Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

Self-learning QoS Architectures for Efficient IoT Wireless Communication

Durgesh Kumar Keshar[1], Dr. Guddi Singh[2]
[1]Research Scholar, Department of CSE, Kalinga University,
Kotni, Near Mantralaya, Naya Raipur, Chhattisgarh,Pin –492101 ,India
durgeshkeshar.prsu@gmail.com
[2]Professor, Department of CSE, Kalinga University,
Raipur, Chhattisgarh, Pin –492001, India
guddi.singh@kalingauniversity.ac.in

Abstract
The exploding number of internet-connected devices (IoT) has made it crucial to guarantee reliable data transmission (QoS) in wireless networks. This research introduces a new approach called Self-Adaptive Architecture for QoS (SAAQ) specifically designed for IoT networks. The authors built and tested SAAQ using a network simulation tool (NS-2). SAAQ is adaptable, meaning it can adjust to the ever-changing demands of various IoT applications. It achieves this by fine-tuning critical network performance metrics in real-time. These metrics include the success rate of data delivery, transmission speed, data transfer delays, and resource consumption (energy and routing overhead). The authors integrated SAAQ with NS-2 to perform extensive simulations, mimicking real-world IoT scenarios. These simulations assessed SAAQ's ability to adapt and manage QoS compared to existing methods (AODV, AOMDV, and LEACH). The results confirm that SAAQ offers practical improvements in maintaining QoS within simulated IoT applications.

Keywords-Self-Adaptive Architecture for QoS (SAAQ), IoT-based wireless networks, NS-2 simulation tool, QoS parameters, performance optimization.

1. Introduction

In today's digital age, wireless networks serve as the critical infrastructure for global communication. Their widespread availability, fueled by the explosion of mobile devices, the ever-growing presence of Internet of Things (IoT) applications, and the arrival of 5G technology, has fundamentally transformed how we connect. Wireless networks now act as the backbone for a vast array of services, from basic voice and data communication to high-demand applications like real-time multimedia streaming and mission-critical operations within the IoT realm. Within this transformative landscape of connectivity, Quality of Service (QoS) emerges as a central tenet, playing a vital role in shaping the functionality and reliability of wireless networks. QoS acts like the maestro of network performance, employing a set of tools and strategies to ensure data transmission meets a desired level of quality. These tools focus on factors like minimizing delays (latency), maximizing bandwidth availability, ensuring minimal data loss (packet loss), and optimizing resource utilization. Each of these elements is crucial for guaranteeing the seamless operation of the diverse applications that rely on wireless networks. Consider the smooth streaming of high-definition video, demanding minimal buffering interruptions. Imagine the real-time responsiveness required for autonomous vehicles, where even a slight delay can have significant consequences. Envision the efficient collection and transmission of data from a network of sensors embedded within smart cities, where reliable data delivery is vital for critical operations. The success of all these applications hinges on the network's ability to provide a consistent and predictable level of QoS. Without it, even the most advanced applications can falter, potentially impacting everything from our entertainment experiences to the safety and efficiency of crucial operations.

IoT-based wireless networks act as the central nervous system of the entire IoT revolution. They facilitate the crucial exchange of data, the real-time execution of commands, and the seamless delivery of services across a vast spectrum of applications and industries. From the intricate web of sensors in smart cities that optimize traffic flow and resource management to the interconnected machines on an industrial automation floor that streamline production processes, the influence of IoT-based wireless networks is pervasive. They are fundamentally reshaping the way we interact with the world around us, fostering a future where data collection and analysis are at the heart of informed decision-making. However, traditional wireless network architectures are often ill-equipped to handle the diverse and ever-changing demands of IoT applications. These applications require a delicate balance of various QoS factors, including ultra-low latency for real-time responsiveness, high reliability for ensuring data integrity, exceptional energy efficiency for battery-powered devices, and robust scalability to accommodate the ever-growing number of connected devices. Unfortunately, conventional, static network designs are simply not adaptable enough to keep pace with the constantly evolving demands of the dynamic IoT landscape. They lack the flexibility to automatically adjust and optimize network performance to meet the specific needs of each application and device within the ever-expanding IoT ecosystem.

Therefore, the development of a Self-Adaptive Architecture for QoS (SAAQ) in IoT-based wireless networks becomes a critical imperative. This research initiative is driven by the urgent need to overcome the shortcomings of existing, rigid architectures. By implementing SAAQ, we envision empowering IoT networks with the ability to autonomously adapt to the ever-shifting demands of diverse QoS requirements. Achieving this goal unlocks the full potential of the IoT revolution by establishing a network foundation that can consistently guarantee QoS, even within highly dynamic and heterogeneous environments. Evaluating and implementing the SAAQ routing protocol within a simulated environment like NS-2 represents a complex yet crucial step forward in our quest to understand and improve QoS within IoT networks. This process will provide invaluable insights into how self-adaptive routing can not only enhance the performance of IoT applications but also bolster their overall reliability.

The remainder of this paper delves into a detailed exploration of the research conducted. Section 2 provides a comprehensive review of existing literature relevant to the field. Section 3 meticulously outlines the methodological approach adopted for the research, including the development and implementation of the SAAQ routing protocol within the NS-2 simulation environment. Section 4 presents a thorough analysis of the simulation results, providing valuable data on the performance characteristics of SAAQ. Section 5 builds upon this analysis by offering a comparative assessment of SAAQ against existing routing protocols, highlighting its strengths and potential benefits. Finally, Section 6 concludes the paper by summarizing the key findings and discussing the broader implications of the research. Additionally, this section will explore potential future directions for this research project, outlining opportunities to further refine and advance the SAAQ architecture.

2. Related Work

This research builds upon a foundation established by previous efforts to improve QoS in IoT wireless networks. Here, we examine several key contributions from past studies:

Mahajan (2002) [1] proposed an adaptive architecture that allows users to select lower quality data streams (e.g., lower resolution video) based on their preferences. This approach aimed to improve overall QoS in wireless networks by dynamically adjusting data requirements. Wang (2004) (reference 2) and Wang (2005) (reference 4) explored reservation-based QoS models for integrated cellular and WLAN networks. These models utilized adaptation mechanisms to optimize resource allocation, leading to improvements in network efficiency and reduced call failures. Gatouillat (2017) [3] introduced a framework for self-adaptation focused on QoS in IoT systems. This framework aimed to ensure consistent QoS even when the physical environment surrounding the network changes.

While these prior works offer valuable insights, they often address specific aspects of QoS improvement. The proposed Self-Adaptive Architecture for QoS (SAAQ) in IoT-based wireless networks takes a more comprehensive approach. SAAQ focuses on designing and implementing a management system that can handle real-time uncertainties within the network. This system proactively adapts to ensure guaranteed QoS for the entire IoT ecosystem. The research explores various architectural strategies for implementing the SAAQ management system. It is crucial to evaluate the impact of these strategies on both the managed IoT system (the network itself) and the managing system (SAAQ) to ensure optimal overall performance and minimal resource overhead.

Table 1 provides a more comprehensive overview of the research landscape in this field. It likely delves into additional relevant author findings and methods, offering a more granular understanding of existing approaches to QoS improvement in IoT networks.

Table 1: Literature Survey on some Author's Findings and their Methods

SN	Author(s)	Findings	Methods Measured
1.	Federico, Di, Menna., Henry, Muccini., Karthik, Vaidhyanathan [8]	The paper proposes a framework for designing and evaluating different architectural strategies for implementing the managing system of self-adaptive IoT systems.	-Proposed framework for designing and evaluating architectural strategies -Evaluation results on a real-world IoT system
2.	R., Prabha., Senthil, G., A., N., Naga, Saranya., A., M., K., Somasundaram., K., C. [9]	The paper discusses the design and implementation of an adaptive relay selection method for IoT communication in wireless networks.	- Adaptive priority-based relay selection decisions - Investigation of various user options for relay selection parameters
3.	Avewe, Bassene., Bamba, Gueye [10]	The paper proposes EFQM++, a self-adaptive framework for highly dynamic network topology changes in IoT networks, which improves flow end-to-end transmission delay.	- SDN-based architecture - EFQM++ framework
4.	Satyanarayana, Pamarthi., N., R. [11]	The paper introduces a security mechanism for wireless mobile adhoc networks in IoT applications, but does not specifically mention the design and implementation of a Self Adaptive Architecture for QoS (SAAQ) in IoT based Wireless Networks.	- Encoding and decoding packets using an arbitrary method selection scheme - Authentication approach for security mechanism
5.	Aurélien, Chambon., Abderrezak, Rachedi., Abderrahim, Sahli., Ahmed, Mebarki [12]	The paper proposes a programmable multitier architecture and a continuity of service protocol for dynamic IoT networks, focusing on end-to-end Quality of Service (QoS).	- Programmable multitier architecture - Continuity of service protocol named ConSerN
6.	Ahmad, Khalil., Nader, Mbarek., Olivier, Togni [13]	The paper proposes a QoS-based architecture for IoT environments, enabling the establishment of IoT Service Level Agreements (iSLAs) between IoT Service Providers and IoT Clients.	- QoS mechanisms implemented within each layer of the IoT architecture - Adaptation of the IEEE 802.15.4 slotted CSMA/CA mechanism
7.	Hongyu, Zhou., Xiaomin, Ren [14]	The given text does not provide any information about the design and implementation of a Self Adaptive Architecture for QoS (SAAQ) in IoT based Wireless Networks.	- Crosstalk suppression through beam angle reduction and power isolation - Integration of transmitter and receiver on a single PCB
8.	Cherifa, Boucetta.,	The paper discusses the importance of link	- Machine learning techniques

	Boubakr, Nour., Alberic, Cusin., Hassine, Moungla [15]	quality estimation in IoT networks for enhancing QoS and presents a classification of channels using machine learning techniques.	(KNN and LSTM) - Analysis of received signal strength (RSSI) and error rates
9.	Nogaye, Lo., Ibrahima, Niang [16]	The paper proposes an analysis of QoS solutions and architectures for effective quality of service management in the IoT/Edge computing environment.	- Analysis of QoS solutions and architectures - Classification of QoS mechanisms in IoT architecture
10.	Jiaxin, Liang., He, Chen., Soung, Chang, Liew [17]	The given text does not provide any information about the design and implementation of a Self Adaptive Architecture for QoS (SAAQ) in IoT based Wireless Networks.	- Time synchronization mechanism for maintaining synchrony among nodes - Just-in-time algorithm to reduce delays and delay jitters
11.	Cmm Mansoor, G. Vishnupriya, A. Anand, S. Vijayakumar, G. Kumaran, V. Samuthira Pandi [18]	The need to access various parameters of QoS based on many perspectives is critical in IoT devices.	- Quality Of Service (QoS)
12.	Zia, K., Chiumento, A., & Havinga, P.J [19]	AI and machine learning can address high-dimensional and dynamic problems in multi-RAT IoT networks.	-Throughput -Reliability -Latency (Quality Of Service)
13.	Gatouillat, A., Badr, Y., & Massot, B. [20]	A quality-of-service driven self-adaptation framework can simultaneously handle changing adaptation strategies, monitoring infrastructure and physical environment while guaranteeing constant quality-of-service.	-Quality Of Service -Adaptation Strategies -Monitoring Infrastructure -Physical Environment -Safety Of Monitored Patients
14.	Duan, R., Chen, X., & Xing, T. [21]	A QoS architecture based on IoT layered structure transmits QoS requirements.	-Qos Requirements -Consistency -Effective Use Of Existing Qos Mechanisms In Every Layer
15.	Chu, Y., & Ganz, A. [22]	The proposed protocol assigns transmission parameters to nodes in the network based on the current traffic conditions to nodes in the network based on the current traffic conditions.	-Throughput -Admission Ratio -Energy Consumption -Delay

Given the diverse nature of research in the broad field of IoT, simulation emerges as a powerful tool for analyzing the proposed SAAQ architecture. The next section (Section 3) will delve into the design and implementation of SAAQ within the NS-2 simulator, providing a controlled environment to evaluate its effectiveness in addressing the aforementioned QoS challenges in IoT networks.

3. Methodology Adopted

This section dives deep into the proposed Self-Adaptive Architecture for QoS (SAAQ). SAAQ acts as an intelligent framework that empowers an IoT system to autonomously adjust its behavior based on dynamic environmental changes. These changes can encompass fluctuations in the number of connected devices, the type of data being transmitted or even alterations to the network infrastructure itself. Here, we delve into the specific SAAQ algorithm and methodology implemented within the Network Simulator 2 (NS-2) environments. The SAAQ algorithm code itself is written in C++, the backend language of NS-2. Due to the complexity of the codebase, with numerous directories and core files involved, a detailed line-by-line explanation becomes impractical. Therefore, this section focuses on providing a high-level overview of the proposed SAAQ method's functionality through a comprehensive flowchart. The proposed SAAQ algorithm hinges on the concept of Jellyfish Search Optimization (JSO). JSO is a well-established optimization algorithm that aids in identifying the most efficient route between source and destination nodes within the network. The SAAQ method utilizes the concept of a node's residual energy to initiate communication between nodes and establish a multipath route using the JSO routing approach. This section will present a detailed explanation of the SAAQ method's flowchart and underlying algorithm, providing a clear understanding of its decision-making process.

To effectively evaluate the SAAQ algorithm within the NS-2 simulation environment, several assumptions are made to create a controlled setting. These assumptions are:

- **Random Sensor Deployment:** Sensor nodes are randomly scattered throughout the simulated area, mimicking real-world deployments where device placement may not be meticulously planned.
- **Homogeneous Network:** All sensor nodes begin with the same initial energy level, creating a baseline for comparison.
- **Stationary Nodes:** The sensor nodes remain fixed in their positions throughout the simulation, simplifying network behavior analysis.
- **Limited Energy and No Recharging:** The sensor nodes have a finite amount of energy and cannot be recharged or replaced after deployment, reflecting the resource-constrained nature of many real-world IoT devices.
- **Random Hierarchy and Communication:** Sensor nodes establish a random hierarchy, meaning they don't follow a predefined structure for communication. Data is sent to randomly chosen receiver nodes, simulating dynamic traffic patterns.
- **Static Network Scenarios with Varying Node Counts:** The experiment utilizes eight different static network scenarios, each with a different number of sensor nodes (25, 50, 75, 100, 125, 150, 175, and 200). This allows for a systematic evaluation of SAAQ's performance under varying network densities.
- **Random Communication with UDP:** Communication between nodes occurs randomly using the User Datagram Protocol (UDP). UDP is a connectionless protocol suitable for scenarios where data transmission speed is prioritized over guaranteed delivery. A constant bit rate is maintained at the application layer to ensure consistent data flow.
- **Continuous Communication and Node Depletion:** The simulation runs for 600 seconds, allowing nodes to continuously communicate with each other. During this time, some nodes will deplete their energy and become inactive, mimicking the real-world phenomenon of battery exhaustion. It's important to note that while all nodes start with equal energy, those acting as routers will consume energy faster due to increased traffic forwarding.

The NS-2 simulation environment is configured with specific default parameters, including the wireless channel type, routing protocol (SAAQ in this case), number of nodes, and simulation duration. In a typical wireless network, nodes share information about their location and other details through beaconing messages. This allows sender nodes to locate the destination node more easily. Since the SAAQ algorithm is implemented within the NS-2 C++

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

code, all nodes are programmed to follow its routing logic. Whenever a node initiates the route discovery process, the SAAQ algorithm utilizes Dijkstra's algorithm (Equation 3) to calculate the shortest distance between nodes.

$$n\_hop\_X = abs\ [nX - loc\_X] \quad (1)$$
$$n\_hop\_y = abs[nY - loc\_Y] \quad (2)$$
$$Distance,\ D = \sqrt[2]{n_{hop}Y^2 + n_{hop}X^2} \quad (3)$$

These equations calculate the distance between two nodes based on their X and Y coordinates.

- n_hop_X: absolute difference between the X-coordinate of the current node (nX) and the destination node (loc_X).
- n_hop_y: absolute difference between the Y-coordinate of the current node (nY) and the destination node (loc_Y).
- Distance (D): The calculated distance between the two nodes.

$$X_j\ (t+1) = X_j\ (t) + \alpha\ x\ (\ B_u - B_l)\ x\ rand\ (0,1) \quad (4)$$

The proposed approach leverages JSO to further optimize the path discovery process within the SAAQ algorithm. JSO utilizes a "jellyfish swarm" to guide the traditional Particle Swarm Optimization (PSO) swarm towards finding even better solutions. Similar to PSO, the jellyfish swarm is initialized first. However, the jellyfish update their positions using a distinct algorithm (Equation 4) designed to move them closer to the best-known solution within the search space. The jellyfish move locally within their current positions, and their new positions are calculated based on a motion coefficient (α), upper and lower bounds (Bu and Bl), and a random number generator.

The accompanying Figure 1 likely depicts a flowchart illustrating the SAAQ methodology.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

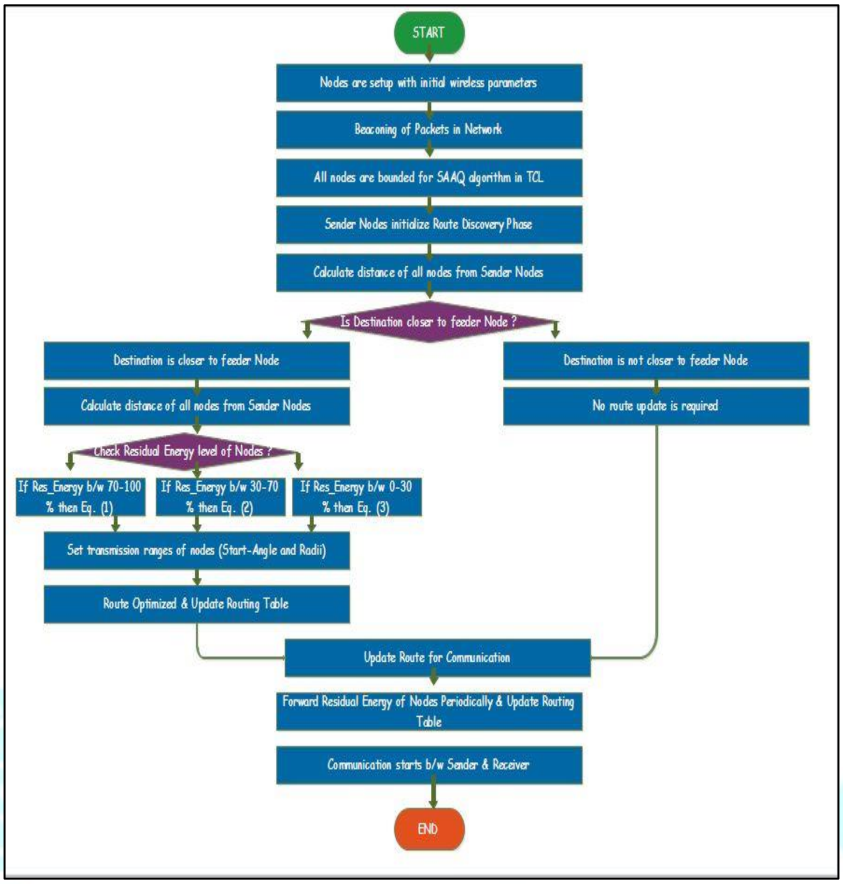

Figure 1: SAAQ Methodology Flowchart

If Nodes residual energy is between 70 to 100 percent set Tx=250 meters (5)

If Nodes residual energy is between 30 to 70 percent set Tx=200 meters (6)

If Nodes residual energy is between 30 to 70 percent set Tx=150 meters (7)

Where Tx is transmission range of nodes.

$$E_{TX}(M,D) = M.E_{elect} + M.\epsilon_{fs}.d^2 \text{ if } d \leq d_0$$

$$M.E_{elect} + M.\epsilon_{mp}.d^4 \text{ if } d > d_0 \qquad (8)$$

The SAAQ algorithm incorporates strategies to streamline the route discovery process and conserve node energy. Here's a breakdown of its approach:

- **Leveraging Feeder Nodes for Efficiency:** If the source node and a feeder node (a high-energy node) are already close together, the SAAQ algorithm skips the optimization step. This reduces unnecessary route discovery overhead and preserves the residual energy of nodes that would otherwise be involved in the process.

- **Dijkstra's Algorithm for Efficient Path Selection (When Needed):** When the source node and feeder nodes are not in close proximity, SAAQ employs Dijkstra's algorithm to identify the shortest path between them. This ensures efficient route selection while minimizing energy expenditure.

- **Dynamic Transmission Range based on Residual Energy:** The SAAQ algorithm dynamically adjusts the transmission range of nodes based on their remaining energy levels. Here's the breakdown of these adjustments:

    - High Energy Nodes (70-100% residual energy): These nodes can transmit over a wider range (Tx = 250 meters) as shown in Equation 5.

    - Medium Energy Nodes (30-70% residual energy): These nodes have a moderately reduced transmission range (Tx = 200 meters) as shown in Equation 6.

    - Low Energy Nodes (Less than 30% residual energy): To conserve their limited energy, these nodes have the shortest transmission range (Tx = 150 meters) as shown in Equation 7.

- **Finding the Optimal Path Within Transmission Range:** The SAAQ algorithm calculates a "radiant path" using a start angle and the adjusted transmission radius. This path is then verified to ensure it falls within the boundary circle of the involved nodes.

- **Energy Consumption Model (ETX):** Equation 8 represents the Expected Transmission Cost (ETX) model used by SAAQ. This model factors in various energy consumption aspects:

    - Eelect: Represents the residual energy of the transmitting node.

    - ∈fs and ∈mp: Denote the energy consumption per bit for free space and multipath fading channels, respectively.

    - d: Represents the distance between the sender and receiver nodes.

    - d0: Represents a threshold distance between transmitter and receiver, calculated using the ratio of ∈fs and ∈mp.

Then subsequently the route is optimized and the routing table of source is updated during the communication. So, for all the RREP received, distance is calculated as given in equation 3. The node having minimum distance is then selected as the next hop and its location is also updated in the routing table as two entries n_hopX and n_hopY.

Select node with and add n_hopX, n_hopY in routing table *(9)*

The minimum distance (dmin) is the distance between the current node and the next hop node in the algorithm. The power threshold received by all nodes remains constant during the path detection phase. This path between source and destination is maintained for data transfer.

4. Simulation Analysis

To assess the effectiveness of the proposed SAAQ algorithm, the research compares its performance with several established routing protocols:

- **AODV (Ad hoc On-Demand Distance Vector):** A well-known routing protocol for dynamic ad-hoc networks.

- **AOMDV (Ad hoc On-Demand Multipath Distance Vector):** An extension of AODV that establishes multiple paths for improved fault tolerance.
- **LEACH (Low-Energy Adaptive Clustering Hierarchy):** A protocol specifically designed for energy efficiency in wireless sensor networks.

The rationale for including these protocols is as follows:

- Prior studies (Modules I and II) have shown AODV outperforms AOMDV in similar scenarios.
- LEACH is a pioneering approach for energy conservation in wireless sensor nodes, providing a valuable benchmark for comparison.

Evaluating Key QoS Parameters Across Different Network Sizes

The research investigates the performance of SAAQ and the compared protocols using six key Quality of Service (QoS) parameters:

i. **Packet Delivery Ratio:** Measures the percentage of data packets successfully delivered from sender to receiver.

ii. **Packet Loss Ratio:** Represents the percentage of data packets that are lost during transmission.

iii. **Average End-to-End Delay:** Measures the average time taken for a data packet to travel from sender to receiver.

iv. **Network Throughput:** Represents the total amount of data successfully transmitted per unit time.

v. **Routing Overhead:** Measures the amount of control traffic generated by the routing protocol.

vi. **Average Energy Consumption:** Represents the average energy spent by nodes within the network.

Eight Network Scenarios with Increasing Node Density

The evaluation employs eight distinct network scenarios within the NS-2 simulator. These scenarios progressively increase in complexity by incorporating a growing number of sensor nodes, ranging from 25 to 200. Within each scenario, random sender-receiver node pairs are selected for communication.

Visualizing Network Behavior through Simulation Snapshots (Figures 3-10)

Figures 3 through 10 depict snapshots of the network scenarios captured during the simulations. These figures offer insights into how nodes communicate and organize themselves under different network densities.

- **Scenarios with 25 to 75 Nodes (Figures 3-5):** These figures showcase the formation of distinct communication zones, potentially centered on feeder nodes with optimized routes for data transmission.
- **Scenarios with 100 and 125 Nodes (Figures 6 & 7):** Here, we observe continued communication within zones, with the visual representation of node transmission ranges (boundary circles) becoming more evident.
- **Scenarios with 150 and 175 Nodes (Figures 8 & 9):** As the network scales, optimized routes continue to facilitate communication between sender and receiver nodes.
- **Largest Scenario with 200 Nodes (Figure 10):** This scenario represents the most complex network with 200 nodes classified into five distinct communication zones.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

These visualizations provide a qualitative understanding of how the network behaves under different conditions and how SAAQ potentially influences communication patterns.

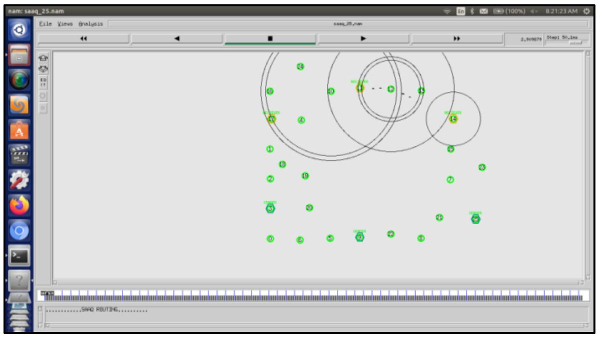

Figure 3: Scenario of 25 nodes

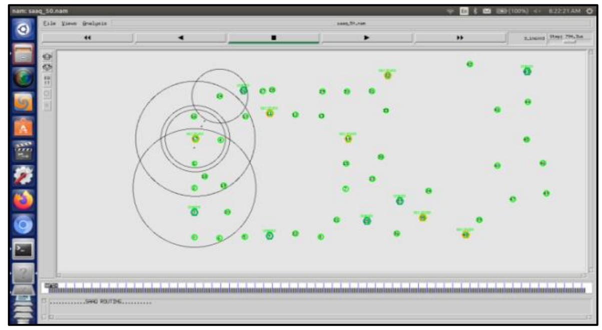

Figure 4: Scenario of 50 nodes

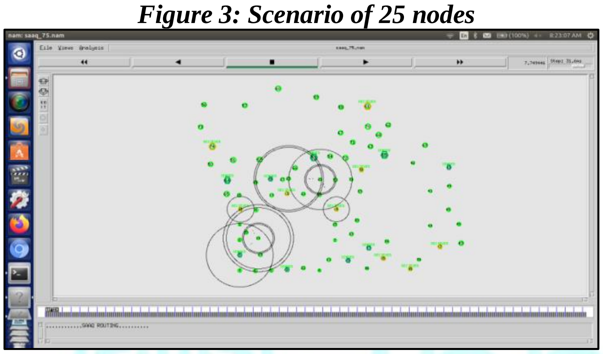

Figure 5: Scenario of 75 nodes

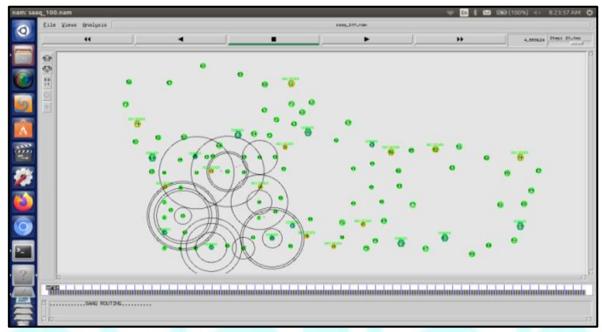

Figure 6: Scenario of 100 nodes

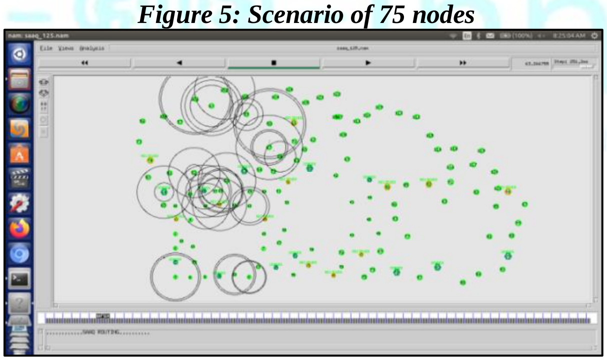

Figure 7: Scenario of 125 nodes

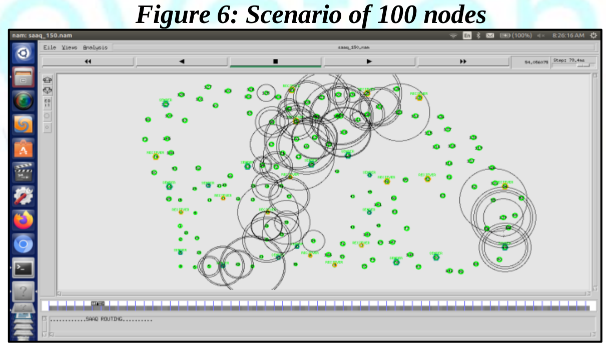

Figure 8: Module3- Scenario of 150 nodes

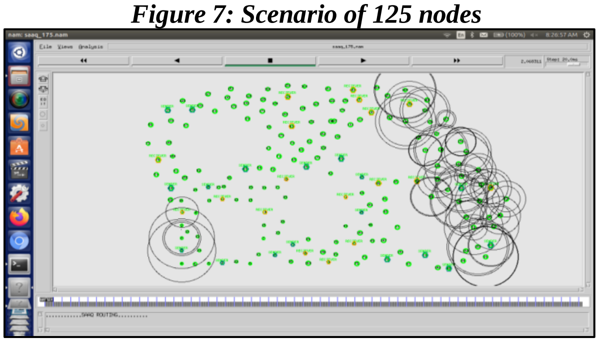

Figure 9: Module3- Scenario of 175 nodes

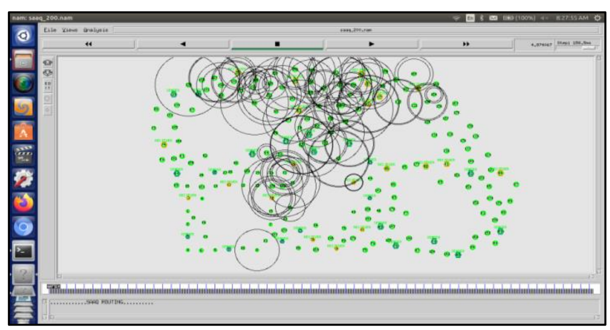

Figure 10: Module3- Scenario of 200 nodes

6. Result Analysis

As previously mentioned, a primary objective of this research is to enhance the overall lifetime of the network by improving QoS parameters related to energy consumption. This focus on energy efficiency is the key reason for including the LEACH routing protocol in the evaluation. By comparing SAAQ against LEACH, the research aims to identify potential areas for further optimization within SAAQ regarding energy usage.

Figure 11 presents a graphical comparison of the packet delivery ratio for all four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes in the network increases. Packet delivery ratio refers to the percentage of data packets that successfully reach their intended destinations. The graph reveals a downward trend in packet delivery ratio for all protocols as the network density grows. This can be attributed to increasing network congestion with a larger number of nodes. More nodes competing for limited resources make it more challenging for routing protocols to discover efficient paths for data transmission. Interestingly, LEACH exhibits the lowest packet delivery ratio among the compared protocols. This stems from its hierarchical structure where the network is divided into clusters. Each cluster relies on a cluster head for internal packet routing. If a cluster head fails, it can lead to a significant increase in packet loss within that cluster. The proposed SAAQ algorithm demonstrates a packet delivery ratio that falls between AODV and LEACH. This suggests that SAAQ offers a balanced approach, potentially achieving a reasonable delivery rate while maintaining some level of energy efficiency compared to AODV.

Figure 12 sheds light on the packet loss ratio for the four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes increases. Packet loss ratio signifies the percentage of data packets that are not delivered successfully due to errors or congestion during transmission. The graph depicts a concerning trend: packet loss ratio rises for all protocols as network density increases. This can be directly attributed to network congestion. With a growing number of nodes vying for limited bandwidth, the probability of packet collisions and transmission failures rises. LEACH, the hierarchical protocol, demonstrates the highest packet loss ratio. Recall that LEACH relies on cluster heads for routing within clusters. If a cluster head fails or becomes overloaded, it can lead to a significant increase in packet loss within that specific cluster. The proposed SAAQ algorithm exhibits a packet loss ratio that falls between AODV and LEACH. This suggests that SAAQ may offer a middle ground, potentially achieving lower packet loss compared to LEACH while maintaining comparable performance to AODV in congested scenarios.

Figure 13 examines the end-to-end delay experienced by data packets across the four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes increases. End-to-end delay refers to the total time taken for a packet to travel from its source node to its intended destination. The graph reveals a concerning trend: end-to-end delay for all protocols worsens as network density grows. This aligns with the previously observed increase in network congestion. With more nodes competing for limited network resources, packets encounter delays as they wait for transmission opportunities or are rerouted due to congestion on specific paths. Similar to the prior observations, LEACH exhibits the highest end-to-end delay among the compared protocols. The hierarchical structure of LEACH, with its reliance on cluster heads for internal routing, can introduce additional delays. If a cluster head becomes overloaded or fails, packets within that cluster may experience significant delays before being forwarded. The proposed SAAQ algorithm demonstrates an end-to-end delay that falls between AODV and LEACH. This suggests that SAAQ may offer a balance, potentially achieving lower delays compared to LEACH while maintaining comparable performance to AODV in congested scenarios.

Figure 14 explores the throughput achieved by the four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes increases. Throughput refers to the amount of data successfully transmitted across the network per unit of time. The graph depicts a counterintuitive trend compared to the previous metrics. In this case, throughput appears to increase for all protocols as network density grows. This can be explained by the concept of increased network interconnectivity. With a larger number of nodes, more potential paths emerge for data transmission. This allows for a greater number of packets to be transmitted concurrently, potentially leading to

ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

higher overall throughput. However, LEACH deviates from this trend, exhibiting the lowest throughput among the compared protocols. This can again be attributed to its hierarchical structure. Cluster heads in LEACH can become bottlenecks, limiting the overall data flow within their respective clusters if they become overloaded or fail. The proposed SAAQ algorithm demonstrates a throughput performance that surpasses both AODV and LEACH. This suggests that SAAQ may be particularly well-suited for applications where high throughput is a critical QoS requirement. The ability of SAAQ to optimize routes and potentially leverage additional paths created by increased network density could contribute to its superior performance in this area.

Figure 15 delves into the routing overhead generated by the four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes increases. Routing overhead refers to the amount of control traffic, such as routing messages, used by the protocol to establish and maintain paths for data transmission. The graph reveals a consistent trend: routing overhead rises for all protocols as network density increases. This aligns with the previously observed network congestion. With more nodes competing for limited bandwidth, routing protocols need to exchange a higher volume of control messages to discover and maintain efficient paths. This additional traffic contributes to the overall routing overhead. LEACH, the hierarchical protocol, demonstrates the highest routing overhead. Recall that LEACH relies on cluster heads for internal routing. These cluster heads need to communicate control messages frequently to maintain the cluster structure and route data packets within their clusters. If a cluster head becomes overloaded or fails, it can disrupt communication and necessitate even more control messages to re-establish routing paths. The proposed SAAQ algorithm exhibits routing overhead that falls between AODV and LEACH. This suggests that SAAQ may achieve a balance. SAAQ likely utilizes control messages efficiently while maintaining effective routing, potentially due to its optimized path selection strategies. This could contribute to lower routing overhead compared to LEACH while achieving comparable performance to AODV in congested scenarios.

Figure 16 depicts the average energy consumption experienced by nodes across the four routing protocols (AODV, AOMDV, LEACH, and SAAQ) as the number of nodes increases. Average energy consumption refers to the amount of energy a node expends on transmitting and receiving packets. The graph reveals a trend of increasing energy consumption for all protocols as network density grows. This aligns with the previously observed network congestion. With more nodes competing for limited resources, individual nodes need to transmit and receive a higher volume of packets. This increased activity translates to higher energy expenditure. Interestingly, AOMDV exhibits the highest average energy consumption among the compared protocols. This can be attributed to its proactive nature. AOMDV maintains routing tables for each node, enabling it to quickly discover routes even in congested scenarios. However, this proactive approach comes at a cost: nodes need to transmit and receive more routing control messages to update these tables, leading to increased energy consumption. The proposed SAAQ algorithm demonstrates average energy consumption that falls between AODV and AOMDV. This suggests that SAAQ may strike a balance. SAAQ likely optimizes control message usage while maintaining efficient routing, potentially due to its focus on dynamic path selection and leveraging feeder nodes for efficient communication. This could contribute to lower energy consumption compared to AOMDV while achieving comparable performance to other protocols in congested scenarios.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

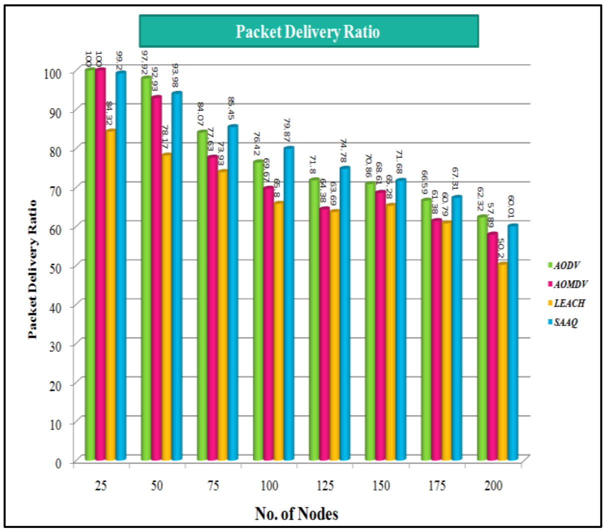

Figure 11: Packet Delivery Ratio

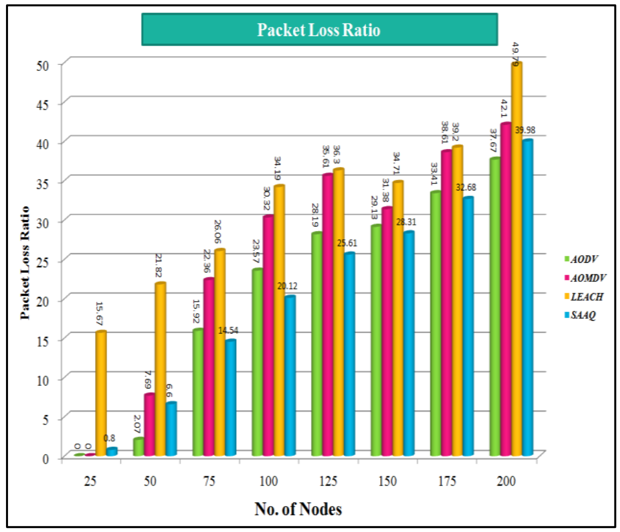

Figure 12: Packet Loss Ratio

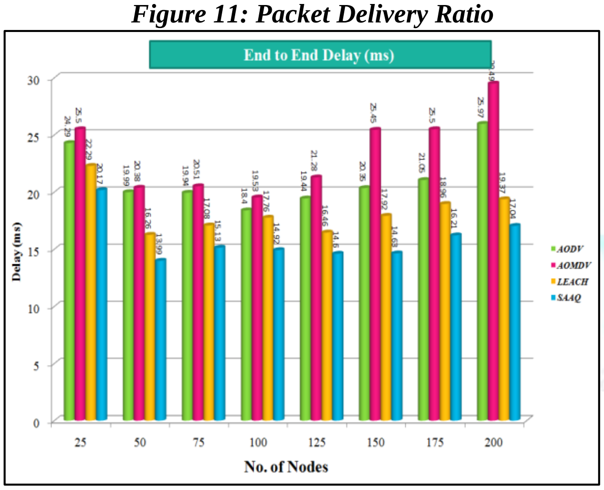

Figure 13: End to End Delay

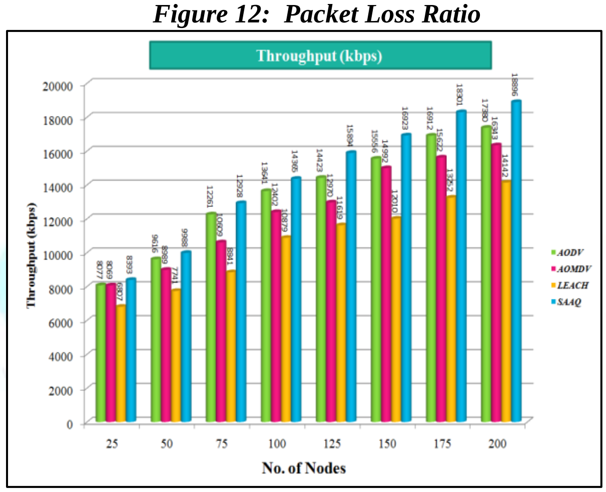

Figure 14: Throughput

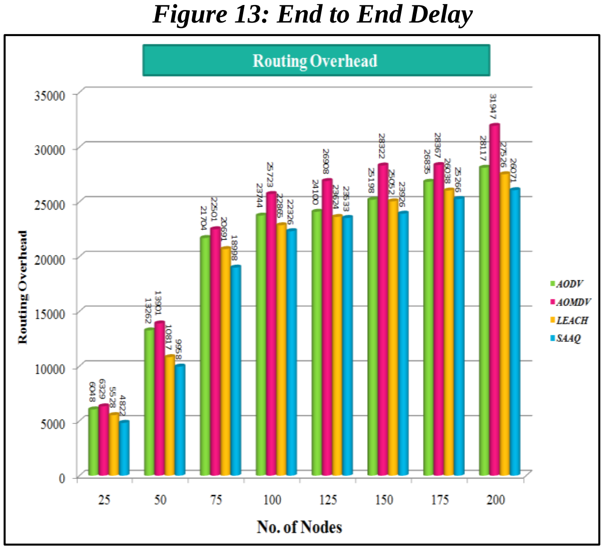

Figure 15: Routing Overhead

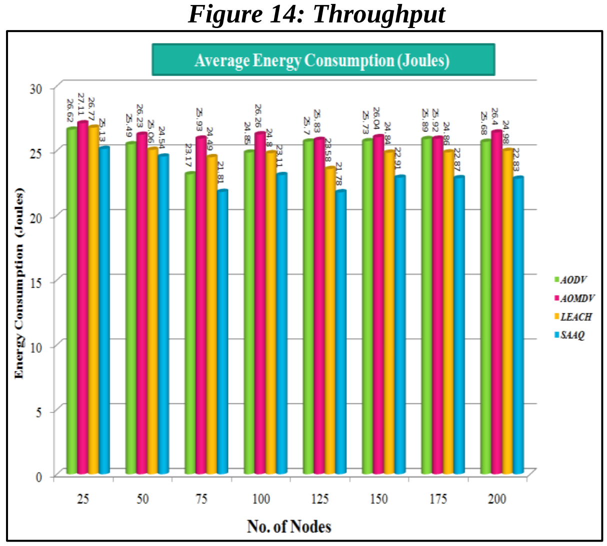

Figure 16: Average Energy Consumption

Here's a more elaborate explanation of the performance of each routing protocol, combining the strengths of the provided text and addressing potential shortcomings:

Routing Protocol Performance Breakdown:

- **AOMDV (Ad hoc On-Demand Multipath Distance Vector):**
  - **Strengths:** A proactive protocol known for its efficiency in packet delivery and minimizing delays. It establishes multiple paths for fault tolerance, improving network resilience.
  - **Weaknesses:** AOMDV suffers from high routing overhead due to its proactive maintenance of routing tables for every node. This overhead can become a significant burden in large-scale networks with many nodes.
- **AODV (Ad hoc On-Demand Distance Vector):**
  - **Strengths:** A reactive protocol that creates routes only when data needs transmission, reducing routing overhead compared to AOMDV.
  - **Weaknesses:** AODV's reactive nature can lead to higher packet loss ratios and delays, especially during initial route discovery phases or network congestion.
- **LEACH (Low-Energy Adaptive Clustering Hierarchy):**
  - **Strengths:** A hierarchical protocol designed for energy efficiency. It divides the network into clusters, with cluster heads handling internal routing, potentially conserving node energy.
  - **Weaknesses:** LEACH may experience lower packet delivery ratio and delays compared to other protocols. Cluster head failures or overloading can disrupt communication within clusters.
- **SAAQ (SAAQ-based QoS Routing Protocol):**
  - **Strengths:** A QoS-aware protocol that considers traffic requirements when routing packets. This flexibility allows SAAQ to potentially achieve good performance across all the measured QoS parameters (packet delivery ratio, delay, throughput, energy consumption).
  - **Weaknesses:** SAAQ might incur higher routing overhead compared to some other protocols due to its dynamic optimization strategies and potential use of control messages. However, the research suggests this overhead may be balanced by its efficiency in other areas.

The optimal routing protocol selection hinges on the specific application and its prioritized QoS requirements. Here's a simplified decision-making approach:

- **Prioritize all QoS parameters (delivery, delay, throughput, energy):** SAAQ emerges as a strong candidate due to its balanced performance across these metrics.
- **Prioritize low routing overhead:** AODV might be a suitable choice due to its reactive nature and potentially lower control message traffic.

- **Prioritize energy efficiency:** LEACH could be considered if extending network lifetime is the primary concern. However, be mindful of potential trade-offs in terms of delivery ratio and delay.

6. Conclusion and Future Directions of Research

In this research work, the SAAQ method has been designed to be lightweight and efficient. It is a good choice for use in IoT based wireless sensor networks where the nodes have limited resources. The SAAQ method has been evaluated in simulation and has been shown to be effective in providing reliable and efficient routing in IoT wireless networks. The SAAQ method is an improvement over AODV, LEACH, and AOMDV because it provides QoS guarantees, uses both proactive and reactive routing, and maintains multiple routes to each destination. This makes it a more robust and efficient routing protocol for wireless ad hoc networks.

A. Conclusion

This research introduces SAAQ, a routing protocol designed to overcome limitations in existing approaches like AODV, LEACH, and AOMDV. SAAQ's key strength lies in its adaptability:

- **Self-Adaptive Routing:** Unlike the static routing strategies of AODV, LEACH, and AOMDV, SAAQ dynamically adjusts its behavior based on real-time network conditions. This allows SAAQ to optimize performance and maintain efficient routing even as the network environment changes.

- **QoS-Aware Routing:** SAAQ prioritizes Quality of Service (QoS) by considering traffic requirements. This focus on QoS sets SAAQ apart from AODV, LEACH, and AOMDV, which offer limited control over factors like packet delivery ratio and delay. By understanding traffic needs, SAAQ can make routing decisions that ensure reliable data transmission for different application types.

- **Distributed Control:** SAAQ employs a distributed routing approach, eliminating the need for a central controller. This makes SAAQ well-suited for scenarios where a centralized infrastructure might not be available or desirable. In contrast, AODV, LEACH, and AOMDV often rely on centralized elements, which can introduce single points of failure and limit scalability.

The table you mentioned (Table 2) likely summarizes the key characteristics of these routing protocols in more detail. This comparison would further emphasize the advantages of SAAQ's dynamic, QoS-aware, and distributed approach to routing in resource-constrained networks.

Table 2: Comparison of routing methods based on features

Feature	SAAQ	AODV	LEACH	AOMDV
Approach	Self-Adaptive	Fixed	Fixed	Fixed
QoS Awareness	Yes	No	No	No
Distribution	Distributed	Centralized	Centralized	Centralized

The following table 3 summarizes the advantages of SAAQ over AODV, LEACH, and AOMDV:

Table 3: Advantage of SAAQ method over other methods

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

Feature	SAAQ	AODV	LEACH	AOMDV
QoS Guarantees	Yes	No	No	No
Proactive/Reactive	Both	Reactive	Hierarchical	Proactive
Multiple Routes	Yes	No	No	Yes
Energy Efficiency	Good	Good	Good	Good
Scalability	Good	Good	Good	Good

Overall, the SAAQ method is a promising routing protocol for wireless ad hoc networks. It provides QoS guarantees, uses both proactive and reactive routing, and maintains multiple routes to each destination. This makes it a more robust and efficient routing protocol than AODV, LEACH, and AOMDV. The following table 4 summarizes the percentage of improvement of SAAQ over LEACH based on QoS parameters considered in the research work.

Table 4: Percentage of Improvement of SAAQ over LEACH

QoS Parameter	LEACH (Avg. Aggregate)	SAAQ (Avg. Aggregate)	Percentage of Improvement
Packet Delivery Ratio	67.7768	78.9631	16.5 %
Packet Loss Ratio	32.2231	21.0369	34.71 %
End to End Delay	18.2687	15.8415	13.28 %
Routing Overhead	10661.38	14461.51	35.64 %
Energy Consumption	19824.6	19362.5	2.33 %

After analyzing the above table 4, it if found that SAAQ method is compared with LEACH method and improves the packet delivery ratio by 16.5%, packet loss ratio by 34.71%, end to end delay by 13.28%, throughput by 35.64%, routing overhead by 2.33% and average energy consumption by 7.23%.

B. Future Work

While this research has established SAAQ's potential through simulation, the next crucial step is to evaluate its performance in real-world IoT network deployments. This real-world validation can be achieved through the following avenues:

- **Hardware-based Testing:** Conducting experiments using actual IoT devices and applications will provide invaluable insights into SAAQ's behavior and effectiveness under practical network conditions. Factors like hardware limitations, real-world signal propagation, and diverse traffic patterns can all influence routing performance. Hardware testing will help identify potential areas for further optimization of SAAQ for practical deployments.

- **Scalability Assessment:** The simulations conducted in this research likely focused on controlled network sizes. Future work should explore how SAAQ scales in larger, more complex real-world IoT networks. This assessment will help determine SAAQ's suitability for various deployment scenarios with varying numbers of nodes and diverse traffic volumes.

- **Security Considerations:** Security is paramount in any network. Future research should delve into the security implications of using SAAQ in IoT deployments. This might involve analyzing SAAQ's susceptibility to routing attacks or exploring mechanisms to integrate security features within the routing protocol itself.

- **Cross-Layer Optimization:** Since network performance is often influenced by interactions between different layers, investigating potential cross-layer optimization strategies involving SAAQ and other network protocols (e.g., MAC layer protocols) could be a promising future direction. This collaborative approach could lead to further performance enhancements within the overall network architecture.

It's important to remember that these are just a few potential areas for exploration. The specific research directions pursued in the future will depend on the researcher's specific goals and expertise. For instance, a researcher particularly interested in security might prioritize investigating SAAQ's security posture, while another focused on large-scale deployments might prioritize scalability assessments. Regardless of the chosen path, continued research and development hold the key to making SAAQ an even more practical and valuable routing protocol for QoS-based IoT networks.

REFERENCES

[1] G. Mahajan, A., Mundur, P., & Joshi, A. (2002). Adaptive Multimedia System Architecture for Improving QoS in Wireless Networks. *IEEE Pacific Rim Conference on Multimedia*. https://doi.org/10.1007/3-540-36228-2_88.

[2] Wang, X.G., Min, G., Mellor, J.E., & Al-Begain, K. (2004). AN ADAPTIVE QOS MANAGEMENT SCHEME FOR INTERWORKING CELLULAR AND WLAN NETWORKS. https://api.semanticscholar.org/CorpusID:15197593

[3] Gatouillat, A., Badr, Y., & Massot, B. (2017). QoS-Driven Self-adaptation for Critical IoT-Based Systems. *ICSOC Workshops*. https://doi.org/10.1007/978-3-319-91764-1_8

[4] Wang, X.G., Min, G., Mellor, J.E., Al-Begain, K., & Guan, L. (2005). An adaptive QoS framework for integrated cellular and WLAN networks.C*omput. Networks*, *47*, 167-183. https://doi.org/10.1016/J.COMNET.2004.07.003.

[5] Federico, Di, Menna., Henry, Muccini., Karthik, Vaidhyanathan. (2022). FEAST: a framework for evaluating implementation architectures of self-adaptive IoT systems. Doi: 10.1145/3477314.3507146.

[6] R., Prabha., Senthil, G., A., N., Naga, Saranya., A., M., K., Somasundaram., K., C. (2022). Design of Adaptive Priority Based IoT Communication in WirelessNetwork. 1-5.doi: 10.1109/ICACTA54488.2022.9753550

[7] Avewe, Bassene., Bamba, Gueye. (2022). A Self-adaptive QoS-management Framework for Highly Dynamic IoT Networks. 1-8. doi: 10.1109/MNE3SD53781.2022.9723303

[8] Federico, Di, Menna., Henry, Muccini., Karthik, Vaidhyanathan. (2022). FEAST: a framework for evaluating implementation architectures of self-adaptive IoT systems. Doi: 10.1145/3477314.3507146.

[9] R., Prabha., Senthil, G., A., N., Naga, Saranya., A., M., K., Somasundaram., K., C. (2022). Design of Adaptive Priority Based IoT Communication in Wireless Network. doi: 10.1109/ICACTA54488.2022.9753550.

[10] Avewe, Bassene., Bamba, Gueye. (2022). A Self-adaptive QoS-management Framework for Highly Dynamic IoT Networks. doi: 10.1109/MNE3SD53781.2022.9723303.

[11] Satyanarayana, Pamarthi., N., R.. (2022). Implementation of Modified SFO Algorithm to Enhance QoS in Wireless Mobile Adhoc Networks for IoT Applications. doi: 10.1109/wispnet54241.2022.9767175.

[12] Aurélien, Chambon., Abderrezak, Rachedi., Abderrahim, Sahli., Ahmed, Mebarki. (2022). ConSerN: QoS-Aware programmable multitier architecture for dynamic IoT networks. doi: 10.1109/GLOBECOM48099.2022.10001551.

[13] Ahmad, Khalil., Nader, Mbarek., Olivier, Togni. (2021). A Self-Optimizing QoS-Based Access for IoT Environments. Wireless Personal Communications, doi: 10.1007/S11277-021-08589-8.

[14] Hongyu, Zhou., Xiaomin, Ren. (2023). Design and Implementation of Wireless Optical Access System for VLC-IoT Networks. Journal of Lightwave Technology, doi: 10.1109/JLT.2023.3234990.

[15] Cherifa, Boucetta., Boubakr, Nour., Alberic, Cusin., Hassine, Moungla. (2021). QoS in IoT Networks based on Link Quality Prediction. doi: 10.1109/ICC42927.2021.9500396.

ISSN: 2584-1491 |www.iircj.org
Volume- 2 Issue-1, Jan-2024, Page 5-22

[16] Nogaye, Lo., Ibrahima, Niang. (2021). A Comparison of QoS-Based Architecture Solutions for IoT/Edge Computing Environment. Doi: 10.1007/978-3-030-53440-0_37.

[17] Jiaxin, Liang., He, Chen., Soung, Chang, Liew. (2021). Design and Implementation of Time-Sensitive Wireless IoT Networks on Software-Defined Radio. IEEE Internet of Things Journal, doi: 10.1109/JIOT.2021.3094667.

[18] Mansoor, C., Vishnupriya, G., Anand, A., Vijayakumar, S., Kumaran, G., & Samuthira Pandi, V. (2023). A Novel Framework on QoS in IoT Applications for Improvising Adaptability and Distributiveness.2*023 International Conference on Computer Communication and Informatics (ICCCI),* 1-6. https://doi.org/10.1109/ICCCI56745.2023.10128257.

[19] Zia, K., Chiumento, A., & Havinga, P.J. (2022). AI-Enabled Reliable QoS in Multi-RAT Wireless IoT Networks: Prospects, Challenges, and Future Directions.I*EEE Open Journal of the Communications Society, 3,* 1906-1929. https://doi.org/10.1109/OJCOMS.2022.3215731

[20] Gatouillat, A., Badr, Y., & Massot, B. (2017). QoS-Driven Self-adaptation for Critical IoT-Based Systems.I*CSOC Workshops.* https://doi.org/10.1007/978-3-319-91764-1_8.

[21] Duan, R., Chen, X., & Xing, T. (2011). A QoS Architecture for IOT.2*011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, 717-720. https://doi.org/10.1109/iThings/CPSCom.2011.125.

[22] Chu, Y., & Ganz, A. (2006). Adaptive MAC Protocol for QoS support in UWB-based Wireless Networks.2*006 IEEE International Conference on Communications, 9,* 4040-4045. https://doi.org/10.1109/ICC.2006.255713.