Model of the Signal of the Galileo Satellite Navigation System

M. Džunda, S. Čikovský & L. Melníkova
Technical University of Kosice, Kosice, Slovakia

ABSTRACT: This article presents an analysis of the Galileo E1 signal and its sensitivity to different types of interference. The research involved modeling white noise, chaotic impulse interference, and narrowband interference and the effects of these interfering signals on the E1 signal. Based on the available information, spectral structures were created for the mentioned types of interference, and subsequently, these interferences were integrated into the E1 signal in the Matlab program environment. A Kallman filter was used to filter out white noise from the additive mixture of the E1 signal and white noise. The research aimed to analyze the influence of white noise, chaotic impulse interference, and narrowband interference on the spectral power density of the E1 signal. The results of this work can be used in the design of robust receivers and signal structures capable of withstanding these types of interference.

1 INTRODUCTION

he Satellite Navigation System Galileo was launched in a joint project between the European Union's Satellite Navigation and the European Space Agency. It is part of the Trans-European Transport Network, which aims to solve navigational and geographic issues. The Galileo project is an ambitious European venture aimed at creating the most advanced global positioning satellite system in the world. Its objectives are to create an autonomous system that provides guaranteed global positioning services, as well as interoperable compatibility with other global positioning systems, such as GPS and GLONASS. In this work, we will focus on the signals transmitted by GNSS Galileo. We will briefly describe the individual signals. We will show and mathematically analyze signal models. For each signal, we will visualize and describe the structure of the signal. The signals of Galileo or other navigation systems have already been described in some literature. In the literature [3,4], models of measurement signals for some navigation systems are derived. The authors used these models to evaluate the accuracy of navigation systems. At the beginning of the system Galileo development, the system structure and signal models were described in the literature [9], where the author described the frequency plan and signal structure. The atmosphere of the Earth also has a great influence on the accuracy of signals. The author of the article [1,2] describes how signals behave when passing through the ionosphere. In the article [7,8], the author describes how it is possible to use more than three frequencies for decimeter positioning accuracy using Galileo and BeiDou signals. Models of measurement signals for selected communication systems are presented in the literature [5]. The mentioned models make it possible to simulate the signal processing of communication systems under interference conditions. Galileo's navigation signals are coherent and transmitted across three frequencies in the L band, namely E1, E5, and E6. The E6 signal is transmitted at a carrier frequency

of 1278.75 MHz and consists of three E6 components: E6-A, E6-B, and E6-C. On the other hand, the E5 signal, which has a center frequency of 1191.795 MHz, includes two separate signals called E5a and E5b. These two signals share a carrier frequency of 1176.45 MHz but are modulated independently. The E5a data and pilot components can be found below 15.345 MHz on the E5 carrier frequency. Meanwhile, the E5b signals are modulated on two different carrier frequencies within the E5 band, allowing them to be monitored separately [6]. Long codes can be useful in monitoring weak signals, such as those found inside buildings, but they can be difficult to obtain because the receiver detects signals by looking for delays in the received code, and long codes have more options than short codes. Short codes are good for quick fixes, but they can lead to incorrect satellite positioning when the receiver is exchanging signals between two satellites. This is because the receiver's ability to distinguish between two different codes is inversely proportional to the length of the codes. Signal length may not be suitable for all types of users, with internal and static users preferring long codes while external and fast-moving users preferring shortcodes. To address this issue, alternative codes with different properties have been provided for different Galileo signals. This is one of the reasons why Galileo has so many signals. Another reason for the abundance of signals is that the receiver can estimate the ionospheric delay error, which is caused by the delay experienced by navigation signals passing through the ionosphere. This delay can cause the receiver-measured distance from the satellite to the user to appear larger than it is, resulting in large position errors if not corrected. However, this delay is proportional to the frequency of the signal, with low-frequency signals experiencing longer delays than high-frequency signals. As a result, by combining measurements from two different frequencies on the same satellite, a new measurement can be created that removes the ionospheric delay error. The greater the distance between the two frequencies, the more effective this cancellation will be. This is why Galileo services are typically implemented using signal pairs [11].

2 GALILEO SIGNAL E1

The Galileo E1 signal uses BOC modulation, which employs carrier shift modulation to shift the energy away from the center of the band. This is significant because it enables multiple GPS systems to use the same band. BOC modulation utilizes two independent parameters, namely the carrier frequency of the auxiliary signal (fs) in MHz and the code rate of the code shift (fc) in mega chips per second. This gives the signal two parameters that can be adjusted to manipulate the signal's power in specific ways to reduce interference from other signals on the same band. Furthermore, the redundant upper and lower sidebands of BOC modulations offer advantages in signal processing for receiver acquisition, carrier tracking, code tracking, and data demodulation [12].

The entire transmitted Galileo E1 signal consists of the following components [13]:
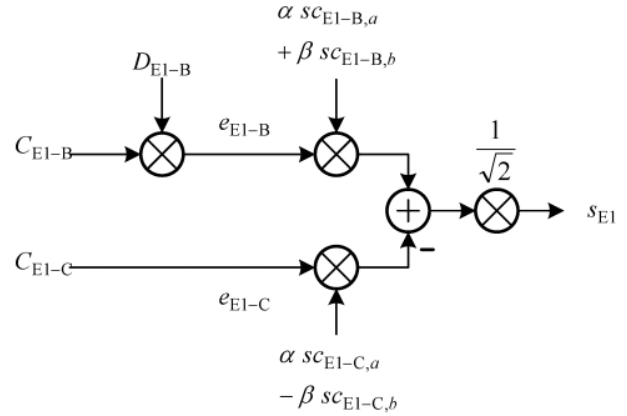

Figure 1. E1 Signal Modulation scheme [13].

Figure 1 shows the modulation scheme of signal E1. E1 open service data channel $e_{E1-B}(t)$ is generated from I/NAV navigation data stream $D_{E1-B}(t)$ and measurement code $C_{E1-B}(t)$, which are then modulated by subcarriers $SC_{E1-B, a}(t)$, and $SC_{E1-B,b}(t)$. The open service pilot channel E1 $e_{E1-C}(t)$ is generated from the range code $C_{E1-C}(t)$, including its secondary code, which is then modulated by subcarriers $SC_{E1-C, a}(t)$, and $SC_{E1-C,b}(t)$, in antiphase [13].

The Galileo E1 signal is modulated at a medium frequency such as [11] :

$$s(t) = A(C_D(t-\tau)d_D(t-\tau)CBOC_-(t-\tau) - \\ - c_P(t-\tau)S_c(t-\tau)CBOC_+(t-\tau)x\cos(2\pi f_{IF}t+\theta) \quad (1)$$

where:

$$CBOC_-(t) = \sqrt{10/11}\,BOC_1(t) - \sqrt{1/10}\,BOC_6(t) \quad (2)$$

$$CBOC_+(t) = \sqrt{10/11}\,BOC_1(t) + \sqrt{1/10}\,BOC_6(t) \quad (3)$$

$$BOC_x(t) = sign\left(\sin\left(x.2\pi.1.023e^6.t\right)\right). \quad (4)$$

A is the amplitude of the input signal at the input of the correlator,
c_P a C_D are extended sequences that carry pilot and data components,
d_D represents the navigation message symbol of the I/NAV modulated data component.
S_c represents the secondary code present on the pilot component,
τ is a sequenced delay,
f_{IF} is the center frequency,
θ is the phase shift of the carrier frequency [12].

GPS C / A and Galileo BOC (1,1) share the L1 / E1 spectrum, which is shown in Figure 1. The mean frequency of the E1 / L1 signal is 1575.42 MHz. It is important to remember that the current E1 band was given the name L1 band for a long time, analogous to GPS, until 2008, when the name of the L1 signal was changed to the current E1.

Based on the formulas (1-4), the simulation of the E1 signal was performed in the Matlab programming

environment. The following parameters of the E1 signal were used in the modeling:

$$fsa = 1.023 \cdot 10^6 ; \qquad fsb = 6 \cdot 1.023 \cdot 10^6$$

Define the subcarrier frequencies for BOC(1, 1) and BOC(6, 1). The value of fsa is set to 1.023 MHz and fsb is set to 6 · 1.023 MHz.

F = row space (-20, 20, 40000);

Generates a frequency vector for a PSD plot. The frequency vector ranges from -20 MHz to 20 MHz with 400000 points.

$$t \;=\; 0:1/ fsb : 10^{-3} \qquad (5)$$

Generates the time vector for the signal. The time vector ranges from 0 to 10 ms with a step size of 1/fsb.

$$x1 = \sin\left(2\pi \cdot fsa \cdot t\right) \qquad (6)$$

$$x6 = \sin\left(2\pi \cdot fsb \cdot t\right) \qquad (6)$$

They generate signals for BOC(1, 1) and BOC(6, 1). Signals are generated by multiplying partial carrier frequencies by a time vector and evaluating a sine function.

sca = character(x1)

scb = char(x6)

These lines generate secondary codes for BOC(1, 1) and BOC(6, 1). Secondary codes are generated by accepting the sign of the BOC signals.

$$alpha = \sqrt{10/11} \qquad (8)$$

$$beta = \sqrt{1/11} \qquad (9)$$

$$scB = alpha \cdot sca + beta \cdot scb \qquad (10)$$

$$scC = alpha \cdot sca - beta \cdot scb \qquad (11)$$

These lines generate the primary codes for E1-B and E1-C. Primary codes are generated by mixing secondary codes with alpha and beta coefficients.

$$eE1B = \cos\left(2\pi \cdot 1575.42 \cdot 10^6 \cdot t\right) \qquad (12)$$

$$eE1C = \sin\left(2\pi \cdot 1575.42 \cdot 10^6 \cdot t\right) \qquad (13)$$

These lines generate the E1-B and E1-C signals. Signals are generated by multiplying the carrier

frequency by the time vector and evaluating the cosine and sine functions.

$$X = fft\left(sE1, length\left(f\right)\right)$$

$$PSD = 10 \cdot \log_{10}\left(\frac{|X|^2}{length\left(f\right)}\right)$$

These lines calculate the FFT of the Galileo E1 signal and convert it to the PSD. The PSD is calculated as above, where X is the FFT of the signal and f is the frequency vector.

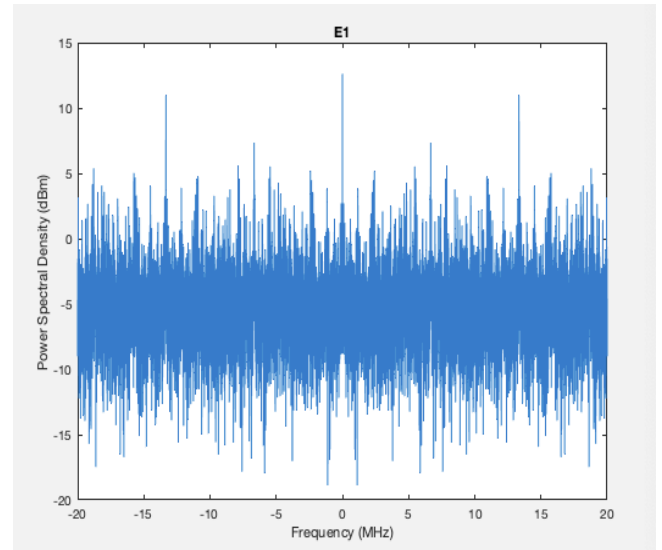The results of the E1 signal simulation are shown in Figure 2.

Figure 2. Structure of signal Galileo E1.

Figure 2 shows the power spectral density (PSD) for the E1 signal, which is part of the Galileo navigation system. On the x-axis is the frequency range from -20 MHz to 20 MHz, while on the y-axis is the calculated PSD in dBm. Frequency 0 on the x-axis represents the center frequency of the signal.

The source signal E1 is created by modulating the carrier wave using PRN (Pseudo Random Noise) coding. Therefore, the PSD for the E1 signal shows a characteristic frequency structure that consists of several layers. The highest layer represents the signal power in the area of the main carrier frequency of 1575.42 MHz, which occurs at frequency 0 on the x-axis. This peak has a calculated value of about -162 dBm/Hz.

Additional bands of power are visible around the main carrier frequency, 1.023 MHz apart, known as "sidebands" and "adjacent bands". These bands are part of the Galileo signal modulation and are responsible for the design of the PRN coding and other signal parameters.

The overall PSD waveform for the E1 signal shows how the signal power is distributed over the entire frequency axis. Since this signal occurs close to other satellites and transmission channels, it is important to know its power spectral density and its characteristic structure to avoid interference and guarantee the reliable use of Galileo.

3 SIMULATION OF INTERFERING SIGNALS

Research has confirmed that the measurement signals of the Galileo system are degraded by interfering signals. These signals can significantly affect the results of navigational measurements by the Galileo system. Therefore, we performed the simulation of interfering signals such as white noise, chaotic impulse interference, and narrowband interference. White noise, chaotic impulse interference, and narrowband interference models were created in the Matlab programming environment. We proceeded in the following way when creating interference models. First, we created an interference model, and then an additive mixture of the E1 signal and interference. The simulation results are shown in images no. 2 - 11.

3.1 *White noise*

We created a white noise model and then simulated this noise in the Matlab program environment. The following algorithm was used to generate white noise:

$$Noise = randn(1, N) \cdot noise\_level \tag{14}$$

This line generates a Gaussian noise signal with a mean of 0, and a standard deviation is given by the noise level vector.

Where:

fs = 40,000

This line sets the sample rate of the signal to 40000 MHz.

$T = 10^{-3}; \%$

This line sets the signal duration time to 10ms.

N = 400

This line sets the number of samples in the signal to 400.

$f = row\ space(-20 \cdot 10^6, 20 \cdot 10^6, N);$

This line creates a vector of N evenly spaced frequencies ranging from -20 MHz to 20 MHz.

x = rowspace(-T/2, T/2, N);

This line creates a vector of N evenly spaced time values ranging from -T/2 to T/2.

noise_level = linspace(-1, 6, N);

This line creates a vector of N evenly spaced noise levels ranging from -1 to 6.

$$PSD = \frac{\left| fftshift\left(fft\left(noise \right) \right)^2 \right|}{fs \cdot N} \tag{15}$$

This line calculates the PSD of the noise signal using the FFT, taking the absolute value and squaring it, and dividing it by the product of the sample rate and the number of samples.

$$PSD_{dBm} = 10 \cdot \log_{10}\left(PSD \cdot 10^3 \right) - 11 \tag{16}$$

This line converts the PSD from V²/Hz to dBm units using a reference power of 1 milliwatt and subtracts 11 to adjust for noise level.

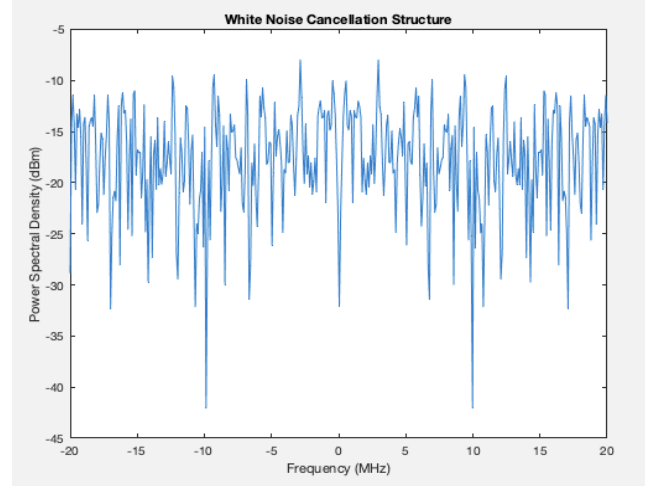The results of the white noise simulation are shown in Figure 3.

Figure 3. Structure of White noise.

Figure 3 shows the PSD graph of the Gaussian noise signal with the specified noise level. PSD is calculated using the Fast Fourier Transform (FFT) and converted from V²/Hz units to dBm units. The resulting PSD is then plotted as a function of frequency in MHz. The plot shows a symmetrical bell-shaped curve centered at zero frequency, which represents the PSD of the Gaussian noise signal. The curve has a maximum value at zero frequency, indicating that the noise signal has the highest power at low frequencies. The curve drops off rapidly with increasing frequency, indicating that the power of the noise signal decreases at higher frequencies. The x-axis of the graph represents frequency in MHz, while the y-axis represents PSD in dBm. The graph is labeled with the appropriate axis labels and the title "White Noise Suppression Structure", which describes the nature of the analyzed signal. The plot provides a visual representation of the frequency content of a Gaussian noise signal, which is useful in a variety of signal processing and communications applications.

3.1.1 *The additive mixture of E1 signal and white noise*

If in the previous steps, we generated the Galielo E1 signal and subsequently white noise, the next step is to generate the noisy E1 signal. E1 signal with the addition of white noise is possible using the following relationship:

$$sE1 = \frac{1}{\sqrt{2}} \cdot \left(eE1B \cdot (alpha \cdot scB + beta \cdot scB) - eE1C \cdot (alpha \cdot scC - beta \cdot scC) \right) + noise \tag{17}$$

Add white Gaussian noise to the signal:

$$sE1 = sE1 + noise \tag{18}$$

$$X = fft(sE1, length(f)) \tag{19}$$

Calculates the FFT of the sE1 signal with a length of (f) points

$$PSD = \frac{10}{3} \log_{10}\left(\frac{|X|^2}{length(f)}\right) \qquad (20)$$

Calculates the power spectral density of the signal.

The results of the simulation of the additive mixture of the useful signal E1 and white noise are shown in Figure 4.
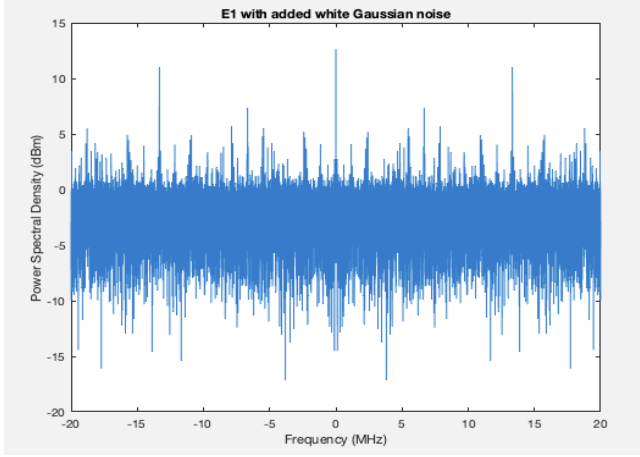

Figure 4. Signal E1 with added White noise.

The white Gaussian noise added to the sE1 signal can be seen in Figure no. 4. The effect of adding noise to a signal is that it increases the power spectral density (PSD) of the signal at all frequencies, including those of the original signal and the noise itself. The PSD is calculated using the fast Fourier transform (FFT) of the noise signal sE1, which is represented by the variable X. The PSD is then plotted against the frequency f in the image. The noise level is specified using the noise_level variable, which controls the standard deviation of the white Gaussian noise added to the signal. A higher level of noise results in a higher PSD of the noisy signal, which means that the signal is harder to detect and more accurately decoded, especially if the signal-to-noise ratio is low. Adding noise to the signal also introduces errors in the decoding of the spreading codes and the original signals, which can affect the overall performance of the communication system. In particular, the accuracy of timing and frequency synchronization can be affected, which can lead to further degradation of signal quality.

3.2 Chaotic impulse interference

We created a model of chaotic impulse interference and then simulated this interference in the Matlab environment. Chaotic impulse interference is created by combining a Lorentzian chaotic signal with an impulse response. In relation (23), the impulse response is defined as a vector of zeros with one 1 at index 100:

$$pulse = zero\ (size\ (t)) \qquad (21)$$

$$impulse(100) = 1 \qquad (22)$$

The conv() function is then used to convolve this impulse response with the first component of the Lorenz signal (x(:,1)):

$$interference = conv(x(:,1),\ impulse\ ) \qquad (23)$$

The resulting interference signal is a version of the Lorenz signal with an added pulse at index 100.

This relationship was used to generate the structure of impulsive chaotic interference, the spectrum of which is shown in Figure 5.

Parameters for the Lorenz chaotic system: sigma to 10, beta to 8/3, rho to 28, and the initial state vector x0 to [1;1;1]. The ode45 function is then used to solve the Lorenz system for the given time range t and initial conditions x0.

$$dxdt = sigma \cdot \big(x(2) - x(1)\big); x(1) \cdot \big(rho - x(3)\big) - x(2); x(1) \cdot x(2) - beta \cdot x(3) \qquad (24)$$

A function called "Lorenz" defines the differential equations for the Lorenz chaotic system. It takes in the input arguments "~" and "x", which are not used in the function. It also takes in the parameters "sigma", "beta", and "rho", which are used to define the Lorenz system equations.

The output "dxdt" is a vector of the same size as "x", which defines the rate of change of each state variable in the Lorenz system at a given time. The first element of "dxdt" is the rate of change of x(1), the second element is the rate of change of x(2), and the third element is the rate of change of x(3). The equations used in this function are the classic Lorenz equations, which are commonly used in chaos theory to study the behavior of nonlinear dynamical systems.

Parameters for the Lorenz chaotic system: sigma to 10, beta to 8/3, rho to 28, and the initial state vector x0 to [1;1;1]. The ode45 function is then used to solve the Lorenz system for the given time range t and initial conditions x0.

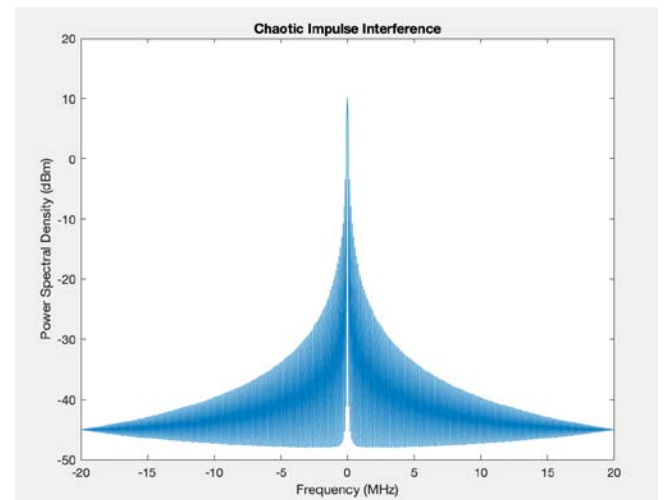The results of the chaotic impulse interference simulation are shown in Figure 5.

Figure 5. Structure of Chaotic Impulse Interference.

Figure 5 shows the power spectral density (PSD) of chaotic impulse interference as a function of

frequency. The horizontal axis shows the frequency in MHz, while the vertical axis shows the PSD in dBm.

On the graph, there is a suitable peak near the frequency of 0 MHz, which represents the DC component, i.e., alternating current with zero frequency. Next, we will see peaks that occur in different frequency bands, which are the frequency chaotic behavior of the system. Chaotic impulse interference contains many different frequency components that are propagated into the environment and affect other neighboring systems.

3.2.1 *The additive mixture of E1 signal and chaotic impulsive noise*

We created a model of chaotic impulse interference and then simulated this interference in the Matlab environment. We then created an additive mixture of the useful signal E1 and the chaotic impulse interference using the following algorithms:

$$sE1 = sE1 + interference \tag{25}$$

The disturbance is generated by the convolution of the first variable (x(:,1)) of the output of the Lorentzian impulse response system. The resulting interference signal is then added to the sE1 signal. The simulation results of the additive mixture of useful signal E1 and chaotic impulse interference are shown in Figure 6.
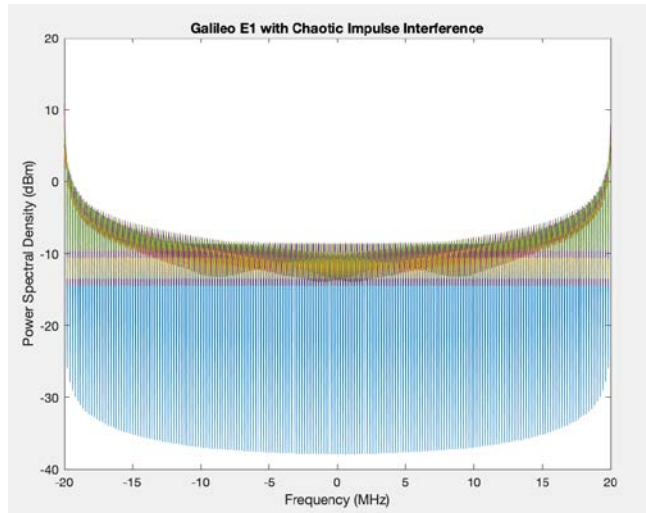

Figure 6. Structure of Galielo signal E1 with added Chaotic impulse Interference.

Figure 6 shows the PSD of the Galileo E1 signal, which is subsequently affected by chaotic interference. This process can have a major impact on signal quality and cause interference and communication disruptions. A Lorenz chaotic system is a differential equation that describes chaotic behavior. In this case, the system consists of three equations that describe the evolution of three variables over time. These variables represent the state of the system and change depending on time and other parameters. The chaotic disturbance in this case is generated by the convolution of the signal from the Lorentz system with the impulse response. The result of this convolution is a time-shifted signal that can cause interference and communication breakdowns. The

effect of interference on the Galileo E1 signal can be observed using the power spectral density (PSD), which is calculated using the Fourier transform. PSD shows the distribution of signal power as a function of frequency. In this case, we can see that chaotic interference causes interference in a wide frequency band, which can cause disturbances in receiving navigation information. The result of this process is a signal that is affected by chaotic interference, which manifests itself as interference in the entire frequency band. The resulting signal may be of lower quality and cause communication problems. Therefore, it is important to ensure a sufficient level of protection against interference and to minimize the effect of chaotic interference on the signal.

3.3 *Narrowband interference*

Narrowband interference is a type of interference that occurs when there is a strong signal at a frequency equal to the frequency of the E1 signal. This interference signal causes errors in the navigation measurements of the Galileo system. We generated the narrowband interference according to the following algorithms:

$$x1 = 10 \cdot cos\left(2\pi \cdot \left(f0 + delta_f\right) \cdot t + delta_{phi}\right) \tag{26}$$

That relationship generates narrowband interference by the sum of N sine waves with random frequency and phase variations around the center frequency f0.

Where:

$$delta_f = (rand(1, N) - 0.5) \cdot 2 \cdot 10^6 \tag{27}$$

Generates N random values between -1 MHz and 1 MHz. These values represent frequency variations.

$$delta_{phi} = rand(1, N) \cdot 2\pi \tag{28}$$

Generates N random values between 0 and 2π. These values represent phase variations.

f0 = 1575.42 · 10^6

Sets the center frequency to 1575.42 MHz.

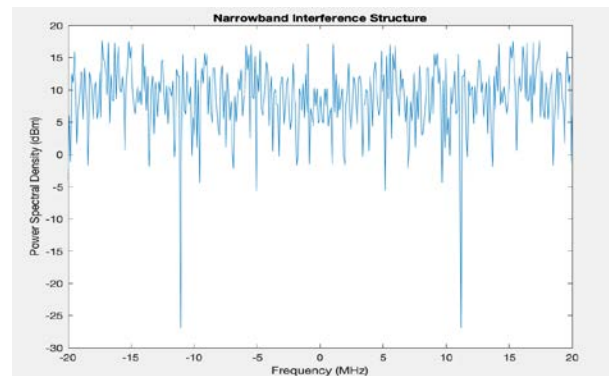The results of the narrowband interference simulation are shown in Figure 7.

Figure 7. Narrowband interference structure.

The power spectral density (PSD) of a narrowband interference signal with a random frequency and phase variations is shown in Figure 7. The x-axis represents frequency in MHz and the y-axis represents the PSD in dBm. The center frequency of the signal is 1575.42 MHz, and the frequency variations around this center frequency are randomly generated with a range of +/- 2 MHz. The phase of the signal also varies randomly. The resulting PSD plot shows a main lobe centered at the center frequency and several smaller lobes on either side due to the random frequency variations. The height and width of the lobes depend on the amplitude and duration of the interference signal. The plot can be used to identify the frequency and strength of the interference signal, which is important for mitigating its effects on the system.

3.3.1 The additive mixture of E1 signal and narrowband interference

We used the programming environment of the Matlab program to create an additive mixture of the E1 signal and narrowband noise. Interference is added to the original signal (Figure 2) using the relation:

$$sE1\_with\_interference = sE1 + x1 \tag{29}$$

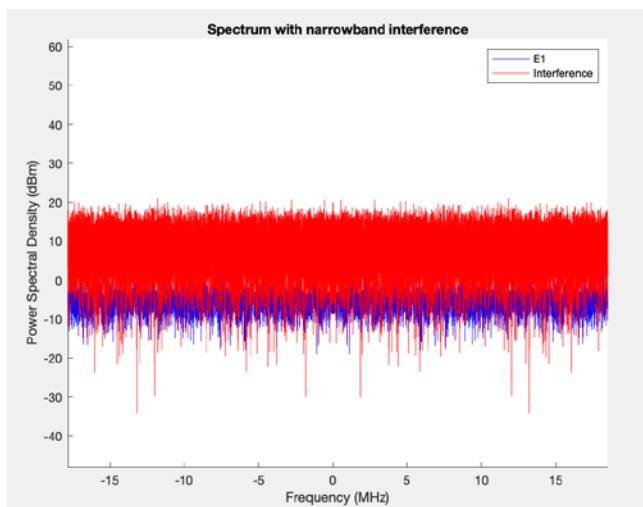The result of adding narrowband interference to the E1 signal can be seen in Figure 8.

Figure 8. Structure of Signal E1 with added Narrowband interference.

Figure 8 shows the resulting power spectral density (PSD) of the E1 signal and narrowband interference with random frequency and phase deviations. PSD indicates the distribution of signal power concerning the frequency and is expressed in units of dBm/MHz. The E1 signal is a blue curve and corresponds to the global navigation satellite system Galileo, which transmits at a frequency of 1575.42 MHz. The blue curve has a maximum value in the middle of the spectrum, which is chosen as a reference value for the entire spectrum and is therefore placed at zero. The red curve shows narrowband interference, which is characterized by random frequencies and phase deviations. This type of interference can arise, for example, from sources with an unstable frequency, such as generators with short-

term deviations, or the influence of external interference. Thus, the figure shows that narrowband interference has a significant effect on signal quality if the transmitted signal and the interference are on the same frequency. Therefore, it is important to protect the transmitted signals from interference and avoid overlapping frequency bands of different signals and interference.

4 USING KALLMAN FILTRE

White noise has a pronounced effect on the original structure of the E1 signal. It is therefore important to remove this noise from the signal. In this chapter, we will describe the effect of using a Kallman filter to filter out white noise from a noisy signal (Figure 4.)

Defining the system matrices for the Kalman filter:
A = 1; state transition matrix
H = 1; observation matrix
Q = 0.01; process noise covariance
R = 0.1; covariance of measurement noise
PO = 1; initial covariance state

In this block of code, the matrices that define the Kalman filter are defined. A is the state transition matrix, H is the observation matrix, Q is the process noise covariance, R is the measurement noise covariance, and P0 is the initial state covariance.

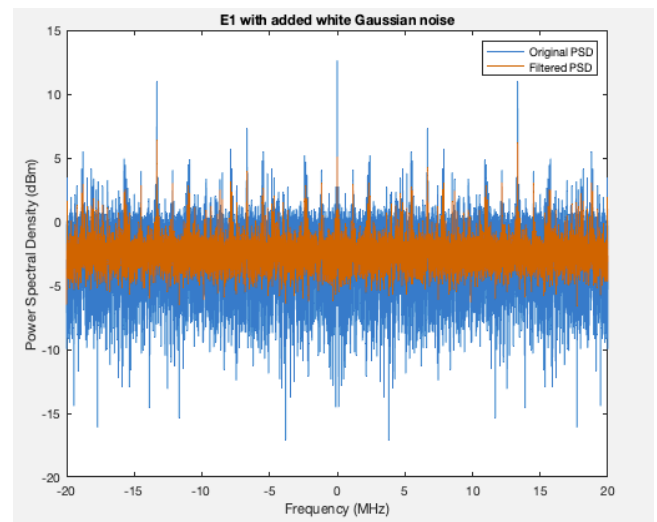The simulation results are shown in Figure 9.

Figure 9. Filtered signal E1 using the Kallman filter.

In this case, white noise was suppressed using a Kalman filter (Figure 9.) A Kalman filter is a mathematical algorithm that uses a series of measurements observed over time to estimate the state of a linear system. In this case, the state is the actual power spectral density of the signal, and the measurements are the power spectral density values obtained from the noisy signal. The Kalman filter works by predicting the state of the system at each time step based on a previous estimate of the system's state and dynamics, and then updating the state estimate based on the current measurement. The filter also estimates the uncertainty in the state estimate, which is used to weigh the importance of the prediction and measurement in the overall estimate.

In this implementation, the state transition matrix A was set to 1, indicating that the system does not change over time. The observation matrix H was also set to 1, indicating that the measurement is a direct observation of the state. The covariance Q of the process noise and the covariance R of the measurement noise was set to 0.01 and 0.1, respectively, which are assumptions about the noise variance in the system. A filter was applied to each frequency bin of the power spectral density using a loop. At each cycle iteration, the state and covariance estimates were predicted using the state transition model. Then, the state estimate was updated using the current measurement and the Kalman gain, which is a weighting factor that balances the contribution of prediction and measurement to the overall estimate. Finally, the filtered value of the power spectral density was stored in the array. The filtered power spectral density values were then plotted against the original power spectral density values to show the effectiveness of the filter in suppressing white noise.
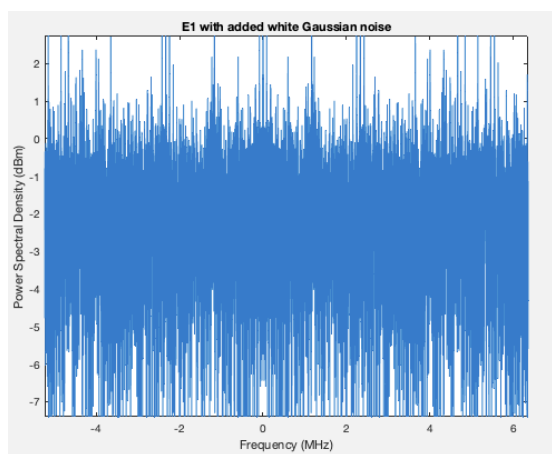

Figure 10. Detailed Figure 4.

Figure 10 shows a more detailed spectrum of the additive mixture of the useful signal E1 and white noise.
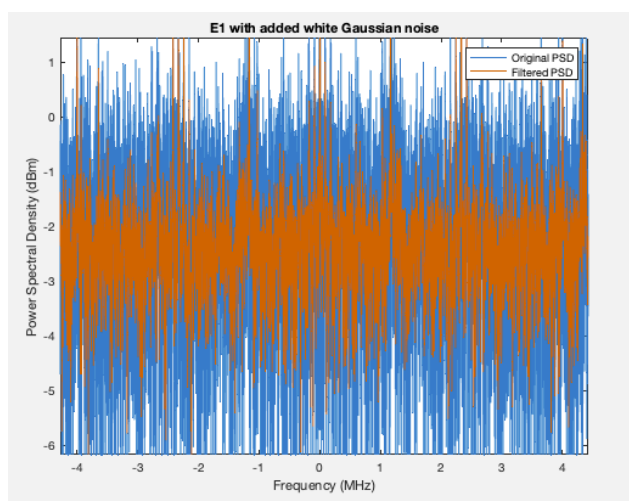

Figure 11. Detailed Figure 9.

Figure 11 shows the spectrum of the additive mixture of signal E1 and white noise after passing the signal through the Kallman filter. The Kalman filter suppressed the white Gaussian noise in the additive mixture of the E1 signal and white noise. It is clear from the picture that the (orange color) filter suppressed the effect of noise on the useful E1 signal from the Galileo system.

5 CONCLUSIONS

In this paper, mathematical models of the measurement signal E1 of the Galileo system were created to simulate intentional interference and the influence of atmospheric conditions on signal propagation. The study also presents the determination of the Galileo satellite navigation system, including the frequency and structure of the E1 signal and its block diagram for signal generation. The modeling results can be used to evaluate the immunity of the Galileo system to interference. The results of the simulations of the measurement signal E1 with white noise, chaotic impulsive interference, and narrowband interference allow a spectral analysis of the structure of this signal. Through simulation, we found that the frequency spectrum of the additive mixture of the E1 signal and interference is significantly different from the frequency spectrum of the E1 signal itself. This fact indicates that signal measurements that are degraded by interference can cause errors in navigation measurements by the Galileo system. The simulation results confirmed that chaotic impulse interference is the most dangerous type of interference. We can explain it by the fact that chaotic impulse interference has a wide frequency spectrum. Its influence on the original signal has destructive effects, and when processing the measurement signal E1, it is necessary to eliminate this interference as much as possible. A Kallman filter was designed to filter E1 signal interference. This filter was used to filter out white noise from the additive mixture of the E1 signal and white noise. The simulation results confirmed that such a filter is effective for suppressing white noise and can be used at the receiver input for E1 signal processing. The simulation results confirmed that when synthesizing algorithms for processing the E1 measurement signal from the Galileo system, it is necessary to pay attention to the influence of white noise, chaotic impulse interference, and narrowband interference on the measurement results. Dual frequency is used to improve the measurement results of the Galileo system. Galileo's use of dual frequency positioning provides an advantage over other satellite systems, particularly in dense urban areas or forested environments. The use of dual frequency allows for more accurate positioning.

REFERENCES

[1] Bałdysz, Z., Szołucha, M., Nykiel, G., & Figurski, M. (2017, June). Analysis of the impact of Galileo's observations on the tropospheric delay estimation. In 2017 Baltic Geodetic Congress (BGC Geomatics) (pp. 65-71). IEEE. DOI: 10.1109/bgc.geomatics.2017.22.
[2] Bidaine, B. (2006). Ionosphere Crossing of GALILEO Signals.
[3] Dzunda, M; Kotianova, N; Dzurovčin, P., et all.: Selected Aspects of Using the Telemetry Method in Synthesis of RelNav System for Air Traffic Control. Jan 2020

INTERNATIONAL JOURNAL OF ENVIRONMENTAL RESEARCH AND PUBLIC HEALTH 17 (1)

[4] Dzunda, M; Dzurovcin, P; et all. : Selected Aspects of Navigation System Synthesis for Increased Flight Safety, Protection of Human Lives, and Health. Mar 2020, INTERNATIONAL JOURNAL OF ENVIRONMENTAL RESEARCH AND PUBLIC HEALTH 17 (5)

[5] Dzunda, M.; Kotianova, N.; Holota, K.; et al.: Use of Passive Surveillance Systems in Aviation. ACTIVITIES IN NAVIGATION: MARINE NAVIGATION AND SAFETY OF SEA TRANSPORTATION. Published: 2015. Pages: 249- 253

[6] Galileo navigation signals and frequencies [online]. Available from: https://www.esa.int/Applications/Navigation/Galileo/Galileo_navigation_signals_and_frequencies

[7] Geng, J., Guo, J. Beyond three frequencies: an extendable model for single-epoch decimeter-level point positioning by exploiting Galileo and BeiDou-3 signals. J Geod 94, 14 (2020). https://doi.org/10.1007/s00190-019-01341-y

[8] Hadas, T., Kazmierski, K., & Sośnica, K. (2019) Performance of Galileo-only dual-frequency absolute positioning using the fully serviceable Galileo constellation. GPS Solutions, 23(4), 108. DOI: 10.1007/s10291-019-0900-9.

[9] Hein, Guenter & Godet, Jeremie & Issler, Jean-Luc & Martin, Jean-Christophe & Lucas-Rodriguez, Rafael & Pratt, Tony. (2001). The Galileo Frequency Structure and Signal Design. Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation.

[10] European Commission (2010), European GNSS (Galileo) Open Service – Signal-In-Space Interface Control Document Issue 1, February.

[11] EUROPEAN GNSS (GALILEO) OPEN SERVICE SIGNAL- IN-SPACE INTERFACE CONTROL DOCUMENT Issue 2.0, January 2021

[12] Olivier Julien, Christophe Macabiau, Emmanuel Bertrand. Analysis of Galileo E1 OS unbiased BOC/CBOC tracking techniques for mass market applications. NAVITEC 2010, 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals, Dec 2010, Noordwijk, Netherlands. pp 1-8, 10.1109/NAVITEC.2010.5708070 . hal-01022203

[13] Rodríguez, J.A,A. Galileo Signal Plan, University FAF Munich, Germany, 2011