The Catch Me if You Can Signal: Mathematics of FHSS

Introduction

Have you ever wondered how all wireless systems, such as Bluetooth, Wi-Fi, PlayStation, baby monitors, radio, etc., work together simultaneously without interfering with or jamming one another? Well, I have, and if you have too, then this paper is made for you. To fully understand the technology behind these systems, it's important first to understand how it was invented.

While Hedy Lamarr is best known for her acting career, she always had a keen interest in science and technology and co-invented a radio guidance system called "frequency hopping" during World War II. At the time, radio-controlled torpedoes were rumored to be in development by the Allies. Lamarr was aware that these torpedoes could be easily jammed by the enemy, making them ineffective.

Radio jamming involves the intentional transmission of radio signals on the same frequency as another communication channel, with the goal of disrupting that communication. This posed a significant threat to the Navy. As a result, she believed that if the frequency used to control these torpedoes was constantly changing, it would be much more difficult for the enemy to jam the signal and stop the torpedo from reaching its target

Inspired by the piano, George Antheil and Hedy Lamarr used paper rolls to control changes in frequency. The fascinating thing is that in fully developed frequency hopping, the transmitter and receiver rapidly switch between different channels in a predetermined

sequence, which is like a secret code shared only between them. Lamarr and Antheil handed this idea over to the Navy, but it was sadly rejected at the time. It wasn't until the 1960s that their ingenious idea of frequency hopping was implemented in the modern technologies we use and rely on today.

Let's clarify what frequency hopping really means.
Frequency hopping is nothing more than the switching of channels during transmission to avoid interception of the message and prevent interference. For example, when you press a button, the controller transmits the signal by hopping across multiple frequencies, so your game doesn't lag, even if the room is full of Wi-Fi and Bluetooth devices. This is frequency hopping spread spectrum (FHSS) in action

But how is it decided what frequency to switch to next?
The change in the frequency is completely random and unprecedented. A pseudo-random sequence is generated that both the transmitter and the receiver know.

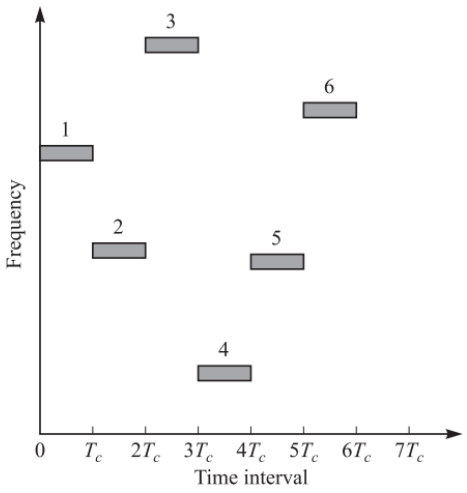

FIGURE 12.3–1
An example of a frequency-hopped (FH) pattern.

Whenever a signal, text, or audio is being transmitted to or from you, it is always broken up into small chunks and bits and then sent over different frequencies within a given time interval, as shown in the diagram.

This seemingly simple process has a lot of ideas behind it to ensure that our information is not jammed or interfered with. It involves math, but not as you see it. So, I will be exploring some of the mathematics behind this concept and showing just how reliable it is. Let's start our journey now!

1. Cosine Wave with Random Phase

Imagine a carrier wave with the signal:

$$\boldsymbol{s(t) = Acos(2\pi f_n t + \emptyset)}$$

Where A = amplitude (Higher amplitude, stronger signal)
f = frequency (number of cycles per second, how fast the signal oscillates)
t = time
Ø= phase shift (for simplicity's sake, the phase change is 0)

This cosine function plays a crucial role in demonstrating and explaining the properties of a carrier signal. Electrical Engineers use this wave function to generate the signal that hops frequencies over time. Chips like those in Bluetooth chips generate this cosine wave.

Let's take some values for a demonstration

$f_1, \ f_2, f_3$	$3, 6, 2$
A_1, (strength of signal may vary in real world communication system)	$2{,}3\ 1$
t	$0.1s$

$$s(t) = \begin{cases} 2 \times cos(2\pi \times 3t), 0 \leq t < 0.1, \\ 3 \times cos(2\pi \times 6t), 0.1 \leq t < 0.2 \\ 1 \times cos(2\pi \ \times \ 2t), 0.2 \leq t < 0.3 \end{cases}$$

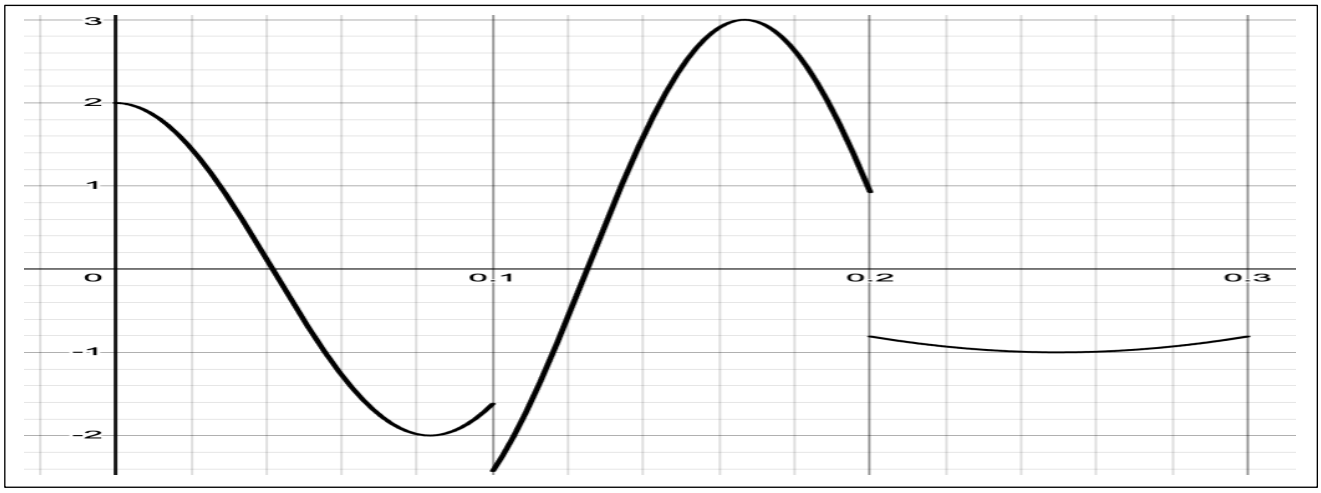

Figure 1: Wave

The wave allows engineers to visualize the carrier signal more clearly and analyze how the sinusoidal carrier's frequency changes over time as it hops across the available channels. This formula is a tool for defining and structuring FHSS, enabling devices to operate as they do.

2. Calculating the energy of each wave

Now, looking at the graph and the formula, we can see that for a signal, the power is given by:

$$p(t) = s^2(t)$$

$$p(t) = A^2 cos^2(2\pi f_n t)$$

It is known that the formula for energy is given by,

$$E_{hop} = \int_t^{T_h} p(t)dt$$

Where the duration of the hop is from t to T_h

$$E_{hop} = \int_t^{T_h} A^2 cos^2(2\pi f_n t)dt$$

Using double-angle identity,

$$cos^2(\theta) = \frac{cos(2\theta) + 1}{2}$$

$$E_{hop} = \int_t^{T_h} A^2 \frac{cos(4\pi f_n t) + 1}{2} dt$$

$$E_{hop} = \frac{A^2}{2} \int_t^{T_h} [cos(4\pi f_n t) + 1]dt$$

Split the integral into two parts,

$$E_{hop} = \frac{A^2}{2} \int_t^{T_h} cos(4\pi f_n t)dt \ + \int_t^{Th} 1dt$$

$$E_{hop} = \frac{A^2}{2} \int_t^{T_h} cos(4\pi f_n t)dt \ + \int_t^{Th} 1dt$$

First, we can integrate the sin term,

$$\int_t^{T_h} cos(4\pi f_n t)dt \ = \frac{sin(4\pi f_n T_h) - sin(4\pi f_n t)}{4\pi f_n}$$

Now, we can integrate the second term,

$$\int_t^{Th} 1dt = T_h - t$$

Now combine the terms,

$$E_{hop} = \frac{A^2}{2}\left(\frac{sin(4\pi f_n T) - sin(4\pi f_n t)}{4\pi f_n} + T_h - t\right)$$

Since we have derived the energy formula for each hop, we can calculate the individual energy,

$$\text{Hop 1: } \frac{4}{2}\left(\frac{sin(4\pi \times 0.3) - sin(0)}{12\pi} + 0.1\right) = 0.169J$$

$$Hop\ 2: \frac{9}{2}\left(\frac{sin(4\pi \times 1.2) - sin(4\pi \times 0.6)}{24\pi} + 0.1\right) = 0.428J$$

$$\text{Hop 3: } \frac{1}{2}\left(\frac{sin(4\pi \times 0.6) - sin(4\pi \times 0.4)}{8\pi} + 0.1\right) = 0.0878J$$

What does this energy mean?

This energy we just calculated plays a major role in determining how evenly or unevenly energy is spread across the hop set. In FHSS, energy is ideally spread evenly across all frequencies to prevent over-reliance on one frequency and make the energy harder to detect. If hop energies vary too much, engineers may make changes by modifying the hopping sequence, improving the transmitter's linearity, or assigning a shorter hop time for each bit.

But even when energy is uniformly distributed across multiple channels, there is still a chance that someone might try to intercept the message, or two channels might try to use the same channel at the same time.

3. Collision Probability Naturally and Interception

Let's consider a simple case: if there are N channels available and another device picks a channel at the same time, what is the chance it picks the same channel as we do?

$$P(collision) = \frac{1}{N}$$

Mathematically, it is just probability.
Let's take an example:

Bluetooth, which operates in the 2.4Ghz band, uses 79 different channels for FHSS, meaning N = 79

$$P(collision) = \frac{1}{79} = 0.01265822784 \approx 0.0127$$

There is only 1.27% chance of a collision
Now imagine an eavesdropper trying to intercept the message over all 3 hops. The probability that they pick the correct channel for each hop will be very low or high, depending on the number of hops.
Hops are independent, so the probability of intercepting the entire k-hop message is:

$$P_{intercept} = \prod_{i=1}^{k} P_i\,(intercept) = \prod_{i=1}^{k} \frac{1}{N} = (\frac{1}{N})^k$$

We know that the signal's detectability also depends on energy per hop. Since both events are independent of each other, using probability, it can be refined as:

$$P_i = \frac{E_i}{E_{max}} \times \frac{1}{N}$$

Therefore, the total probability across all hops becomes:

$$P_i^{eff}(total) = \prod_{i=1}^{k} P_i^{eff}\,(intercept) = \prod_{i=1}^{k} (\frac{E_i}{E_{max}} \times \frac{1}{N})$$

For Bluetooth with 3 hops with $E_1 = 0.169 \quad E_2 = 0.428 \quad E_3 = 0.0878$

$$P_i^{eff}(total) = \frac{0.169}{0.428} \times \frac{1}{79} \times \frac{0.428}{0.428} \times \frac{1}{79} \times \frac{0.0878}{0.428} \times \frac{1}{79}$$

$$P_i^{eff} = 1.64 \times 10^{-7}$$

This shows that, despite the uneven energy, the eavesdropper has only a chance of 1.64×10^{-5} % intercepting the message, making it practically impossible!

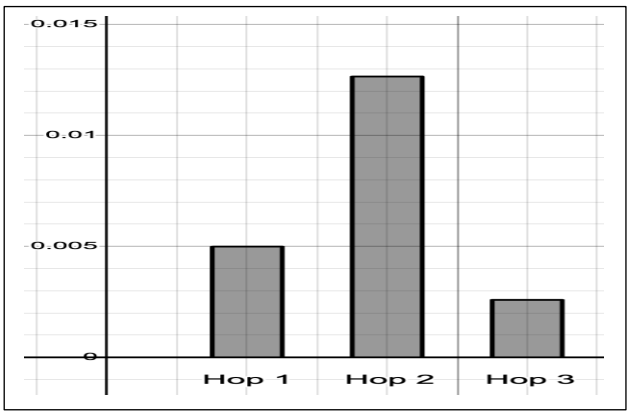

Figure 2:
individual Probability of each Hop

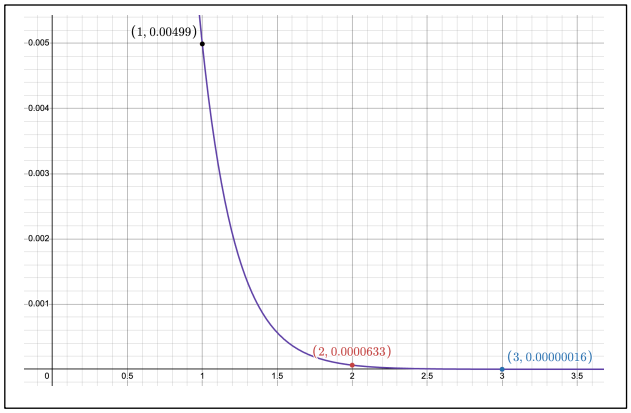

Figure 3:
Cumulative Probability

From this, we learn that as the number of hops increases, the probability of interception and collision naturally decreases exponentially. This significantly prevents interference (a concept known as “hop clash”). With a larger hop set, such as Bluetooth or military radio (100s or 1000s of channels), each hop occupies a small fraction of the spectrum, making the chance of interference unlikely. This prevents transmitted data from being corrupted.

Why is this necessary to know?

Knowing about the chances of interception is necessary for engineers to know how likely a wireless system is to be jammed. Let’s say that if many hops are corrupted, Bluetooth audio may stutter or WiFi might become slower than average. However, FHSS can recover from these interferences and has an error-correction system. So even if one hop is corrupted, the system can still transmit the message clearly. This makes FHSS highly resilient and prevents messages from being jammed or corrupted.

4. Markov Chains

A Markov chain is a mathematical model that describes how a system moves between a set of states over time, where the next state depends only on the current state and not on the previous history. In FHSS, a two-state Markov model can be used to decide whether a channel is “good” (low interference) or “bad” (high interference). The model computes the probability that the system will switch between these states. This model is known as the Gilbert-Elliot model and helps in better analyzing and understanding how the system operates.

$P(G \rightarrow G) =$ 1-a
$P(G \rightarrow B) =$ a (Probability of falling from good to bad)
$P(B \rightarrow G) =$ b (Probability of recovering from bad to good)
$P(B \rightarrow B) =$ 1-b

In a realistic environment, a channel falling from a good to a bad state has a probability of 0.2, and a channel recovering has a probability of 0.4.
a= 0.2
b= 0.4

$$P = \begin{bmatrix} a & 1-a \\ b & 1-b \end{bmatrix}$$

$$P = \begin{bmatrix} 0.2 & 0.8 \\ 0.4 & 0.6 \end{bmatrix}$$

Using this matrix, the steady-state probabilities can be calculated:

$$\pi_G = \frac{\pi_{BG}}{\pi_{BG} + \pi_{GB}}$$

$$\pi_G = \frac{0.4}{0.4 + 0.2}$$

$$\pi_G = \frac{2}{3} \approx 0.666\ldots$$

$$\pi_B = \frac{\pi_{GB}}{\pi_{GB} + \pi_{BG}}$$

$$\pi_B = \frac{0.2}{0.2 + 0.4}$$

$$\pi_B = \frac{1}{3} \approx 0.333\ldots$$

The found values are:

$$\pi_B = 0.33 \; and \; \pi_G = 0.67$$

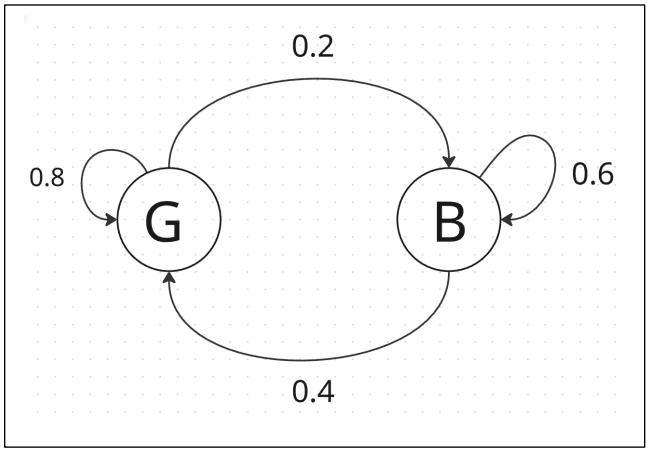

Figure 4: Gilbert-Elliot Model

What this tells us is that 66.7% of the time, the system stays in good channels, whereas the rest 33.3%, the system stays in bad channels. The Markov model is especially useful for engineers to determine how often a channel will be useful. In conventional FHSS, Markov models like this one are used to model the channel state and analyze performance metrics such as collision probability and error rates. This lets engineers know roughly how reliable the channels will be in real conditions.

5. Bit Error Rate (BER)

The bit error rate in FHSS represents the proportion of the bits transmitted that are received incorrectly. The Markov model gives the probability of the system being in the good or bad

channel, whereas the BER indicates the fraction of bits received incorrectly. When the system is in a good state, the error rate value is low. However, when the system is in the bad channel, the error rate value is significantly higher.

One of the formulas for calculating BER is given as:

$$BER = \frac{number\ of\ bits\ recieved\ in\ error}{total\ number\ of\ bits\ transmitted}$$

Let's assume that over 3 hops, there are 500 bits (a small value for demonstration) being transmitted
From our Markov model, we know,
P(G)= 0.667
P(B)= 0.333

$$\text{Good bits: } 0.667 \times 500 = 333.5 \approx 334$$

$$\text{Bad bits: } 0.333 \times 500 = 166.5 \approx 166$$

The probability of error in the good and bad states is:
P(e|G)= 10^{-12}
P(e|B)= 10^{-2}

$$\text{errors in the good state: } 334 \times 10^{-12}$$

$$\text{errors in the bad state: } 166 \times 10^{-2}$$

When we insert these values in the BER formula, we get:

$$BER = \frac{(334 \times 10^{-12} + 166 \times 10^{-2})}{500} = 3.3 \times 10^{-3} = 0.33\%$$

Therefore, from the Markov model, we derived a BER of 0.33%. In real communication systems, the ratio of the incorrectly received bits to the total transmitted bits is typically aimed to be below 10^{-5} for wireless communications. This significantly low BER indicates that the system is successfully avoiding or mitigating any interference, ensuring that the majority of the bits are transmitted correctly.

Conclusion

In conclusion, results from the calculations show that FHSS is a highly secure and useful method for the transmission of signals. Using the cosine function, energy calculations, collision and interference probabilities, the Gilbert-Elliott Markov chain, and Bit error rate, we gained some insight about FHSS's tight security.

A little Note

If you've read this far, I want to thank you for reading my first mini research report. It was loads of fun to learn more about this topic and do the maths. I hope you found this informative.

Works cited

1. https://arnabiitk.wordpress.com/wp-content/uploads/2013/02/proakis-digital-communications-4th-ed.pdf

2. https://www.mangoud.com/EENG373_files/Book-Sklar.pdf

3. https://ggnindia.dronacharya.info/Downloads/Sub-info/RelatedBook/4thSem/Communication-System-text-book-6.pdf

4. https://www.sciencedirect.com/topics/engineering/bit-error-rate

5. https://news.sparkfun.com/6147

6. https://daskalakispiros.com/files/Ebooks/digital-communication-proakis-salehi-5th-edition.pdf

7. https://www.sciencedirect.com/topics/engineering/signal-energy#:~:text=1.2%20Energy%20and%20Power%20of,2%20(%20t%20)%20d%20t%20.

8. https://www.mdpi.com/2076-3417/15/4/1935#:~:text=Markov%20chains%20are%20widely%20used,state%20to%20the%20bad%20state;

9. https://people.computing.clemson.edu/~jmarty/projects/lowLatencyNetworking/papers/APPFEC/GEModelForLossinTheRTInternet.pdf

10. https://pdfs.semanticscholar.org/618f/7c8a4230a7d75eca826be844a3638354d161.pdf

11. https://wyldnetworks.com/blog/hedy-lamarr-frequency-hopping-spread-spectrum