



# TTWiFi: Time-Triggered WiFi for Mobile Robotics in Human Environments

Carl Lusty<sup>1</sup> , Vladimir Estivill-Castro<sup>2</sup> , and René Hexel<sup>1</sup> 

<sup>1</sup> Griffith University, Nathan 4111, Australia

carl.lusty@alumni.griffithuni.edu.au, r.hexel@griffith.edu.au

<sup>2</sup> Universitat Pompeu Fabra, Barcelona 08018, Spain

vladimir.estivill@upf.edu

**Abstract.** WiFi is a ubiquitous protocol, but exhibits flaws that become particularly critical for teams of robots in human environments. We demonstrate that our Time-Triggered WiFi (TTWiFi) protocol allows us to utilise the benefits of the hardware available in mobile robotic systems while ensuring resilience and bounded error detection in the time domain as required by teams of robots to make reliable real-time decisions. Our experiments demonstrate that TTWiFi performs equally well in static and mobile scenarios in retaining its resilience to interference.

**Keywords:** Wireless Communication · Mobile Robotics · IEEE 802.11

## 1 Introduction

In the rapidly evolving landscape of robotics, one of the most transformative advancements has been the integration of WiFi connectivity into mobile robots designed to operate in human environments [31]. This technological leap has sparked a new era of robotics, seamlessly interacting with humans and their surroundings, making them more versatile, adaptable, and valuable in various applications. WiFi plays a significant role in enabling the Internet of Robotic Things (IoRT) [23, 40, 50]. Whether it is the endearing Pepper robot engaging customers in retail stores [36] or tele-operated robots assisting in remote medical procedures [30], the importance of WiFi connectivity cannot be overstated. However, the ubiquity of WiFi in human environments has profound implications on efficiency, safety, and overall human-robot interaction [30, 36].

Mobile robots have become increasingly ubiquitous in various human environments, from hospitals<sup>1</sup> and factories to public spaces and households. These robots are no longer confined to controlled, industrial settings but navigate the complexities of our daily lives. They perform tasks as diverse as delivering goods, providing companionship [18], assisting in healthcare [30], and even conducting

<sup>1</sup> Aethon's TUG autonomous mobile robot delivers medications, laboratory specimens, or other sensitive material within a hospital environment while using WiFi to communicate with elevators, automatic doors, and fire alarms.

surveillance [19]. However, WiFi faces challenges when it comes to reliability and speed, complicating how mobile robots gather real-time information, process data, and swiftly respond to dynamic situations.

Mobile robot WiFi connectivity is so pervasive that it has been proposed and deployed for navigation and localisation in human environments [15, 43, 56] where Global Positioning System (GPS) signals are unreliable or unavailable, such as indoor spaces<sup>2</sup>. Efficient wireless communication is vital for robots to collaborate seamlessly. Although WiFi enables connectivity, severe delays or missing packets jeopardise real-time coordination amongst teams of robots or signals from human pilots [46]. Communicating WiFi nodes often share the wireless communication medium. Thus, transmission attempts can often overlap, potentially resulting in collisions where message payloads are lost [24]. Given this fundamental issue, many coordination algorithms have been proposed [4, 24, 41, 42, 45, 48]. The growth of IoRT applications and the increasing deployment of mobile robots in human environments has stimulated research into the reliability of communication within, to, and from mobile robots [32, 33, 44]. However, communication timeliness remains a significant and open problem [17]. Therefore, applications where robots must share information, receive commands, and promptly respond to human input, can become safety-critical. Low latency and reliability are crucial in applications such as tele-operation, where human operators rely on near-instant feedback to control robots remotely for tasks like surgery, search and rescue, or hazardous material handling. WiFi connectivity also plays a pivotal role in enhancing the quality of human-robot interaction. In scenarios where robots are combined with mixed reality for sophisticated applications in assisting, guiding, or entertaining people, a stable and fast connection is essential for a smooth and natural experience. The Pepper robot, for instance, relies on WiFi to engage with customers in retail stores, answer questions, and provide information. However, the experience when lag and dropped packets are present eliminates the sensation of an interactive and enjoyable experience.

WiFi connectivity has also been used to extend mobile robot applications with cloud computing and data analytics. WiFi enables mobile robots to harness internet knowledge to make informed decisions and adapt to changing environments. However, with WiFi as the go-between, delays are inevitable. WiFi offers unparalleled scalability and flexibility for robot deployment, easy integration, and makes it cost-effective to deploy robots. In robotic environments (as opposed to many cloud services) eventual consistency is insufficient. Teams of robots need timely information, not only of the correct order of events, but delays that make some signals obsolete are simply unacceptable. This paper presents the experimental design for evaluating WiFi protocols for mobile robots in human environments. To the best of our knowledge, this is the first experimental setup where the communicating nodes are moving. We evaluate the TTWiFi protocol [27] against off-the-shelf alternatives.

---

<sup>2</sup> Some commercial solutions, like Aruba’s Meridian, use WiFi infrastructure for indoor positioning ([www.arubanetworks.com](http://www.arubanetworks.com)).

## 2 The Issue of Packet Loss and Packet Delay

The deployment of Wireless Local Area Networks (WLANs) in human environments is now massive. WiFi (the Institute of Electrical and Electronics Engineers (IEEE) 802.11 family) is now the most prevalent WLAN technology. However, many applications, specifically mobile robots, demand new Quality of Service (QoS) requirements for packet loss and communication timeliness [53]. Many reports highlight the significance of timeliness and reliability degradation under interference in wireless communication protocols. The community recognises that enabling real-time communications over license-free bands in open environments, encompassing multiple real-time stations, is particularly challenging [8]. Such is the demand for reliability and timeliness for mobile robotics in wireless environments that many protocols have been suggested, extending available technologies [12, 52, 54]. For instance, several proposals use IEEE 802.15.4 operating on the 2.4 GHz ISM band [16, 34, 37]. Their analyses assume unpolluted environments [8]. However, we focus here on WiFi (also known as IEEE 802.11) because of its wide availability and commonality among mobile robots deployed in human environments, as discussed in the introduction.

Variants for timely WiFi have been analysed theoretically and experimentally [10, 55], with some incorporated as amendments to the IEEE 802.11 standard. We highlight the IEEE 802.11n standard as it provides longer transmission distances than other WiFi standards and its ubiquity over newer standards such as IEEE 802.11ax [13]. Since IEEE 802.11n offers open-source firmware implementations [1, 49], and only two protocols are acknowledged [53] as suitable for timely communication over WiFi, we emphasize those two: the Wireless Flexible Time-Triggered (WFTT) protocol [9, 11] and TTWiFi, which has been recently proposed [26, 27]. However, wireless communication is so pivotal that numerous wireless protocols aim to achieve reliable real-time communication on IEEE 802.11 networks (i.e. RT-WMP [45], SchedWiFi [33], Adaptive TDMA [38], and RT-WiFi [51]). Some efforts aim at solving the problem even higher in the network stack [22, 35, 46]. That is, algorithms are designed at higher levels attempting to be tolerant and robust to the package losses and delays of the communication network. However, this cannot always be achieved.

Here, we report on the experimental setting where the stations move on board simple robotic platforms. The impact of packet loss has been studied for higher-level tasks, such as re-electing a leader [29]. Previous experimental research with indoor WiFi communication amongst a team of mobile robots explored WFTT behaviour when the access point changes or under a joining mesh and infrastructure networks [12]. That research emphasises the tight requirements of timely communication and reliability (detection and minimisation of packet loss). Still, it does not perform the experimental analysis we present here on single-hop broadcast performance. Moreover, no interference was evaluated [12] since swapping between two networks happens in such cases. We explore a large set of the parameters of the TTWiFi protocol under this experimental setting and contrast its performance against off-the-shelf alternatives at different levels of interference.

**Definition 1.** *A dependable (minimal packet loss, packet loss detection and timeliness) wireless communication protocol for mobile robotics, shall ensure alignment with the context of human environments.*

1. *The communication protocol must remain compatible with existing, widely-accessible mobile robotics platforms in human environments. Examples of such platforms include VGo [3], Giraff [2], Pepper [7], and Nao [5].*
2. *The communication protocol shall maintain functionality and coexistence within diverse wireless environments, including other IEEE 802.11 networks.*
3. *Practical implementation on physical hardware is a prerequisite for the communication protocol. This guarantees that the developed protocol holds relevance within the intended domain and can actively contribute to enhancing wireless communication, as opposed to being purely theoretical.*
4. *The reliability of the protocol's performance must be substantiated through real-world experimentation on tangible hardware across a spectrum of wireless interference scenarios. The experiments must unequivocally establish whether the protocol meets the desired latency and dependability.*

### 3 The TTWiFi Protocol

TTWiFi [26,27] relies on modifications to the IEEE 802.11n MAC layer, similar to other access methods developed for dependable WiFi networks, including RT-WMP [45], WFTT [9,11], SchedWiFi [33], and RT-WiFi [51]. TTWiFi reduces the Interframe Space (IFS) timing, granting TTWiFi priority access to the wireless medium. To meet the deterministic timing requirements of Definition 1, TTWiFi deactivates the Carrier Sense (CS) mechanism of the Distributed Coordination Function (DCF) in the IEEE 802.11 standards. DCF is the first method coordinating access to the communication medium. The second method, the Point Coordination Function (PCF), is optional and involves a central Point Coordinator (PC) node that manages access to the wireless medium.

The rationale for TTWiFi is that DCF is the only mandatory, and fundamental access method within IEEE 802.11 [4], relying on CSMA/CA involving contention-based access. Thus, all nodes employ CSMA/CA as the primary mechanism to decide when to transmit on an IEEE 802.11 LAN, i.e., whether the communication medium is presently unoccupied or if another node is actively transmitting. The Carrier Sense (CS) mechanism encompasses two distinct approaches: the physical and virtual CS functions. The physical CS function listens to the communication medium to detect ongoing transmissions. The virtual CS function relies on the Network Allocation Vector (NAV), containing transmission predictions, to gauge the likelihood of another node using the medium. Mechanisms such as RTS and CTS influence these predictions. The CS mechanism registers the medium as busy when the physical or virtual CS functions determine so, requiring a node to delay its transmission. A backoff timer decreases only when the communication medium is idle. Nodes can transmit only when their CS mechanism ensures medium availability and

their backoff timer reaches zero. For mobile robotics, this mechanism holds significance. First, a busy communication medium delays transmission until the medium becomes available, potentially leading to unpredictable communication delays. Second, variability in transmission timing is influenced by network load and the behaviour of other nodes using the same wireless frequency. These factors introduce dynamic and fluctuating transmission delays, causing inconsistencies in system behaviour. For instance, a remotely controlled robot relying on wireless commands could experience performance degradation and instability due to these random delays [17, 20].

TTWiFi operates as a time-triggered communication protocol, relying on precise transmission timing, thus necessitating altering the DCF to transmit without the mentioned fluctuating transmission delays. Turning off the CS mechanism makes the operation as if the medium were perpetually idle. While this action removes certain wireless collision safeguards from the DCF, TTWiFi nodes are safeguarded by the TDMA schedule, which prevents wireless collisions between synchronised nodes. Collisions between standard wireless nodes and TTWiFi nodes are minimised by the conventional wireless collision avoidance mechanisms. In such cases, standard DCF implementations will wait until TTWiFi transmissions pause before attempting retransmission, a factor that is considered in the timing allocated between TDMA slots. Other extensions, such as Enhanced Distributed Channel Access (EDCA) and Hybrid Coordination Function (HCF) Controlled Channel Access (HCCA), allow for adjustments in transmission priorities and exclusive periods for transmitting nodes.

TTWiFi employs time-triggered communication principles to determine transmission timing and promptly identify transmission failures within well-defined time constraints. Wireless communication medium access is governed by a pre-determined, cyclic Time Division Multiple Access (TDMA) transmission schedule. Consequently, the times of transmission arrivals can be inferred from their scheduled transmission times, along with the inclusion of bounded delays, e.g. the various IEEE 802.11n IFS intervals. This design ensures the detection of missed transmissions within their expected arrival times. TTWiFi is engineered to minimise single points of failure to bolster fault tolerance. As a result, it relies on Ad-Hoc wireless communication rather than a centralised Access Point (AP). TTWiFi assumes that all wireless nodes are located within transmission range, effectively sidestepping issues related to hidden nodes and exclusive groups.

Mobile robots designed for human environments employ General Purpose Operating Systems (GPOS), which poses challenges when scheduling transmission processes. GPOS robots rely on best-effort process schedulers that cannot guarantee precise timing, creating a problem for time-triggered communication, which relies on precise transmission timing. This issue can be addressed using a dedicated Real-time Operating System (RTOS) or processor for managing transmissions. Using an RTOS ensures better performance of TTWiFi, but on a GPOS, TTWiFi takes two steps to reduce OS scheduling jitter. First, when supported, we raise the priority of the TTWiFi transmission process within the OS. TTWiFi also uses process scheduling libraries compatible with multi-

ple OSs, such as `libdispatch`, which allows scheduling tasks at specific times, enabling TDMA scheduling. Here, we show that, even when the nodes are moving, these strategies significantly mitigate the inherent uncertainties associated with GPOS. Typically, time-triggered communication protocols achieve synchronisation by using transmission times alone to deduce each node’s clock drift. Due to the scheduling variability inherent in a GPOS, this approach is not feasible for TTWiFi. Thus, TTWiFi transmits a timestamp captured just before transmission to gauge clock drift among nodes. This clock synchronisation method consumes negligible additional communication bandwidth and reduces the dependence on scheduler precision, thus mitigating the effects of scheduling variability within a best-effort OS. TTWiFi accomplishes clock synchronisation through a two-phase approach. Initially, standard protocols such as the Network Time Protocol (NTP) or Precision Time Protocol (PTP) synchronise all clocks, eliminating extreme outliers. TTWiFi takes over clock synchronisation after this start-up phase, utilising its predefined TDMA communication schedule and disabling NTP and PTP to prevent uncontrolled wireless interference. During runtime synchronisation, TTWiFi evaluates clock variations among all participating nodes, establishing a unified global time that nodes use for scheduling their transmissions. TTWiFi nodes adopt Unix Time [47] as a common temporal reference (widely used in mobile robotics and general computing to measure the elapsed time since a fixed reference point). This way, TTWiFi reaches a consensus on the timing of slots in the TDMA round. Each node can compute the current communication slot by assuming the TDMA cycle commenced at the epoch time. Since the duration of our TDMA round remains constant, dividing the present time by this period yields the number of rounds since the epoch and a modulus operation determine the current offset within the TDMA round. Each node utilises the transmission time stamp and known IFS delays to assess the differences in system clock values of all other TTWiFi nodes to compensate for clock drift and periodically adjust its global clock through the Fault Tolerant Average (FTA) [25], which eliminates the impact of faulty clock values.

TTWiFi employs User Datagram Protocol (UDP)’s broadcast transmissions, offering a simple, well-supported interface on mobile robots for human environments. Unlike Transmission Control Protocol (TCP), UDP does not involve retransmissions or extensive data recovery mechanisms that affect transmission timing. Broadcast transmissions sidestep the Media Access Control (MAC) layer fragmentation timing problems. Ad-hoc frequency scanning, where nodes periodically pause transmissions to locate others for group formation, could disrupt TTWiFi’s predefined transmission timing, especially when scanning nodes experience delays in sending or receiving transmissions. Since TTWiFi defines its communication frequency in advance, it turns off ad-hoc scanning.

We compare our approach with TCP as the de facto standard for wireless communication between mobile robots in human environments, and also the basis of MQTT, named as “The Standard for IoT Messaging”. MQTT runs over any protocol that provides ordered, lossless, bi-directional connections, but the released implementations run on TCP. MQTT is gaining momentum, for example, in applications for robots in human environments, such as robot assistants [14].

## 4 Experiments with Mobile Robots

The Raspberry Pi 3 exhibits commonalities with common mobile robotics platforms for human environments; thus, its previous use for assessing TTWiFi effectiveness under a GPOS and its scheduling strategy [26,27]. TTWiFi clock synchronisation has been evaluated successfully, showing its merits [26, 27]. Moreover, TTWiFi has been shown to be robust to interference in experiments with static nodes [26,27]. Using the Raspberry Pi 3 offers several advantages for implementation and testing. Its CPU clock speed falls within the representative range of CPU speeds, and it provides numerous exposed General-Purpose Input/Output (GPIO) pins, enable measuring and debugging. Additionally, we can select a Network Interface Card (NIC) compatible with open-source firmware options, namely the Alfa Wireless Adapter AWUS036NHA [21]. Our Raspberry Pi nodes will operate using the same Raspberry Pi distribution, Raspbian GNU/Linux 8 (jessie)[6], coupled with version 4.4.38-v7+ of the Linux kernel. It is crucial to emphasise that TTWiFi is not bound to a specific operating system and utilises the POSIX API for portability. Moreover, we confine our implementation to a single Operating System (OS) process to mitigate any potential advantage arising from its multi-core configuration.

Until now, all experiments had employed a fixed node positioning approach, where the position and orientation of each node remained constant across different trials. While static configuration ensures consistency, for wireless communication between mobile robots, it becomes imperative to assess how mobility impacts the performance and reliability of TTWiFi. Hence, we describe a series of experiments devised to execute TTWiFi and compare it with standard baselines within a mobile experimental setup. To our knowledge, this is the first evaluation of WiFi alternatives under such dynamic conditions.

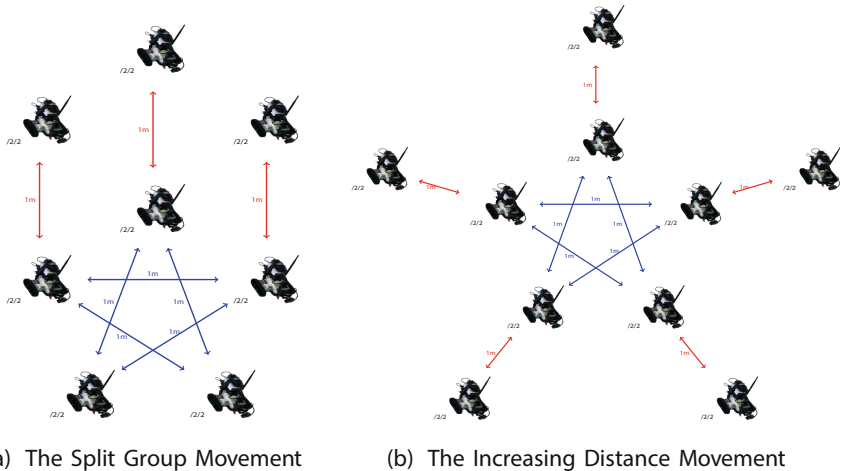
Inspired by mobile robots for human environments such as Giraff [2] and Oculus Prime, we will conduct our mobility experiments mounting the Pi 3 atop a locomotion platform (an EV3 platform common in mobile robotics localisation and map research [28]). The EV3 provides a straightforward USB interface for control and includes encoders on its motors, ensuring that movements are reproducible. Figure 1 depicts our experimental platform, set as a vertically integrated system with the battery-powered EV3 at the bottom attached to a basic track drive configuration, similar to those found in various EV3 design manuals. Atop the EV3, a 5V USB power bank supplies the Pi 3, which is mounted atop inside a standard Pi Foundation enclosure. The NIC is positioned above the Pi.



**Fig. 1.** Mobile NXT robot with a mounted Raspberry Pi 3.

#### 4.1 Split Group Movement Pattern

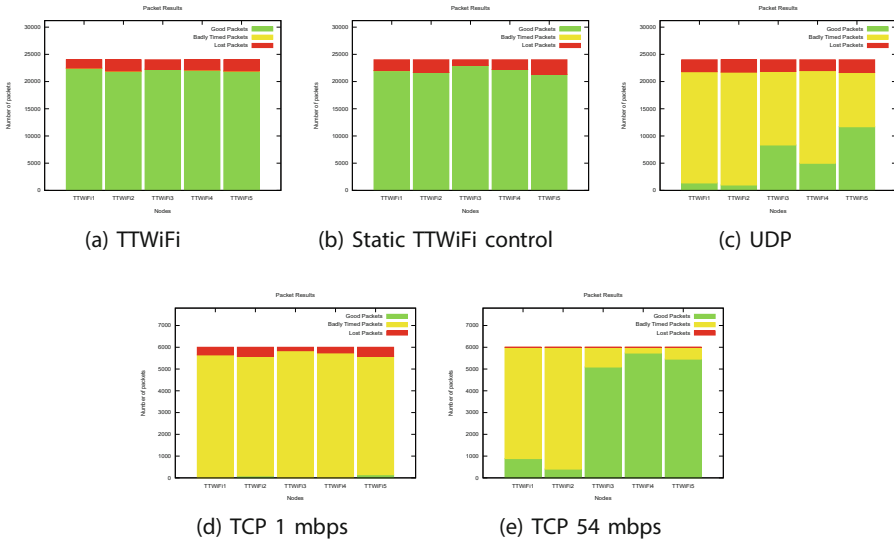
In the initial configuration, we divide the set of nodes into two groups of repeated separating and reuniting trajectories. The mobile nodes involved in this scenario are TTWiFi 2, 3, and 5, while the remaining nodes remain stationary. Figure 2a provides a visual representation of the mobile nodes, with a red line denoting the maximum distance they traverse and their respective direction. These experiments are progressively aligning with the concept of Swarm Robotics [39]. We reproduced all experiments in [27] to contrast this as a baseline for a static configuration with our mobile experiments. The first part of Table 1 shows the parameters, such as frequency (in Hz) and maximum bitrate (megabit per second (Mbit/s)) for this setting. Figure 3 shows typical results from the comparison protocols. Various nodes exhibited superior performance due to poor synchronization, see Fig. 4. The TTWiFi results were much more consistent. Table 2 shows that the difference in well-timed packets between the static and mobile TTWiFi experiments was 0.39%.



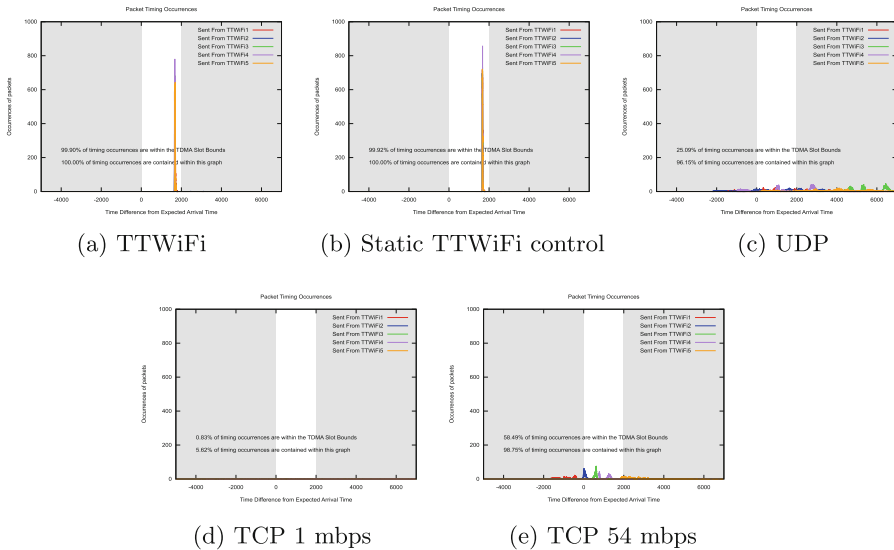
**Fig. 2.** Movement experiments. Blue measurements denote the initial relative positioning. Red measurements denote the extents of the movement pattern. (Color figure online)

**Table 1.** Parameters for the two experimental settings.

Split Group Movement Pattern experiments					Increasing Distance Pattern experiments				
Hz	Max Bitrate	Clock Sync	Protocol	Moving	Hz	Max Bitrate	Clock Sync	Protocol	Moving
100	1 Mbit/s	Active (TTWiFi)	TTWiFi	Yes	100	1 Mbit/s	Active (TTWiFi)	TTWiFi	Yes
100	1 Mbit/s	Active (TTWiFi)	TTWiFi	No	100	1 Mbit/s	Active (TTWiFi)	TTWiFi	No
100	1 Mbit/s	Active (NTP)	UDP	Yes	100	1 Mbit/s	Active (NTP)	UDP	Yes
100	1 Mbit/s	Active (NTP)	TCP	Yes	100	1 Mbit/s	Active (NTP)	TCP	Yes
100	54 Mbit/s	Active (NTP)	TCP	Yes	100	54 Mbit/s	Active (NTP)	TCP	Yes



**Fig. 3.** Individual node experiment results for the Split Group Movement Pattern. The mobile nodes are TTwifi2, TTwifi3 and TTwifi5.



**Fig. 4.** Experiment packets occurrence charts for the Split Group Movement Pattern. The mobile nodes are TTwifi2, TTwifi3 and TTwifi5.

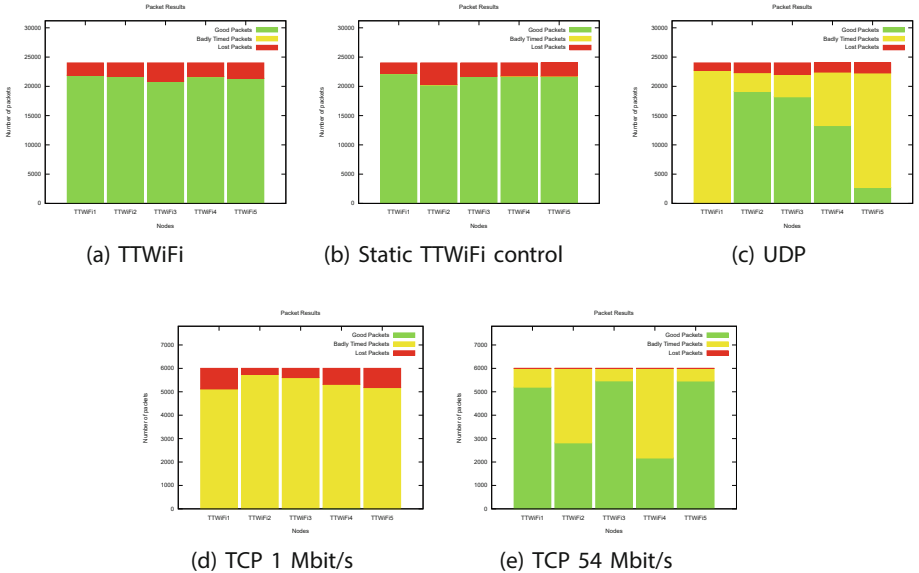
**Table 2.** Packet statistics for the protocols.

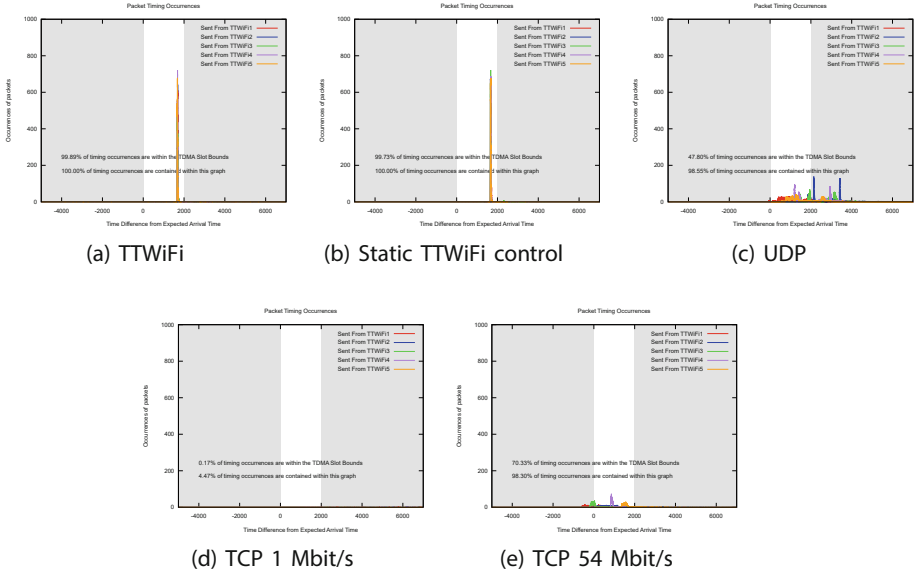
Split Group Movement Pattern experiment					Increasing Distance Movement Pattern experiment				
Protocol	Good Timing	Bad Timing	Lost Packets	% of Good Packets	Protocol	Good Timing	Bad Timing	Lost Packets	% of Good Packets
TTWiFi	110587	107	9306	92.16 % (SD: 0.87 %)	TTWiFi	107441	115	12444	89.53 % (SD: 1.51 %)
Static TTWiFi	110125	93	9782	91.77 % (SD: 2.31 %)	Static TTWiFi	107565	292	12143	89.64 % (SD: 2.75 %)
UDP	27370	81724	10906	22.81 % (SD: 17.10 %)	UDP	53449	58411	8140	44.54 % (SD: 32.85 %)
TCP 1 mbps	235	28159	1606	0.78 % (SD: 1.00 %)	TCP 1 mbps	45	26925	3030	0.15 % (SD: 0.22 %)
TCP 54 mbps	17573	12413	14	58.58 % (SD: 39.28 %)	TCP 54 mbps	21117	8869	14	70.39 % (SD: 23.82 %)

## 4.2 Increasing Distance Pattern

For this mobility pattern, nodes start in the standard static configuration and move directly away from each other. Once each node reaches the extent of the distance denoted in red in Fig. 2b, they reverse course and drive back to their original positions. This movement pattern repeats and forms a star shape of nodes that expands and contracts several times throughout each experiment. As with the first mobility pattern, a baseline control experiment was also conducted at the same time and in the same location using static nodes. The second part of Table 1 shows the parameters.

Table 2 shows only a slight difference between the static and mobile TTWiFi experiments of 0.11%. Based on the individual node results in Fig. 5 and the synchronisation results shown in Fig. 6, the comparison protocols behaved typically as well.

**Fig. 5.** Individual node experiment results for the Increasing Distance Movement Pattern.



**Fig. 6.** Packets occurrences for the Increasing Distance Movement Pattern.

## 5 Discussion

Two series of movement experiments were conducted to evaluate the impact of transmitter position changes and the movement of robots at runtime. In the first series of experiments, three nodes moved away from their initial positions as a group. This resulted in splitting the initial grouping of experiment nodes into two groups, which would later merge again. Compared to the static, no-movement control experiment we only see a slight difference in TTWiFi performance of 0.39%. The second movement pattern was to have the group of experiment nodes spread out. Each node moved away from the group, increasing its transmitter distance from all other nodes, again resulting in only a small difference (0.11%).

These minor differences between the movement patterns and their respective controls, which are well within the standard deviation observed within each set of experiments, demonstrate that there is no major detriment to the performance of TTWiFi within the movement patterns tested. These experiments demonstrate that with smaller movement patterns within communication range, TTWiFi remains a robust and temporally reliable time-triggered communication protocol, even when individual nodes change position and transmitter orientation.

## 6 Conclusions

WiFi connectivity plays a pivotal role for mobile robots to function optimally in human environments. Our experiments have shown that TTWiFi provides

resilience against interference in both the time and value domain in scenarios critical to the reliable operation of teams of robots in human environments.

Since developers of applications for mobile robots in human environments are adopting MQTT, an avenue of further work is to evaluate an application with TTWiFi versus the same application using MQTT.

## References

1. madwifi-project.org - trac (2007). <http://madwifi-project.org>
2. For researchers — Giraff (2014). <http://www.giraff.org/for-researchers/?lang=en>
3. VGo robotic telepresence for healthcare, education and business (2014). <http://www.vgocom.com>
4. IEEE standard for information technology-telecommunications and information exchange between systems local and metropolitan area networks-specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), pp. 1–3534 (2016)
5. Discover Nao, the little humanoid robot from SoftBank Robotics — SoftBank Robotics (2018). <https://www.softbankrobotics.com/emea/en/robots/nao>
6. Frontpage - Raspbian (2018). <https://www.raspbian.org>
7. Pepper, the humanoid robot from SoftBank Robotics, a genuine companion — SoftBank Robotics (2018). <https://www.softbankrobotics.com/emea/en/robots/pepper>
8. Bartolomeu, P., Alam, M., Ferreira, J., Fonseca, J.: Implementation and analysis of wireless flexible time-triggered protocol. *Ad Hoc Netw.* **58**, 36–53 (2017). <https://doi.org/10.1016/j.adhoc.2016.11.016>
9. Bartolomeu, P., Ferreira, J., Fonseca, J.: Enforcing flexibility in real-time wireless communications: a bandjacking enabled protocol. In: 2009 IEEE Conference on Emerging Technologies & Factory Automation, pp. 1–4 (2009). <https://doi.org/10.1109/ETFA.2009.5347177>
10. Bartolomeu, P., Fonseca, J.: Channel capture in noisy wireless contention-based communication environments. In: 2010 IEEE International Workshop on Factory Communication Systems Proceedings, pp. 23–32 (2010). <https://doi.org/10.1109/WFCS.2010.5548640>
11. Bartolomeu, P., Fonseca, J.: Towards flexible time triggered wireless communications. In: 2010 IEEE International Workshop on Factory Communication Systems Proceedings, pp. 203–206 (2010). <https://doi.org/10.1109/WFCS.2010.5548603>
12. Bazzi, A., Haxhibeqiri, J., Jarchlo, E.A., Moerman, I., Hoebeke, J.: Flexible Wi-Fi communication among mobile robots in indoor industrial environments. *Mob. Inf. Syst.* **2018**, 3918302 (2018). <https://doi.org/10.1155/2018/3918302>
13. Bellalta, B.: IEEE 802.11ax: High-efficiency WLANs. *IEEE Wireless Commun.* **23**(1), 38–46 (2016). <https://doi.org/10.1109/MWC.2016.7422404>
14. Diddeniya, I., Wanniarachchi, I., Gunasinghe, H., Premachandra, C., Kawanaka, H.: Human-robot communication system for an isolated environment. *IEEE Access* **10**, 63258–63269 (2022). <https://doi.org/10.1109/ACCESS.2022.3183110>
15. Ferris, B., Fox, D., Lawrence, N.: WiFi-SLAM using Gaussian process latent variable models. In: Proceedings 20th International Joint Conference on Artificial Intelligence, pp. 2480–2485. IJCAI 2007, Morgan Kaufmann, San Francisco, USA (2007)

16. Flammini, A., Marioli, D., Sisinni, E., Taroni, A.: Design and implementation of a wireless Fieldbus for plastic machineries. *IEEE Trans. Industr. Electron.* **56**(3), 747–755 (2009). <https://doi.org/10.1109/TIE.2008.2011602>
17. García-Valls, M., Casimiro, A., Reiser, H.P.: A few open problems and solutions for software technologies for dependable distributed systems. *J. Syst. Archit.* **73**, 1–5 (2017). <https://doi.org/10.1016/j.sysarc.2017.01.007>, <http://www.sciencedirect.com/science/article/pii/S1383762117300310>, special Issue on Reliable Software Technologies for Dependable Distributed Systems
18. Góngora Alonso, S., Hamrioui, S., de la Torre Díez, I., Motta Cruz, E., López-Coronado, M., Franco, M.: Social robots for people with aging and dementia: a systematic review of literature. *Telemed. e-Health* **25**(7), 533–540 (2023/09/10 2018). <https://doi.org/10.1089/tmj.2018.0051>
19. Habib, M.K., Baudoin, Y.: Robot-assisted risky intervention, search, rescue and environmental surveillance. *Int. J. Adv. Rob. Syst.* **7**(1), 10 (2010). <https://doi.org/10.5772/7249>
20. Hokayem, P.F., Spong, M.W.: Bilateral teleoperation: an historical survey. *Automatica* **42**(12), 2035–2057 (2006). <https://doi.org/10.1016/j.automatica.2006.06.027>, <http://www.sciencedirect.com/science/article/pii/S0005109806002871>
21. Inc., A.N.: AWUS036NHA 802.11b/g/n Long-Range USB Adapter (2009)
22. Jalil, A., Kobayashi, J., Saitoh, T.: Performance improvement of multi-robot data transmission in aggregated robot processing architecture with caches and QoS balancing optimization. *Robotics* **12**(3) (2023). <https://doi.org/10.3390/robotics12030087>
23. Kabir, H., Tham, M.L., Chang, Y.C.: Internet of robotic things for mobile robots: concepts, technologies, challenges, applications, and future directions. *Digital Commun. Netw.* (2023). <https://doi.org/10.1016/j.dcan.2023.05.006>
24. Kleinrock, L., Tobagi, F.: Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Trans. Commun.* **23**(12), 1400–1416 (1975)
25. Kopetz, H., Ochsenreiter, W.: Clock synchronization in distributed real-time systems. *IEEE Trans. Comput.* **C-36**(8), 933–940 (1987). <https://doi.org/10.1109/TC.1987.5009516>
26. Lusty, C.: TTWiFi: Improving Wireless Communication Reliability for Mobile Robotics in Human Environments. Ph.D. thesis, School of Communication and Information technology, Griffith University (2023)
27. Lusty, C., Estivill-Castro, V., Hexel, R.: TTWiFi: time-triggered communication over WiFi. In: *Proceedings 11th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, pp. 35–44. DIVANet 2021, Association for Computing Machinery, New York, USA (2021). <https://doi.org/10.1145/3479243.3487298>
28. Mac, T.T., Copot, C., Ionescu, C.M.: Design and implementation of a real-time autonomous navigation system applied to lego robots. *IFAC-PapersOnLine* **51**(4), 340–345 (2018). <https://doi.org/10.1016/j.ifacol.2018.06.088>, 3rd IFAC Conference on Advances in Proportional-Integral-Derivative Control PID
29. Madhevan, B., Sreekumar, M.: Analysis of communication delay and packet loss during localization among mobile robots. In: Berretti, S., Thampi, S.M., Dasgupta, S. (eds.) *Intelligent Systems Technologies and Applications*. AISC, vol. 385, pp. 3–12. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-23258-4\\_1](https://doi.org/10.1007/978-3-319-23258-4_1)
30. Nertinger, S., Kirschner, R.J., Naceri, A., Haddadin, S.: Acceptance of remote assistive robots with and without human-in-the-loop for healthcare applications. *Int. J. Soc. Robot.* (2022). <https://doi.org/10.1007/s12369-022-00931-9>

31. Pahlavan, K., Krishnamurthy, P.: Evolution and impact of Wi-Fi technology and applications: a historical perspective. *Int. J. Wirel. Inf. Netw.* **28**(1), 3–19 (2021). <https://doi.org/10.1007/s10776-020-00501-8>
32. Park, P., Coleri Ergen, S., Fischione, C., Lu, C., Johansson, K.H.: Wireless network design for control systems: a survey. *IEEE Commun. Surv. Tutorials* **20**(2), 978–1013 (2018). <https://doi.org/10.1109/COMST.2017.2780114>
33. Patti, G., Alderisi, G., Lo Bello, L.: SchedWiFi: an innovative approach to support scheduled traffic in ad-hoc industrial IEEE 802.11 networks. In: 2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA), pp. 1–9 (2015). <https://doi.org/10.1109/ETFA.2015.7301460>
34. Pereira, N., Andersson, B., Tovar, E.: WiDom: a dominance protocol for wireless medium access. *IEEE Trans. Industr. Inf.* **3**(2), 120–130 (2007). <https://doi.org/10.1109/TII.2007.898461>
35. Poberezkin, E., Roozbahani, H., Alizadeh, M., Handroos, H.: Development of a robust Wi-Fi/4G-based ROS communication platform for an assembly and repair mobile robot with reliable behavior under unstable network or connection failure. *Artif. Life and Robot.* **27**(4), 786–795 (2022). <https://doi.org/10.1007/s10015-022-00792-5>
36. Roozen, I., Raedts, M., Yanycheva, A.: Are retail customers ready for service robot assistants? *Int. J. Soc. Robot.* **15**(1), 15–25 (2023). <https://doi.org/10.1007/s12369-022-00949-z>
37. Rowe, A., Mangharam, R., Rajkumar, R.: Rt-link: a global time-synchronized link protocol for sensor networks. *Ad Hoc Netw.* **6**(8), 1201–1220 (2008). <https://doi.org/10.1016/j.adhoc.2007.11.008>
38. Santos, F.: An adaptive TDMA protocol for soft real-time wireless communication among mobile computing agents. In: Proceedings of the Workshop on Architectures for Cooperative Embedded Real-Time Systems (satellite of RTSS), pp. 5–8 (2004)
39. Schranz, M., Umlauf, M., Sende, M., Elmenreich, W.: Swarm robotic behaviors and current applications. *Front. Robot. AI* **7** (2020). <https://doi.org/10.3389/frobt.2020.00036>
40. Simoens, P., Dragone, M., Saffiotti, A.: The internet of robotic things: a review of concept, added value and applications. *Int. J. Adv. Rob. Syst.* **15**(1), 1729881418759424 (2018). <https://doi.org/10.1177/1729881418759424>
41. Sobrinho, J.L., Krishnakumar, A.S.: Quality-of-service in ad hoc carrier sense multiple access wireless networks. *IEEE J. Sel. Areas Commun.* **17**(8), 1353–1368 (1999). <https://doi.org/10.1109/49.779919>
42. Song, J., et al.: WirelessHART: applying wireless technology in real-time industrial process control. In: IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 377–386 (2008)
43. Tang, C., Sun, W., Zhang, X., Zheng, J., Sun, J., Liu, C.: A sequential-multi-decision scheme for WiFi localization using vision-based refinement. *IEEE Trans. Mob. Comput.* 1–16 (2023). <https://doi.org/10.1109/TMC.2023.3253893>
44. Tardioli, D., Villarroel, J.L.: Real time communications over 802.11: RT-WMP. In: 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems, pp. 1–11 (Oct 2007). <https://doi.org/10.1109/MOBHOC.2007.4428607>
45. Tardioli, D.: Real time communications in wireless ad-hoc networks. The RT-WMP protocol. Ph.D. thesis, Universidad de Zaragoza (2010)
46. Tardioli, D., Parasuraman, R., Ögren, P.: Pound: a multi-master ROS node for reducing delay and jitter in wireless multi-robot networks. *Robot. Auton. Syst.* **111**, 73 – 87 (2019). <https://doi.org/10.1016/j.robot.2018.10.009>, <http://www.sciencedirect.com/science/article/pii/S0921889017309144>

47. Thompson, K., Ritchie, D.M.: UNIX Programmer's Manual. Bell Telephone Laboratories (1975)
48. Tobagi, F., Kleinrock, L.: Packet switching in radio channels: part II - the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *IEEE Trans. Commun.* **23**(12), 1417–1433 (1975)
49. Vanhoef, M.: modwifi (2014). <https://github.com/vanhoefm/modwifi>
50. O, V., et al.: Internet of robotic things intelligent connectivity and platforms. *Front. Robot. AI* **7**, 104 (2020). <https://doi.org/10.3389/frobt.2020.00104>
51. Wei, Y., Leng, Q., Han, S., Mok, A.K., Zhang, W., Tomizuka, M.: RT-WiFi: real-time high-speed communication protocol for wireless cyber-physical control applications. In: 2013 IEEE 34th Real-Time Systems Symposium, pp. 140–149 (2013)
52. Wichmann, A., Demirelli Okkalioglu, B., Korkmaz, T.: The integration of mobile (tele) robotics and wireless sensor networks: a survey. *Comput. Commun.* **51**, 21–35 (2014). <https://doi.org/10.1016/j.comcom.2014.06.005>
53. Xu, Y., Zhou, S., Cao, Q., Zheng, B., Xiong, Z., Ni, Y.: Time-triggered reservation for cooperative random access in wireless LANs. In: 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), pp. 1–7. IEEE (2023). <https://doi.org/10.1109/VTC2023-Spring57618.2023.10199862>
54. Yan, Z., Jouandeau, N., Cherif, A.A.: A survey and analysis of multi-robot coordination. *Int. J. Adv. Rob. Syst.* **10**(12), 399 (2013). <https://doi.org/10.5772/57313>
55. Zhang, J., Han, G., Qian, Y.: Queuing theory based co-channel interference analysis approach for high-density wireless local area networks. *Sensors* **16**(9) (2016). <https://doi.org/10.3390/s16091348>
56. Zhang, L., et al.: WiFi-based indoor robot positioning using deep fuzzy forests. *IEEE Internet Things J.* **7**(11), 10773–10781 (2020). <https://doi.org/10.1109/JIOT.2020.2986685>