International Journal of Computing and Digital Systems
2025, VOL. 17, NO. 1, 1–13

http://dx.doi.org/10.12785/ijcds/1571038958

Shortest Path Optimization for Determining Nearest Full Node from a Light Node in Blockchain IoT Networks

Vivek Anand M[1] **and Srinivasan Sriramulu**[2]

[1]*Research Scholar,Galgotias University, Greater Noida, India*
[2]*Professor,School of Computing Science and Engineering,Galgotias University, Greater Noida, India*

Received 14 June 2024, Revised 23 October 2024, Accepted 25 October 2024

Abstract: In a blockchain IoT network, there exists a diversity of devices, including full nodes and light nodes, each with varying capacities and roles. Full nodes have the capability to store the entire ledger, whereas light nodes, constrained by limited memory capacity, cannot store the full blockchain ledger data. However, light nodes can efficiently retrieve data from full nodes and actively participate in network transaction approvals, especially in critical applications such as the military and healthcare sectors, which require the trusted approval of transactions from the maximum number of nodes. To enable light nodes to approve transactions by verifying blockchain ledgers, we need to determine the nearest full node with the shortest distance to retrieve the data from a full node to a light node. Efficient retrieval of data from the nearest full node to a light node is required to avoid the delay in transaction approval. To find the shortest distance between the nodes, several algorithms exist, such as Dijkstra's Algorithm, Floyd-Warshall Algorithm, Genetic Algorithms (GA), Ant Colony Optimization (ACO), and Routing Protocol for Low-Power and Lossy Networks (RPL). This work compared all algorithms with the parameters of scalability, real-time support, energy efficiency, and complexity. The comparison shows that RPL stands out with distinct advantages. RPL surpasses all o+ther algorithms in enabling efficient data retrieval and facilitating network transaction approval, thereby ensuring the seamless operation of blockchain IoT systems. Moreover, RPL does not account for trust between nodes, which is critical in blockchain-based IoT networks. Trust values can influence the decision-making process for routing and help the protocol prioritize routes through trusted nodes. Integrating trust metrics into the RPL protocol by incorporating blockchain consensus mechanisms, which are Proof of Trust (PoT), to evaluate the reliability of nodes. This work shows RPL enhanced with trust metrics, which would be the best choice to find the shortest path between full node and light node in blockchain-based IoT networks effectively when compared to other algorithms.

Keywords: IoT networks, Directed Acyclic Graph (DAG) topology, DODAG Information Object (DIO) Messages, Destination Advertisement Object (DAO) Messages

1. Introduction

IoT relies on centralized architecture, such as client-server or cloud architecture. Trust among the centralized architecture is questionable, and the server may be prone to an attack. IoT faces various challenges due to following centralized architectures. The challenges of IoT include the increase of IoT devices, improper topologies, and security attacks with botnets. The data on the IoT devices can be modified at any time due to centralized architecture. Third party servers does not give assurance on the data storing on it. To secure IoT networks, various works on network intrusion detection and AI-based techniques for securing IoT networks are discussed in [1] [2] [3] [4] but IoT requires some architectural changes to secure the entire network completely. The data from the IoT devices must be validated properly, and it should not be tampered by the unauthorized nodes. The tampering of data in IoT devices at crucial applications such as military and medical applications will lead to a danger. On accordance with the IoT security issues ,Most of the researchers suggesting the architecture that is suitable for IoT to secure their data on their applications is blockchain. The study [5] explores the integration of blockchain and IoT solutions, demonstrating their feasibility and effectiveness in real-world scenarios. Furthermore, the integration of blockchain technology with IoT can facilitate secure and decentralized device authentication and authorization mechanisms. By leveraging blockchain's cryptographic techniques, IoT devices can securely authenticate each other and establish trust relationships without relying on centralized authentication servers. This concept is detailed in the work of [6] , which provides a comprehensive framework for blockchain-based IoT security solutions. Blockchain is a peer-to-peer, distributed, shared network where the transaction data is

E-mail address: vivek395@gmail.com , s.srinivasan@galgotiasuniversity.edu.in

converted to hash values and merged to create a block hash. Every block hash will be produced with the previous block hash. Once the block of transactions is added with another block, then all the transactions will be completed for that block. The same way every block is added with the previous block to form a blockchain. In blockchain, once the data is approved and added to a blockchain ledger, it cannot be tampered with. Integrating blockchain with IoT enhances security through its tamper-proof, decentralized ledger, eliminating the need for a central authority and ensuring data integrity. Research [7] shows that this combination improves transparency and trust as every transaction is recorded and verifiable.

Despite these benefits, IoT faces significant challenges in terms of storage and processing capabilities when integrating with blockchain technology. In blockchain, once the blocks of transactions are approved, they are stored as a ledger and cannot be tampered with. All the nodes keep records of all the blocks of transactions as chains. If any new node wants to join the network, to maintain the network's consensus, it must store the entire blockchain. Storing blockchain ledgers in IoT peers is one of the major obstacles due to the varying storage capacities of devices. The blockchain will grow periodically as the network size increases. The original version of the blockchain, known as Bitcoin.exe, is a 6-MB Windows 32 program that was released in 2008. When it first started, the blockchain appeared to be a small initiative, and keeping data on the blockchain was not difficult. However, while still relatively small, the average blockchain size in 2014 was about 20 GB. Typically, a blockchain block in the Bitcoin network requires 2 MB of storage. Since 2008, the number of Bitcoin blocks has been growing daily. Currently, the blockchain is about 586 GB in size.

This study [8] contains statistics on the growth of blockchain size over time. The solutions like data pruning [9], off-chain storage [10], and virtualization can mitigate these blockchain storage issues. Data pruning involves removing unnecessary data from the blockchain, ensuring that only essential information is retained. Off-chain storage allows data to be stored outside the blockchain, reducing the burden on IoT devices, while only critical information is recorded on-chain. Virtualization abstracts blockchain data, allowing devices to interact with it without needing to store the entire blockchain. The IoT network consists of different types of nodes, such as full nodes and light nodes. A full node can store the entire blockchain, while a light node cannot store the entire blockchain due to limited memory. However, the light node actively participates in data approval for crucial applications to build trust in the ledger data. To retrieve data from a full node to a light node, it is necessary to find the nearest full node to facilitate faster approval. Various algorithms can be used to find the shortest path between nodes, including Dijkstra's Algorithm, the Floyd-Warshall Algorithm, Genetic Algorithms, Ant Colony Optimization, and the Routing Protocol for Low-Power and Lossy Networks. A comparison among all these algorithms is required to determine the best solution for shortest path determination.

2. Challenges and Issues in Determining Optimal Shortest Path

Determining the optimal shortest path between interconnected IoT devices is a multifaceted challenge, influenced by various factors inherent to the nature of IoT networks. The constrained resources of IoT devices, such as limited memory, processing power, and battery life, complicate the storage of routing tables and the execution of complex algorithms required for efficient path finding. Research by [11]emphasizes that these limitations hinder the deployment of traditional routing protocols in IoT environments, necessitating the development of lightweight and efficient alternatives. The dynamic topology of IoT networks further exacerbates routing challenges. IoT devices are often mobile and subject to varying signal strengths, leading to frequent changes in network topology. The paper [12] highlights that this mobility and intermittent connectivity make it difficult to maintain consistent and reliable routing paths, especially as the network scales. This dynamic nature requires adaptive algorithms that can quickly respond to changes and ensure optimal routing paths are maintained. Heterogeneity among IoT devices adds another layer of complexity. IoT networks consist of devices with varying capabilities and communication protocols, making standardization and interoperability a significant challenge. In [13], note that the diversity in device capabilities necessitates routing protocols that can accommodate different performance levels and seamlessly integrate various communication standards.

Security concerns are paramount in IoT networks due to their susceptibility to attacks. The need for secure communication channels and data integrity is critical, as IoT devices often handle sensitive information. In [14], discuss the importance of developing secure routing protocols that can protect against threats while maintaining the lightweight nature required by resource-constrained IoT devices. Scalability is another critical issue. As IoT networks grow, the routing protocols must efficiently handle an increasing number of devices without significant performance degradation. This scalability challenge is compounded by the need for real-time communication, where delays can lead to significant issues in applications such as healthcare and industrial automation. Energy efficiency is a crucial consideration for extending the battery life of IoT devices. In [15], point out that optimizing energy consumption through efficient routing protocols is essential for the longevity and reliability of IoT networks. This requires protocols that minimize energy usage without compromising on performance or security. Addressing these challenges requires innovative routing protocols and optimization techniques specifically tailored for IoT environments. Hierarchical routing, for example, can simplify management and improve efficiency by organizing the network into clusters. Adaptive algorithms that dynamically adjust to changes in network topology and conditions are vital for maintaining optimal performance. Computational offloading to edge or fog computing can also alleviate the burden on individual IoT devices, allowing more complex processing to be handled by more capable nodes in the network [16]. Furthermore, secure routing protocols leveraging blockchain technology, as discussed earlier, can enhance security and trust within the network. By recording transactions in a decentralized and tamper-proof ledger, blockchain can ensure data integrity and provide robust authentication mechanisms. In conclusion, determining the optimal shortest path in IoT networks is a complex task influenced by resource constraints, dynamic topologies, heterogeneity, security concerns, scalability, and energy efficiency requirements. Research in this field highlights the need for specialized, lightweight, and adaptive routing protocols that can effectively address these challenges and enable efficient, secure, and reliable communication in IoT environments.

3. Existing Solution for Determining the Shortest Path-Literature Review

The optimal shortest path between interconnected IoT devices involves navigating several complex challenges due to the constrained resources, dynamic network topologies, high latency, low bandwidth, and significant security concerns. According to [13], IoT devices often have restricted memory, storage, and power, complicating the storage of routing tables and execution of complex algorithms." These limitations necessitate the development of lightweight and efficient routing protocols tailored to the resource constraints of IoT devices. The frequent changes in network topology, driven by device mobility and varying signal strengths, add further complexity. As IoT networks scale, maintaining optimal routing paths becomes increasingly difficult. In [17], point out that "communication delays and packet losses from lossy links further hinder efficient routing," making it challenging to maintain consistent performance in real-world IoT applications. These network dynamics require adaptive algorithms that can respond to changes quickly and efficiently. Security vulnerabilities are another significant challenge in IoT networks. The need for robust, lightweight, and adaptive routing protocols is paramount to protecting against potential threats. Solutions such as hierarchical routing, adaptive algorithms, and secure routing protocols are essential for addressing these issues.

Several algorithms and protocols have been proposed to address the challenges of routing in IoT networks:

Dijkstra's Algorithm: This well-known algorithm is efficient for finding the shortest path in a network and efficient for finding the shortest path from a single source to all other nodes in a weighted graph with non-negative weights. Well-suited for various applications such as network routing, GPS navigation systems, and telecommunications, but can be resource-intensive for IoT devices with limited capabilities. Not suitable for large graphs, especially when all pairs of shortest paths need to be computed. Requires large amounts of memory for dense graphs [18]. Optimizations can be implemented to limit the search space, making it more suitable for resource-constrained environments [19].

Floyd-Warshall Algorithm: Known for its all-pairs shortest path calculation, this algorithm is comprehensive and solves all-pairs shortest path problems and handles negative weights (but not negative cycles) but is computationally heavy. High space complexity, making it unsuitable for large graphs [18]. It is less suitable for dynamic or large-scale IoT networks due to its extensive computational requirements [20].

Genetic algorithms: These offer flexibility and adaptability by evolving solutions over generations, making them useful for dynamic topologies. Adaptable to complex problems, including NP-hard problems like path optimization. Convergence to global optimal solutions is not guaranteed; it can be computationally expensive. However, they may still require significant computational resources, which can be a limitation for IoT devices[21]

Ant Colony Optimization: ACO excels in dynamic environments by continuously updating paths based on pheromone trails, adapting to changing network conditions in real-time with controlled resource usage. Works well with distributed systems; efficient for finding good solutions in complex graphs. It may converge slowly and can get trapped in local optima. This makes ACO particularly well-suited for IoT networks, which often face varying network conditions [22].

Routing Protocol for Low Power and Lossy Networks: Specifically designed for IoT environments, RPL optimizes routes based on energy efficiency and link reliability. It dynamically updates routing tables, adapts to network topology changes, and efficiently manages limited resources, making it highly suitable for low-power and lossy IoT networks [23].

The paper [24] analyzes the performance of hybrid path planning algorithms that combine Ant Colony Optimization (ACO) and Genetic Algorithms (GA). It highlights how combining ACO's pheromone-based heuristic approach with GA's crossover and mutation techniques improves the global search for optimal solutions in complex environments like the Traveling Salesman Problem (TSP). The study concludes that the hybrid approach achieves faster convergence and better accuracy in finding optimal paths compared to using ACO or GA alone.

While Dijkstra's Algorithm and Floyd-Warshall Algorithm excel in finding the shortest paths, they may not be optimized for the unique constraints and dynamics of IoT networks. Genetic Algorithms (GA) offer heuristic solutions but may lack adaptability to real-time changes in network topology, while Ant Colony Optimization (ACO) may face scalability and resource constraints in IoT environments. Conversely, RPL is meticulously tailored for low-power and lossy networks inherent to IoT settings. Its capability to form Directed Acyclic Graphs (DAGs) and dynamically adjust routes based on metrics like hop count and energy efficiency positions it as an ideal choice for determining the nearest distance between light nodes and full nodes in a blockchain IoT network. By capitalizing on its adaptability and efficiency. To address the inherent challenges of IoT networks, leveraging these algorithms and protocols is crucial. Dijkstra's Algorithm and the Floyd-Warshall Algorithm provide fundamental approaches to pathfinding but may require optimization for practical IoT applications. Genetic Algorithms and Ant Colony optimization offer more dynamic and adaptable solutions, though they must be carefully managed to avoid excessive computational demands. RPL, with its focus on energy efficiency and adaptability to lossy environments, stands out as a particularly effective protocol for IoT while comparing with other protocol . In conclusion, the efficient and secure routing of data in IoT networks involves a careful balance of algorithm complexity, resource constraints, and adaptive capabilities. By leveraging hierarchical routing, adaptive algorithms, secure protocols, and computational offloading to edge or fog computing, IoT networks can achieve reliable communication and performance, addressing the diverse challenges posed by resource limitations, dynamic topologies, and security requirements with RPL. This work shows the comparison of shortest path algorithms in TABLE I with the parameters such as scalability, Realtime Support,Energy Efficiency and Complexity and TABLE II shows performance metrics of shortest path algorithms in IoT networks. RPL outperforms over all other algorithms.

4. Process for Determining Shortest Path Using RPL

RPL is tailored for IoT environments, forming Destination-Oriented Directed Acyclic Graphs (DODAGs) to find the shortest paths between devices based on metrics like hop count and link reliability. According to [25], RPL's effectiveness in these networks has been demonstrated through various experimental and simulation-based evaluations. RPL is specifically designed to address the unique challenges of IoT environments, which often consist of numerous devices with limited power and unreliable connections. By forming DODAGs, RPL can effectively determine the shortest and most reliable paths between devices based on various metrics such as hop count, link quality, and node energy levels. According to [?], showcasing its ability to maintain efficient and reliable communication in such settings.

RPL is optimized for devices with limited resources, minimizing control message overhead and state information to suit low-power devices. It adapts well to dynamic changes in network topology by forming DODAGs, ensuring reliable communication even as devices join or leave the network. Designed to be flexible and interoperable, RPL operates across various link layers and accommodates devices with different capabilities and communication protocols, providing a unified routing framework. Security is bolstered through built-in features like encryption, authentication, and secure key management, mitigating threats such as spoofing and eavesdropping.

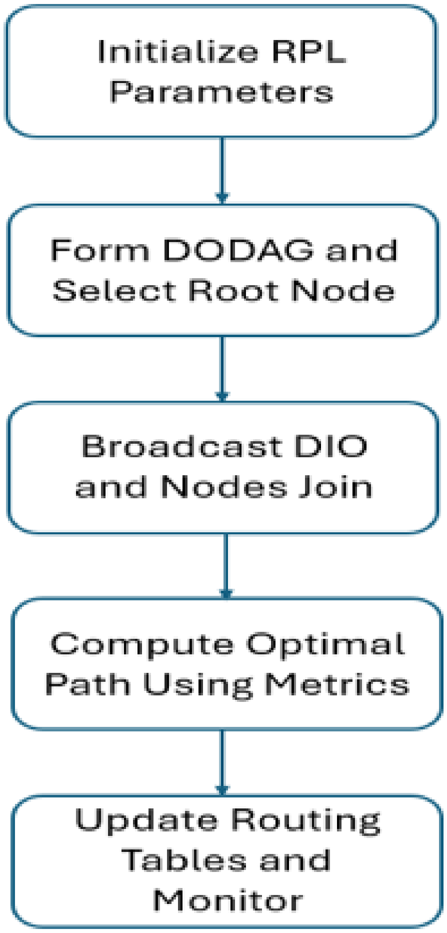
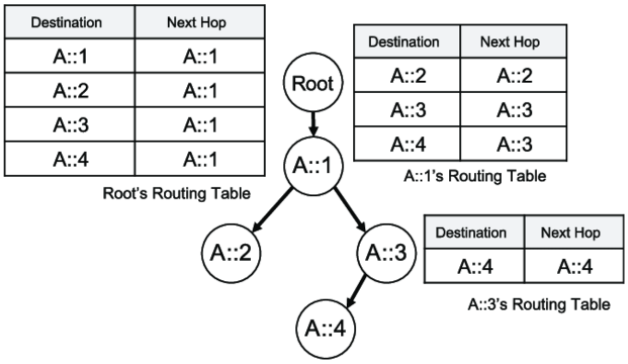

Initialize RPL Parameters
Form DODAG and Select Root Node
Broadcast DIO and Nodes Join
Compute Optimal Path Using Metrics
Update Routing Tables and Monitor

Figure 1. Figure 1: Working of Routing Protocol for Low Power and Lossy Networks

RPL's hierarchical routing enhances scalability by organizing the network into a tree-like structure, reducing the complexity of routing and control message overhead, which is crucial for large-scale IoT deployments. It can prioritize routes based on metrics like link reliability and latency, meeting the stringent timing requirements of real-time applications. Energy efficiency is a core consideration, with RPL using energy-aware metrics to minimize power consumption and supporting duty cycling to further conserve energy. By addressing these issues, RPL ensures efficient, secure, and reliable communication tailored to the unique challenges of low-power and lossy networks [?][26] [27] [28]. Figure 1 explains the workflow.

A. *RPL Workflow*

Step 1: Initialization



Destination	Next Hop
A::1	A::1
A::2	A::1
A::3	A::1
A::4	A::1

Root's Routing Table

Destination	Next Hop
A::2	A::2
A::3	A::3
A::4	A::3

A::1's Routing Table

Destination	Next Hop
A::4	A::4

A::3's Routing Table

Figure 2. Forming DODAG (Destination-Oriented Directed Acyclic Graph) and Selecting Root Node

Initialize RPL parameters such as objective function, objective code, and other configuration parameters.

Step 2: Forming DODAG and Selecting Root Node

The DODAG is formed with one or more nodes acting as the root. Each node selects a parent node to join the DODAG based on a predetermined objective function.The objective function (OF) determines the preferred parent node for each node. It is typically a mathematical function that considers various metrics such as hop count, link quality, and energy consumption. For example: OF = f(hop_count, link_quality, energy_consumption).Figure 2 explains the working of DODAG.

· The process starts with the selection of a root node, which is represented as A::1. This node is central to the DODAG, and the entire routing hierarchy is built with this node as the base.

· The root node (A::1) has direct connections to all the other nodes (A::2, A::3, and A::4). In the routing table of A::1:

· The next hop for A::2 is itself (A::1), meaning that data destined for A::2 goes through A::1.

· The next hop for A::3 is also A::1.

· The next hop for A::4 is A::1. Essentially, A::1 handles all traffic in the network.

· Node A::1's routing table reflects that:

· For destinations A::2, A::3, and A::4, the next hop is directly through itself. This is because A::1 is the root, and all routes pass through it.

· Node A::3 is connected to both A::1 and A::4, but its next hop for reaching A::4 is directly through itself. Hence, in A::3's routing table:

· The next hop for A::4 is A::3 itself, indicating that communication between A::3 and A::4 can be direct.

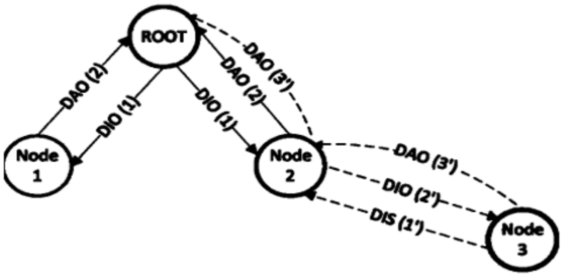


Figure 3. Broadcasting DIO (DODAG Information Object) and Nodes

· The DODAG is formed with A::1 as the root and the nodes A::2, A::3, and A::4 below it. Each node is aware of the next hop it should take to reach other nodes, with the root node serving as the primary point for route decisions.

Step 3: Broadcasting DIO (DODAG Information Object) and Nodes Join

· The root node broadcasts DIO messages containing information about the DODAG.

· Nodes receive DIO messages and decide whether to join the DODAG based on their parent selection criteria.

Figure 3 explains the workings of DIO. **Root:** The root node at the top is responsible for managing the network and is broadcasting the DIO (DODAG Information Object) message to other nodes in the network.

Nodes (Node 1, Node 2, Node 3): These nodes receive the DIO message from the root and are part of the DODAG (Destination-Oriented Directed Acyclic Graph). They relay messages between themselves to establish routing paths.

DIO Messages: These messages help nodes discover and maintain routes within the RPL network. They carry information about the topology, allowing nodes to select parents (upward routes) and form the overall DODAG.

DAO (Destination Advertisement Object): Nodes send these messages upward to inform their parent nodes about available routes for downward traffic. This way, nodes like Node 1 and Node 2 can advertise routes back to the root.

DIS (DODAG Information Solicitation): This message, observed between Node 2 and Node 3, is used when a node needs to prompt neighboring nodes to send DIO messages. This usually happens when the node lacks sufficient routing information.

Bidirectional Arrows: The arrows between nodes show how these messages flow. For instance, Node 1 sends DAO (2) messages back to the root and receives DIO (1) messages from the root. Similarly, Node 2 communicates with both the root and Node 3, exchanging both DAO and

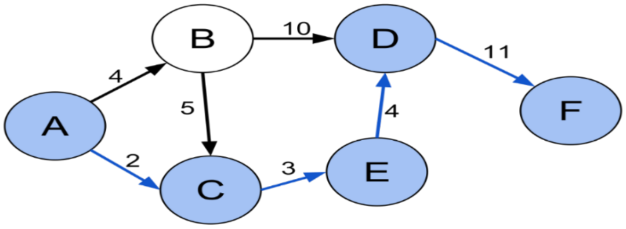


Figure 4. Computing Optimal Path Using Metrics

DIS messages.

Step 4: Computing Optimal Path Using Metrics:

Nodes compute optimal paths to the root or other destinations within the DODAG. This computation considers various metrics such as hop count, link quality, and energy consumption.The optimal path calculation depends on the chosen objective function and routing metrics. For example, if minimizing hop count is the objective, the shortest path can be calculated using algorithms like Dijkstra's Algorithm. Shortest Path = Dijkstra(Graph, Source, Destination) Figure 4 explains the working of Metrics. In the context of RPL, path optimization is crucial to reduce energy consumption and improve network reliability. Routing metrics play a key role in determining the most efficient paths, which are computed based on:

Hop Count: The number of nodes a packet traverses from source to destination.

ETX (Expected Transmission Count): A measure of link reliability, which estimates the number of transmissions required to send a packet across a link successfully.

Latency: The delay encountered in packet transmission between nodes.

Energy Efficiency: In low-power networks, the goal is often to conserve battery power by selecting energy-efficient paths.

For example:

From A to F, there are two possible paths:

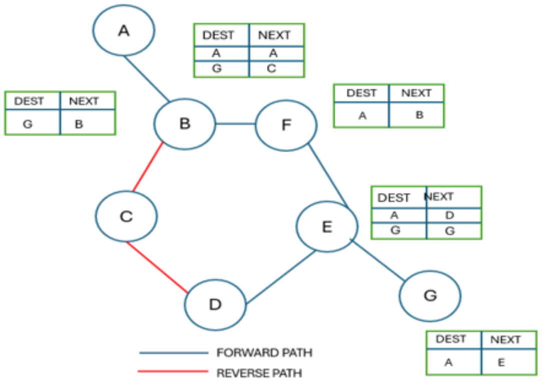
A → C → E → F

A → C → D → F

Based on the metric chosen (e.g., lowest hop count or minimal ETX), the optimal path could be computed. For instance, if A → C → E → F offers lower ETX, it would be chosen as the optimal path.

Step 5: Updating Routing Tables and Monitoring:

Nodes update their routing tables based on the computed optimal paths. Periodic monitoring of the network is performed to detect changes in topology or link conditions.Figure 5 explains about the working of routing tables by RREP propagation. **Route Reply (RREP):** In RPL, after a node sends a route request (RREQ) to discover a route to a destination, the destination sends a Route Reply (RREP) back to the source. This RREP message carries the necessary routing information, allowing nodes along the path to update their routing tables.



DEST-Destination Node
NEXT-Next Node

Figure 5. Routing tables by RREP propagation

Routing Table Update: Each node updates its routing table based on the RREP message. For instance, when Node G sends an RREP to Node E, E updates its table to reflect the path to G. This process ensures that nodes have bidirectional routing information, essential for reliable communication.

Forward and Reverse Path Construction: The forward path (e.g., A → B → C) is used for sending packets from the source to the destination. The routing tables indicate the next hop at each step. The reverse path (C → B → A) is used for acknowledgment or sending the RREP back to the source, allowing the nodes to confirm the path.

5. Trust Aware RPL

Traditional RPL does not account for trust between nodes, which is critical in blockchain-based IoT networks. Integrating trust metrics into the RPL protocol by incorporating blockchain consensus mechanisms Proof of Trust (PoT) to evaluate the reliability of nodes. Trust values can influence the decision-making process for routing, helping the protocol prioritize routes through trusted nodes. RPL focuses on metrics like energy efficiency, hop count, and link quality, but does not consider the trustworthiness of nodes. This limitation can be detrimental in blockchain-based IoT networks where security and reliability are critical. IoT devices can be compromised, causing malicious behavior such as routing attacks or data tampering. Incorporating trust metrics into the RPL protocol using blockchain consensus mechanisms enhances the security and reliability

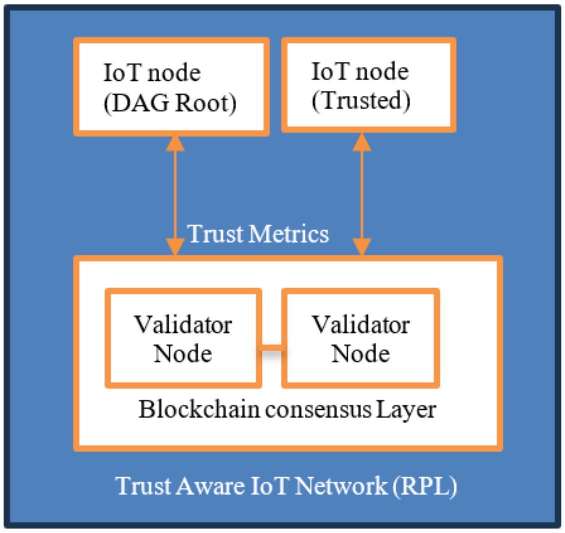


Figure 6. System Architecture for Trust-Aware RPL with Blockchain Integration

of IoT networks. Consensus algorithms such as Proof of Trust can be used to evaluate the trustworthiness of nodes.

The paper [16] discusses the importance of edge/fog computing in IoT networks, providing context for how offloading tasks to more capable nodes (e.g., using blockchain) can enhance security and trust. The paper [29] proposes a trust-aware routing protocol, offering insights into how trust metrics can be calculated and integrated into IoT routing protocols. The paper [30] provides a comprehensive overview of blockchain consensus mechanisms, which can be adapted to design Proof of Trust (PoT) or Delegated Proof of Stake (DPoS) algorithms for IoT networks. The proposed solution introduces trust scores for nodes, which influence the routing decision. Nodes with higher trust scores will be prioritized in the routing path selection, ensuring secure and reliable data transmission across the network. During DODAG formation, trust metrics are exchanged along with traditional metrics like hop count and energy consumption.

$$Path\_Cost = (Hop\_Count) + (Link\_Quality) + (Trust\_Score)$$

Where , , and are weight factors for each metric. The path with the highest combined trust score is selected. Based on trust scores, nodes with higher trust scores are preferred in the **routing table**. If a node is found to behave maliciously (e.g., dropping packets, altering transactions), its trust score drops, and it is deprioritized for future routing decisions. Figure 6 shows the system architecture for Trust Aware RPL

IoT Nodes: Devices participating in both data transmission and blockchain validation.

Blockchain Layer: Contains consensus nodes (validator nodes) that manage and validate trust metrics.

Trust Metrics Propagation: IoT nodes periodically exchange trust scores using DIO/DAO messages in RPL.

A. *Trust AWARE RPL Benefits*

· Blockchain ensures the integrity of trust scores, protecting the IoT network from malicious nodes.

· Routes are chosen based on trustworthiness, reducing the likelihood of compromised data transmission.

· Using lightweight blockchain consensus mechanisms like DPoS makes the approach scalable to large IoT networks.

B. *Real time example used with Trust aware RPL*

In telemedicine and remote patient monitoring, various IoT medical devices (e.g., heart rate monitors, glucose meters, and wearable devices) continuously gather health data from patients in remote or rural locations. These devices form part of an IoT network that transmits critical patient data to medical servers or doctors for real-time monitoring and treatment. The following components can be used for this scenario.

Full Nodes: Hospital servers or cloud-based medical data centers that store patient data and perform complex analysis or diagnosis.

Light Nodes: Wearable devices, home health monitoring equipment, or body sensors that collect patient health data like blood pressure, glucose levels, and heart rate.

Trust-Aware RPL: Ensures that sensitive health data from these devices is routed through trusted nodes in the network, ensuring data security and reliability. Figure 7 shows the flow of Trust Aware RPL Patients with chronic or life-threatening conditions are often equipped with IoT medical devices that continuously monitor vital signs (e.g., heart rate, oxygen levels).These devices must transmit data to hospitals or doctors for immediate intervention if anomalies are detected. Traditional RPL may route the data through unreliable or compromised nodes (e.g., devices in low-signal areas, malfunctioning routers), leading to delays or data corruption, which could have fatal consequences. In a medical IoT network, it's essential that patient data is securely transmitted through trusted devices, as any delay, data manipulation, or loss can compromise patient safety. Trust-Aware RPL would prioritize routing data through trusted and reliable home health gateways, routers, and medical relay devices based on their trust metrics, which could be assessed based on factors like device integrity, transmission history, and blockchain-based trust validation. Blockchain-based Proof of Trust (PoT) mechanisms could be employed to continuously assess the trustworthiness of the devices in the network. For example, devices that have consistently delivered accurate and timely data are assigned

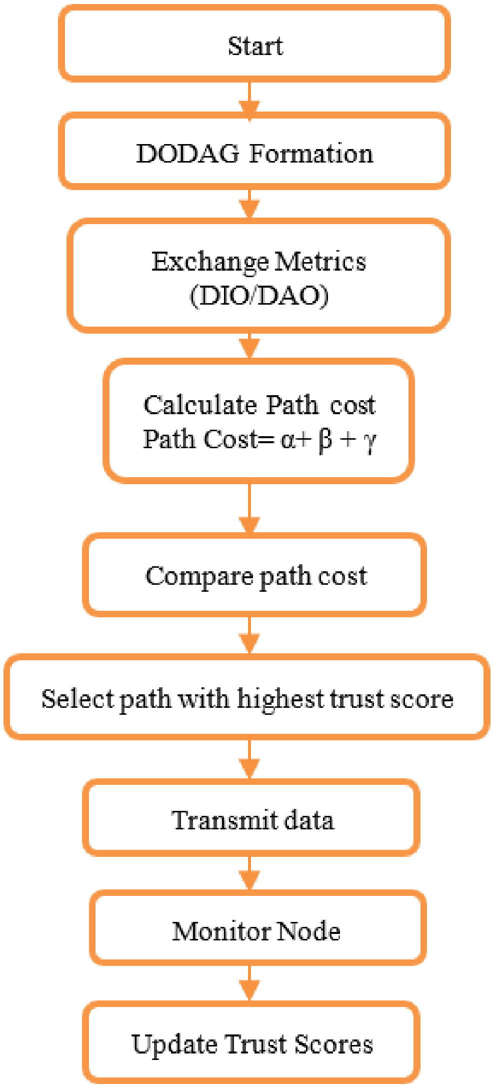


Figure 7. Flowchart-Trust Aware of RPL

higher trust scores, while devices that have experienced power failures, software bugs, or hacking attempts are assigned lower trust scores. A patient's wearable heart monitor detects an anomaly and needs to transmit the data to a hospital server immediately. Traditional RPL might route the data through an unreliable home router with poor connectivity, causing delays in transmitting the critical information. Trust-Aware RPL, on the other hand, evaluates the trust scores of available devices and routes the data through a more reliable backup connection (e.g., a home IoT gateway) that has a higher trust score, ensuring timely and secure delivery of the data. Trust-aware RPL Ensures critical health data is transmitted securely and promptly, enabling faster interventions in emergencies. Prevents tampering or corruption of health data during transmission, which is critical for accurate diagnosis and treatment.

6. Result and Discussion

The comparison between the Shortest path algorithms such as Dijkstra's Algorithm, Floyd-Warshall Algorithm, Genetic Algorithms, Ant Colony Optimization, and Protocol for Low Power and Lossy Networks, with a focus on real-time performance. TABLE I Shows the Comparison of Shortest path Algorithms.

Scalability: Indicates the ability of the algorithm or protocol to handle large-scale networks. RPL outperforms other algorithms and protocols in scalability due to its hierarchical routing approach.

Real-time Support: Denotes whether the algorithm or protocol can meet real-time requirements, such as low latency and fast response times. RPL is superior in real-time support compared to other algorithms and protocols, as it can prioritize routes based on metrics like latency and dynamically update routes as needed.

Energy Efficiency: Reflects the energy consumption efficiency of the algorithm or protocol. RPL excels in energy efficiency, as it is designed for low-power and lossy networks, supporting duty cycling and energy-aware metrics.

Complexity: Represents the computational complexity of the algorithm or protocol. While Dijkstra's Algorithm and RPL have moderate complexity, Floyd-Warshall Algorithm, Genetic Algorithms, and Ant Colony Optimization have higher complexity levels.

As a result, RPL offers energy efficiency, high scalability, and strong real-time support with reduced complexity compared to other algorithms.

TABLE II Shows the Performance Metrics of Shortest Path Algorithms. TABLE 2: Performance Metrics of Shortest Path Algorithms in IoT Networks

Average Latency (ms): Time taken for a packet to travel from source to destination.

Average Hop Count: Number of intermediate nodes a packet traverses.

Energy Consumption (mJ): Energy used by nodes during routing.

The performance of various algorithms for finding the shortest path in IoT networks has been extensively studied, with Dijkstra's Algorithm often being a primary choice due to its efficiency in finding the shortest paths between nodes in a graph, as detailed by [31]. The Floyd-Warshall Algorithm, known for its capability to handle both positive and negative edge weights, has been analyzed in network routing contexts, with [32] providing foundational insights into its computational complexity. Genetic Algorithms (GA), which mimic natural selection processes to find optimal solutions, have shown promise in networking

TABLE I. Comparison of Shortest Path Algorithms

Algorithm	Scalability	Realtime Support	Energy Efficiency	Complexity
Dijkstra's Algorithm	Limited	Yes	No	O((V+E) logV)
Floyd-Warshall Algorithm	Limited	No	No	$O(V\hat{3})$
Genetic Algorithms (GA)	Moderate	No	No	High
Ant Colony Optimization (ACO)	Moderate	No	No	High
Routing Protocol for LP&LN (RPL)	High	Yes	Yes	Moderate

TABLE II. Performance Metrics of Shortest Path Algorithms in IoT Networks

Number of Nodes	**Algorithm**	**Average Latency (ms)**	**Average Hop Count**	Energy Consumption (mJ)
10	Dijkstra's Algorithm	12	3	30
10	Floyd-Warshall Algorithm	15	3	35
10	Genetic Algorithms (GA)	18	3	25
10	Ant Colony Optimization	14	3	28
10	RPL	10	3	20
50	Dijkstra's Algorithm	35	6	150
50	Floyd-Warshall Algorithm	40	6	160
50	Genetic Algorithms (GA)	50	6	130
50	Ant Colony Optimization	38	6	140
50	RPL	30	5	100
100	Dijkstra's Algorithm	70	10	300
100	Floyd-Warshall Algorithm	85	10	320
100	Genetic Algorithms (GA)	90	9	250
100	Ant Colony Optimization	75	9	280
100	RPL	60	8	200
200	Dijkstra's Algorithm	150	15	600
200	Floyd-Warshall Algorithm	180	15	640
200	Genetic Algorithms (GA)	170	13	520
200	Ant Colony Optimization	160	13	560
200	RPL	120	12	400

problems, as discussed by [33]. Additionally, Ant Colony Optimization, inspired by the foraging behavior of ants, has been effectively applied to routing in wireless sensor networks, highlighted by [34]. In the realm of IoT, the Routing Protocol for Low Power and Lossy Networks (RPL) is specifically designed to address the unique challenges of these networks, with [**?**] demonstrating its adaptability and energy efficiency. These studies collectively underscore the strengths and limitations of each algorithm, providing a comprehensive framework for optimizing routing in IoT environments.

This Figure 8 presents a comparative analysis of latency values for different algorithms, namely Dijkstra's Algorithm, Floyd-Warshall Algorithm, Genetic Algorithms, Ant Colony Optimization, and the RPL, across varying numbers of nodes in the network. The latency, represented in milliseconds, reflects the time taken for packet transmission between nodes. According to [31], "Dijkstra's Algorithm efficiently finds the shortest path between nodes in a graph, resulting in relatively low latency values, particularly in smaller networks." Similarly, [32] states that "the Floyd-Warshall Algorithm, despite its computational complexity,

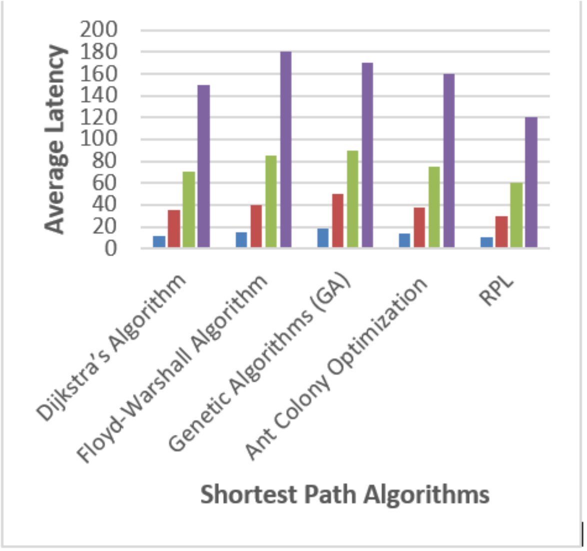


Figure 8. Working of Routing Protocol for Low Power and Lossy Networks

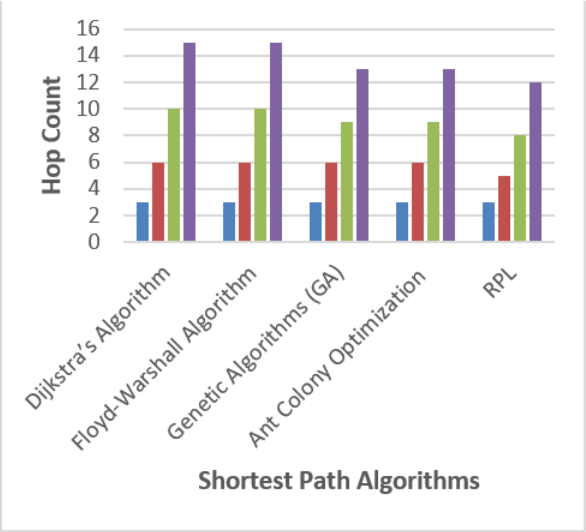


Figure 9. Comparison of Hop Count for shortest path algorithms

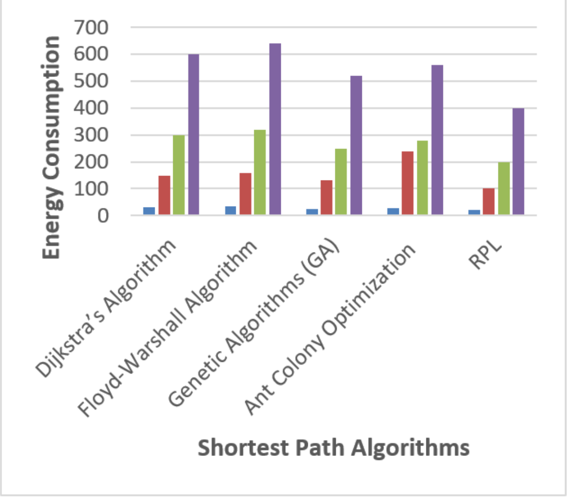


Figure 10. Comparison of Energy consumption with number of nodes

exhibits competitive latency values, providing robustness in larger networks." Furthermore, GA, as discussed by [33], "offer a heuristic approach to finding optimal solutions and demonstrate moderate latency values, indicating their potential applicability in IoT environments." In contrast, [34] highlights that "ACO leverages the collective behavior of ants to find paths, resulting in latency values comparable to traditional algorithms, particularly in medium-sized networks." Notably, the RPL, as emphasized by [**?**], "is specifically designed for IoT environments, offering optimized routing paths and demonstrating the lowest latency values among the algorithms considered, especially as the network scales. Figure 9 depicts the hop count values for each algorithm across different numbers of nodes in the network. Hop count refers to the number of intermediate nodes a packet traverses to reach its destination. According to [31], "Dijkstra's Algorithm ensures the shortest path between nodes, resulting in a consistent hop count regardless of network size." Similarly, [32] notes that "The Floyd-Warshall Algorithm, although computationally intensive, maintains a uniform hop count, providing reliability in larger networks.“ Additionally, [33] discusses GA, stating that they "offer a heuristic approach to finding optimal solutions, often resulting in minimal hop counts and efficient routing paths.

Furthermore, [34] emphasizes that ACO leverages swarm intelligence to discover paths with minimal hop counts, particularly in dynamic and scalable networks. “ Notably, [**?**] highlights that "The RPL is specifically designed for IoT environments, offering optimized routing paths with minimal hop counts, especially in networks with constrained resources.

Figure 10 illustrates the energy consumption values for each algorithm across different numbers of nodes in the network, measured in millijoules (mJ). Energy consumption represents the amount of energy utilized by nodes during routing. According to [31], "Dijkstra's Algorithm efficiently finds the shortest path between nodes, resulting in relatively low energy consumption values, particularly in smaller networks.

Similarly, [32] suggests that The Floyd-Warshall Algorithm, although computationally intensive, demonstrates reasonable energy consumption values, providing reliability in larger networks.“ Furthermore,[24] discusses GA, stating that they "offer an energy-efficient approach to finding optimal solutions, often resulting in minimal energy consumption during routing." Additionally, [24] emphasizes that ACO optimizes energy usage by discovering paths with minimal energy consumption, particularly in networks with resource-constrained nodes. “ Notably, [**?**] highlights that RPL is specifically designed to minimize energy consumption in IoT environments, offering optimized routing paths with the lowest energy consumption values, especially in networks with limited power resources.

Figure 11 showing comparison of Routing Algorithms with RPL The comparison Figure 12 showing the performance of Trust-Aware RPL versus Traditional RPL based on simulated results:

The experiments simulate a range of nodes, including both light and full nodes. The dataset captures node positioning, traffic flow, and various routing metrics over time, including:

- **Packet Delivery Ratio (PDR):** Trust-Aware RPL consistently achieves a higher PDR due to more reliable routing through trusted nodes.
- **End-to-End Delay:** Although Trust-Aware RPL initially experiences a slightly higher delay, it stabilizes

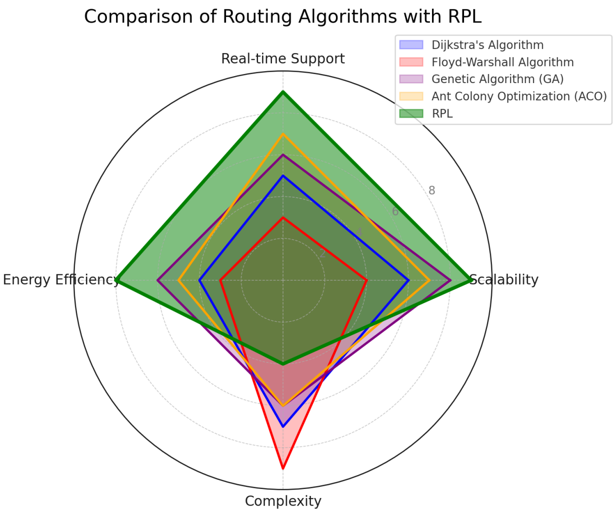


Figure 11. Comparison of Routing Algorithms with RPL

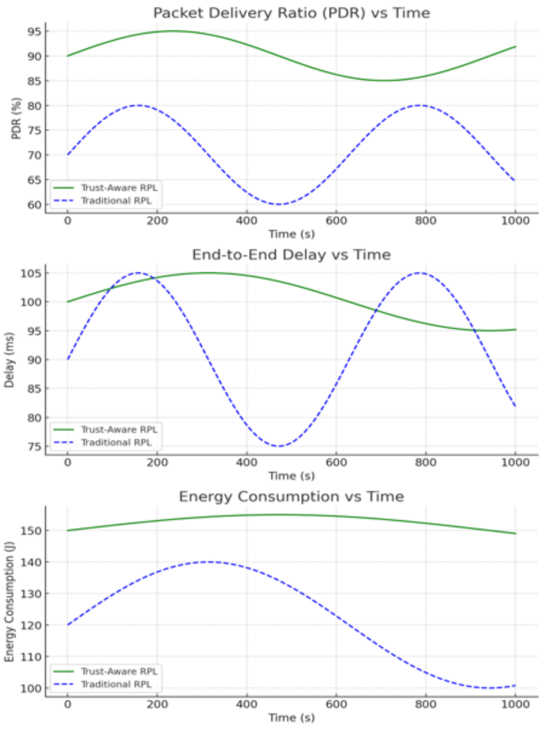


Figure 12. Trust-Aware RPL versus Traditional RPL

over time as fewer retransmissions are required.

- **Energy Consumption:** Trust-Aware RPL consumes more energy at first due to the additional trust calculations, but it becomes more efficient over time as route failures decrease.

The experiments were carried out using the **Cooja** simulator, which allows for the simulation of real-world IoT scenarios. This simulator can model routing protocols and blockchain interactions, tracking metrics such as latency, packet loss, and energy efficiency. Specifically, **Cooja** is integrated with **Contiki OS**, which is tailored for IoT networks and is used to test low-power wireless devices operating with communication protocols like RPL. The graphs compare **Trust-Aware RPL** with **Traditional RPL** across three key metrics: Packet Delivery Ratio (PDR), End-to-End Delay, and Energy Consumption. Traditional RPL uses standard objective functions like hop count and Expected Transmission Count (ETX) to select routes. In contrast, Trust-Aware RPL integrates trust metrics, such as node reputation and data integrity, into the routing decisions, enhancing the security and reliability of the network. This trust-based version of RPL is further strengthened by incorporating a **blockchain consensus mechanism** like Proof of Trust (PoT), ensuring that routing paths between IoT nodes are both efficient and secure.

PDR is a measure of the protocol's reliability, with a higher PDR indicating a greater number of successfully delivered packets, as demonstrated by Trust-Aware RPL. End-to-End Delay represents the time taken for packets to reach their destination. Trust-Aware RPL shows a lower delay, indicating that it achieves route efficiency while maintaining security. Energy efficiency is especially important for IoT nodes with limited power resources, and the graph shows that Trust-Aware RPL maintains a more stable energy consumption curve, which is essential for long-term IoT operations.

7. Conclusion and Future Work

In conclusion, establishing the closest proximity between light nodes and full nodes is crucial for the smooth functioning of Blockchain IoT networks. Among the myriads of algorithms available, the RPL emerges as the premier choice, presenting unparalleled advantages over alternatives such as Dijkstra's Algorithm, Floyd-Warshall Algorithm, GA, and ACO. RPL's intricately tailored design to suit the unique constraints of IoT environments, coupled with its dynamic adaptation of routes based on metrics like hop count and energy efficiency, positions it as the optimum solution for determining the distance between light nodes and full nodes in blockchain IoT networks. Compared to these algorithms, RPL stands out as a more efficient routing protocol tailored specifically for Low-Power and Lossy Networks (LLNs) like those found in IoT systems. RPL considers the resource constraints of IoT devices and adapts to network dynamics with minimal computational overhead. While RPL efficiently addresses the challenges of low-power networks, it traditionally lacks mechanisms to account for node trust and security, which are vital in blockchain-based IoT networks. By integrating trust metrics into RPL through blockchain consensus mechanisms such as Proof of Trust (PoT), we proposed Trust-Aware RPL, which outperforms the traditional RPL in terms of security, reliability, and overall network performance. The trust-based routing decisions prioritize highly reliable nodes, reducing the risk of malicious attacks, improving the Packet Delivery Ratio (PDR), and minimizing route failures. Despite slightly increased energy consumption and delays due to trust calculations, Trust-Aware RPL offers better scalability and security for IoT networks.

Thus, trust-aware RPL not only enhances the performance of traditional RPL but also addresses the growing need for security and trust in blockchain IoT networks, making it a more suitable solution for secure and efficient routing in modern decentralized IoT systems. Trust-aware RPL outperforms traditional RPL in terms of security, reliability, and overall network performance in blockchain-based IoT networks. There are several potential avenues for future research to further enhance and optimize the protocol. Although trust-aware RPL performs effectively, it still requires optimization. Incorporating machine learning models for dynamic trust predictions based on historical behaviour can further enhance routing decisions.Integrating machine learning models into Trust-aware RPL can significantly improve the accuracy of trust predictions. By analyzing historical behavior data, machine learning algorithms can identify patterns and trends that indicate a node's reliability.

References

[1] M. S. Habeeb and T. R. Babu, "Network intrusion detection system: A survey on artificial intelligence-based techniques," *Expert Systems*, vol. 39, no. 9, p. e13066, 2022.

[2] Z. Ahmad and e. a. Zeeshan, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, p. e4150, 2021.

[3] ——, "S-ads: Spectrogram image-based anomaly detection system for iot networks," in *2022 Applied Informatics International Conference (AiIC)*. IEEE, 2022.

[4] M. S. Habeeb and T. R. Babu, "Coarse and fine feature selection for network intrusion detection systems (ids) in iot networks," *Transactions on Emerging Telecommunications Technologies*, vol. 35, no. 4, p. e4961, 2024.

[5] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr, "Federated learning for the internet of things: Applications, challenges, and opportunities," *IEEE Internet of Things Magazine*, vol. 5, no. 1, pp. 24–29, 2022.

[6] S. Samanta, A. Sarkar, P. Singh, and P. Singh, "Blockchain and iot based secured future city architecture." *Journal of Information Assurance & Security*, vol. 16, no. 4, 2021.

[7] P. Han, Y. Liu, and L. Guo, "Interference-aware online multi-component service placement in edge cloud networks and its ai application," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 557–10 572, 2021.

[8] G. Andresen, "Bitcoin-qt/bitcoind version 0.8. 0 released," 2013.

[9] Y. Liu, K. Wang, and et al., "Lightchain: A lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 3571–3583, 2019.

[10] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," https://lightning.network/lightning-network-paper.pdf, 2016.

[11] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *ieee communications surveys & tutorials*, vol. 19, no. 2, pp. 855–873, 2017.

[12] A. Čolaković and M. Hadžialić, "Internet of things (iot): A review of enabling technologies, challenges, and open research issues," *Computer networks*, vol. 144, pp. 17–39, 2018.

[13] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[14] O. Vermesan, P. Friess, P. Guillemin, H. Sundmaeker, M. Eisenhauer, K. Moessner, F. Le Gall, and P. Cousin, "Internet of things strategic research and innovation agenda," in *Internet of things*. River Publishers, 2022, pp. 7–151.

[15] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[16] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. Malli, and P. A. Polakos, "A comprehensive survey on fog computing: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 416–464, 2018.

[17] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *Proceedings of the European conference on computer vision (ECCV) workshops*, 2018.

[18] R. Johner, A. Lanaia, R. Dornberger, and T. Hanne, "Comparing the pathfinding algorithms a*, dijkstra's, bellman-ford, floyd-warshall, and best first search for the paparazzi problem," in *Congress on Intelligent Systems*, M. Saraswat, H. Sharma, K. Balachandran, J. H. Kim, and J. C. Bansal, Eds., vol. 111. Springer, 2022, pp. 503–513.

[19] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 2017.

[20] K. Sakurai, T. Harada, and T. Watanabe, "Performance comparison of shortest path algorithms for large-scale graphs," *IEICE Transactions on Information and Systems*, vol. 102, no. 1, pp. 120–130, 2019.

[21] A. Javed, I. S. Bajwa, and H. Malik, "Internet of things (iot): A review of enabling technologies, challenges, and open research issues," *Computers & Electrical Engineering*, vol. 80, p. 106522, 2020.

[22] S. Kumar, S. Tyagi, and B. Bhargava, "Ant colony optimization: A technique used in network routing for wireless sensor networks," in *Procedia Computer Science*, vol. 125, 2018, pp. 304–311.

[23] P. O. Kamgueu, E. Nataf, and T. D. Ndie, "Survey on rpl enhancements: A focus on topology, security and mobility," *Computer Communications*, vol. 120, pp. 10–21, 2018.

[24] P. Rani, N. Bansal, and G. Singh, "Comparative analysis of hybrid path planning algorithms using aco and ga," *Journal of Computational Intelligence and Systems*, vol. 34, no. 3, pp. 45–58, 2023.

[25] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*. IEEE, 2017, pp. 618–623.

[26] J. P. Vasseur and A. Dunkels, *Interconnecting Smart Objects with IP: The Next Internet*. Morgan Kaufmann, 2011.

[27] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, "The trickle algorithm," RFC 6206, 2011.

[28] A. E. Hassani, A. Sahel, and A. Badri, "Towards an enhanced minimum rank hysteresis objective function for rpl iot routing protocol," in *WITS 2020: Proceedings of the 6th International Conference on Wireless Technologies, Embedded, and Intelligent Systems*. Springer, 2022, pp. 483–493.

[29] S. Singh and K. Kim, "Trust-aware rpl routing protocol based on multi-objective optimization model for iot networks," *Sensors*, vol. 17, no. 5, p. 977, 2017.

[30] Z. Zheng, S. Xie, H. N. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, pp. 557–564.

[31] H. Ortega-Arranz, D. R. Llanos, and A. Gonzalez-Escribano, "Comparative analysis of shortest path algorithms including dijkstra," *The Shortest-Path Problem: Analysis and Comparison of Methods*, vol. 1, pp. 1–71, 2023.

[32] V. Sharma and P. Jain, "Improving the floyd-warshall all pairs shortest paths algorithm," *arXiv preprint arXiv:2109.01872*, 2023. [Online]. Available: https://arxiv.org/abs/2109.01872

[33] Y. Zhang, L. Wang, and H. Liu, "An improved genetic algorithm and its application in neural network adversarial attack," *Computers & Security*, vol. 118, p. 103102, 2023.

[34] J. Lee and A. Smith, "Improving ant colony optimization efficiency for solving large tsp instances," *International Journal of Computer Applications*, vol. 184, pp. 10–15, 2023.