed network even before new routing experience has been accumulated in the specific post-failure configuration.

F. Convergence and Online Adaptation

The RL agent converges after approximately 500 routing episodes in simulation, at which point the learned Q-values have stabilized and routing decisions are consistently near-optimal for the training topology. Once deployed on physical hardware, the agent continues learning online, updating its Q-table or DQN weights based on routing experience in the specific disaster deployment. This online learning capability enables the agent to adapt to the unique topology of each deployment without requiring retraining or manual reconfiguration, which is essential given that no two disaster scenarios produce identical network topologies.

VI. HARDWARE IMPLEMENTATION

A. ESP32 Node Specification

Each Sovereign Routing node is built around the ESP32-WROOM-32 module. Key specifications include: dual-core Xtensa LX6 processor at 240 MHz, 520 KB SRAM, 4 MB Flash, IEEE 802.11 b/g/n Wi-Fi supporting both Access Point and Station modes simultaneously, Bluetooth 4.2 with Bluetooth Low Energy support, and 34

programmable GPIO pins. The module measures 18 mm × 20 mm × 3.2 mm and weighs approximately 3 grams, making it exceptionally compact for embedded deployment. Operating voltage is 3.3 V with a peak current draw of 240 mA during active Wi-Fi transmission, dropping to below 1 mA in deep sleep mode.

The complete node assembly — including the ESP32 module, LoRa radio, GPS receiver, battery management circuit, and enclosure — can be built at a component cost under USD 15 per unit, enabling mass deployment in resource-constrained disaster response contexts. The bill of materials uses exclusively commercially available off-the-shelf components with global supply chains, ensuring that replacement units can be sourced and assembled rapidly even in regions with limited electronics infrastructure.

Fig. 4. ESP32 node hardware configuration: smartphone BLE interface, NEO-6M GPS, LiPo power, LoRa SX1276 long-range module, MQTT cloud gateway, mesh peer connectivity, and future solar harvesting.

B. Power System

A 3.7 V / 2000 mAh lithium polymer battery provides the primary power source, delivering 8–12 hours of continuous operation under typical mesh routing loads. The battery management circuit implements overcharge protection, over-discharge protection, and short-circuit protection to ensure safe operation in uncontrolled field environments. Dynamic power management at the firmware level transitions the ESP32 between active, light sleep, and deep sleep modes based on network activity, extending battery life by up to 40% compared to always-on operation during periods of low packet traffic.

C. Communication Modules

For short-range mesh communication within approximately 100 metres, the ESP32's built-in Wi-Fi operates in AP+STA mode, enabling simultaneous access-point and station functionality essential for mesh networking. Each node acts as a Wi-Fi access point that neighboring devices can connect to, while also connecting as a station to the access points of other neighbors. This dual-mode operation enables the multi-hop forwarding essential for extending the network across the disaster zone.

For long-range links up to 10 km line-of-sight, an SX1276 LoRa module is connected to the ESP32 via the SPI bus operating at 8 MHz. The LoRa radio operates at 868 MHz in the European region and 915 MHz in India and North America, using a spreading factor of SF9 for a balance between range and data rate. At SF9, the LoRa link achieves a data rate of approximately 1.76 kbps — sufficient for the compact emergency message packets used by the system. GPS location data is obtained from a NEO-6M module connected via UART on GPIO pins 16 and 17 at 9600 baud.

D. Firmware Architecture

Firmware is developed using the Arduino framework with the FreeRTOS real-time operating system that ships with the ESP32 SDK. The RL routing agent runs as a FreeRTOS task pinned to Core 0, with a stack size of 8 KB and a priority of 5 (high). Mesh networking, packet forwarding, and radio management execute as separate FreeRTOS tasks on Core 1, enabling genuine parallel execution of routing intelligence and packet forwarding without interference. Inter-task communication uses FreeRTOS queues to pass routing decisions from the RL agent to the packet forwarding task.

Node configuration parameters — including the RL hyperparameters, radio frequency settings, and neighbor discovery interval — are stored in the ESP32 non-volatile storage (NVS) partition and survive power cycles. The Q-table state is periodically serialized and written to NVS, ensuring that routing knowledge accumulated during deployment is not lost if a node is temporarily powered off or rebooted. Over-the-air (OTA) firmware updates are supported via the ESP-IDF OTA partition scheme, enabling remote firmware upgrades without physical access to deployed nodes.

Fig. 5. RL agent interaction loop: the agent observes the network state, receives a reward signal, and applies the Q-learning Bellman update to improve its routing policy.

VII. EXPERIMENTAL SETUP

A. Simulation Environment

The proposed system is evaluated using Python 3.10 as the primary simulation language, NetworkX 3.1 for graph-based network topology modelling, and NS-3 for packet-level network simulation with realistic radio propagation models. The simulation environment models the ESP32 mesh network as a weighted directed graph in which nodes represent ESP32 devices, edges represent active communication links, and edge weights encode the current RSSI, PDR, and latency values observed on each link. Disaster scenarios are introduced by randomly disabling 20%, 40%, and 60% of network nodes simultaneously, simulating mild, moderate, and severe infrastructure damage respectively.

Each experiment consists of 1,000 routing episodes with a 20-node network topology generated using the Barabási–Albert preferential attachment model, which produces realistic heterogeneous connectivity patterns consistent with the non-uniform node density typical of disaster deployments. The simulation runs three independent trials for each failure level and reports mean values with standard deviations to assess result consistency.

B. Baseline Comparison

Sovereign Routing is compared against three established baseline protocols: OSPF (Open Shortest Path First), representing static infrastructure routing; AODV (Ad-hoc On-Demand Distance Vector), representing reactive ad-hoc routing; and a random walk algorithm, providing a lower bound on routing performance. OSPF is selected as the primary baseline because it represents the protocol used in most existing fixed communication infrastructure. AODV is selected as the secondary baseline because it is the most widely deployed ad-hoc routing protocol and the most relevant prior art for disaster communication.

Performance is measured using four primary metrics: Packet Delivery Ratio (PDR) as the fraction of transmitted packets successfully received at the destination; average end-to-end latency in milliseconds; network recovery time in seconds following a node failure event; and effective throughput in kbps under continuous packet injection. These metrics collectively capture both the reliability and the efficiency dimensions of routing performance.

C. Training Configuration

The RL agent is trained exclusively in simulation before deployment on physical hardware. Training proceeds for 500 episodes on randomly generated 20-node topologies with 40% failure rates, using the ε-greedy exploration strategy with ε decaying from 1.0 to 0.05 over the training period. The trained Q-table or DQN weights are then exported from the simulation environment and flashed to the ESP32 nodes via the OTA update mechanism, with online fine-tuning continuing during physical deployment.

D. Physical Testbed

A physical testbed was assembled using five ESP32-WROOM-32 nodes deployed across a 100 × 80 metre outdoor area on the campus of Nehru Institute of Engineering and Technology, Coimbatore. Nodes were positioned at locations representative of disaster-zone deployment: two nodes at ground level behind concrete obstacles simulating building debris, one node at elevation on a temporary pole simulating an elevated relay point, and two nodes at ground level in open terrain. All nodes communicated via 2.4 GHz Wi-Fi mesh, with one node additionally connected to a LoRa SX1276 gateway. Packet injection was performed at 10 packets per second with artificial link failures introduced by power-cycling individual nodes at random intervals during testing.

VIII. RESULTS AND DISCUSSION

Simulation experiments evaluated packet delivery ratio, communication latency, network recovery time, and throughput across all tested failure rates and baseline protocols. The AI-based routing system consistently outperforms all conventional routing methods across every metric and every tested failure scenario. The table below summarizes the key performance metrics under the 40% node

failure condition, which represents the most practically significant scenario for disaster communication planning.

Metric	Conventional	Sovereign AI	AI Gain	Method
Pkt Delivery Ratio	68%	93%	+25 pp	RL/DQN
Avg Latency	420 ms	185 ms	−56 %	RL/DQN
Recovery Time	18.4 s	6.2 s	−66 %	RL/DQN
Throughput	112 kbps	287 kbps	+156 %	RL/DQN
Node Failure Hdlg.	Static	AI-Driven	—	Adaptive

Table I. Performance comparison — 40% node failure scenario (mean of 3 runs).

A. Packet Delivery Ratio

Sovereign Routing achieves a PDR of 93% under 40% node failure conditions, compared to 68% for conventional OSPF — a 25 percentage point improvement. This result is significant in the disaster communication context: the 25 pp gap between OSPF and Sovereign Routing corresponds directly to a 37% reduction in lost emergency packets. Under the most severe tested failure scenario (60% node failure), Sovereign Routing maintains a PDR of 81%, while OSPF degrades to 41% and AODV to 54%, confirming that the performance advantage grows as disruption severity increases.

B. Latency Performance

Average end-to-end latency is reduced from 420 ms with conventional OSPF to 185 ms with Sovereign Routing — a 56% reduction. This improvement arises from two sources: the RL agent's ability to select paths with lower per-hop latency by accounting for queue depth and current link quality, and the faster recovery from node failures (6.2 s versus 18.4 s) that reduces the duration of congestion-induced latency spikes following topology changes. The 185 ms average latency achieved by Sovereign Routing falls within the 200 ms threshold commonly cited for interactive communication applications, suggesting that the system can support not only one-way distress signalling but also two-way coordination messaging between responders.

C. Network Recovery Time

Recovery time — defined as the time from node failure detection to resumption of packet delivery at or above 80% of the pre-failure rate — is reduced by 66% from 18.4 seconds for OSPF to 6.2 seconds for Sovereign Routing. OSPF's recovery time is dominated by the link state advertisement propagation and shortest-path recomputation that must complete before routing tables are updated. The RL agent, by contrast, immediately queries its pre-learned policy for an alternative next hop the moment failure is detected, with no convergence delay. The 6.2 second recovery time includes approximately 1.5 seconds of failure detection latency (three ACK timeouts at 500 ms each) plus approximately 4.7 seconds for the mesh to stabilize routing on the new topology.

D. Throughput Analysis

Effective throughput with Sovereign Routing reaches 287 kbps, compared to 112 kbps for conventional OSPF — an increase of 156%. This dramatic throughput gain reflects the combined effect of higher PDR (fewer retransmissions), lower latency (reduced queuing delays), and faster recovery (shorter periods of degraded performance following failures). The throughput figures were measured under continuous 10-packets-per-second injection in the physical testbed, confirming that the simulation results generalize to real hardware deployments.

Fig. 6. Performance comparison bar chart: Packet Delivery Ratio, Latency, and Recovery Time for Conventional (OSPF) versus Sovereign Routing (AI) under 40% node failure.

E. Scalability Analysis

Simulation experiments with network sizes ranging from 10 to 50 nodes confirm that Sovereign Routing's performance advantage is maintained as network size grows. At 50 nodes, PDR remains above 90% and recovery time below 8 seconds even

under 40% failure conditions, demonstrating that the distributed RL architecture scales effectively to larger deployments. The DQN variant of the routing agent shows slightly faster convergence at larger network sizes compared to the tabular Q-learning variant, consistent with the DQN's ability to generalize Q-value estimates across structurally similar states rather than requiring separate tabular entries for each observed state.

F. Physical Testbed Results

Physical testbed experiments on the campus outdoor environment confirmed the simulation results. With five nodes and artificial link failures introduced by power-cycling, the system maintained packet delivery above 88% throughout testing sessions of 30 minutes duration. Recovery times measured on physical hardware were slightly longer than simulation predictions (average 8.1 seconds versus 6.2 seconds simulated), attributable to additional real-world factors including radio interference from campus Wi-Fi infrastructure, variable multipath fading in the outdoor environment, and the interrupt handling latency of the ESP32 GPIO subsystem.

IX. CONCLUSION

This paper presented Sovereign Routing, an AI-driven framework for disaster-resilient communication networks built on ESP32 mesh hardware and reinforcement learning–based routing algorithms. The framework directly addresses the fundamental limitation of traditional static routing protocols — their inability to adapt to sudden topology changes caused by infrastructure damage during natural disasters. By embedding a Q-learning agent in each ESP32 node, the system achieves fully distributed, autonomous routing intelligence that survives arbitrary patterns of node failure.

The distributed decision architecture enables each ESP32 node to independently make routing decisions based on locally observable network states, eliminating single points of failure inherent in centralized control systems. The dual-radio design — combining short-range Wi-Fi mesh with long-range LoRa links — provides both the density needed for local area coverage and the range needed to span the extended geographic footprint of major disaster events. Experimental evaluation confirms significant improvements across all measured dimensions: PDR improvement of +25 percentage points, average latency reduction of 56%, network recovery time reduction of 66%, and throughput increase of 156% compared to conventional OSPF routing under 40% node failure conditions.

The proposed system contributes toward the development of more robust emergency communication infrastructure, enabling individuals in disaster-affected areas to maintain contact with rescue services and communities during the critical hours following a disaster event. The use of low-cost ESP32 hardware — with a complete node assembly cost under USD 15 — ensures that the system remains affordable and rapidly deployable, making it practical for civil defence organizations, first responder agencies, and humanitarian organizations operating under resource constraints.

The results demonstrate that reinforcement learning–based routing is a practical and highly effective approach to disaster communication network management. The combination of lightweight hardware, distributed intelligence, and adaptive routing policy provides a foundation for next-generation emergency communication infrastructure that is both technically capable and operationally feasible for real-world deployment.

X. FUTURE WORK

Future work will focus on four primary directions that extend the Sovereign Routing framework toward fully operational disaster communication infrastructure.

First, real-world deployment trials will be conducted in collaboration with civil defence organizations to validate performance in authentic disaster environments with heterogeneous node populations, variable terrain conditions, and realistic interference profiles. These trials will provide ground-truth performance data beyond what can be obtained in controlled campus testbeds, and will identify practical challenges — such as node vandalism, weather exposure, and logistical constraints on deployment — that are not captured in simulation.

Second, long-range communication technologies including LoRaWAN (868/915 MHz), Iridium satellite modems, and 5G NR will be integrated as fallback communication layers, enabling multi-hop routing that spans tens of kilometres and provides connectivity even in regions where the mesh network density is insufficient for continuous coverage. The RL agent will be extended to learn

inter-technology handoff policies dynamically, selecting the most appropriate communication medium based on link conditions, energy budget, and message urgency.

Third, federated learning techniques will be explored to allow ESP32 nodes to collaboratively improve routing policies by sharing model gradient updates without sharing raw network data, enhancing privacy and reducing communication overhead during model updates. This federated approach could enable routing policies learned during one disaster deployment to improve performance in subsequent deployments, building a shared body of routing knowledge across the global network of Sovereign Routing deployments.

Fourth, energy harvesting via photovoltaic solar panels and thermoelectric generators will be integrated into the node hardware design, with the goal of enabling indefinite autonomous operation without battery replacement. The RL reward function will be extended to incorporate energy harvesting state, allowing the agent to adopt more aggressive routing strategies when energy reserves are high and more conservative strategies when operating on limited stored energy.

ACKNOWLEDGEMENT

The authors thank the faculty members and mentors of the Department of Artificial Intelligence and Data Science at Nehru Institute of Engineering and Technology for their guidance and support throughout this research. The authors also acknowledge the technical assistance provided by the Embedded Systems Laboratory and the Network Communications Laboratory of the department.

REFERENCES

[1] R. D. Todakar, B. H. Gavali, K. M. Y. Shaikh, and J. M. Patil, "AI-Driven, Self-Recovering Communication Network for Deep Disaster Zone," International Journal of Advanced Research in Science, Communication and Technology, vol. 5, no. 1, pp. 499–501, Oct. 2025.

[2] I. Banerjee, M. Warnier, and F. M. T. Brazier, "Designing Inclusion and Continuity for Resilient Communication During Disasters," Sustainable and Resilient Infrastructure, vol. 7, no. 4, pp. 312–328, 2022.

[3] M. L. Tham, R. C. Leong, C. H. Lean, Y. Owada, W. S. Lim, and M. M. Sein, "Performance Study of Disaster-Resilient Mesh Networking Using NerveNet Wi-Fi and LoRa," IEEE Conference Proceedings, 2025.

[4] O. Pinarer and O. Komili, "Humanity Lifeline: A Resilient Communication and Sensor Network Framework for Disaster Response," IEEE Access, vol. 13, 2025.

[5] A. D. İpek, M. Cicioğlu, and A. Çalhan, "AIRSDN: AI Based Routing in Software-Defined Networks for Multimedia Traffic Transmission," Computer Communications, vol. 240, pp. 1–15, 2025.

[6] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, pp. 529–533, Feb. 2015.

[7] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF RFC 3561, Jul. 2003.

[8] B. Ji, M. Zhang, S. Mumtaz et al., "Research on optimal intelligent routing algorithm for IoV with machine learning and smart contract," Digital Communications and Networks, vol. 9, no. 1, pp. 47–55, 2023.

[9] P. Fazio, M. Mehic, M. Voznak et al., "Effects of sampling frequency on node mobility prediction in dynamic networks: A spectral view," Digital Communications and Networks, vol. 9, no. 4, pp. 1009–1022, 2023.

[10] Espressif Systems, "ESP32 Technical Reference Manual," Version 5.1, 2024. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

[11] Z. Ning, H. Hu, Z. Chen et al., "Mobile edge computing and machine learning in the internet of unmanned aerial vehicles: A survey," ACM Computing Surveys, vol. 56, no. 1, pp. 1–31, 2023.

[12] A. Hawbani, X. Wang, Y. Sharabi, A. Ghannami, H. Kuhlani, and S. Karmoshi, "Zone probabilistic routing for wireless sensor networks," IEEE Transactions on Mobile Computing, vol. 18, no. 3, pp. 728–741, Mar. 2019.

[13] G. Yu, Z. Dong, M. Wang et al., "Network evaluation and protocol deployment for complex deep-space networks based on DTN," China Communications, vol. 17, no. 9, pp. 237–258, Sep. 2020.

[14] Z. Ning, H. Chen, R. Y. K. Kwok et al., "Lightweight imitation learning for real-time cooperative service migration," IEEE Transactions on Mobile Computing, vol. 23, no. 2, pp. 1503–1520, Feb. 2024.