Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/6/20230899

Noise signal review and analysis on a digital accelerometer in a real-world circuit system

Yizhou Fang

Northeastern University, Shenyang, Liaoning, China, 110167

20190708@stu.neu.edu.cn

Abstract. In recent years, the applications of MEMS inertial sensors have expanded significantly. The majority of applications challenges the accuracy, stability, and sensitivity of inertial sensors. Noise characteristic is a crucial performance indicator for sensors. though diverse methods for analyzing and classifying sensor noise from various perspectives are proposed, t his study examines a prominent noise analysis technique known as Allan Variance (AVAR). According to the results of several computations, this approach classifies inertial sensor noise into five groups. In addition, a set of algorithms are proposed to minimize noise in order to limit sensor noise's impact on the system. In general, these algorithms disregard the type of noise and assume that it is random. In this paper, a quick introduction to these algorithms will be provided. A comprehensive investigation of accelerometer noise will conclude this paper. The noise signal is gathered under various conditions for comparison. Typically, this research illustrates a potential difficulty with microcontroller noise signal reception accuracy. The impact and potential explanation of such a problem on noisy signal reception will also be discussed.

Keywords: noise, sensor, algorithm, accelerometer, Allan Variance.

1. Introduction

MEMS is an acronym for microelectromechanical systems. It can be utilized to manufacture micrometer-scale structures, electronics, and systems. Inertial sensors based on MEMS technology are rapidly becoming ubiquitous due to its incorporation into numerous types of consumer electronics, such as smart phones, tablets, gaming systems, TV remotes, toys, power tools, and wearable sensors. Accelerometers and gyroscopes are the most extensively used inertial sensors.

Historically, the Allan Variance (AVAR) was used to determine the frequency stability of clocks, oscillators, and amplifiers. With the increasing use of Inertial Sensors, AVAR is used to measure the sensors' properties. In reality, AVAR focuses on the intrinsic noise of the sensors, particularly the physical units of the sensors. Extrinsic noise originating from circuits and environs is excluded from the measurement. However, we can still examine the circuit as a full system and study noise at a higher level using the AVAR. In terms of such measurements, the final results summarize the noise signals with similar properties.

Fundamental components of numerous control systems are noise reduction algorithms. Some algorithms offer excellent responsiveness, while others compromise it for extreme accuracy. Numerous algorithms have been implemented and expanded in various systems and domains. Most MEMS Inertial sensors in the real world have noise reduction modules.

DOI: 10.54254/2755-2721/6/20230899

The ADXL346 accelerometer is an ultralow power digital accelerometer. It is suitable for mobile device applications and a variety of embedded usage. It contains around thirty registers for configurations and data buffers that are user-specific. In addition, it includes a sequence of interruptions for various events, including tap and free fall. In this experiment, we intend to retrieve updated data from sensor data registers. We consider the sampling, filtering, ADC, and other steps as a system and calculate noise signals using PSD and AVAR.

This paper reviews AVAR and noise signal reduction algorithms. We typically investigate the sorts of noise each algorithm handles. PSD and AVAR were used to detect noise on the ADXL346 accelerometer. In this experiment, the system configuration processes and detection results are unique.

2. Review

2.1. Power Spectral Density

Power Spectral Density (PSD) is a fundamental technique for digital signal processing [1]. In terms of MEMS inertial sensor noise, the connection between density and frequency represents the properties of various types of noise, as defined by function S(f). PSD can be easily determined using the square root of the absolute value of the signal's Fourier Transform. Normalization of the factor can be applied to the result in order to stabilize the output amplitude regardless of the sampling frequency.

2.2. Allan Variance and noise types

Allan Variance(AVAR) is a time-domain analysis technique that is adapted to the analysis of MEMS inertial sensor noise [2]. It clusters the digital noise signal into a series of consecutive groups. Each of these groups contains T samples. T is a variable whose value determines the AVAR result σ(T) given the digital signal.

The relationship between σ(T) and T defines different types of noise from a mathematical perspective. An important property of AVAR is its tight relationship with the PSD of the signal.

$$\sigma^2(T) = 4\int_0^\infty S(f)\frac{sin^4(\pi fT)}{(\pi fT)^2}df$$

That relationship indicates that AVAR does not provide more information than the PSD in that it can be calculated directly from the PSD. However, as the AVAR can be processed merely in the time domain, it can be applied to more signals than the PSD. An example is a specific type of noise called ‘rate ramp’ which will be shown in the following pages. In addition, the time complexity of the FFT algorithm is much greater than that of the AVAR calculation, and the PSD approximation algorithms may result in varying degrees of result distortion. These advantages contribute to AVAR's appeal.

AVAR can identify five different types of noise [3], including: Quantization Noise(QN), Angle/Velocity Random Walk, Bias Instability, Rate/Acceleration Random Walk, and Rate Ramp.

2.2.1. Quantization noise. Quantization noise is a type of noise introduced by the quantization of analog signal. During this process, an analog value is rounded to the nearest digital value, leading to a slight distortion in the signal. It can be calculated that the AVAR of quantization noise can be expressed as below:

$$\sigma^2(T) = \frac{3Q^2}{T^2}$$

While Q stands for the quantization coefficient given by

$$Q^2 = \frac{q^2}{12}$$

q stands for the quantization step size(resolution) of ADC [4]. Such type of noise can be reduced by increasing the resolution of the ADC.

2.2.2. Angle/Velocity Random Walk. Angle Random Walk refers to the random walk of a gyroscope’s measurement. Similarly, Velocity Random Walk adapts to the accelerometers. Random walk refers to a

DOI: 10.54254/2755-2721/6/20230899

mathematical model of a stochastically random process. Such a process accumulates the system white noise and adds the summation of noise to the true value. The source of the random walk is suggested to be the thermo-mechanical noise inside a system. The AVAR of this type of noise can be calculated as

$$\sigma^2(T) = \frac{N^2}{T}$$

where N is the angle(velocity) random-walk coefficient given by

$$S(f) = N^2$$

2.2.3. Bias Instability. This type of noise refers to the flicker noise. It is a widespread noise whose origin is still unclear. A series of complex models have been proposed to analyze the flicker noise of semiconductors, ADC circuits, and many other different systems. The AVAR of such noise is calculated as

$$\sigma^2(T) = \frac{2B^2}{\pi} ln2$$

where B is the is the bias instability coefficient given by

$$S(f) = \begin{cases} \frac{B^2}{2\pi f}, f \le f_0; \\ 0, \quad f > f_0. \end{cases}$$

2.2.4. Rate/Acceleration Random Walk. This is a type of noise whose origin is also not unclear. The PSD of such noise is similar to that of a Brown noise. The integration of such noise will also affect the angle measurement of a gyroscope or the velocity measurement of an accelerometer. Allan noise of such type of noise is shown as

$$\sigma^2(T) = \frac{K^2 T}{3}$$

where K is the rate/acceleration random walk coefficient given by

$$S(f) = \left(\frac{K}{2\pi}\right)^2 \frac{1}{f^2}$$

2.2.5. Rate Ramp. Rate ramp describes a systematic error that the measurement of the sensor drifts as time passes. Such a noise cannot be measured by PSD for it does not meet the prerequisite of Fourier Transformation. However, AVAR can still be applied to it and is shown as

$$\sigma^2(T) = \frac{R^2 T^2}{2}$$

where R is the rate ramp coefficient given by

$$N_{rate\_ramp} = Rt$$

2.3. Noise Reduction Algorithm

2.3.1. Recursive Least Square. Least Square is a technique that can be used to simulate the linear system function and reduce sensor noise. The Recursive Least Square is an adaptation of the Least Square algorithm. This improvement eliminates the requirement to construct the inverse matrix for each sample while maintaining a minimal system delay. Such an algorithm disregards any noise feature properties. It forecasts the output stochastically based on historical inputs. There are algorithms comparable to Recursive Least Square. The Least Mean Mean Square employs Mean Square Error, a distinct measurement or so-called 'loss function', to calculate and control system error [5]. Enhanced Recursive Least Square algorithms are also presented for various optimizations [6-7].

2.3.2. Kalman Filter. The Kalman Filter is an extremely popular algorithm [8]. It is currently integrated inertially into sensors and several control systems. The Kalman filter assumes the summation of noise

DOI: 10.54254/2755-2721/6/20230899

follows a Gaussian distribution. Note that the Kalman Filter disregards the PSD, AVAR, and other noise features.

There exist a series of algorithms that extend the Kalman Filter. The Extend Kalman Filter and the Incremental Kalman Filter are two examples[9-10].

2.3.3. Various Machine Learning and Deep Learning Algorithms. Every adaptive algorithm can be seen from a machine-learning perspective. These algorithms include the "training" and "predicting" steps based on signal datasets. In this term, many popular machine learning and deep learning algorithms are also used to reduce sensor noise. For example, the SVM, RNN, and their variations have been studied on some sensors[11-12]. However, their implementations are limited, partly due to the time complexity and space complexity.

2.3.4. Wavelet Threshold. Wavelet Threshold is based on wavelet decomposition. With wavelet decomposition, a signal is iteratively decomposed into a tree-like structure, where each leaf consists of wavelet coefficients. Meanwhile, the algorithm defines wavelet threshold according to signal characteristics and applies hard threshold processing and soft threshold processing to each wavelet coefficient. The original signal can then be reconstructed with these new coefficient values.

Wavelet Threshold have been mainly applied to reduce Gaussian white noise in speech and image signals [13-14]. There are also studies that try to improve the performance of this algorithm on non-Gaussian noise.

3. Experiment

3.1. Basic Ideas

We conducted an experiment about sensor noise detection on an accelerometer, ADXL346. The experiment was held at room temperature. The average supply voltage is about 2.68V. The accelerometer was placed horizontally on standing. We changed the configuration of the sensor and recorded its noise signal.

3.2. System Device

The ADXL346 accelerometer is a 3-axis digital accelerometer. It can transmit data through SPI and I2C digital interfaces. It supports user-selectable measurement ranges and output data rates. Users can write new values into several configuration registers to configure the sensor behaviors.

The data of the sensor are saved in 6 data registers (2 for the x-axis, 2 for the y-axis, and 2 for the z-axis). The data for each axis are saved in int16_t format. We used an STM32F103 microcontroller to read sensor data through the SPI interface and transmit data to the PC through the USART interface. A serial port monitor software on the PC is used to save and print out the data.

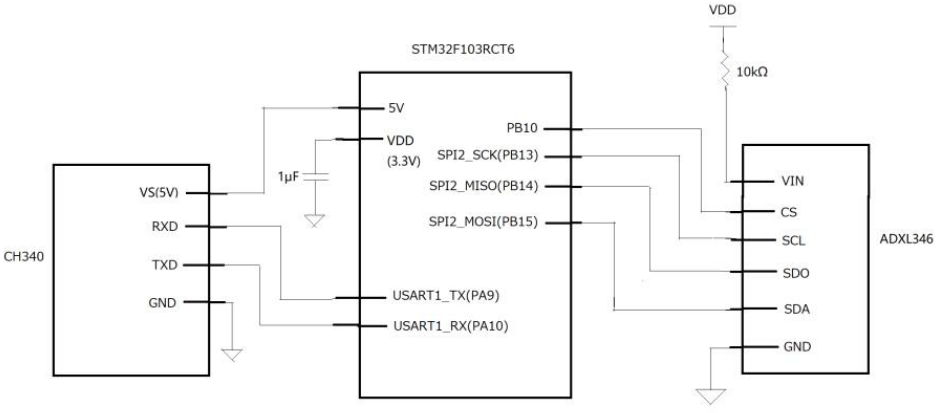

Figure 1. Circuit Configuration.

DOI: 10.54254/2755-2721/6/20230899

During each measurement, we only read and receive the noise signal of a single axis. The data consists of three bytes: a flag byte, the first data register byte, and the second data register byte. The flag byte is used to mark up the coming of new register data. The resolution and the range of the sensor guarantee that '0xCC' will never appear as a value of data registers, shown as figure 2.

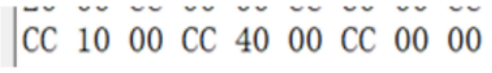

Figure 2. Data Flow.

3.3. System configuration

A question of this system is that the microcontroller does not know when the sensor has updated its data. In other words, the data read operation conducted by the microcontroller is independent of the data update operation conducted by the sensor. The mismatch of the data read rate and update rate will cause serious distortion in the signal.

We dealt with this question in this experiment by applying different configurations and observing the received data. Specifically, we firstly set the SPI baud rate to 2.25MHz, and then configure the data update rate; next, we configure the USART baud rate, and finally count the data repetitive times. The results are shown in table 1.

Table 1. several attributes under different configured data update rate.

configured data update rate (Hz)	USART1 baud rate (Hz)	Received bits per sec (bits)	same data repeat times	Actual data update rate (Hz)
25	57600	42000	72	24.31
25	19200	14000	24	24.31
100	19200	14000	6	97.22

The Relationship between each variable is shown below:

$$actual\ data\ update\ rate\ = \frac{received\ bits\ per\ sec}{data\ bits\ \cdot\ same\ data\ repeat\ times}$$

The 'data bits' refers to the number of bits transmitted to the PC in every cycle whose value is 24 in this experiment. The observation is listed as:

$$actual\ data\ update\ rate\ = 0.972\ \cdot\ configured\ data\ update\ rate$$

$$USART\ baud\ rate\ = 32\ \cdot\ configured\ data\ update\ rate$$

The first relationship shows that the actual data update rate is slightly lower than the configured data update rate due to system delay. The second relationship shows that the USART1 baud rate should be configured to about 32 times the data update rate of the sensor.

Under this configuration, the noise brought by the mismatch of data read rate and data update rate can be partially limited.

3.4. Noise detection and analysis

The first sets of y-axis noise data are measured under the condition of

SPI baud rate = 2.25MHz

USART baud rate = 102400Hz

Data update rate = 3200Hz

DOI: 10.54254/2755-2721/6/20230899

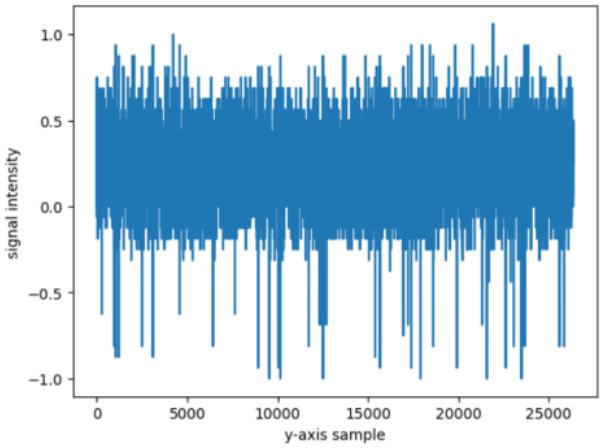

Figure 3. noise signal at 102400 Hz USART Baud Rate.

Without the DC bias, the PSD of the signal is shown as below

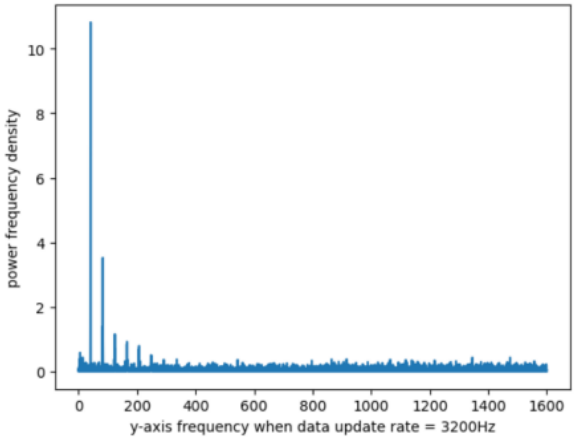

Figure 4. PSD of signal at 102400 Hz USART Baud Rate.

The AVAR of the signal is shown as below, figure 5.

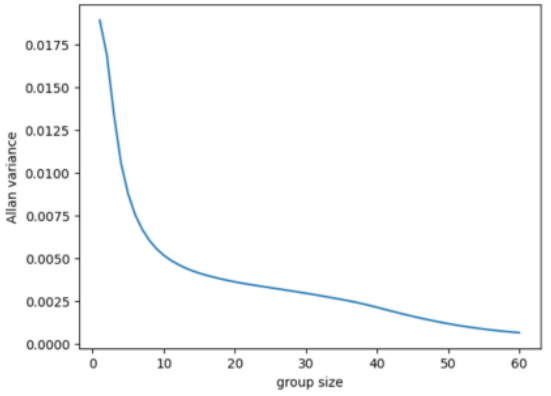

Figure 5. AVAR of signal at 102400 Hz USART Baud Rate.

We model the AVAR with formula

$$\sigma(T) = \sum_{i=1}^{5} a_i \, T^{i-3}$$

We use the Least Square to calculate a1 ~ a5. And the result is shown in figure 6.

DOI: 10.54254/2755-2721/6/20230899

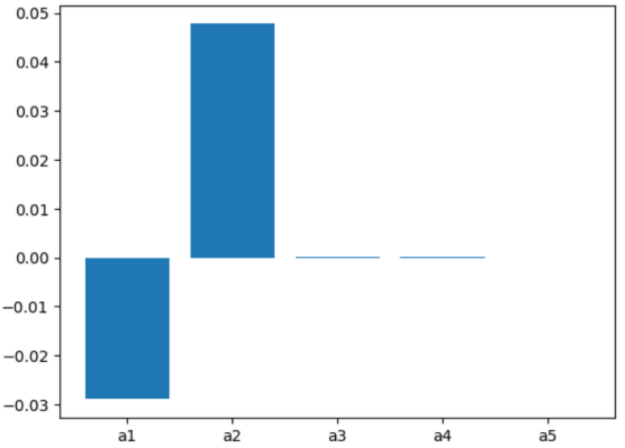

Figure 6. coefficients of model at 102400 Hz USART Baud Rate.

According to the AVAR, the Quantization Noise and the Velocity Random Walk are the two main types of noise. However, the PSD shows a number of peaks spread among the low-frequency noise. These peaks are weird and appear at certain frequency.

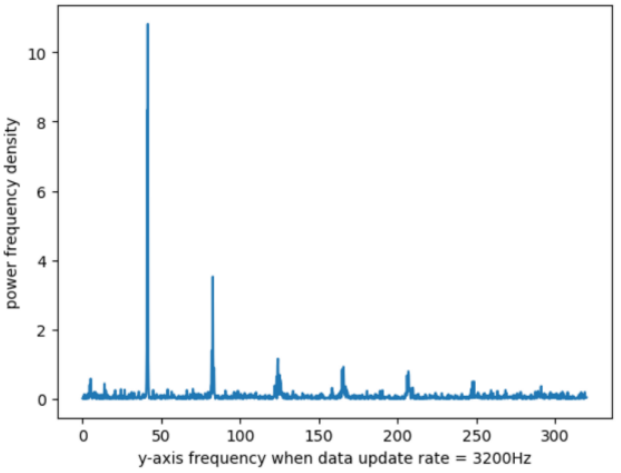

Figure 7. augmented vision of PSD at 102400 Hz USART Baud Rate.

In the third part of this section, we introduced the mismatch of data read rate that will bring about additional distortion to the signal. We suggested that the peak intervals have relationship with the mismatch. To prove it, we changed the data read rate and observed whether the intervals changed accordingly.

We configured the data read rate at different values by regulating the USART baud rate. The next figure 8-9 depicts the noise signal PSD when the USART baud rate is configured at 103600Hz. Under this configuration, the peaks of the PSD seem to 'converge' at lower density.

DOI: 10.54254/2755-2721/6/20230899

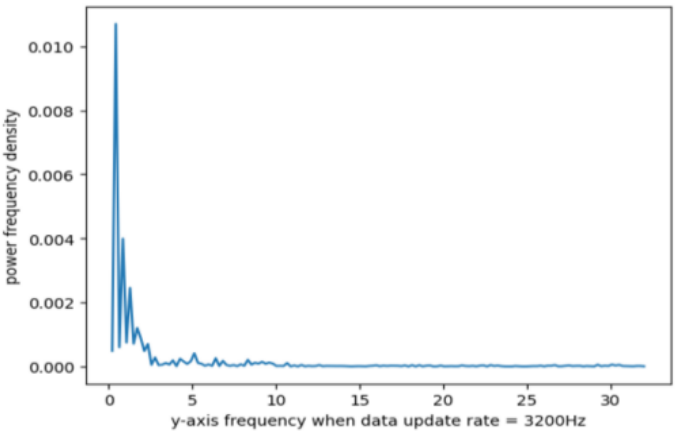

Figure 8. PSD of signal at 103600 Hz USART Baud Rate.

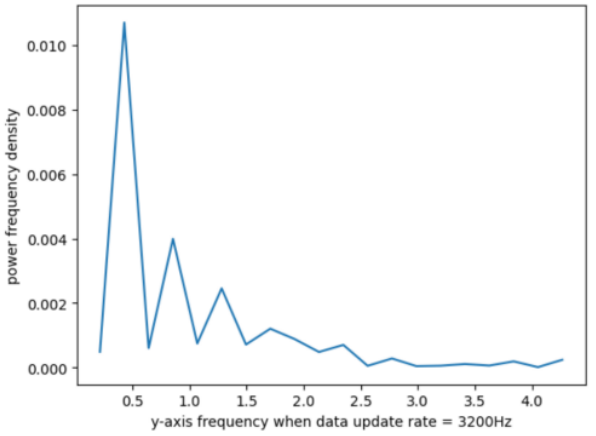

Figure 9. augmented vision of PSD at 103600 Hz USART Baud Rate

The AVAR of this situation is shown in figure 10.

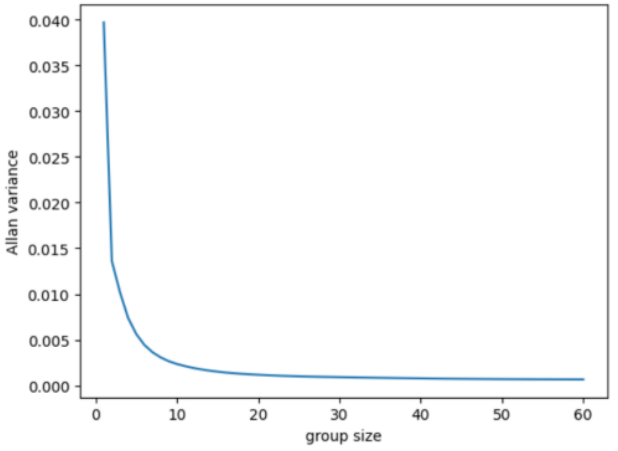

Figure 10. AVAR of signal at 103600 Hz USART Baud Rate

The coefficient of a1~a5 in previous model is calculated as below figure 11.

DOI: 10.54254/2755-2721/6/20230899

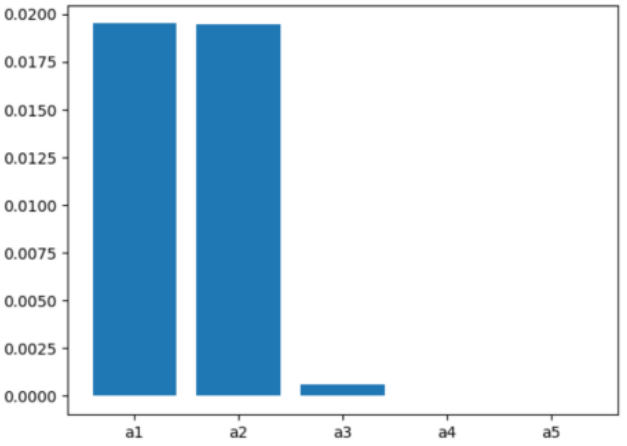

Figure 11. coefficients of model at 103600 Hz USART Baud Rate.

The figure 11 shows that due to the different peak intervals, the values of a1 and a2 varies significantly under different USART baud rate. The distortion of noise signal also affects the shape of AVAR function graph.

We recorded the peak intervals under different USART baud rate configurations. The following table 2 shows a series of measurement. These data prove that the peak intervals are under control of the difference between data read rate and data update rate.

Table 2. Intervals under different USART baud rate.

USART baud rate (Hz)	Interval (Hz)
105400	53.57
103800	5.38
103700	5.56
103600	0.42
103500	4.07
103400	3.95
103300	8.15
103200	13.55
102800	22.79

Although the phenomena has been seen, no definitive conclusion can be drawn on the origins of these noise peaks. Unknown is also the mathematical explanation for the influence of the discrepancy between data read rate and data update rate. However, we propose that the difference between data read rate and data update rate separately shifts different types of noise. The model of the suggestion is displayed below.

The original noise source N(f) consists of 2 signals as

$$N(f) = k_1 QN(f) + k_2 VRW(f) + \cdots$$

Under the mismatch of data read rate and data update rate, the noise signal is distorted as

$$N(f) = k_1 QN(\lambda_1 f + \mu_1) + k_2 VRW(\lambda_2 f + \mu_2) + \cdots$$

While Using Least Square to calculate the coefficient of each type of noise, The model is translated as

$$N(f) = k_1' QN'(f) + k_2' VRW'(f) + \cdots$$

It is difficult to keep that k1 = k1' and k2 = k2' after these steps.

4. Conclusion

This study begins by discussing the AVAR and its application to the analysis of MEMS inertial sensors. AVAR can be derived simply from a signal's PSD, indicating that it does not include additional information. However, because PSD is a frequency-domain analysis tool, it has inherent limitations. In particular, PSD cannot be employed to rate ramp noise since the Fourier Transform is inappropriate for

DOI: 10.54254/2755-2721/6/20230899

this sort of noise. Since AVAR's calculations are performed in the time domain, this constraint does not apply. Finally, AVAR can recognize five distinct types of system noise. We also describe the origins of these sounds. Typically, the cause of Rate/Acceleration Random Walk noise remains obscure.

This study reviews the strategies used to minimize MEMS inertial sensor noise. Typically, we investigate the forms of noise applied by each method. The majority of random noise reduction algorithms concentrate on Gaussian noise or white Gaussian noise. With the use of these techniques, the Velocity Random Walk can be significantly inhibited in terms of the accelerometer.

In the final experiment, we detected and recorded the noise signal of the ADXL346 accelerometer. We demonstrate the challenge of recording the noise signal precisely in this setup. The data read rate of the microcontroller and the data update rate of the sensor must coincide to prevent signal noise distortion. In addition, we investigate this element's impact on the signal and provide a mathematical model to simulate the outcome.

References

[1] R. N. Youngworth, B. B. Gallagher, and B. L. Stamper, "An overview of power spectral density (PSD) calculations," in Optical Manufacturing and Testing VI, H. P. Stahl, Ed., vol. 5869, International Society for Optics and Photonics. SPIE, 2005, p. 58690U. [Online]. Available: https://doi.org/10.1117/12.618478

[2] N. El-Sheimy, H. Hou, and X. Niu, "Analysis and modeling of inertial sensors using allan variance," IEEE Transactions on Instrumentation and Measurement, vol. 57, no. 1, pp. 140–149, 2008.

[3] F. Mohd-Yasin, D. J. Nagel, and C. E. Korman, "Noise in mems," Measurement Science and Technology, vol. 21, no. 1, p. 012001, nov 2009. [Online]. Available: https://dx.doi.org/10.1088/0957- 0233/21/1/012001

[4] R. Walden, "Analog-to-digital converter survey and analysis," IEEE Journal on Selected Areas in Communications, vol. 17, no. 4, pp. 539–550, 1999.

[5] A. Abdullah, M. Yusof, and S. Baki, "Adaptive noise cancellation: a practical study of the least-mean square (lms) over recursive least-square (rls) algorithm," in Student Conference on Research and Development, 2002, pp. 448–452.

[6] M. Z. A. Bhotto and A. Antoniou, "Robust recursive least-squares adaptive-filtering algorithm for impulsive-noise environments," IEEE Signal Processing Letters, vol. 18, no. 3, pp. 185–188, 2011.

[7] "New improved recursive least-squares adaptive-filtering algorithms," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 60, no. 6, pp. 1548–1558, 2013.

[8] P. Gui, L. Tang, and S. Mukhopadhyay, "Mems based imu for tilting measurement: Comparison of complementary and kalman filter based data fusion," in 2015 IEEE 10th conference on Industrial Electronics and Applications (ICIEA). IEEE, 2015, pp. 2004–2009.

[9] K. Fujii, "Extended kalman filter," Refernce Manual , pp. 14–22, 2013.

[10] H. Chu, T. Sun, B. Zhang, H. Zhang, and Y. Chen, "Rapid transfer alignment of mems sins based on adaptive incremental kalman filter," Sensors, vol. 17, no. 1, p. 152, 2017.

[11] H. Xing, B. Hou, Z. Lin, and M. Guo, "Modeling and compensation of random drift of mems gyroscopes based on least squares support vector machine optimized by chaotic particle swarm optimization," Sensors, vol. 17, no. 10, p. 2335, 2017.

[12] C. Jiang, S. Chen, Y. Chen, B. Zhang, Z. Feng, H. Zhou, and Y. Bo, "A mems imu de-noising method using long short term memory recurrent neural networks (lstm-rnn)," Sensors, vol. 18, no. 10, p. 3470, 2018.

[13] S. Tabibian, A. Akbari, and B. Nasersharif, "Speech enhancement using a wavelet thresholding method based on symmetric kullback–leibler divergence," Signal Processing, vol. 106, pp. 184–197, 2015.

[14] A. Dixit and P. Sharma, "A comparative study of wavelet thresholding for image denoising," IJ Image, Graphics and Signal Processing, vol. 12, pp. 39–46, 2014.