

Article

Improved Spectrum Coexistence in 2.4 GHz ISM Band Using Optimized Chaotic Frequency Hopping for Wi-Fi and Bluetooth Signals

Ashraf A. Elthoth 

National Telecommunication Institute, Cairo 11765, Egypt; ashraf.elthoth@nti.sci.eg

Abstract: Efficiently managing coexistence is crucial for achieving high-quality wireless communication in the Industrial, Scientific, and Medical (ISM) band where multiple wireless communication systems operate. Coexistence problems between Wi-Fi and Bluetooth Low Energy (BLE) signals are especially significant due to their shared frequency band, which often leads to interference and a reduced performance for both systems. Therefore, effective coexistence management strategies are essential for ensuring the optimal performance of Wi-Fi and Bluetooth signals in the ISM band. In this paper, the authors conducted a study to investigate coexistence management in the ISM band by evaluating four frequency hopping techniques: random, chaotic, adaptive, and an optimized chaotic technique proposed by the authors. The optimized chaotic technique aimed to minimize interference and ensure zero self-interference among hopping BLE nodes by optimizing the update coefficient. Simulations were conducted in an environment with existing Wi-Fi signal interference and interfering Bluetooth nodes. The authors compared several performance metrics, including the total interference rate, total successful connection rate, and trial execution time for channel selection processing time. The results indicated that the proposed optimized chaotic frequency hopping technique achieved a better balance between reducing interference with Wi-Fi signals, achieving a high success rate for connecting BLE nodes, and requiring minimal trial execution time. This makes it a suitable technique for managing interference in wireless communication systems. While the proposed technique had a higher interference than the adaptive technique for small numbers of BLE nodes, for larger numbers of nodes it had a much lower interference than the adaptive technique. The proposed optimized chaotic frequency hopping technique provides a promising solution for effectively managing coexistence in the ISM band, particularly between Wi-Fi and BLE signals. It has the potential to improve the performance and quality of wireless communication systems.



Citation: Elthoth, A.A. Improved Spectrum Coexistence in 2.4 GHz ISM Band Using Optimized Chaotic Frequency Hopping for Wi-Fi and Bluetooth Signals. *Sensors* **2023**, *23*, 5183. <https://doi.org/10.3390/s23115183>

Academic Editors: Yiliang Liu, Yanyu Cheng, Matthew Hamilton, Qiang Ye and Wei Wang

Received: 6 April 2023

Revised: 27 April 2023

Accepted: 2 May 2023

Published: 30 May 2023



Copyright: © 2023 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: spectrum coexistence; frequency hopping; Wi-Fi; BLE; ISM band; logistic map

1. Introduction

The wireless frequency spectrum has become increasingly congested due to the growing demand for wireless communication services. The coexistence of multiple wireless systems in the same frequency band can cause interference and performance degradation. Various methods have been proposed in the literature to address the coexistence issue in heterogeneous cognitive networks, including spectrum sensing, spectrum sharing, and spectrum allocation [1,2].

Cognitive radio networks require spectrum sensing to detect the presence and absence of primary users. Energy detection, cyclostationary detection, and matched filter detection are commonly used sensing techniques but they have limitations, such as a low detection accuracy, high computational complexity, and low robustness in the presence of noise and fading [3].

Spectrum sharing is another approach that has been widely studied, where available spectrum resources are shared among different wireless systems. Static and dynamic

spectrum access, auction-based spectrum allocation [4], and game-theoretic approaches [5] are some of the most used spectrum sharing methods.

Spectrum allocation has been proposed as an approach to minimize interference between wireless systems in heterogeneous cognitive networks. This involves assigning specific frequency bands to different wireless systems using interference-aware or interference-free methods [6]. However, this approach is not a complete solution to the problem of interference in such networks.

Despite the proposed spectrum allocation approach, the coexistence of different wireless systems in heterogeneous cognitive networks remains a challenging problem. Further research is needed to develop more efficient and robust techniques that can address this issue [7]. The research trend in this area is focused on enhancing the utilization of spectrum resources, reducing interference, and improving the performance of wireless systems.

The shared frequency band in the unlicensed ISM frequency band of 2.4 GHz used by wireless communication systems such as Wi-Fi and Bluetooth makes them highly susceptible to interference and coexistence issues leading to performance degradation [8]. Several approaches have been developed to mitigate coexistence issues, including spectrum sharing, adaptive frequency hopping, and statistical signal transmission [1,4,8–11].

Chaotic frequency hopping and adaptive frequency hopping techniques have been proposed to improve the coexistence of Wi-Fi and Bluetooth signals. Chaotic frequency hopping introduces randomness in the hopping sequence, making it difficult for interfering signals to predict and track the system's frequency. Adaptive frequency hopping dynamically adjusts the hopping pattern based on the current communication environment. The use of chaotic frequency hopping sequences is a promising approach that has shown potential for improving coexistence, but further research is needed to develop and evaluate new techniques for enhancing wireless communication system coexistence.

Recent studies have explored the potential of combining multiple coexistence strategies to enhance wireless communication system performance [9,12], including frequency-to-time mapping-based frequency hopping receivers [13] and cognitive radio techniques [14]. There is no single solution for enhancing coexistence in wireless communication systems in the 2.4 GHz ISM frequency band and a combination of approaches may be necessary.

This paper proposes using an optimized chaotic frequency hopping technique to address self-interference and interference with existing Wi-Fi signals in wireless communication systems. The researchers were motivated to explore chaotic frequency hopping due to its simplicity of generation and improved control over interference compared to random frequency hopping methods.

Generating a chaotic frequency hopping sequence using linear logistic maps is a relatively simple process that requires only one parameter to adjust the output of the sequence. This simplicity allows for the generation of a frequency hopping sequence that appears to be random while avoiding self-interference and interference with other wireless networks. By adjusting the “r” parameter, the frequency hopping sequence can be optimized to reduce the chances of self-interference and limit any interference with existing Wi-Fi signals. In contrast, random frequency hopping sequences require a much more complex generation process that typically involves the use of pseudo-random number generators. The use of these generators can lead to correlations between generated sequences or inconsistencies in the generated sequences, leading to self-interference or interference with other wireless networks. Therefore, the use of chaotic frequency hopping sequences offers a simpler and potentially more effective solution to interference issues in wireless communication systems.

The proposed optimized chaotic frequency hopping method is evaluated using simulations to analyze the coexistence between these two wireless systems in the ISM 2.4 GHz band. The results of this study are compared to those obtained from conventional frequency hopping and non-hopping systems. The main contributions of this paper include offering insights into the challenges of coexistence issues between wireless systems and proposing an alternative solution to mitigate mutual interference between Wi-Fi and Bluetooth signals.

Several performance metrics can be used to analyze the coexistence between Wi-Fi and Bluetooth, including channel occupancy, spectral purity, and throughput. Channel occupancy measures the fraction of time a channel is occupied by either Wi-Fi or Bluetooth and serves as an indicator of the available spectrum resources. Spectral purity measures the extent to which Wi-Fi and Bluetooth signals occupy the same frequency band, while throughput measures the amount of data transmitted per unit time. The simulation results provide a detailed analysis of these performance metrics to evaluate the effectiveness of the proposed optimized chaotic frequency hopping method for coexistence enhancement.

The main contributions of this study include providing insights into coexistence issues between wireless systems and proposing a novel approach to mitigate mutual interference between Wi-Fi and Bluetooth signals. Moreover, the proposed solution is not limited to Wi-Fi and BLE coexistence but is also applicable to other personal wireless communication systems operating on the 2.4 GHz band and using frequency hopping. This includes systems such as Zigbee, Bluetooth Classic, and Bluetooth Low Energy (BLE), among others. These systems are widely used in various applications, such as home automation, industrial control, and healthcare monitoring. The proposed solution utilizes frequency hopping to mitigate the interference from other wireless communication systems operating in the same band, which can improve the overall performance and reliability of these systems. Therefore, our proposed solution has the potential to be applied to a wide range of personal wireless communication systems operating in the 2.4 GHz band and using frequency hopping.

1.1. Data Collection

We conducted practical measurements to collect data from the 2.4 GHz ISM band using a spectrum analyzer. The measurements were performed in the National Telecommunication Institute building in Cairo, Egypt over 5 working days, resulting in 40,000 data instances for the 40 channels in the 2.4 GHz ISM band.

The data collection step aimed to encompass a diverse range of wireless network conditions to realistically model interference and self-interference. We also performed a data preparation step using the Gaussian mixed model approach to estimate the noise threshold and make a decision on the availability of each channel. This helped ensure that the data used in our simulations accurately reflected the wireless environment in which the proposed technique would be applied.

The use of a spectrum analyzer enabled us to measure the signal strength of existing Wi-Fi signals and identify potential sources of interference. We also recorded the duration of each detected signal, enabling the accurate characterization of the wireless environment's activity over time. The data collection and preparation are depicted in Algorithm 1.

Algorithm 1 Data Collection and Preparation Algorithm.

Data: Raw signal data from spectrum analyzer.
Result: Interference map for the 2.4 GHz ISM band.
for *channel* in 2.4 GHz ISM band **do**
 Collect raw data for channel using the spectrum analyzer;
 Save the data using the GPIB interface;
 Use a Gaussian mixed model to estimate the noise threshold for the channel;
 if Wi-Fi signals above threshold **then**
 | Mark the channel as busy;
 end
 else
 | Mark the channel as empty;
 end
end
Generate an interference map based on channel availability decisions;

The above algorithm summarizes the key steps involved in our data collection and preparation processes. We collected 40,000 data instances for the 40 channels in the 2.4 GHz ISM band using a spectrum analyzer and recorded the collected data using the GPIB interface. We then analyzed the collected data to determine the noise threshold for each channel using a Gaussian mixed model approach, ensuring a high accuracy. Based on the estimated noise threshold, we determined the channel availability, with a channel being considered busy or empty depending on the presence or absence of Wi-Fi signals above the estimated noise threshold. We generated an interference map based on these channel availability decisions, providing an estimate of the interference experienced in the 2.4 GHz ISM band and enabling the simulation of practical applications.

Overall, the data collection step provided a rich set of data, which was utilized in our simulation study to realistically model the wireless environment. We believe that the use of practical measurements and GPIB data storage ensures the accuracy and reliability of our dataset, providing a valuable foundation for the simulation study's subsequent data preparation and analysis steps.

Therefore, by basing the simulations on actual measurements of existing Wi-Fi signals and using a well-established statistical approach to evaluate the decision threshold, the proposed optimized chaotic frequency hopping technique became more promising. These measures helped to establish a more reliable basis for evaluating the effectiveness of the proposed technique in practical applications.

1.2. Research Methodology

The system model that includes four tested BLE frequency hopping techniques, three performance metrics for each technique, and the proposed optimized chaotic technique is depicted in Figure 1.

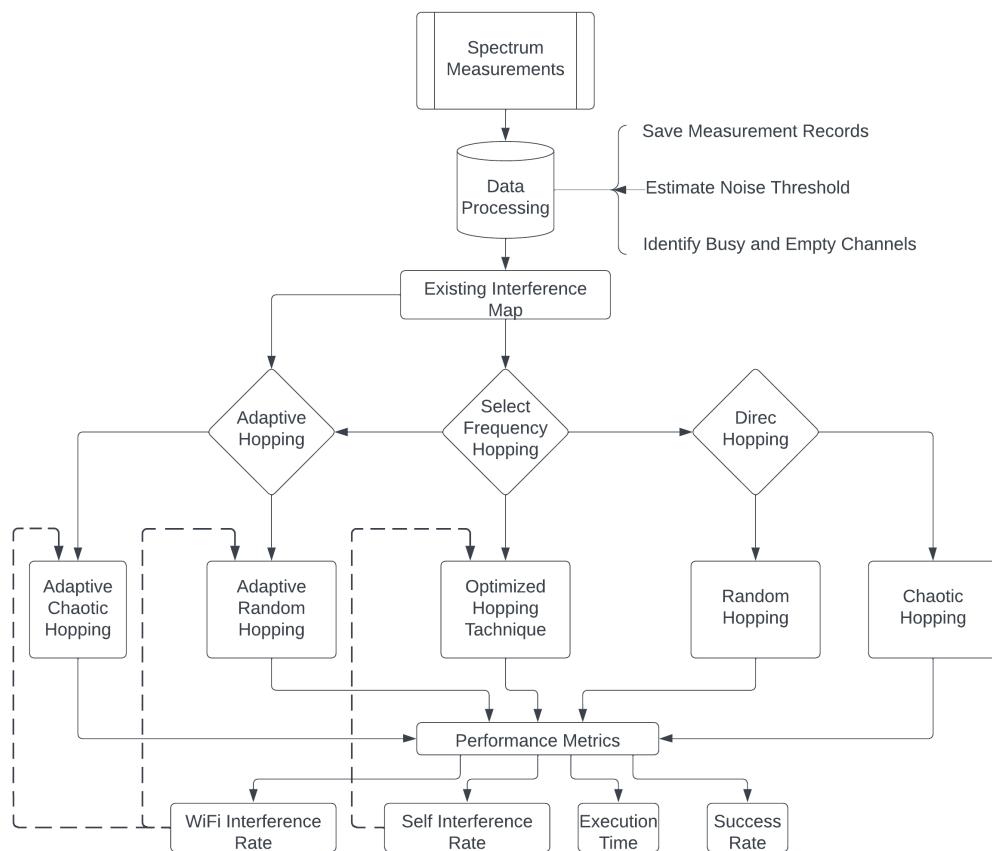


Figure 1. Research methodology.

The research methodology was based on measuring the signal power for five working days using a spectrum analyzer and estimating the noise threshold using a Gaussian mixed model. An existing interference map was then produced based on the threshold and four BLE frequency hopping techniques were simulated. For each implemented technique, the execution time was tested, as well as the Wi-Fi and BLE self-interference rates, and successfully connected the BLE nodes rates. The performance metrics for each frequency hopping technique were then compared based on these factors. Finally, an optimized chaotic frequency hopping technique was proposed based on the results of the performance comparison, and the selection among the tested techniques was made based on desired performance criteria. By including multiple frequency hopping techniques and testing them across three relevant metrics, this methodology ensured that the proposed optimized technique was well-informed and thoroughly tested against other frequency hopping techniques.

2. Bluetooth and Wi-Fi Coexistence

Bluetooth and Wi-Fi are popular wireless technologies used in consumer electronics that share the 2.4 GHz ISM band for RF transmission. However, they differ in their transmission methods, with Bluetooth using a slotted transmission approach for power efficiency while Wi-Fi continuously transmits. This creates interference problems as Wi-Fi signals look like noise to Bluetooth receivers, and vice versa [15].

IoT gateways and hubs can benefit from supporting multiple wireless protocols, such as Wi-Fi, Zigbee, Thread, and Bluetooth, to accommodate a diverse range of end nodes. However, since these protocols use the same 2.4 GHz ISM band, performance degradation can occur when multiple radios operate simultaneously. Wi-Fi coexistence strategies can help mitigate these issues and minimize the interference between collocated radios.

Wi-Fi coexistence enables multiple 2.4 GHz technologies, including Wi-Fi, Zigbee, Thread, and Bluetooth, to operate simultaneously without interfering with each other's signals. Interference can lead to a reduced device responsiveness and increased power consumption due to message failures and retries. However, achieving coexistence is more challenging with current gateways and hubs, which support a higher throughput, transmit at higher power levels, and integrate up to 4 2.4 GHz radios, requiring additional steps to ensure a reliable wireless performance.

The 2.4 GHz ISM band frequency plan is illustrated in Figure 2, with Wi-Fi operating on 3 non-overlapping 20 MHz channels, and Bluetooth operating on up to 40 BLE channels with 2 MHz spacing.

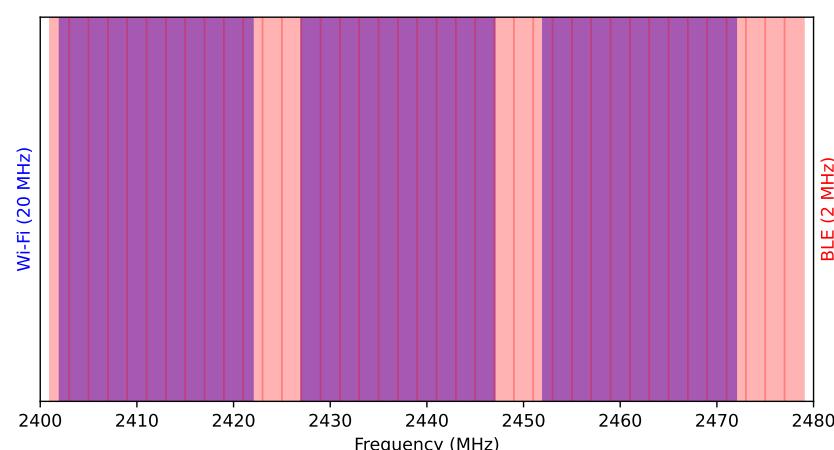


Figure 2. Frequency Plan for 2.4 GHz ISM band.

A Wi-Fi channel is 20 MHz wide, while a BLE channel is 2 MHz wide, allowing approximately 10 BLE channels to fit into the range of a Wi-Fi channel. Of the 40 Bluetooth

channels available for the Frequency Hopping Spread Spectrum (FHSS), 10 (25%) appear within the same frequency space as a given Wi-Fi channel, as depicted in Figure 2.

For example, in a situation where a device is using collocated Wi-Fi and Bluetooth with both active simultaneously, such as making a Wi-Fi phone call while using a Bluetooth headset, a Bluetooth channel hopping into the middle of a transmitting Wi-Fi packet can corrupt it. After a few such instances, the Wi-Fi transmitter will back off to a lower speed, resulting in more time to deliver a complete packet and a decreased throughput. The probability of another collision with that packet is 25%. Should that collision happen, the Wi-Fi will further reduce the throughput. Similarly, if the Wi-Fi signal corrupts the Bluetooth packet, Bluetooth will hop to the next channel and try again if the Bluetooth transmission is asynchronous, resulting in a reduced throughput.

The coexistence of Wi-Fi and Bluetooth in the 2.4 GHz ISM band requires careful consideration and implementation of strategies to minimize interference and maximize wireless performance [12].

2.1. The Wi-Fi Physical Layer (PHY)

The 802.11a/g/n/ac Wi-Fi signal model is commonly employed in contemporary Wi-Fi systems. It utilizes Orthogonal Frequency Division Multiplexing (OFDM) modulation, which divides the frequency band into multiple subcarriers and applies distinct modulation schemes and coding rates for each subcarrier [16]. The 802.11a/g/n/ac model comprises various parameters, such as:

- Carrier frequency: This is the center frequency of the Wi-Fi signal, typically in the range of 2.4 GHz to 5 GHz.
- Channel bandwidth: This is the width of the frequency band used by the Wi-Fi signal, typically 20 MHz, 40 MHz, or 80 MHz.
- Subcarrier spacing: This is the frequency spacing between adjacent subcarriers, typically 312.5 kHz.
- Modulation scheme: This determines how data are mapped onto the subcarriers, typically using Quadrature Amplitude Modulation (QAM) with different constellation sizes (e.g., 16-QAM, 64-QAM, 256-QAM).
- Coding rate: This determines how much redundancy is added to the data to improve error correction, typically using convolutional or Reed-Solomon codes with different code rates (e.g., 1/2, 3/4, 5/6).
- Guard interval: This is a time interval inserted between adjacent OFDM symbols to prevent inter-symbol interference, typically 800 ns or 400 ns.

Various characteristics of Wi-Fi signals, such as data rates, susceptibility to noise and interference, and resistance to channel fading and multipath propagation, can be modeled by adjusting the parameters of Wi-Fi signal models such as the 802.11a/g/n/ac model. The Wi-Fi physical layer in the 2.4 GHz band incorporates preambles and headers in its signal structure to aid in signal detection and synchronization. The preamble establishes the timing and frequency of the signal, while the header contains information about the data type being transmitted [17].

2.2. Bluetooth Physical Layer Signal

The 2.4 GHz physical layer of Bluetooth adopts frequency hopping spread spectrum (FHSS) technology to minimize the interference from other devices and provide high-level security. FHSS is a radio transmission technology that uses rapidly changing frequencies. In BLE, data are transmitted over 40 frequency channels in the 2.4 GHz ISM band with a 2 MHz bandwidth each. The hopping sequence is established through a master-slave communication model where the master device determines the hopping pattern, while the slave devices follow it [18].

To encode data, the Bluetooth physical layer employs a Gaussian Frequency Shift Keying (GFSK) modulation, which involves slightly shifting the frequency of the carrier signal. The amount of frequency shift is proportional to the amplitude of the data signal.

Additionally, the Bluetooth physical layer uses a preamble and header similar to Wi-Fi to aid in synchronization and signal detection. The preamble is used to establish the timing and frequency of the signal, while the header carries information about the transmitted data type.

3. Bluetooth LE Channel Selection Algorithms

Effective channel selection is essential to ensure the quality and reliability of Bluetooth LE communication. Numerous algorithms have been proposed to tackle this challenge with the goal of optimizing communication quality and minimizing interference. During the assignment of a carrier frequency to a BLE signal, it may either collide with an existing Wi-Fi signal or be assigned to a vacant channel depending on certain selection criteria that are discussed later. Some of the vacant channels may be unoccupied despite an interference event between the BLE and Wi-Fi signals, as depicted in Figure 3.

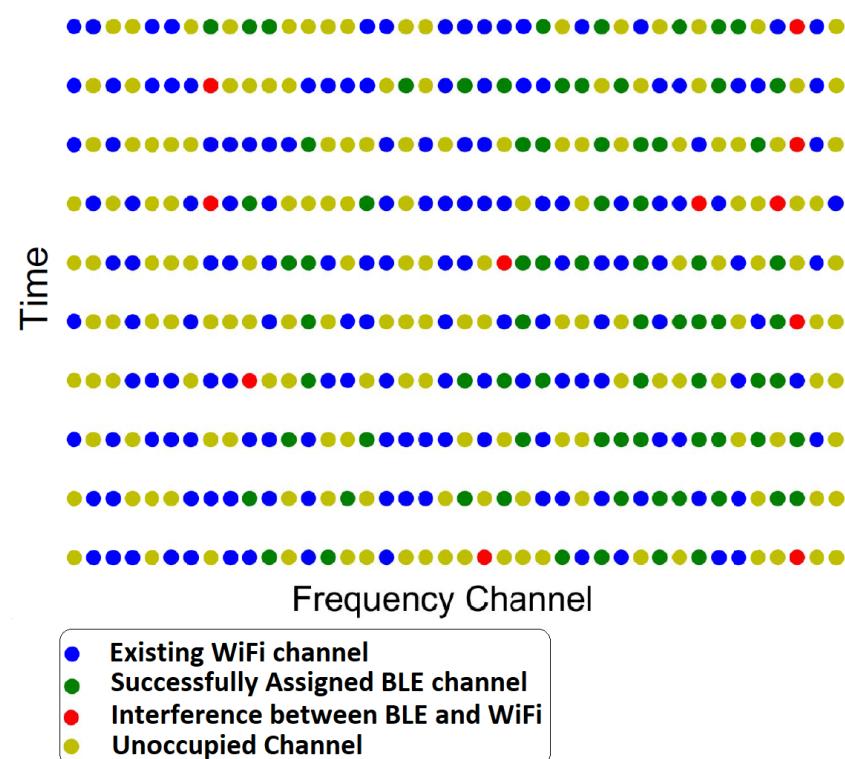


Figure 3. BLE Frequency Hopping to Coexist with Wi-Fi.

There are three main categories of Bluetooth LE channel selection algorithms: fixed, dynamic, and adaptive. While fixed channel selection is a straightforward approach, it may not be suitable in environments with high levels of interference. In contrast, dynamic and adaptive channel selection algorithms can adjust to changing interference conditions in real-time, resulting in improved communication quality and reliability [1]. Studies have extensively analyzed the performance of various channel selection algorithms based on metrics such as communication quality, reliability, and energy efficiency. These investigations demonstrate that the selection of a particular algorithm can significantly affect the overall performance of the Bluetooth LE communication [13].

To mitigate the impact of interference and avoid collision, Bluetooth Low Energy uses a frequency hopping function that distributes the transmission signal over several frequencies, thereby reducing the chance of signal collisions.

3.1. Random Frequency Hopping

Random frequency hopping is a technique that aims to improve the security of wireless transmissions by preventing interference. It involves rapidly switching between different radio frequencies in a random manner during transmission.

The main concept behind random frequency hopping is that by constantly changing the transmission frequency, it becomes difficult for interfering signals or eavesdroppers to disrupt the communication. This technique is commonly used in military communications and other applications where secure and reliable wireless communication is crucial [14].

Frequency selection in traditional frequency hopping is based on a well-known pseudo-random sequence. As these sequences are fixed, they are not capable of adapting to the current state of the wireless environment. This can lead to frequency overlaps and channel congestion, impacting the signal quality and throughput.

Let f_{base} represent the base frequency of the BLE signal and $f_{offsets}$ denote a set of offset frequencies. The set of valid frequencies for the random frequency hopping scheme is defined as $F = \{f_{base} + f_{offsets}\}$, where $+$ represents the set union operation. The length of F is typically limited to a small number of channels (e.g., 37 for BLE).

At each time slot, a random frequency is chosen from F and the next hop occurs at the selected frequency. The randomness can be achieved by employing a pseudo-random number generator (PRNG) to obtain a random index in F .

Assuming N is the length of the hopping sequence, $h_n \in F$ represents the frequency used for the n th time slot, $1 \leq n \leq N$ where the sequence can be defined as:

$$h_n = f_{base} + f_{offsets}_{r(n)} \quad (1)$$

where $r(n)$ is a pseudo-random number generator (PRNG) that maps values in the range $1, \dots, |F|$ to a specific value in F for each time slot (n).

The pseudo-random mapping function should be secure and unpredictable to prevent attackers from predicting the hopping sequence. Common examples of PRNGs include the Linear Congruential Generator (LCG) or the Mersenne Twister PRNG [19].

While both LCG and Mersenne Twister generate random-looking sequences of numbers, they differ in various ways, including speed, period length, randomness, and security.

LCG is a straightforward and fast PRNG that generates a sequence of numbers through a linear recurrence relation. It has the following form:

$$x_{n+1} = (ax_n + c) \bmod m \quad (2)$$

where x_n is the n th generated number, a is the multiplier, c is the increment, m is the modulus, and \bmod is the modulo operation. LCGs are simple to implement and require only a small amount of memory. However, they have some significant drawbacks, including low randomness and predictability, which can make the frequency hopping sequence vulnerable to attacks.

On the other hand, Mersenne Twister PRNG is a much more sophisticated and secure PRNG with a much longer period length. It is based on a twisted generalized feedback shift register that can generate a sequence of 32-bit integers. The Mersenne Twister PRNG offers better randomness and security than LCG due to its more complex design. The Mersenne Twister PRNG is also more computationally intensive, requiring more memory and processing power, but offers a much longer period length and better statistical randomness. Thus, it is simulated in this research for the frequency assignment of BLE signals.

Using random frequency hopping for frequency selection of BLE signals can be implemented by Algorithm 2.

Algorithm 2 BLE frequency hopping simulation and performance evaluation.

```

Input: frequencies from 0 to 39, communication attempts  $N = 40,000$ .
Output: Assigned map of Bluetooth signals.
Define  $M_s = 40$ ;
for each  $M$  in  $M_s$  do
    Generate a random  $M$ -dimensional vector  $x$  of floats between 0 and 1;
    for each row  $i$  in hop_sequence do
        for each column  $j$  in hop_sequence do
            Choose a random frequency from frequencies with replacement and
            assign it to hop_sequence[ $i, j$ ];
            Update  $x[j]$  by calling the function random_hopping;
        end
    end
    for each row  $i$  in interference_map do
        for each column  $j$  in interference_map do
            if frequencies[ $j$ ] == hop_sequence[ $i, k$ ] then
                | interference_map[ $i, j$ ] = 1;
            end
        end
    end
end

```

3.2. Chaotic Frequency Hopping

Chaotic frequency hopping is a technique that introduces chaos into the frequency hopping pattern to increase the security of wireless communications. This technique utilizes a chaotic map to generate the hopping sequence, which is used to select the frequency channel for each transmission [20].

A chaotic function in frequency hopping refers to a mathematical function that generates a random-like sequence of frequencies. In comparison to the previous pseudo-random sequence, these sequences are dynamic and continuously changing, adapting to the current state of the wireless environment. Chaotic sequences allow BLE devices to hop over previously used frequencies, enabling diversified transmissions and mitigating interference effects.

Various chaotic maps can be used for frequency hopping, including the logistic map, the Hénon map, and the Lorenz system. Each of these maps has different properties and can be used to generate different types of chaotic sequences. A comparison of different chaotic frequency hopping techniques is presented in Table 1.

Table 1. Comparison of Chaotic Frequency Hopping Techniques.

Technique	Advantages	Disadvantages	Complexity
Logistic Map	Easy to implement	Sensitivity to initial conditions	Low
Hénon Map	High degree of randomness	Complex to implement	Moderate
Lorenz System	Highly complex behavior	Difficult to implement	High
Hybrid Techniques	High degree of randomness and security	Complex to implement	High

In general, using chaotic frequency hopping can be a beneficial technique to enhance wireless communication security but it can also pose implementation challenges and increase system complexity. Therefore, it is crucial to carefully weigh the benefits and drawbacks of this approach. However, the logistic map is a simple chaotic map that can effectively generate random sequences, which makes it a potential option for wireless communication systems seeking to implement chaotic frequency hopping.

The logistic map is a mathematical function that is often used as an example of how chaotic behavior can arise from a simple non-linear system. The map is defined by the following equation:

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n) \quad (3)$$

where x_n is the current value at time step n and r is a control parameter that determines the rate of change. The logistic map exhibits chaotic behavior for certain values of r , resulting in an unpredictable and aperiodic sequence of values of x_n .

To apply the logistic map to frequency hopping, we first map the generated values of x_n to the frequency range we wish to hop over. This can be accomplished by defining the range of valid frequencies as $F = \{f_1, f_2, \dots, f_m\}$, with m being the number of available channels, and mapping the frequency values x_n to this valid range by:

$$h_n = F(x_n) = f_{\lfloor (x_n \times (m-1)) + 1 \rfloor} \quad (4)$$

where h_n is the corresponding frequency selected at time step n and $\lfloor \cdot \rfloor$ is the floor function used to round the value of $(x_n \times (m-1)) + 1$ to the nearest integer.

The logistic map chaotic frequency hopping has the benefit of flexibility and adaptiveness, as the hopping sequence is dependent on the current state of the wireless environment. Attackers cannot predict the hopping sequence since it is generated from the chaotic behavior of the logistic map; therefore, it is much more secure than traditional hopping methods.

The logistic map chaotic frequency hopping for frequency selection of BLE signals can be implemented using Algorithm 3.

Algorithm 3 Bluetooth Signal Assignment using Chaotic Frequency Hopping.

Input: $n_channels$: number of channels.
 n_slots : number of slots per channel.
 $n_signals$: number of Bluetooth signals.
 r : parameter for logistic map.
Output: Assigned map of Bluetooth signals.

```

interference_map ← random  $n\_channels \times n\_slots$  array;
x ← random number between 0 and 1;
channel_sequence ← empty list;
for i ← 1 to  $n\_slots \times n\_signals$  do
    x ← logistic_map(x, r);
    channel_index ← integer part of  $x \times n\_channels$ ;
    append channel_index to channel_sequence;
end
assigned_map ←  $n\_channels \times n\_slots$  array filled with zeroes;
for i ← 1 to  $n\_signals$  do
    start_slot ← random integer between 0 and  $n\_slots - 1$ ;
    for j ← 1 to  $n\_slots$  do
        slot_index ← ( $start\_slot + j - 1$ ) mod  $n\_slots$ ;
        if assigned_map[channel_sequence[(i - 1) ×  $n\_slots + j$ ]][slot_index] == 0
            then
                assigned_map[channel_sequence[(i - 1) ×  $n\_slots + j$ ]][slot_index] ← i;
            end
        else
            assigned_map[channel_sequence[(i - 1) ×  $n\_slots + j$ ]][slot_index] ←
                -1;
            end
        end
    end
end

```

The logistic map has a variety of properties that make it useful in many applications. One example is its ability to generate pseudo-random numbers, which is useful in cryptography and other areas where randomness is important.

In the context of chaotic frequency hopping, the logistic map can be used to create a sequence of pseudo-random numbers that determine the frequency hopping pattern. Using a unique initial condition for each transmission, a sequence of frequencies that appears to be random and unpredictable can be generated, making it difficult for an attacker to intercept the communication.

However, it is important to note that the logistic map also has some limitations. For instance, it can be sensitive to initial conditions, which means that even small changes in the initial state of the system can result in significantly different outcomes. Additionally, the map can exhibit period-doubling bifurcations, which can make the behavior of the system more challenging to anticipate.

3.3. Adaptive Frequency Hopping

Adaptive frequency hopping is a technique used in wireless communication systems to dynamically adjust the hopping sequence based on the interference and noise conditions of the channel. The idea is to monitor the channel quality and adapt the hopping sequence to avoid the frequency bands with high interference or noise [21].

The adaptive frequency hopping technique typically involves several steps. First, the quality of the channel is monitored to detect the frequency bands with high interference or noise. This can be performed by measuring the signal-to-interference-plus-noise ratio (SINR) or other quality metrics. Then, a new hopping sequence is selected that avoids those frequency bands. This can be performed by randomly selecting a new sequence that avoids the problematic frequency bands or using a predefined algorithm to generate the new sequence. The new hopping sequence is updated in the system and used for future transmissions. The system continues to monitor the channel quality and adapt the hopping sequence as needed [22].

The advantage of adaptive frequency hopping is that it can improve the reliability and performance of the wireless communication system in the presence of interference and noise. It can also increase the security of the system by making it more difficult for eavesdroppers to intercept the transmitted data. Additionally, the use of the logistic map can simplify the implementation of adaptive frequency hopping by generating a pseudo-random sequence that avoids problematic frequency bands. The disadvantage is that it requires additional hardware and software to implement, and can increase the complexity and cost of the system [23].

The use of adaptive frequency hopping in wireless communication systems is becoming increasingly popular due to its ability to dynamically adjust the hopping sequence based on interference and noise conditions. There are several adaptive frequency hopping techniques that can be used, and a comparison of some of the most common techniques is as follows.

The threshold-based technique involves setting a threshold value for the interference or noise level and changing the hopping sequence if the measured interference or noise level exceeds the threshold. The advantage of this technique is its simplicity and minimal computational resource requirements. However, it may not be effective in all situations, as interference and noise levels can vary widely.

The learning-based technique involves using machine learning algorithms to learn the interference and noise patterns in the channel and adapt the hopping sequence accordingly. The advantage of this technique is that it can adapt to a wide range of interference and noise conditions and improve the performance over time. However, it requires significant computational resources and may not be practical for all systems.

The distributed technique involves using a distributed approach to adapt the hopping sequence, where each node in the network monitors the channel quality and communicates with other nodes to exchange information and adapt the hopping sequence. The advantage

of this technique is that it can be more robust and adaptive to changing conditions than centralized approaches. However, it requires more communication and coordination between nodes, which can increase the overhead and complexity of the system.

The hybrid technique combines multiple adaptive frequency hopping techniques, such as threshold-based and learning-based approaches, to improve the overall performance of the system. The advantage of this technique is that it can leverage the strengths of each approach and adapt to a wide range of interference and noise conditions. However, it can be more complex and require more computational resources than single-technique approaches.

The choice of the adaptive frequency hopping technique depends on the specific requirements and constraints of the wireless communication system. Threshold-based techniques may be suitable for simple systems with low interference and noise levels, while learning-based and distributed techniques may be more appropriate for complex systems with high interference and noise levels. Hybrid techniques may offer the best performance and adaptability, but require more resources and complexity to implement.

4. Proposed Algorithm

A proposed chaotic frequency hopping technique based on optimizing the update coefficient r to minimize interference and ensure zero self-interference among hopping BLE nodes could work as follows:

1. Define a set of N hopping frequencies to be used by the BLE nodes.
2. Initialize the update coefficient r to a random value between 3.6 and 4.0.
3. For each BLE node, generate a chaotic sequence using the logistic map with the current value of r . This sequence will be used to select the hopping frequencies for that node.
4. Calculate the interference matrix I of size $N \times N$ for the current set of hopping frequencies, where $I_{i,j} = 1$ if frequency i interferes with frequency j and $I_{i,j} = 0$ otherwise.
5. For each BLE node, calculate the interference caused by that node on all other nodes using the interference matrix I . If the node is interfering with itself, adjust its chaotic sequence to select a different frequency.
6. Update the value of r using an optimization algorithm such as gradient descent or simulated annealing to minimize the interference among the nodes. Repeat steps 3–6 until interference is minimized and self-interference is eliminated.

This technique uses chaos theory to generate random and unpredictable hopping sequences for each BLE node, which helps to avoid predictable patterns and reduce interference. By optimizing the update coefficient r , the chaotic behavior can be fine-tuned to minimize interference and ensure zero self-interference among the nodes. However, this approach may require significant computational resources and may be more complex to implement than simpler frequency hopping techniques. The proposed algorithm is implemented as depicted in the flowchart in Figure 4.

To compare among the four frequency hopping techniques (random, chaotic, adaptive, and proposed), we use the following criteria:

- Interference reduction: This technique should reduce the amount of interference among hopping nodes.
- Self-interference avoidance: This technique should ensure that each hopping node does not interfere with itself.
- Frequency diversity: This technique should use a wide range of frequencies to avoid congested frequency bands.
- Synchronization: This technique should ensure synchronization among hopping nodes to avoid collisions.
- Adaptivity: This technique should adapt to changes in the environment, such as the presence of new devices or interference sources.
- Complexity: This technique should be simple and easy to implement.

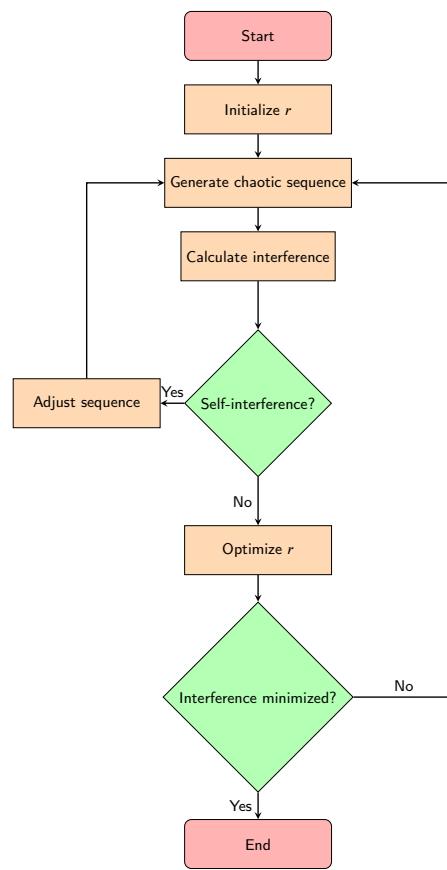


Figure 4. Flowchart of the proposed chaotic frequency hopping technique.

Based on these criteria, we compare the four techniques as follows in Table 2.

Table 2. Comparison of frequency hopping techniques.

Criteria	Random	Chaotic	Adaptive	Proposed
Interference reduction	Moderate	Low	High	High
Self-interference avoidance	Low	Low	High	High
Frequency diversity	Moderate	High	High	High
Synchronization	Low	Low	Moderate	High
Adaptivity	Low	Low	High	High
Complexity	Moderate	Simple	Complex	Moderate

From Table 2, we see that the proposed technique offers the highest levels of interference reduction, self-interference avoidance, frequency diversity, synchronization, and adaptivity, while still being relatively simple to implement. The chaotic and adaptive techniques also offer high levels of interference reduction and frequency diversity, but they may be more complex to implement. The random technique is the simplest to implement but offers low levels of interference reduction and frequency diversity.

5. Simulation Results

To simulate the effect of Bluetooth interference on Wi-Fi signals, Bluetooth and Wi-Fi signals are simulated according to the following parameters.

First, set up the Wi-Fi signal simulation parameters, including the number of packets = 1000, the number of bits per packet = 1000 and SNR values ranging from 0 to 30 dB. The Wi-Fi carrier frequencies are chosen among the three mentioned non-overlapping channels, (2412, 2437, 2462) each with 20 MHz bandwidth. The first carrier frequency is chosen for simulation. Then, the Bluetooth signal is simulated with a hopping carrier frequency within the

10 available channels to coexist with the simulated Wi-Fi signal with 2 MHz bandwidth. The Bluetooth packets are simulated via 1600 time slots per packet.

For the effect of interference on the bit error rate or packet error rate of a Wi-Fi signal we used an AWGN channel model and Monte Carlo simulation to estimate the error rates for different levels of interference as a function of the Bluetooth carrier frequency location within the Wi-Fi signal bandwidth, as well as the interference level and duration with respect to the Wi-Fi packet. The error vector magnitude (EVM) is a measure of how accurately the demodulated signal matches the original transmitted signal. It is typically expressed as a percentage or in decibels (dBs).

A high EVM indicates that the demodulated signal has a high level of distortion or error compared to the original transmitted signal. This can lead to increased bit error rates and a reduced system performance. A low EVM, on the other hand, indicates that the demodulated signal closely matches the original transmitted signal and that the system is performing well.

Figure 5 depicts the effect of the Bluetooth carrier frequency location within the Wi-Fi signal bandwidth on the Wi-Fi signal in terms of the EVM.

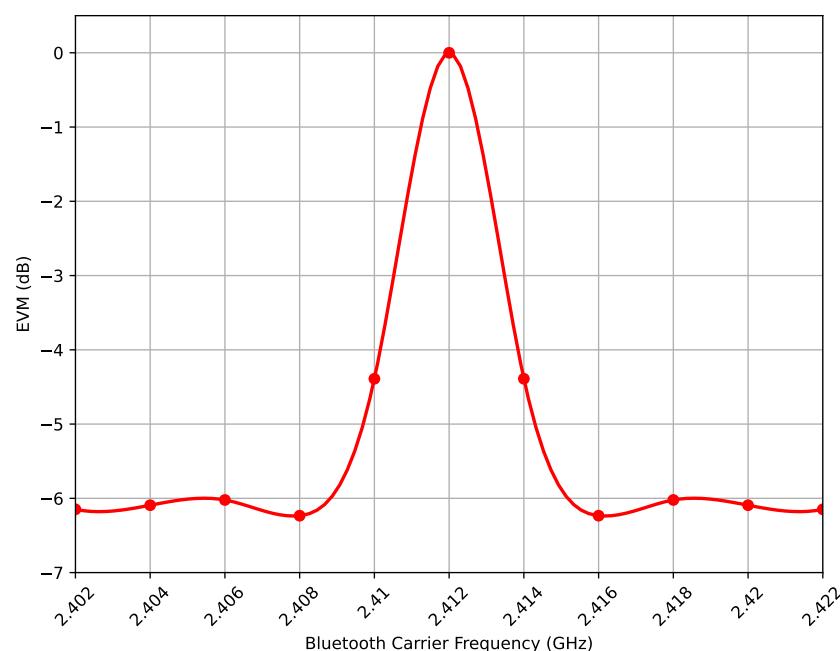


Figure 5. BLE Frequency Hopping to Coexist with Wi-Fi.

As the carrier frequency of the interfering BLE signal is near to the Wi-Fi carrier frequency, the EVM increases and vice versa. If the BLE carrier frequency is equal to the Wi-Fi carrier frequency, the $EVM = 0$ dB, indicating a total Wi-Fi packet loss. Thus, in the upcoming simulations, an interference is only considered if the BLE and Wi-Fi signals have the same carrier frequency.

Frequency hopping techniques for BLE signals are simulated for minimal interference with Wi-Fi signals; the simulation setup is illustrated in Algorithm 4.

Different types of frequency hopping algorithms represented in earlier sections are compared with the proposed algorithm to compare the existing algorithms in terms of interference avoidance and success rate, in addition to the computational complexity. In this section, we presented the results of our simulations for the different types of frequency hopping techniques that we studied, including random, chaotic, adaptive random, and adaptive chaotic. We also proposed an optimized chaotic frequency hopping technique that outperforms the other techniques in terms of reducing interference with Wi-Fi signals and achieving a high success rate for connecting Bluetooth nodes.

Algorithm 4 BLE frequency hopping simulation and performance evaluation.

Input: frequencies from 0 to 39, communication attempts $N = 40,000$.
Output: A CSV file containing the performance measures.
Define $M_s = 40$;
for each M in M_s **do**
 Generate a random M -dimensional vector x of floats between 0 and 1;
 for each row i in hop_sequence **do**
 for each column j in hop_sequence **do**
 Choose a random frequency from frequencies with replacement and
 assign it to $hop_sequence[i, j]$;
 Update $x[j]$ by calling the function random_hopping;
 end
 end
 for each row i in interference_map **do**
 for each column j in interference_map **do**
 if $frequencies[j] == hop_sequence[i, k]$ **then**
 $interference_map[i, j] = 1$;
 end
 end
 end
 Write $interference_map$ to a CSV file “interference_map.csv”;
 Calculate various performance measures such as the number of successful BLE
 signal transmissions, the number of BLE transmissions that interfere with an
 ongoing Wi-Fi communication, and the self-interference count;
 Store the performance measures in $performance_measures$;
end
 Write the $performance_measures$ list to a CSV file;

We conducted simulations in an environment that includes existing interference from Wi-Fi signals and interfering Bluetooth nodes. We compared several performance metrics, including the total interference rate, total successful connection rate, and trail execution time for channel selection processing time. These metrics were plotted for different numbers of accessing Bluetooth nodes.

The total interference rates for different frequency hopping techniques are depicted in Figure 6.

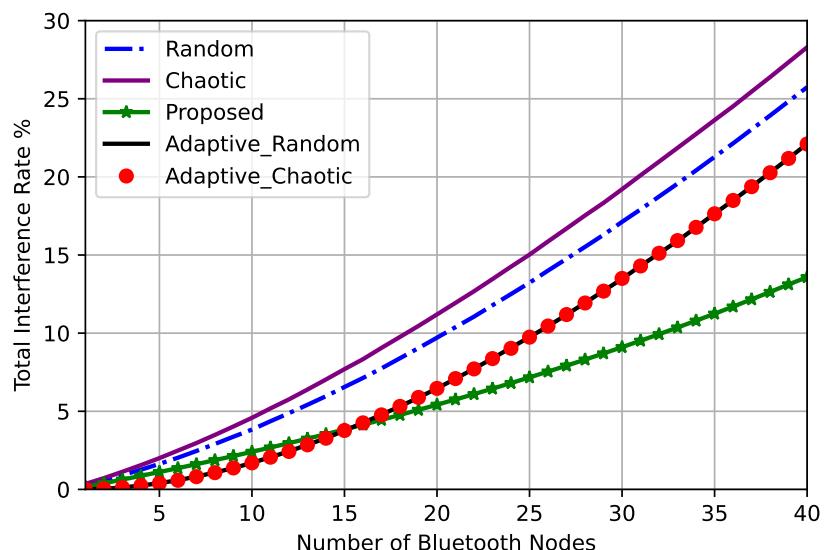


Figure 6. Total interference rates for different frequency hopping techniques.

It is obviously observed that the total interference rate increases as the number of accessing Bluetooth nodes increases for all frequency hopping techniques. The chaotic frequency hopping technique had the highest total interference rate compared to the other techniques, while the adaptive frequency hopping technique had a lower interference rate than both the chaotic and random techniques, as shown in Figure 6. However, our proposed optimized chaotic frequency hopping technique had a slightly higher interference than that of the adaptive technique for small numbers of Bluetooth nodes. As the number of Bluetooth nodes increased, our proposed technique had a much lower interference than the adaptive technique.

The total successful connection rate was defined as the number of successfully assigned frequencies to the Bluetooth nodes without interfering with the Wi-Fi signals. The total successful connection rate comparison for different frequency hopping techniques is shown in Figure 7.

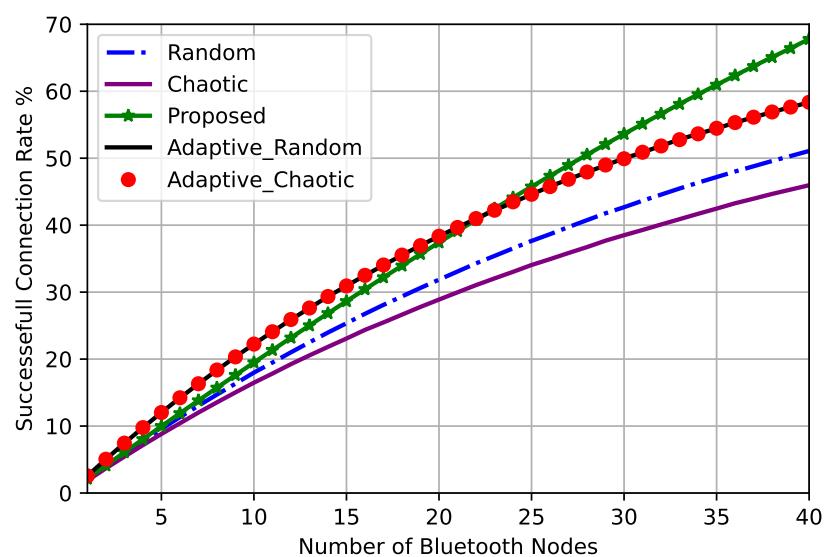


Figure 7. Total successful connection rates for different frequency hopping techniques.

It is clear that the total successful connection rate increases as the number of accessing Bluetooth nodes increases for all frequency hopping techniques. The chaotic frequency hopping technique had the lowest success rate compared to the other techniques, while the adaptive frequency hopping technique had a higher success rate than both chaotic and random techniques, as shown in Figure 7. However, the proposed optimized chaotic frequency hopping technique demonstrated a superior performance compared to all other techniques, including the adaptive technique, particularly when the number of Bluetooth nodes was large. However, for a node count of up to 20, the proposed technique did not outperform the adaptive technique.

The Mersenne Twister PRNG and the logistic map have different computational complexities, which can be compared using Big O calculations.

The Mersenne Twister PRNG has a time complexity of $O(n)$ for generating n pseudo-random numbers. This complexity is due to the initialization process that sets the initial state of the PRNG. Once the initialization is complete, generating each number takes a constant time. Therefore, the computational complexity of generating N random numbers using the Mersenne Twister is $O(N)$, i.e., linear time, while the logistic map is a nonlinear function in which the current value is calculated based on the previous value. The time complexity of a single iteration of the logistic map is $O(1)$, as it only involves basic arithmetic operations.

The trial execution time is also measured for the channel selection processing time. The trial execution time comparison for different frequency hopping techniques is shown in Figure 8.

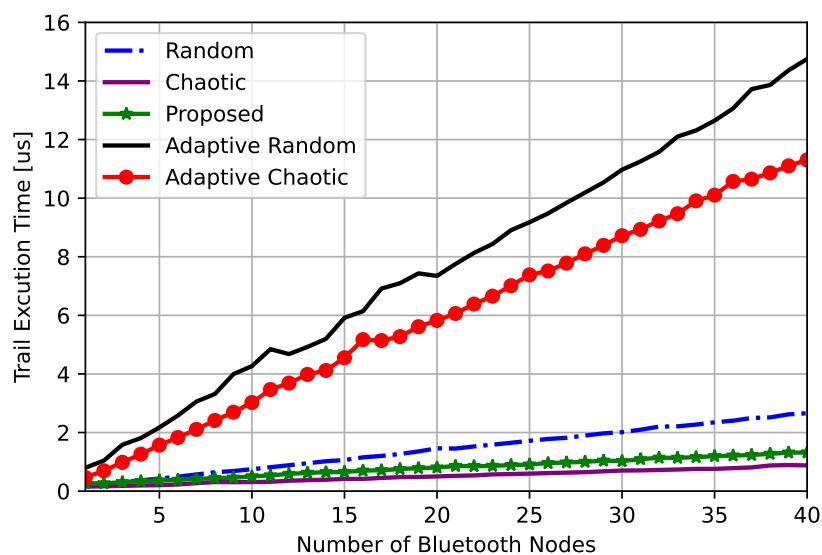


Figure 8. Trial execution time comparison for frequency hopping techniques.

It is found that the chaotic frequency hopping technique had the lowest execution time, demonstrating the net advantage of chaotic techniques due to its simplicity, as shown in Figure 8. The adaptive techniques required much longer execution times, with the adaptive random technique requiring the highest among all. The proposed technique required a very close time to that of the conventional chaotic technique, which was much shorter than that of all other techniques.

Overall, our results demonstrated that the proposed optimized chaotic frequency hopping technique achieved a better balance between reducing interference with Wi-Fi signals, achieving a high success rate for connecting Bluetooth nodes, and requiring minimal trial execution times, making it a suitable technique for managing interference in wireless communication systems.

It is worth mentioning that both adaptive techniques have the same interference and success rates, while the adaptive chaotic hopping requires notably less execution time than that of the adaptive random technique.

6. Conclusions

In this study, we proposed an optimized chaotic frequency hopping technique for managing interference in wireless communication systems. We compared this technique with several other frequency hopping techniques, including random, chaotic, adaptive random, and adaptive chaotic. Our simulations were conducted in an environment that included existing interference from Wi-Fi signals and interfering Bluetooth nodes. Our results showed that the total interference rate increases as the number of accessing Bluetooth nodes increases for all frequency hopping techniques. The chaotic frequency hopping technique had the highest total interference rate compared to the other techniques, while the adaptive frequency hopping technique had a lower interference rate than both the chaotic and random techniques. However, our proposed optimized chaotic frequency hopping technique had a slightly higher interference than that of the adaptive technique for small numbers of Bluetooth nodes. As the number of Bluetooth nodes increased, our proposed technique had a much lower interference than the adaptive technique.

We also found that the total successful connection rate increases as the number of accessing Bluetooth nodes increases for all frequency hopping techniques. The chaotic

frequency hopping technique had the lowest success rate compared to the other techniques, while the adaptive frequency hopping technique had a higher success rate than both chaotic and random techniques. However, our proposed optimized chaotic frequency hopping technique outperformed all other techniques, including the adaptive technique, especially at large numbers of Bluetooth nodes.

Finally, we measured the trial execution time for the channel selection processing time. We found that the chaotic frequency hopping technique had the lowest execution time, demonstrating the net advantage of chaotic techniques due to its simplicity. The adaptive techniques required much longer execution times, with the adaptive random technique requiring the highest among all. The proposed technique required a very close time to that of the conventional chaotic technique, which was much shorter than that of all other techniques. Overall, our results demonstrated that the proposed optimized chaotic frequency hopping technique achieves a better balance between reducing interference with Wi-Fi signals, achieving a high success rate for connecting Bluetooth nodes, and requiring minimal trial execution time. This makes it a suitable technique for managing interference in wireless communication systems.

The paper makes several contributions in the field of wireless communication systems. First, it proposes a new optimized chaotic frequency hopping technique that minimizes interference with Wi-Fi signals while achieving a high success rate for connecting Bluetooth nodes. Second, it compares this technique with existing random, chaotic, and adaptive frequency hopping techniques using several performance metrics, such as interference rate, successful connection rate, and trial execution time. Third, it demonstrates through simulation that the proposed technique outperforms all other techniques, especially at large numbers of Bluetooth nodes. Fourth, it shows that the proposed technique requires minimal trial execution time, making it a suitable solution for managing interference in wireless communication systems. Overall, the paper contributes new knowledge and techniques that can enhance the performance of wireless communication systems in real-world environments with existing interference from other wireless signals.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Xu, T.; Zhang, M.; Zeng, Y.; Hu, H. Harmonious coexistence of heterogeneous wireless networks in unlicensed bands: Solutions from the statistical signal transmission technique. *IEEE Veh. Technol. Mag.* **2019**, *14*, 61–69. [[CrossRef](#)]
2. Hassan, W.A.; Jo, H.S.; Ikki, S.; Nekovee, M. Spectrum-Sharing Method for Co-Existence between 5G OFDM-Based System and Fixed. *IEEE Access* **2019**, *7*, 77460–77475. [[CrossRef](#)]
3. Nasser, A.; Al Haj Hassan, H.; Abou Chaaya, J.; Mansour, A.; Yao, K.C. Spectrum Sensing for Cognitive Radio: Recent Advances and Future Challenge. *Sensors* **2021**, *21*, 2408. [[CrossRef](#)] [[PubMed](#)]
4. Zhan, S.C.; Chou, C.T.; Chang, S.C. Auction-based Spectrum Sharing Among Heterogeneous Secondary Networks. In Proceedings of the 2015 IEEE International Conference on Communications (ICC), London, UK, 8–12 June 2015.
5. Abass, A.A.A.; Kumbhkar, R.; Mandayam, N.B.; Gajic, Z. WiFi/LTE-U coexistence: An evolutionary game approach. *IEEE Trans. Cogn. Commun. Netw.* **2019**, *5*, 44–58. [[CrossRef](#)]
6. He, H.; Shan, H.; Huang, A.; Cai, L.X.; Quek, T.Q.S. Proportional Fairness-Based Resource Allocation for LTE-U Coexisting with Wi-Fi. *IEEE ACCESS* **2017**, *5*, 4720–4731. [[CrossRef](#)]
7. Gao, B.; Yang, Y.; Tech, V. A Taxonomy of Coexistence Mechanisms for Heterogeneous Cognitive Radio Networks Operating in TV White Spaces. *IEEE Wirel. Commun.* **2012**, *19*, 41–48. [[CrossRef](#)]
8. Wang, Y.; Zhang, Q.; Wei, Z.; Kui, L.; Liu, F.; Feng, Z. Performance Analysis of Uncoordinated Interference Mitigation for Automotive Radar. *IEEE Trans. Veh. Technol.* **2023**, *72*, 4222–4235. [[CrossRef](#)]
9. Zhou, Z.; Jia, Y.; Chen, F.; Tsang, K.; Liu, G.; Han, Z. Unlicensed Spectrum Sharing: From Coexistence to Convergence. *IEEE Wirel. Commun.* **2017**, *24*, 94–101. [[CrossRef](#)]
10. Zhao, Z.; Wu, X.; Zhang, X.; Zhao, J.; Li, X.Y. ZigBee vs. WiFi: Understanding Issues and Measuring Performances of their Coexistence. In Proceedings of the 2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC), Austin, TX, USA, 5–7 December 2014.

11. Yang, J.; Hu, L.; Liu, P.; Zhou, R. Design and Implementation of a new kind of Chaotic Frequency Hopping Sequences. In Proceedings of the 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, Dalian, China, 12–14 October 2008.
12. Kosek-Szott, K.; Gozdecki, J.; Loziak, K.; Natkaniec, M.; Prasnal, L.; Szott, S.; Wagrowski, M. Coexistence Issues in Future WiFi Networks. *IEEE Netw.* **2017**, *31*, 86–95. [[CrossRef](#)]
13. Sun, Y.; Li, J.; Chen, J.; Wu, G. Performance Analysis on Interference and Frequency-to-Time Mapping Based Frequency Hopping Receiver. *IEEE Photonics J.* **2023**, *31*, 86–95. [[CrossRef](#)]
14. Kadhim, A.A.; Sadkhan, S.B. Cognitive Radio Performance Enhancement Based on Frequency Hopping System. In Proceedings of the 2020 1st Information Technology to Enhance E-Learning and Other Application (IT-ELA), Baghdad, Iraq, 12–13 July 2020; pp. 109–114. [[CrossRef](#)]
15. Bulusu, S.S.C.; Renfors, M.; Yli-Kaakinen, J.; Shaiek, H.; Roviras, D. Enhanced multicarrier techniques for narrowband and broadband PMR coexistence. *Trans. Emerg. Telecommun. Technol.* **2017**, *28*, e3056. [[CrossRef](#)]
16. Park, J.; Lee, E.; Park, S.H.; Raymond, S.S.; Pyo, S.; Jo, H.S. Modeling and Analysis on Radio Interference of OFDM Waveforms for Coexistence Study. *IEEE Access* **2019**, *7*, 35132–35147. [[CrossRef](#)]
17. Wagh, S.S.; More, A.; Kharote, P.R. Performance Evaluation of IEEE 802.15.4 Protocol under Coexistence of WiFi 802.11b. *Procedia Comput. Sci.* **2015**, *57*, 745–751. [[CrossRef](#)]
18. Maso, M.; Debbah, M.; Vangelista, L. A Distributed Approach to Interference Alignment in OFDM-Based Two-Tiered Networks. *IEEE Trans. Veh. Technol.* **2013**, *62*, 1935–1949. [[CrossRef](#)]
19. Matsumoto, M.; Nishimura, T. Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator. *ACM Trans. Model. Comput. Simul.* **1998**, *8*, 3–30. [[CrossRef](#)]
20. Chen, Y.J.; Wu, J.; Xu, H.; Long, M.; Bao, Y.F. A Novel Method for Designing Chaotic Frequency Hopping Sequence. *Dianxun Jishu/Telecommun. Eng.* **2010**, *50*, 24–27.
21. Tanbourgi, R.; Elsner, J.P.; Jäkel, H.; Jondral, F.K. Adaptive frequency hopping in ad hoc networks with rayleigh fading and imperfect sensing. *IEEE Wirel. Commun. Lett.* **2012**, *1*, 185–188. [[CrossRef](#)]
22. Lee, S.H.; Lee, Y.H. Adaptive frequency hopping for bluetooth robust to WLAN interference. *IEEE Commun. Lett.* **2009**, *13*, 628–630. [[CrossRef](#)]
23. Liang, W.; Liu, S.; Yang, Y.; Li, S. Research of Adaptive Frequency Hopping Technology in WIA-PA Industrial Wireless Network. In Proceedings of the 6th China Conference, CWSN 2012, Huangshan, China, 25–27 October 2012; pp. 248–262.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.