Received: 26 June 2025 | Revised: 16 August 2025 | Accepted: 17 August 2025 | Vol. 1 | Issue 2| December 2025
e-ISSN: 3105-2169

International Journal of Mechatronics, Robotics, and Artificial Intelligence
Original Article

Empirical Evaluation of MQTT, CoAP and HTTP for Smart City IoT Applications

Rafid M. Khaleefah*[1], Ali A. Abed[2], Mahmood A. Al-Shareeda[3,4]
[1] Information and Communication Technology Centre, University of Basrah, Basrah, Iraq.
[2]Mechatronics Engineering Department, University of Basrah, Basrah, Iraq.
[3]Department of Electronic Technologies, Basra Technical Institute, Southern Technical University, 61001, Basra, Iraq.
[4]College of Engineering, Al-Ayen University, Thi-Qar, 64001, Iraq

Correspondence
*Rafid Mohammed Khaleefah
Information and Communication Technology Centre, University of Basrah, Basrah, Iraq
Email: rafeed.rmk@uobasrah.edu.iq

Abstract
Smart city applications demand lightweight, efficient and dependable communication protocols to facilitate the functioning of resource-limited Internet of Things (IoT) devices. This work performs an extensive empirical study of the three most prominent IoT standards; Message Queuing Telemetry Transport (MQTT), the Constrained Application Protocol (CoAP) and Hypertext Transfer Protocol (HTTP) by emulating real-world smart city use cases using a Raspberry Pi based testbed. The primary metrics based on which the protocols are analyzed are latency, message overhead, delivery rate and energy consumption. ANOVA and Tukey's HSD tests are used to determine the statistical significance of experimental data. The test results indicate that CoAP under (QoS-1 reliability) shows the least latency and energy consumption and MQTT due to its support for Quality of Service (QoS) is the most reliable. Among the others, HTTP is in general performance terms certainly at the bottom of all metrics mainly for its verbosity and synchronous nature. The paper then also suggests a decision flowchart for developers to choose the suitable protocol according to application requirements. The results are more than just numbers on a graph, and the research can be deployed for advice for protocol selection in practice, where this study helps identify issues with encryption overhead (over 75\%) while showcasing multi-hop network scalability and adaptive switch mechanisms as areas that remain to be resolved. Such findings can be used as a basis for design approaches to construct secure, efficient and scalable communication protocols in urban IoT settings.

Keywords
Internet of Things (IoT), Smart Cities, MQTT, CoAP, HTTP, Protocol Comparison, Energy Efficiency, Latency, Quality of Service (QoS), Secure Communication.

I. INTRODUCTION

The Internet of Things (IoT) has revolutionized urban infrastructure, enabling smart cities to deliver intelligent services such as adaptive traffic control, real-time environmental monitoring, automated waste management, and energy-efficient utility distribution [1]-[3]**.** These advancements rely on massive networks of distributed sensors and actuators, often deployed in bandwidth-limited and power-constrained environments [4]. Ensuring efficient communication in such scenarios requires protocols that minimize latency, reduce energy consumption, mitigate cyberattacks, and support scalable architectures [5]-[7].

Traditional web protocols, particularly Hypertext Transfer Protocol (HTTP), are widely adopted due to their interoperability and ease of implementation [8]-[10]. However, their verbose structure and reliance on persistent Transmission Control Protocol (TCP) sessions introduce excessive overhead, rendering them suboptimal for constrained IoT deployments [11]. In response, lightweight protocols like Message Queuing Telemetry Transport (MQTT) and the Constrained Application Protocol (CoAP) have gained prominence. MQTT uses a publish/subscribe model with configurable Quality of Service (QoS) levels, while CoAP, designed for Representational State Transfer (RESTful) interactions over User Diagram Protocol (UDP), offers reduced message overhead and supports asynchronous communication [12]-[14].

Present studies like Gemirter et al. [15] and Koutras et al. and although others like [16] have considered comparing protocols for IoT applications against one another, they tend to stop short of gathering and evaluating real-world testbed validation results or energy profiling overall as well as any statistical rigor. In comparison, this work also presents an end-to-end statistical comparison of MQTT and CoAP with the HTTP using a physical Raspberry Pi based testbed. It also presents a practical protocol selection framework for smart city applications, which to some extent also fulfills the literature gap.

This study addresses that gap by performing a detailed performance analysis of MQTT, CoAP, and HTTP using a controlled Raspberry Pi-based testbed. Key metrics such as latency, data overhead, delivery rate, and energy consumption are evaluated, and results are statistically validated to ensure robustness. The aim is to provide practical guidance for IoT protocol selection in smart city systems based on empirical evidence. To address these gaps, this paper presents the following key contributions:

- **Empirical Evaluation on a Physical Testbed**: A real-world performance comparison of MQTT, CoAP, and HTTP protocols is conducted using a Raspberry Pi-based testbed that emulates smart city conditions.
- **Comprehensive Metric and Statistical Analysis:** Core metrics—latency, message overhead, delivery success rate, and energy consumption—are quantitatively assessed. Statistical significance is validated through one-way ANOVA and Tukey's HSD tests, with visualizations (e.g., time-series plots, histograms) illustrating protocol stability.
- **Protocol Selection Framework and Future Outlook:** A decision flowchart is proposed to guide protocol selection based on application constraints. The paper also outlines future directions, including the integration of secure communication overhead and adaptive, ML-driven protocol switching.

The rest of the paper is organized as follows: Section II reviews related work on IoT protocol evaluation. Section III details the experimental setup and methodology. Section IV presents and analyzes the results. Section V discusses research gap and future directions. Section VI concludes the paper and outlines directions for future research.

II. RELATED WORK

Previous works on IoT communication protocols have identified a suite of trade-offs in latency, reliability and energy efficiency. However, most work focuses on particular protocol behaviors or does not define standardized conditions for a fair comparison. D'Ortona et al. [17] introduce a fully integrated IoT system that includes: BLE, ESP32, MQTT and open source tools for the monitoring of Vulnerable Road Users. It summarizes very well the architecture, technologies used and also gives a feasibility analysis. While it is one of the best options for smart mobility, a clearer structure and more original form compared to many existing solutions can be beneficial. Hu et al. [18] proposes a certificate-less secure MQTT transmission method based on group encryption optimized for IoT. It is a delightful mix of cryptographic efficiency and light-weight design. We liked how the performance gains were quantified, but felt that the abstract could be slightly more structured; and some form of demonstration of working with IoT existing ecosystem could have been handy. Kaganurmath et al. [19] present DLKS-MQTT, a lightweight and safe technique for sharing a key in MQTT-based communications in Internet of Things (IoT). It specifies the technical advances, key performance indicators and safety advantages in a clear manner. It provides a structured and detailed summary, however needs to be equipped with few sentences hinting the feasibility of implementation or potential real-world use which would make it more pragmatic for energy-constrained IoT. Khalil et al. [20] gives the abstract of a Cooja Hardware Accelerator for CoAP in IoT Using FPGA. It does a great job of pointing out the additional performance power, latency and energy that all be gained. The abstract focuses on virtual routing channels and dynamic resources allocation, but it could more clearly mention some practical deployment challenges or scaling across various IoT solutions. Oliver et al. [21] presents a lightweight mutual authentication scheme for IoT based on CoAP in order to increase the efficiency and security of this environment. It addresses major pros and cons as well as resistance to attacks but redundancy and flaws in grammar. By structuring it properly, tightening up the language and emphasising technical novelty this paper will be more readable and better convey its contribution to IoT security research. Yang et al. [22] explain a containerized smart street lighting system that is scalable, secure and which allows for flexible data handling in this paper. The combination of the forefront of edge computing, NoSQL and encrypted communication does this well. Some cleanup on the language could be done, as well as more explicit technical novelty. Faleh et al. [23] introduces SECRE-MEN as an efficient and post-quantum-ready mutual authentication architecture that could be implemented within 6G IoT and MEC systems. It clearly lays out a description of its modular design and security. On the other hand, clarity suffers from overbearing phrasing and acronym-heavy language. Alyami et al. [24] present a secure and robust post-quantum encryption scheme for 6G-based smart city systems, based on DGFEA with the help of hybrid bio-inspired optimization. It effectively covers methods and metrics. But the more abstract is very technical, and makes awkward wording and does not highlight what 's new about the modeless of this current work for subsequent reader comprehension.

Although previous works provide valuable contributions in terms of the security and performance of protocols, they often do not hold evaluations against a common testbed with empirical results. In this paper, we address these gaps by presenting a comprehensive statistically supported evaluation of MQTT, CoAP and HTTP under real-world IoT settings and delivering a practical edge for selecting the messaging protocol for smart city deployment.

III. METHODOLOGY

To ensure a rigorous and reproducible evaluation of IoT communication protocols, a physical testbed was developed using Raspberry Pi 4 devices to simulate smart city nodes. The methodology is organized into three key components: testbed setup, data collection procedures, and performance metric definitions, as shown in Fig. 1.

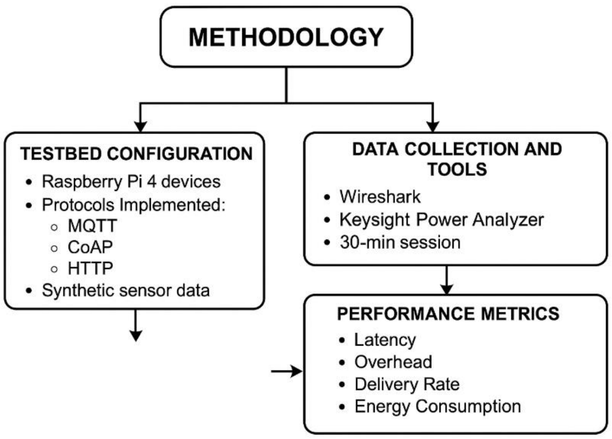

Fig 1. Methodology process.

A. Testbed Configuration

To quantify the performance of these three protocols under realistic IoT conditions, we set up a physical testbed using Raspberry Pi 4 devices. In this implementation, each device acts as a sensor node in a simulated smart city scenario. The testbed configuration includes the following items.

- **Hardware:** Raspberry Pi 4 Model B units (4GB RAM) running Raspberry Pi OS Lite. Studies presenting applications or evaluations of LLMs in cybersecurity [5].
- **Network Environment:** All devices communicated over an isolated 802.11n Wi-Fi network, established using a TP-Link Archer C7 router to ensure stable and interference-free conditions
- **Protocol Implementations:**
  - **MQTT:** Configured using Eclipse Mosquito broker (version 2.0.15).
  - **CoAP:** Implemented using the libcoap library (version 4.3.2).
  - **HTTP:** Simulated using a Flask web server (version 2.3.3).
- **Data Generation:** Synthetic environmental data (temperature and humidity) was generated with a payload size of approximately 50 bytes and transmitted at 5-second intervals.
- **Deployment:** All components are deployed locally to ensure controlled, reproducible test conditions.

This setup allows for an apples-to-apples comparison of the protocols, since every single run is guaranteed to use the same hardware, payload and transmission interval. Specifically, this setup cannot scale to multi-hop network topologies, but it sufficiently isolates protocol behavior for initial baseline measurement. Future work will focus on more complex topologies (such as multi-hop or mobility.

B. Data Collection and Tools

The experiments with each of the protocols were performed every 30 minutes throughout the data collection period. These sessions used a battery of tools to study network activity, power consumption and communication dependability.

- **Wireshark:** Used to capture and analyze network traffic, enabling precise measurement of latency, packet overhead, and delivery success rates [25][26].
- **Keysight N6705B DC Power Analyzer:** Deployed to measure instantaneous and average power consumption during message transmission and reception cycles [27].
- **Measurement Conditions:**
  - Operating voltage was fixed at 5V across all Raspberry Pi devices.
  - Current was monitored continuously, and average values were computed for each protocol session.
  - Each node transmitted synthetic data at 5-second intervals.
- **Monitoring Duration:** For each protocol, data was collected over 30 minutes to ensure adequate sampling for statistical analysis.

As a result, all protocol evaluations will be constructed similarly to provide insights into resource usage and communication performance in comparable test settings. This data was then used to compute latency, energy usage, delivery rate and protocol overhead further in the analysis.

C. Performance Metrics

MQTT, CoAP, and HTTP, the three protocols are evaluated based on four performance metrics. The metrics selected for evaluation were chosen to explain communication efficiency and energy consumption aspects crucial in IoT applications especially deployed in smart city environment:

- **Latency:** Defined as the time delay between the transmission of a message by a sender node and the reception of the acknowledgment by the same node. It includes network propagation time and server processing delay.
- **Message Overhead:** Total size of each transmitted message, including application payload and protocol headers. This reflects protocol efficiency in bandwidth usage.
- **Delivery Rate (%):** The reliability of message transmission, calculated using the formula:

$$Delivery\ Rate\ = \left(\frac{Messages\ Received}{Messages\ Sent}\right) \times 100$$

- **Energy Consumption (mWh):** Estimated based on power usage during the transmission and reception of each message using the formula:

$$E = V \times I \times t$$

Where E is the energy consumption in (millwatt-hours) (mWh), V is the operating voltage (5V), I is the average current (in millamperes), t is the duration of message activity (in hours).

These performance metrics offer a complete picture of how well each protocol performs, and they make any speed-versus-efficiency-and-reliability balancing problems very clear. While this is beyond the scope of this study on unsecured protocol transmissions, future work will extend these metrics to assess encrypted communication scenarios as well.

IV. RESULTS AND ANALYSIS

This section presents the experimental results obtained for MQTT, CoAP, and HTTP protocols under standardized conditions. The performance of each protocol is evaluated based on the metrics defined in the previous section. A summary of the results is shown in Table I., followed by graphical comparisons and statistical validation.
Table I. summarizes the average values of latency, message overhead, delivery rate, and energy consumption recorded during the 30-minute experimental period for each protocol.

TABLE I.
COMPARATIVE PERFORMANCE METRICS OF MQTT, COAP, AND HTTP

Protocol	Latency (ms)	Overhead (bytes)	Delivery Rate (%)	Energy (mWh)
MQTT	22	168	99.7	0.49
CoAP	18	140	99.3	0.42
HTTP	87	610	96.1	1.12

CoAP systematically had the best latency/energy performance, while MQTT provided the most reliable delivery of messages. This was the worst performing protocol across all of the metrics, due to the verbose message format and requirement for persistent TCP sessions.

Additionally, latency trends over the 30-minute test period are shown in Fig 3. CoAP and MQTT maintained stable latencies, while HTTP showed higher variability and spikes under identical conditions.

To validate the importance of any performance variations observed. we performed a one-way Analysis of Variance (ANOVA) test for each metric. All the resulting p-values were less than 0.05 corresponding to statistically significant differences in protocols. Post Hoc Analysis Tukey, Honest Significant Difference (HSD) came out with the:

- CoAP and MQTT significantly outperform HTTP in latency and energy consumption.
- MQTT has a significantly higher delivery rate than HTTP, though the difference with CoAP was marginal.

A. *Latency Distribution*

Fig. 4. shows histograms of latency distribution for each protocol. CoAP exhibits the tightest distribution around the mean, suggesting consistent message timing. HTTP displays a broader distribution, indicating instability under constrained conditions.

B. *Discussion*

The results of the experiments are clear justifying that efficient communication, and optimized usage of resources in IoT-based smart city systems is dependent on protocols that could be selected. CoAP is suitable for environment monitoring, traffic control and other real-time applications because it has the lowest latency (18ms) and energy consumption (0.42mWh). The fact it operates over UDP and also has a lightweight header structure, makes it fit in tiny devices (and networks) nicely but means there may need to be some application-specific mechanism built on top of CoAP for reliability, as no transport guarantees are provided natively.

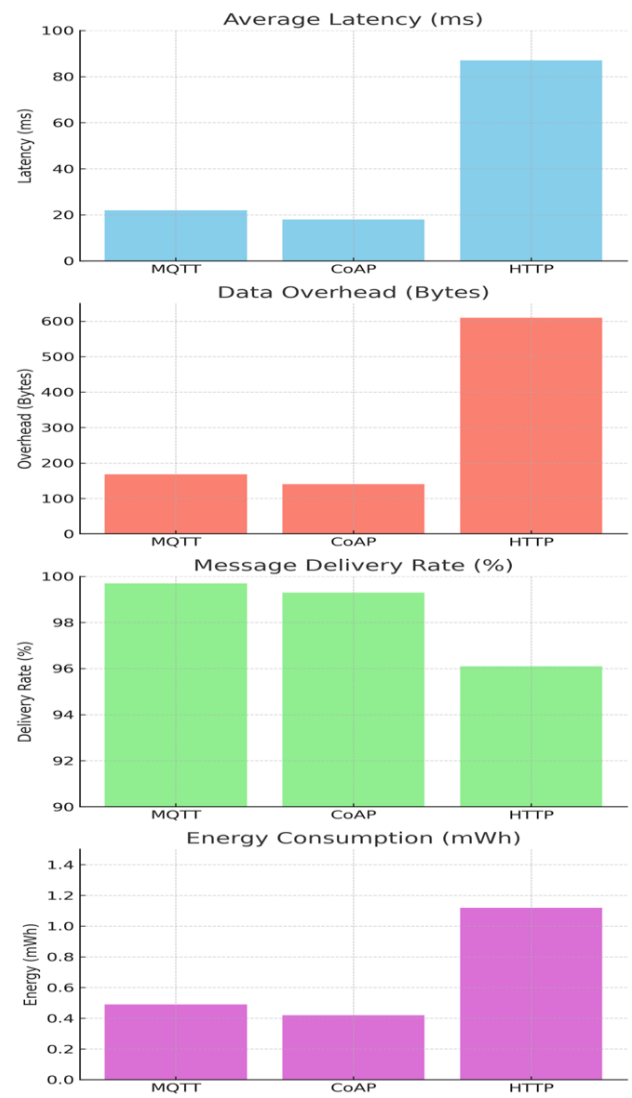

Fig.2. Protocol comparison: latency, overhead, delivery rate, and energy consumption.

One pretty good in-between option is using MQTT, which strikes a nice balance between efficiency and reliability. This delivery rate of 99.7\% combined with the ability to configure message retransmission and low, medium and high Quality of Service (QoS) levels makes it an attractive alternative for applications that require reliable message supply (e.g., utility metering or smart grid systems). While it does have a slightly higher overhead (168 bytes) and energy use (0.49mWh), its features of publish-subscribe support and fault tolerance capabilities justify this overhead.

In Particularly constrained conditions, the performance in this regard of (HTTP) is lower: 87ms latency, 610 bytes overhead and consumes the highest energy (1.12 mWh). This is due to its longwinded headers, the TCP request/response handshake in conjunction with synchronous requests.

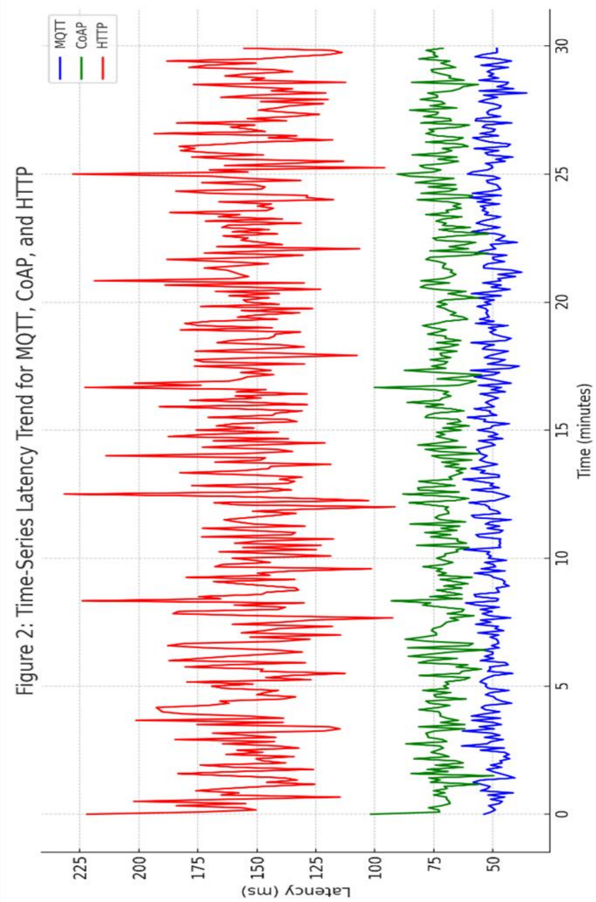

Fig. 3. Time-series latency trend for MQTT, CoAP, and HTTP.

However, there are cases where integration with traditional web services remains a necessity and when resource constraints are not very tight so HTTP still finds its role. One should also take into account that this evaluation did not consider the secure transmission mechanisms such as TLS/DTLS, which may affect latency and energy more alike. Although the results offer a good groundwork, the authors have to consider using encrypted communication and multi-hop topologies in their future work to mimic realistic smart city settings

In the end, CoAP outperforms MQTT in pure raw efficiency while MQTT comes on top of CoAP in term of reliability but the right protocol is obviously very dependent on your specific application needs with a trade-off between energy savings, latency and reliability and interoperability. This study incorporates a protocol selection map (Fig. 5) based on performance trade-offs and environmental constraints to facilitate such choices.

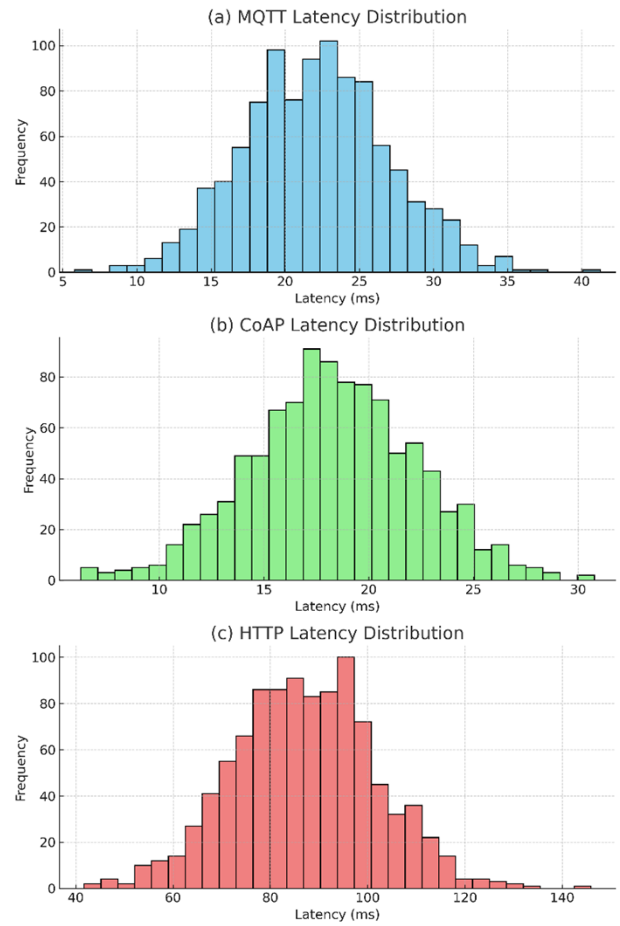

Fig. 4. Latency distribution histograms for (a) MQTT, (b) CoAP, and (c) HTTP

V. RESEARCH GAPS AND FUTURE DIRECTIONS

Even with the research trending on IoT communication protocols and smart city integration, there are still lots of issues to address. If solved, improving these weak links could be key to boosting IoT efficiency and robustness. Based on the review, the following research gaps are proposed to inform future investigations:

1. **Secure Protocol Performance Analysis**: Most existing comparisons of MQTT, CoAP, and HTTP ignore the overhead introduced by secure communication layers (e.g., TLS/DTLS). Future studies should include security-layer impact on latency, energy consumption, and reliability.
2. **Dynamic Protocol Adaptation Mechanisms:** There is limited implementation of intelligent systems capable of dynamically switching between protocols based on context, traffic load, or energy availability. Machine learning-based protocol adaptation remains underexplored.
3. **Multi-hop and Mobile Topologies:** Current testbeds often focus on static, single-hop

topologies. Real-world deployments involve multi-hop, mobile, and interference-prone environments, which require more realistic experimentation.

4. **Protocol Interoperability Frameworks:** Interoperability between different IoT protocols in hybrid networks is still a significant challenge. A standardized framework to manage translation or bridging between protocols is lacking.
5. **Energy Modeling with Real Workloads:** Existing energy evaluations rely on synthetic data. Real-world sensor data streams, which are often bursty and unpredictable, should be used to model energy behavior more accurately.
6. **Scalability under Urban-Scale Deployment:** There is little empirical validation of protocol performance under large-scale, city-wide deployments involving thousands of nodes and diverse data sources.
7. **Edge-Integrated Protocol Decision Engines**: The potential of edge computing to make real-time protocol selection decisions based on latency, load, and network congestion is largely untapped. Integrating lightweight decision engines at the edge can significantly improve system responsiveness and efficiency.

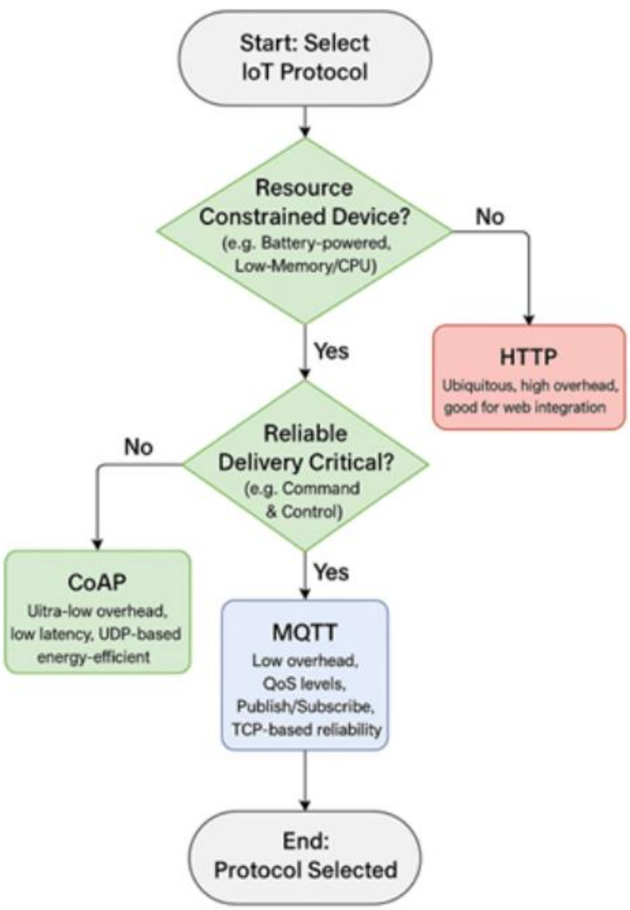

Fig. 5. Flowchart for selecting IoT communication protocol based on device constraints and reliability needs

These gaps point to the importance of heterogeneous cross-discipline solutions in networking, cryptology, and machine learning plus the integration between edge and cloud that are needed for next generation smart city IoT infrastructures.
testbed. It also presents a practical protocol selection framework for smart city applications, which to some extent also fulfills the literature gap.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have performed an extensive empirical analysis to evaluate the performance of three major IoT communication protocols namely MQTT, CoAP and HTTP in Raspberry Pi based testbed under smart city scenario. We have provided qualitative and quantitative insights into both local and network aspects of Paxos to establish a baseline by measuring variance between them with respect to certain key performance metrics such as end-to-end latency, network message overhead (messages/runtime), message delivery rate etc., alongside energy consumed per operation. It reveals that CoAP consumes minimum energy (tolerance)with bringing to bear too much of a burden on latency performance, and also CoAP is more suitable for highly constrained environments.

MQTT although more expensive, has the advantage of guaranteeing the delivery of data and more options for message types. Moreover, while HTTP is used across a large number of devices and implementations, its high overhead and energy requirements led to poor performance on constrained IoT scenarios.

We proposed a protocol selection framework to help developers choose appropriate protocols based on application-specific requirements, such as power availability and reliability constraints to facilitate real-world deployment. Statistical validation also supported the importance of performance differences between the protocols.

In the future work, we will extend this study in four areas. To begin with, we intend to integrate a secure transmission layers (TLS/DTLS), in order to measure how well these modifications might perform with encrypted protocols. Secondly, the testbed will be extended to more realistic deployment scenarios combining multi-hop and mobile topologies. Third, we work on adaptive protocol switching scheme using small machine learning models at the network edge. Finally, a reusable and extensible simulation toolkit is going to be constructed under the project to facilitate wider and reproducible experiments.

Conflict of Interest

One of the authors, Prof. Dr. Ali Ahmed Abed, is the Editor-in-Chief of the International Journal of Mechatronics, Robotics, and Artificial Intelligence (IJMRAI). To ensure transparency and avoid any potential conflict of interest, he was not involved in the editorial processing or peer review of this manuscript. The review process was handled independently by other qualified editors.

REFERENCE

[1] R. Mishra and A. Mishra, “Current research on internet of things (IoT) security protocols: A survey,” *Computers & Security*, p. 104310, 2025, https://doi.org/10.1016/j.cose.2024.104310.

[2] K. Ramani, G.H. Bindu, M. Haritha, N.P. Latha, and I. Suneetha, "IoT protocols", *Innovations in computational Intelligence, Big Data Analytics and Internet of Things*, p. 179, 2024.

[3] B. Mohammed, M. Al-Shareeda, R. Homod, Y. Alkhabra, Z. Al-Mekhlafi, G. Alshammari, A. Alanazi, “Taxonomy-based lightweight cryptographic frameworks for secure industrial IoT: A survey,” *IEEE Internet of Things Journal*, 2025, https://doi.org/10.1109/JIOT.2025.3595649

[4] A. Daniel, N. Krishnaraj, S. Venkatraman, and P. Mah-eswaravenkatesh, “Post-quantum lightweight cryptography algorithms and approaches for IoT and blockchain security,” in *Advances in Computers*, vol. 138, pp. 349–376, Elsevier, 2025, https://doi.org/10.1016/bs.adcom.2025.03.001

[5] Z. Aziz and A. Abed, Harnessing Large Language Models for Enhanced Cybersecurity: A Review of Their Role in Defending Against APT and Cyber Attacks, in International Journal of Mechatronics, Robotics, and Artificial Intelligence (IJMRAI), vol. 1, issue 1, pp. 54-62, June 2025, doi:10.33971/ijmrai.1.1.8

[6] G. Said, A. Ghani, A. Ullah, M. Azeem, M. Bilal, and K. S. Kwak, “Light-weight secure aggregated data sharing in IoT-enabled wireless sensor networks,” *IEEE Access*, vol. 10, pp. 33571–33585, 2022, https://doi.org/10.1109/ACCESS.2022.3160231

[7] M. Rana, Q. Mamun, and R. Islam, “Lightweight cryptography in IoT networks: A survey,” *Future Generation Computer Systems*, vol. 129, pp. 77–89, 2022, https://doi.org/10.1016/j.future.2021.11.011

[8] R. Shah, and S. Correia, “Encryption of data over http (hypertext transfer protocol)/https (hypertext transfer protocol secure) requests for secure data transfers over the internet,” in *2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, pp. 587–590, IEEE, 2021, https://doi.org/10.1109/RTEICT52294.2021.9573978

[9] A.A. Abed, “Internet of Things (IoT): Architecture and design”, *2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)*, IEEE, pp. 1–3, 2016.

[10] K. Tran, A. Phram, N. Nguyen, and P. Dang, "Analysis and performance comparison of IoT message transfer protocol applying in real photovoltaic system, *International Journal of Networked and Distributed Computing*, vol. 12, no. 1, pp. 131–143, 2024, https://doi.org/10.1007/s44227-024-00021-4

[11] M. Talau, T. Herek, M. Fonseca, and E. Wille, “Improving TCP performance over a common IoT scenario using the early window tailoring method,” *Computer Networks*, vol. 234, p. 109875, 2023, https://doi.org/10.1016/j.comnet.2023.109875

[12] P. Sun, S. Shen, Y. Wan, Z. Wu, Z. Fang, and X.-z. Gao, “A survey of IoT privacy security: Architecture, technology, challenges, and trends,” *IEEE internet of things journal*, vol. 11, no. 21, pp. 34567–34591, 2024, https://doi.org/10.1109/JIOT.2024.3372518

[13] H. Jaafer, and A. Abed, “Towards Smart Manufacturing: Implementing PI Control on PLCs in IIoT-Driven Industrial Automation”, in *International Journal of Mechatronics, Robotics, and Artificial Intelligence (IJMRAI)*, vol. 1, issue 1, pp. 19-29, June 2025, https://doi.org/10.33971/ijmrai.1.1.4

[14] V. Quincozes, S. Quincozes, J. Kazienko, S. Gama, O. Cheikhrouhou, and A. Koubaa, “A survey on IoT application layer protocols, security challenges, and the role of explainable AI in IoT (xaiot),” *International Journal of Information Security*, vol. 23, no. 3, pp. 1975–2002, 2024, https://doi.org/10.1007/s10207-024-00828-w

[15] C. Gemirter, S. enturca, and S. Baydere, “A comparative evaluation of amqp, mqtt and http protocols using real-time public smart city data,” in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, pp. 542–547, IEEE, 2021, https://doi.org/10.1109/UBMK52708.2021.9559032

[16] D. Koutras, G . Dimitrakopoulos, V. Malamas, P. Kotzanikolaou, and C. Douligeris, “Comparative analysis and implementation of http3, mqtt, and coap for IoT applications,” in *Proceedings of the 28th Pan-Hellenic Conference on Progress in Computing and Informatics*, pp. 127–132, 2024, https://doi.org/10.1145/3716554.3716830

[17] C. D’Ortona, D. Tarchi, and C. Raffaelli, “Open-source mqtt-based end-to-end IoT system for smart city scenarios,” *Future Internet*, vol. 14, no. 2, p. 57, 2022, https://doi.org/10.3390/fi14020057

[18] X. Hu, J. Li, Y. He, and Y. Wang, “Research and implementation of lightweight security solutions for IoT mqtt protocol,” in *2025 IEEE 7th International Conference on Communications, Information System and Computer Engineering (CISCE)*, pp. 1166–1171, IEEE, 2025, https://doi.org/10.1109/CISCE65916.2025.11065441

[19] S. Kaganurmath, N. Cholli, and M. Anala, “Dlks- mqtt: A lightweight key sharing protocol for secure IoT communications,” *Engineering, Technology & Applied Science Research*, vol. 15, no. 2, pp. 21532–21538, 2025, https://doi.org/10.48084/etasr.10216

[20] K. Khalil, A. Kumar, and M. Bayoumi, “Hardware acceleration of coap protocol for high-speed and low-power internet of things communication,” *IEEE Internet of Things Journal*, 2024, https://doi.org/10.1109/JIOT.2024.3502549

[21] S. Oliver and T. Purusothaman, “Lightweight and secure mutual authentication scheme for IoT devices using coap protocol.,” *Computer Systems Science & Engineering*, vol. 41, no. 2, 2022,

https://doi.org/10.32604/csse.2022.020888

[22] Y. Yang, S. Lee, G. Chen, C. Yang, Y. Huang, and T. Hou, "An implementation of high efficient smart street light management system for smart city," *IEEE access*, vol. 8, pp. 38568–38585, 2020, https://doi.org/10.1109/ACCESS.2020.2975708

[23] M. Faleh, A. Abdulsada, A. Alaidany, M. Al-Shareeda, M. Almaiah, and R. Shehab, "Secremen: A lightweight quantum-resilient authentication framework for IoT-edge networks," *Journal of Robotics and Control (JRC)*, vol. 6, no. 4, pp. 1681–1692, 2025.

[24] R. Alyami, "Secure IoT transmission in 6g smart cities: a quantum-resilient hybrid Galois field and reed-Solomon approach," *The Journal of Supercomputing*, vol. 81, no. 4, pp. 1–37, 2025, https://doi.org/10.1007/s11227-025-07084-2

[25] C. Sanders, "Practical packet analysis: Using Wireshark to solve real-world network problems", no starch press, 2017.

[26] V. Ndatinya, Z. Xiao, V. Manepalli, K. Meng, and Y. Xiao, "Network forensics analysis using Wireshark", *International Journal of Security and Network, vol.10, no.2, pp. 91-106, 2015, https://doi.org/10.1504/IJSN.2015.070421*