Al-Farouk journal of sciences
Volume (2), Issue (3) ( JUNE, 2026)
ISSN: 3135-2359

مجلة الفاروق للعلوم
المجلد (2) العدد 3 يونيو 2026
ردمد: 3135-2359

# A Comprehensive Performance Evaluation of a Wi-Fi-Based Wireless Sensor Network Using Raspberry Pi and ESP8266 Under Multi-Rate Traffic Conditions

[1] **Ali H. BenHusein** [2] **Fatah M. Shakrum** [3] **Ebtisam M. Elgdiri**
[1] Computer Department, College of Electronic Technology, Tripoli
Email: Ali_Hussain_m@yahoo.com
[2] Computer Department, High institute of medical technology, Abosalim, Tripoli
Email: Fatah.trip@gmail.com
[3] Communication Department, College of Electronic Technology, Tripoli
Email: Ebtesam_m7@yahoo.com

تاريخ الاستلام: 2026/04/03 تاريخ المراجعة 02/05/ 2026 تاريخ القبول:2026/05/15- تاريخ النشر: 05/06/ 2026

**Abstract**
In modern Internet of Things environments, Wireless Sensor Networks (WSNs) have become an essential element. particularly, in local monitoring for distributed sensing applications. Smart homes, environmental observation systems, industrial sites, and even small healthcare setups are examples of these applications. In most cases, these networks have traditionally relied on low-power technologies such as ZigBee and Bluetooth Low Energy because those protocols were built with constrained devices in mind. Still, the growing availability of inexpensive Wi-Fi-enabled microcontrollers, especially ESP8266-based boards, has made Wi-Fi a more realistic option for small and medium sensor deployments.
This paper presents an experimental evaluation of a Wi-Fi-based WSN built around a Raspberry Pi gateway configured as a dedicated wireless access point and three ESP8266-based WEMOS D1 R1 sensor nodes. The system was tested under several traffic reporting intervals, namely 1 s, 5 s, and 10 s, and also under a controlled same-channel interference condition. The evaluation focuses on end-to-end latency, packet delivery ratio, throughput, jitter, received signal strength indicator, and gateway resource usage in terms of CPU and memory consumption.
The results suggest that Wi-Fi can support highly reliable sensor communication when reporting intervals are moderate, with packet delivery ratio remaining above 99% in the less aggressive cases. Also, more fast in the reporting will introduce greater channel contention and noticeably less stable timing behavior. Under interference, performance degrades further, particularly in latency and delivery reliability. Taken together, the findings suggest that Wi-Fi is a practical option for small-scale sensor networks, though clearly not without trade-offs. The study also provides a reproducible testbed and grounded deployment insights for edge-based IoT sensing systems.
***Keywords***: Wireless Sensor Networks, ESP8266, Wi-Fi, Raspberry Pi, Internet of Things, Edge Computing, Performance Evaluation.

**الملخص**
تعد شبكات المستشعرات اللاسلكية (WSNs) مكونًا أساسيًا في بنية إنترنت الأشياء (IoT) الحديثة، حيث تمكّن من الاستشعار الموزع والمراقبة في الوقت الحقيقي عبر مجالات مثل المراقبة البيئية والمنازل الذكية والأتمتة الصناعية وأنظمة الرعاية الصحية. تقليديًا، تعتمد نشرات شبكات المستشعرات اللاسلكية على تقنيات الاتصال منخفضة الطاقة مثل ZigBee و Bluetooth Low Energy (BLE)، التي تركز على كفاءة الطاقة ونقل البيانات بمعدل منخفض. ومع ذلك، فإن التوافر

المتزايد للمتحكمات الدقيقة التي تتضمن الواي فاي منخفضة التكلفة، وخاصة الأجهزة القائمة علىESP8266 ، قد حفّز على استكشاف الواي فاي كوسيلة اتصال بديلة لشبكات الاستشعار الصغيرة .
تقدم هذه الورقة تقييمًا تجريبيًا شاملاً لشبكة استشعار لاسلكية (WSN) قائمة على الواي فاي تم تنفيذها باستخدام Raspberry Pi والتي تعمل كبوابة خروج Gateway وكنقطة وصول لاسلكية wireless access point مخصصة وثلاثة عقد استشعار WEMOS D1 R1 تعتمد علي.ESP8266 . ثم تقييم النظام تحت فترات تقارير متعددة (1 ثانية، 5 ثوانٍ، و10 ثوانٍ) وتحت سيناريو تداخل قنوات متحكم فيه. تشمل مقاييس الأداء زمن الوصول من النهاية إلى النهاية ، ونسبة تسليم الحزم(PDR) ، ومعدل النقل، والتقطيع (jitter)، ومؤشر قوة الإشارة المستلمة(RSSI) ، ومعدل استخدام موارد البوابة (وحدة المعالجة المركزية والذاكرة) .
أظهرت النتائج التجريبية أن شبكات الاستشعار اللاسلكية المعتمدة على الواي فاي يمكن أن تحقق تشغيلًا عالي الموثوقية مع معدل تسليم الحزم (PDR) الذي يتجاوز 99٪ تحت فترات الإبلاغ المعتدلة. كما أن معدلات الإبلاغ العالية تزيد بشكل كبير من التنافس والتأرجح.
تكشف تجارب التداخل أيضًا عن تدهور كبير في الأداء، مما يبرز أهمية التخطيط الدقيق للقنوات. توفر الدراسة مجموعات بيانات تجريبية قابلة للتكرار وإرشادات عملية للنشر لأنظمة الاستشعار القائمة على الحافة في إنترنت الأشياء.

**الكلمات المفتاحية:** شبكات المستشعرات اللاسلكية ، انترنت الاشياء ، الشبكات اللاسلكية ، نقاط الوصول اللاسلكية، نظام حوسبة الحافة.

## 1 Introduction

Wireless Sensor Networks have, for quite a while now, been one of the quiet foundations of the Internet of Things. They enable a group of small distributed devices to sense physical conditions and report them in real time, whether the task is monitoring temperature in a greenhouse, detecting smoke in a building corridor, or tracking light and humidity in a smart home environment [1], [2]. The basic idea is not new, but the range of applications keeps expanding.

Most traditional WSNs have been built around low-power communication technologies such as Bluetooth Low Energy, ZigBee, and IEEE 802.15.4. That design choice makes sense. These standards were developed for resource-constrained devices, where battery life matters as much as communication reliability, and sometimes more[3], [4]. Always, the same strengths that make such technologies attractive can also become limitations. Throughput is usually modest, integration with IP-based infrastructure can be less direct, and deployment often requires additional gateways or protocol translation layers [5].

More recently, low-cost Wi-Fi-capable platforms such as the ESP8266 and ESP32 have changed the conversation somewhat. These microcontrollers come with integrated TCP/IP support and direct Internet connectivity, while remaining affordable enough for experimental and even practical deployment [6]. At the same time, small single-board computers like the Raspberry Pi have become common gateway platforms because they combine flexible networking support with enough processing power to handle edge tasks without much trouble[7]. Since these devices operate over IEEE 802.11 networks, they naturally invite the question of whether Wi-Fi itself can serve as the communication backbone of a WSN[8].

That sounds attractive, but it is not entirely straightforward. Wi-Fi was not designed specifically for sensor traffic. Unlike some low-power protocols, it relies on contention-based medium access, meaning nodes compete for channel access. In a lightly loaded system, that may not be a serious issue. Under heavier traffic, however, or in the presence of nearby wireless interference, the behavior becomes less predictable. Latency can increase, jitter becomes harder to control, and packet loss may start to appear[9] .

A number of earlier studies have explored Wi-Fi in IoT contexts. N. Ahmed et al., for example, compared ZigBee and Wi-Fi in IoT applications and found that Wi-Fi offers higher throughput, but at the cost of more contention when traffic becomes dense[10]. Similarly, Raspberry Pi-based edge gateways have been proposed to reduce dependence on cloud processing and lower end-to-end delay in IoT systems[11].

However, most of the existing literature demonstrates relying on setups rather than careful experimental performance analysis. There is still limited work that examines latency variation, jitter, delivery reliability, and interference sensitivity across different reporting rates in a reproducible Wi-Fi-based WSN setting [12].

This paper reviews that gap by providing an experimental evaluation of a Wi-Fi-based sensor network built using one Raspberry Pi gateway and three ESP8266 sensor nodes. Rather than treating Wi-Fi as either obviously suitable or obviously problematic, the work tries to measure what actually happens under realistic operating conditions.

The main contributions are :

1. A reproducible Wi-Fi-based sensor network architecture using widely available, low-cost hardware.
2. A multi-rate traffic evaluation using several sensor reporting intervals.
3. An interference sensitivity analysis under a controlled same-channel wireless interference scenario.
4. A quantitative assessment of latency, jitter, throughput, reliability, RSSI, and gateway resource utilization.

## 2 Related Work

Early WSN research was shaped largely by one central concern, namely how to keep tiny sensor nodes alive for as long as possible. That focus naturally led to energy-aware routing, lightweight communication, and low-duty-cycle operation[13]. ZigBee-based systems built on IEEE 802.15.4 became especially common because they were designed for low-power, low-data-rate environments and could support mesh networking when needed[14]. Bluetooth Low Energy followed a somewhat similar path. BLE was designed for short-range, low-power communication and has been used in many IoT applications where compact data exchange is enough. Gomez et al. gave a useful overview of BLE and its performance characteristics in low-power sensor environments[5]. BLE works well in many cases, though it is not always ideal when continuous connectivity or broader network integration is required [15].

Wi-Fi, in contrast, was often considered a poor fit for sensor networks because of its higher power consumption and heavier protocol stack. That assumption was reasonable, especially given the limited resources of sensor platforms. Still, hardware developments have shifted the trade-off. Platforms based on the ESP8266 have shown that Wi-Fi-enabled sensing can be achieved at relatively low cost, and in some cases with acceptable energy behavior depending on traffic patterns and sleep configuration [16].

Comparative work has highlighted both the strengths and limits of Wi-Fi in this space. Rahman et al. discovered that Wi-Fi presents a clear throughput advantage over ZigBee. They also reported that contention becomes more noticeable as traffic increases[17]. That is not surprising, really. Wi-Fi provides more bandwidth, but it also requires nodes to contend for a shared medium, which can introduce instability. Mangold et al. discussed quality-of-service support within IEEE 802.11 and pointed to the role of traffic prioritization in reducing delay, though practical gains depend a lot on the actual deployment context[18].

Another relevant thread in the literature is edge computing. Instead of pushing all sensor traffic to remote cloud platforms, edge systems process at least part of the data closer to where it is generated. Shi et al. argued that this can reduce latency and improve responsiveness in IoT systems[19]. In practice, a Raspberry Pi gateway is often enough for local aggregation, filtering, dashboarding, or lightweight decision-making, which makes it a practical option for small deployments [20].

At the application layer, protocols such as MQTT and CoAP have also been introduced to make communication more efficient for constrained devices. These protocols affect the overall behavior of the system, but are not substitutes for the underlying wireless technology. They can also reduce overhead or simplify message exchange, mainly in the edge-assisted architectures[21].

Even with these developments, there is still relatively little experimental work that carefully examines Wi-Fi-based WSNs across multiple traffic rates while accounting for latency stability, jitter, packet delivery, and interference in a unified setup. That is where this work positions itself. It does not try to argue that Wi-Fi should replace lower-power technologies in general. It asks a narrower and possibly more useful question: "Under what conditions does a simple Wi-Fi-based sensor network perform well enough to be worth using?"

## 3 System Architecture and Methodology

### 3.1 Hardware Platform

As shown in Figure 1, the experimental platform consists of one Raspberry Pi 4 Model B configured as the gateway and three ESP8266-based WEMOS D1 R1 boards acting as sensor nodes. Each node includes a DHT22 temperature and humidity sensor, a digital light-dependent resistor connected to pin D7, and a flame detection sensor. The combination was chosen to represent a small but realistic multi-sensor monitoring setup rather than a single-purpose test node.

The Raspberry Pi is configured to have two roles in our topology. The first role is to act as the local Wi-Fi access point, and its second function is as the edge gateway, which receives, processes, and stores incoming sensor data. That dual role is useful in practice because it avoids the need for a separate router or cloud relay during experimentation.

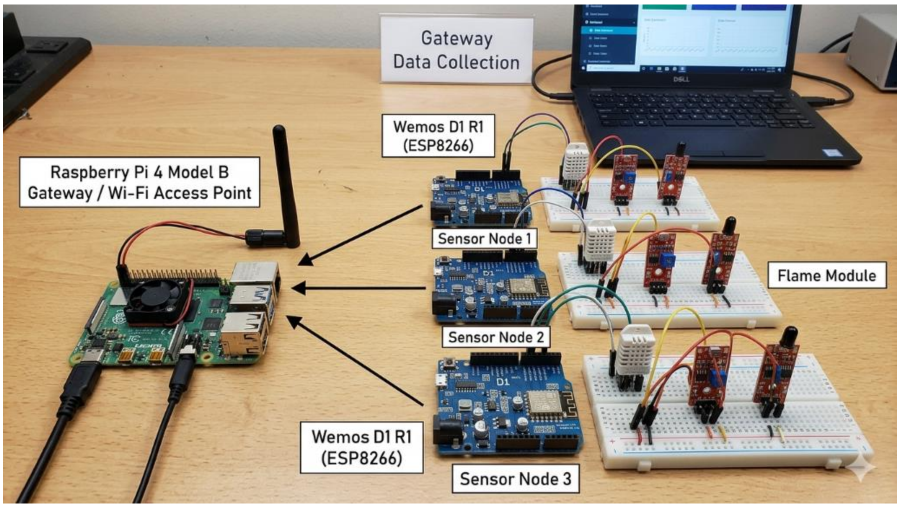


Figure 1: The Experimental platform

### 3.2 Network Configuration

The Raspberry Pi gateway operates as a dedicated wireless access point using **hostapd** and **dnsmasq** services. All ESP8266 nodes connect to the access point and communicate directly with the gateway over a local IPv4 network.

Sensor nodes transmit measurements using HTTP POST requests containing JSON-formatted payloads.

The gateway processes incoming sensor data using Node-RED, a flow-based programming platform widely used for rapid development of IoT applications and edge analytics.

### 3.3 Performance Metrics

To evaluate the communication behavior of the network, the following metrics were measured:
• End-to-End Latency (ms).
• Packet Delivery Ratio (PDR).
• Throughput (kbps).
• Jitter (ms).
• Received Signal Strength Indicator (RSSI).
• Gateway CPU Utilization (%).
• Gateway Memory Usage (MB).

These metrics were chosen to reflect both sides (network-side and gateway-side) performance. Latency, jitter, packet delivery, and throughput describe the quality of communication, while CPU and memory usage help indicate whether the gateway becomes a bottleneck under the tested conditions.

**4 Experimental Scenarios**

Four experimental scenarios were considered:

1. 1-second reporting interval.
2. 5-second reporting interval.
3. 10-second reporting interval.
4. 1-second reporting interval with Wi-Fi interference.

Each scenario was executed for 10 minutes with all three sensor nodes operating simultaneously. The reporting interval was the main independent variable in the baseline experiments, while the interference condition was introduced to examine how a congested wireless environment affects performance under aggressive traffic. The specified intervals reflect a convenient range. The 1-second reporting interval is quite challenging for this type of sensor network, especially when multiple nodes operate simultaneously. On the other hand, the 5-second and 10-second intervals are more forgiving and may better reflect the actual use of many monitoring applications.

## 5 Results and Analysis

### 5.1 Latency Analysis

End-to-end latency is the time required for a node to transmit a packet and for the gateway to receive and process it. Fig.2 shows the mean and 95th percentile latency versus reporting interval.

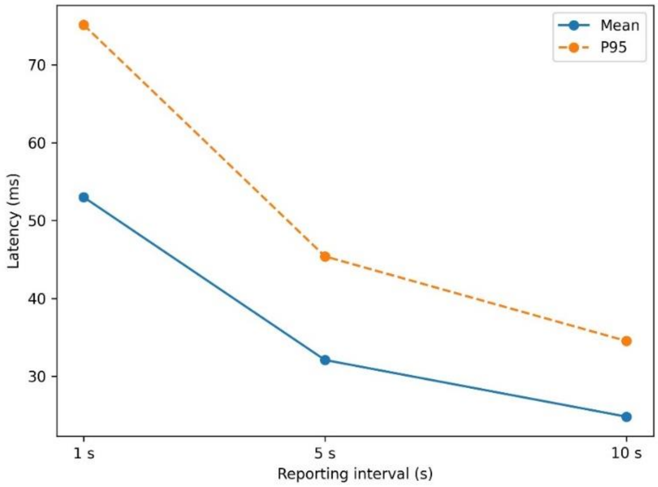


**Figure 2:** Mean and 95th percentile latency vs reporting interval.

The results indicate a clear trend. Latency decreases when the reporting interval increases. The network experiences greater contention at the 1-second interval, which leads to higher mean delay and a wider latency distribution. Also, the system remains functional while it becomes less consistent. With 5-second and 10-second reporting intervals, communication stabilizes noticeably and latency becomes much more predictable. Although that outcome is quite intuitive, it is nonetheless worthwhile to validate experimentally. There is more idle time on the shared channel and fewer packet conflicts or backoff delays when nodes send less frequently.

### 5.2 Packet Delivery Ratio

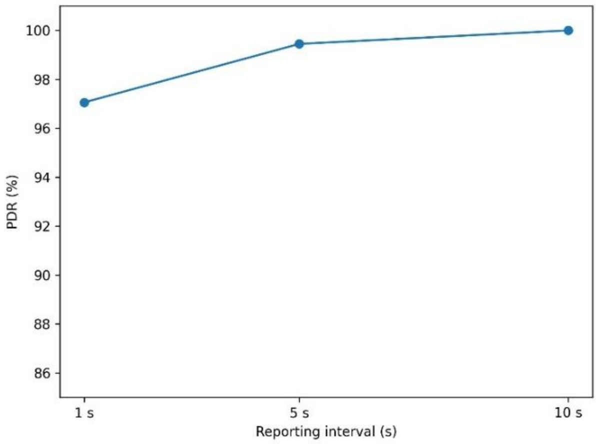


**Figure 3:** Packet Delivery Ratio vs reporting interval.

Packet Delivery Ratio remained above 99% for the 5-second and 10-second scenarios. At the 1-second interval, however, a small but visible decline appears due to increased network contention. The drop is not ruinous, but it is enough to suggest that pushing Wi-Fi-based sensor traffic too aggressively begins to affect reliability.
In a casual monitoring application, that may be acceptable. In a system where every reading subjects, the trade-off becomes harder to ignore.

### 5.3 Throughput Performance

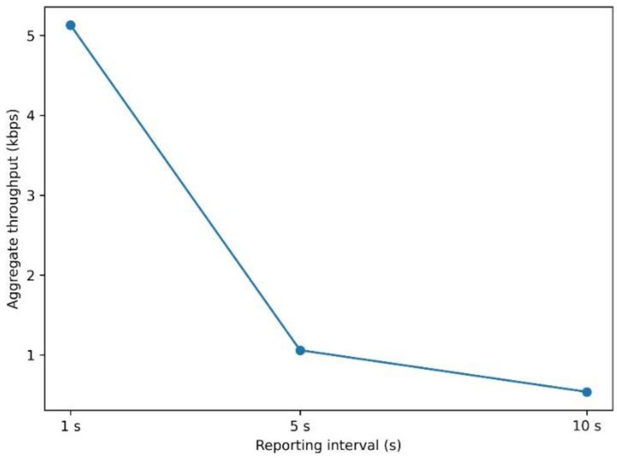


**Figure 4:** Aggregate network throughput vs reporting interval.

As expected, throughput is increasing as the reporting interval is shortened. When transmissions happen more frequently, it means we have more traffic per unit time. Even though there is more throughput, that does not necessarily mean better performance. It can lead to increased channel occupation, increased contention, and less stable timing behavior. That is one of those situations where a single metric can mislead when viewed in isolation. Higher throughput may look favorable, but not if it is accompanied by degraded latency and increased packet loss.

### 5.4 Jitter Analysis

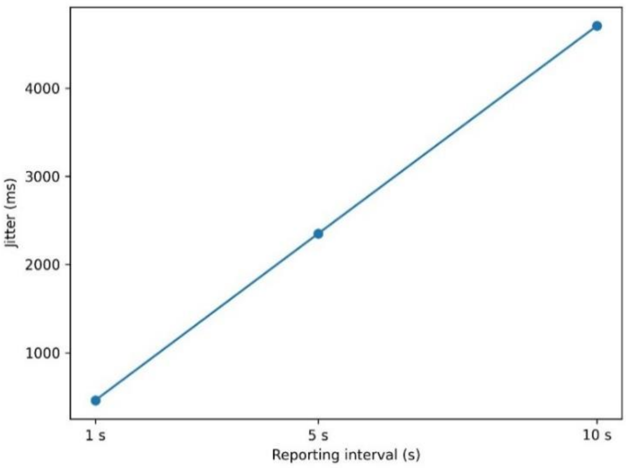


**Figure 5:** Jitter vs reporting interval.

Jitter increases significantly under high-rate reporting. This reflects the variability in packet inter-arrival times caused by channel access delays and medium contention. The 1-second scenario produces the least stable timing behavior, whereas the longer intervals show more consistent packet timing. For some sensing tasks, a bit of jitter may not matter much. For others, especially those involving event correlation or time-sensitive edge responses, it can be more problematic than the average delay itself.

### 5.5 Gateway Resource Utilization

Figures 6 and 7 show CPU and Gateway utilization vs. reporting intervals.

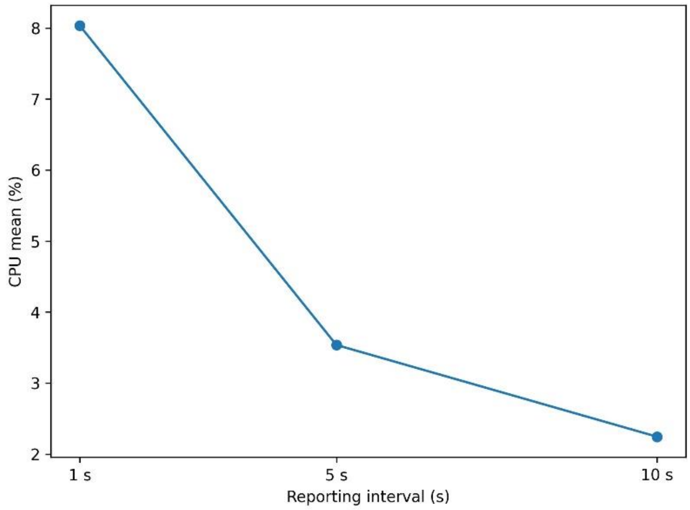


**Figure 6:** CPU utilization vs reporting interval.

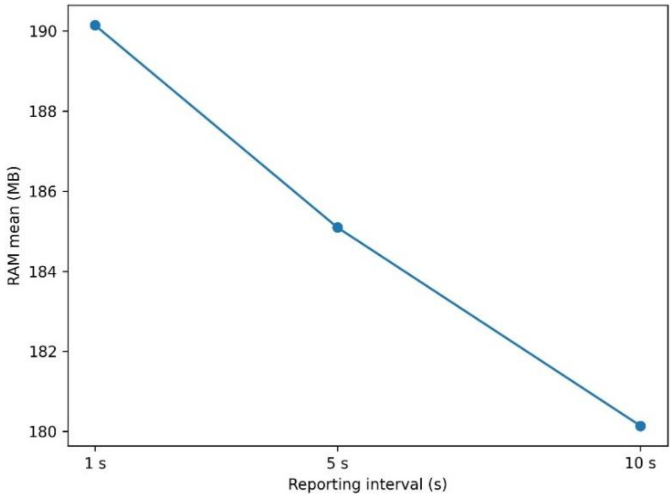


**Figure 7:** RAM utilization vs reporting interval.

Across all implemented scenarios, we can clearly see that the Gateway CPU utilization remains below 10% and that memory usage remains well within the Raspberry Pi's capabilities. This suggests that the gateway is not the limiting element in the current setup. The main restrictions are more likely to come from the wireless medium rather than from local processing capacity. It means that the Raspberry Pi can comfortably support this scale of deployment and perhaps relatively larger ones without becoming overloaded.

### 5.6 RSSI Distribution

RSSI distributions per node are shown in Figure 8.

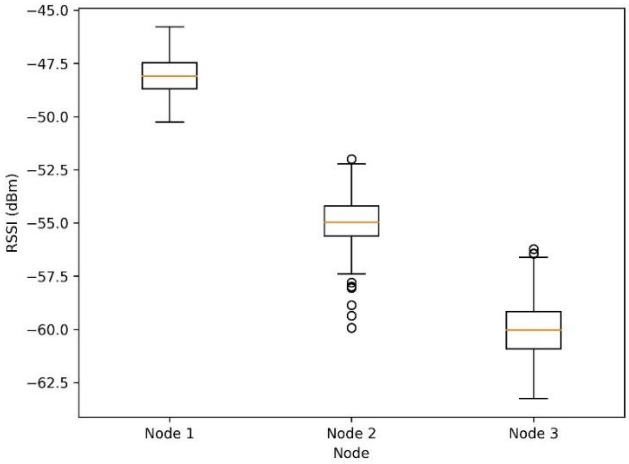


**Figure 8:** RSSI distribution for sensor nodes.

RSSI values remained within acceptable ranges during the baseline experiments, indicating relatively stable signal quality across all nodes. We can also see that the mean RSSI values were similar across all scenarios, indicating that the performance differences were mostly due to traffic conditions rather than drastic changes in radio link quality.

### 5.7 Interference Impact

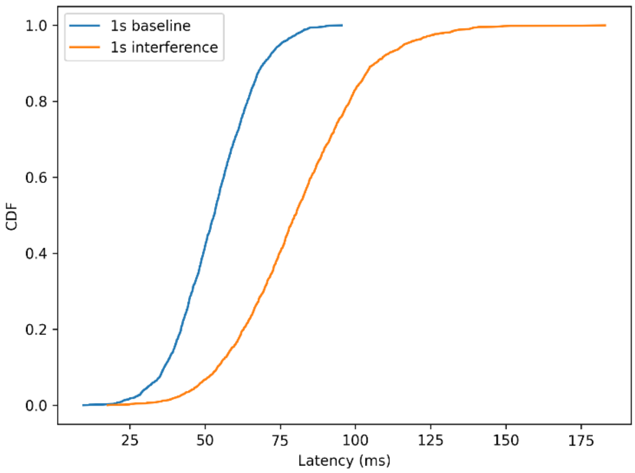


**Figure 9:** Latency cumulative distribution under baseline and interference scenarios.

The interference experiment produced a substantial increase in latency, especially in the tail of the distribution. In other words, not only did packets take longer on average, but extreme delays also became more common. Packet delivery reliability also worsened under interference. This is probably one of the more practically relevant findings in the paper. In a clean lab setting, Wi-Fi can look quite capable. In a real deployment surrounded by neighboring routers, phones, and other access points, the picture may be less tidy. Therefore, channel planning is not a minor detail. It is part of whether the system works well or just barely works.

### 5.8 Aggregate Performance Comparison

Tables 1 and 2 summarize the full results from all scenarios.

We can clearly conclude that a few patterns stand out immediately. The best latency and perfect packet delivery are achieved at the lowest throughput, with 10-second intervals. The 1-second baseline shows more frequent data updates but with reduced reliability and higher delay variation. The interference scenario is the weakest overall, with the highest latency and the lowest PDR. CPU and memory increase slightly under heavier load, but not to a worrying extent.

| Scenario | Interference | RX msgs Total | Pdr Total | Loss rate Total | Latency mean ms | Latency median ms | Latency p95 ms | Latency std ms | Jitter ms |
|---|---|---|---|---|---|---|---|---|---|
| interval_1s_baseline | FALSE | 1748 | 0.970571904 | 0.029428096 | 52.99320883 | 53.02461459 | 75.11079715 | 13.14844653 | 459.4622 |
| interval_5s_baseline | FALSE | 361 | 0.994490358 | 0.005509642 | 32.0929108 | 31.52865757 | 45.39771372 | 7.905689038 | 2350.765 |
| interval_10s_baseline | FALSE | 183 | 1 | 0 | 24.81258021 | 24.3596684 | 34.52653602 | 5.745748667 | 4706.821 |
| interval_1s_interference | TRUE | 1621 | 0.899556049 | 0.100443951 | 80.74221423 | 80.00603976 | 116.2236159 | 21.26323117 | 459.1315 |

Table 1: latency and jitter results from all implemented scenarios

| Scenario | Throughput Kbps | CPU Mean pct | CPU_p95 pct | RAM Mean mb | RAM p95 mb | RSSI Mean dbm | RSSI MIN dbm | RSSI MAX dbm |
|---|---|---|---|---|---|---|---|---|
| interval_1s_baseline | 5.12733 | 8.033304277 | 9.180676 | 190.1409628 | 191.8630618 | -54.3114213 | -63.25278344 | -45.7825194 |
| interval_5s_baseline | 1.058902 | 3.536473766 | 4.66797 | 185.0915492 | 186.8473103 | -54.4948984 | -67.72591626 | -44.8880522 |
| interval_10s_baseline | 0.536796 | 2.244319815 | 3.443964 | 180.1396171 | 182.0099868 | -54.6132468 | -63.99684057 | -43.9955791 |
| interval_1s_interference | 4.754878 | 9.549296251 | 10.74466 | 192.1198343 | 194.0382224 | -54.0342384 | -64.19736692 | -44.7800107 |

Table 2: throughput, CPU, and RSSI results from all experimental scenarios

## 6 Discussion

The experimental results demonstrate that Wi-Fi-based sensor networks can achieve reliable performance for small-scale deployments when reporting intervals are carefully selected. While aggressive reporting intervals improve temporal resolution, they significantly increase contention and jitter.

Additionally, interference experiments highlight the importance of proper channel planning when deploying Wi-Fi-based WSNs. Compared with ZigBee-based networks, Wi-Fi provides higher throughput and native IP connectivity. However, the contention-based medium access mechanism introduces performance variability under high traffic conditions.

## 7 Conclusion

This paper presented an experimental evaluation of a Wi-Fi-based Wireless Sensor Network implemented using a Raspberry Pi gateway and ESP8266 sensor nodes. The system was tested with multiple traffic reporting intervals and controlled interference to assess its communication behavior under realistic load conditions.

The results indicate that Wi-Fi can serve as a viable communication technology for small-scale WSN deployments, particularly when paired with edge computing platforms that can process data locally. Under moderate reporting intervals, the network achieved strong reliability and acceptable latency. Under more aggressive reporting and interference conditions, performance degradation became more pronounced, especially in latency, jitter, and packet delivery ratio.

Future work could explore larger node populations, application-layer messaging alternatives such as MQTT, and a deeper analysis of energy consumption. Those directions would help clarify whether the observed behavior holds as the network scales and as the system moves from controlled experimentation toward more realistic deployment environments.

**References**


[1] S. W. Nourildean, M. D. Hassib, and Y. A. Mohammed, "Internet of things based wireless sensor network: a review," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 1, pp. 246–261, Jul. 2022, doi: 10.11591/IJEECS.V27.I1.PP246-261.

[2] T. A. Singh and J. Chandra, "IOT based Green House monitoring system," *Journal of Computer Science*, vol. 14, no. 5, pp. 639–644, 2018, doi: 10.3844/JCSSP.2018.639.644.

[3] T. Muciaccia and P. Tedeschi, "Future scenarios for the infrastructure digitalization: The road ahead," *Frontiers in the Internet of Things*, vol. 2, Mar. 2023, doi: 10.3389/FRIOT.2023.1140799.

[4] A. Abane, P. Muhlethaler, and S. Bouzefrane, "Modeling and improving named data networking over IEEE 802.15.4," *Annales des Telecommunications/Annals of Telecommunications*, vol. 76, no. 11–12, pp. 839–850, Dec. 2021, doi: 10.1007/S12243-021-00848-W.

[5] S. M. Darroudi, C. Gomez, and J. Crowcroft, "Bluetooth Low Energy Mesh Networks: A Standards Perspective," *IEEE Communications Magazine*, vol. 58, no. 4, pp. 95–101, Apr. 2020, doi: 10.1109/MCOM.001.1900523.

[6] F. Serepas, I. Papias, K. Christakis, N. Dimitropoulos, and V. Marinakis, "Lightweight Embedded IoT Gateway for Smart Homes Based on an ESP32 Microcontroller," *Computers*, vol. 14, no. 9, Sep. 2025, doi: 10.3390/COMPUTERS14090391.

[7] B. Qureshi, K. Kawlaq, A. Koubaa, B. Saeed, and M. Younis, "A Commodity SBC-Edge Cluster for Smart Cities," *2nd International Conference on Computer Applications and Information Security, ICCAIS 2019*, May 2019, doi: 10.1109/CAIS.2019.8769500.

[8] "Wireless Sensor and Actuator Networks for Smart Cities," *Wireless Sensor and Actuator Networks for Smart Cities*, Dec. 2018, doi: 10.3390/BOOKS978-3-03897-424-6.

[9] P. Bosch, S. Latré, and C. Blondia, "IEEE 802.11 latency modeling with non-IEEE 802.11 interfering source," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11618 LNCS, pp. 40–50, 2019, doi: 10.1007/978-3-030-30523-9_4.

[10] N. Ahmed, H. Rahman, and M. I. Hussain, "A comparison of 802.11ah and 802.15.4 for IoT," *ICT Express*, vol. 2, no. 3, pp. 100–102, Sep. 2016, doi: 10.1016/J.ICTE.2016.07.003.

[11] B. Rababah and R. Eskicioglu, "Distributed intelligence model for iot applications based on neural networks," *International Journal of Computer Network and Information Security*, vol. 13, no. 3, 2021, doi: 10.5815/IJCNIS.2021.03.01.

[12] L. Sciullo, A. Trotta, L. Gigli, and M. Di Felice, "Deploying w3c web of things-based interoperable mash-up applications for industry 4.0: A testbed," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11618 LNCS, pp. 3–14, 2019, doi: 10.1007/978-3-030-30523-9_1.

[13] N. Singh, "An Improved Gateway-Based Energy-Aware Multi-Hop Routing Protocol for WSNs.," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 6, no. 5, pp. 647–655, May 2018, doi: 10.22214/IJRASET.2018.5109.

[14] Y. K. Huang, A. C. Pang, and H. N. Hung, "A comprehensive analysis of low-power operation for beacon-enabled IEEE 802.15.4 wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 8, no. 11, pp. 5601–5611, Nov. 2009, doi: 10.1109/TWC.2009.081485.

[15] P. Barker and M. Hammoudeh, "A survey on low power network protocols for the internet of things and wireless sensor networks," *ACM International Conference Proceeding Series*, vol. Part F130522, Jul. 2017, doi: 10.1145/3102304.3102348.

[16] D. Thomas, R. McPherson, G. Paul, and J. Irvine, "Optimizing Power Consumption of Wi-Fi for IoT Devices: An MSP430 processor and an ESP-03 chip provide a power-

efficient solution.," *IEEE Consumer Electronics Magazine*, vol. 5, no. 4, pp. 92–100, Oct. 2016, doi: 10.1109/MCE.2016.2590148.

[17] Y. Zhang and Q. Li, "HoWiES: A holistic approach to ZigBee assisted WiFi energy savings in mobile devices," *Proceedings - IEEE INFOCOM*, pp. 1366–1374, 2013, doi: 10.1109/INFCOM.2013.6566930.

[18] Y. Yang and R. Kravets, "Throughput guarantees for multi-priority traffic in ad hoc networks," *2004 IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 379–388, 2004, doi: 10.1109/MAHSS.2004.1392177.

[19] J. Ren, Y. Pan, A. Goscinski, and R. A. Beyah, "Edge Computing for the Internet of Things," *IEEE Netw.*, vol. 32, no. 1, pp. 6–7, Jan. 2018, doi: 10.1109/MNET.2018.8270624.

[20] Alnnale, T. (2026). Predictive Governance in Digital Enterprises: An LSTM-Enhanced Deep Learning Framework for Economic Optimization of IT Incident Management Using Enriched Process Logs. *Al-Farooq Journal of Sciences*, *2*(3), 86-113.

[21] T. Semwal and S. B. Nair, "AgPi: Agents on Raspberry Pi," *Electronics (Switzerland)*, vol. 5, no. 4, Dec. 2016, doi: 10.3390/ELECTRONICS5040072.

[22] I. F. Akyildiz et al., "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.