Tampere University

Jonathan Shabulinzenze

COMPARISON OF MEMS SENSORS IN MACHINE VIBRATION MONITORING

ABSTRACT

Jonathan Shabulinzenze: Comparison of MEMS sensors in machine vibration monitoring
Master of Science Thesis
Tampere University
Automation Engineering, MSc
March 2020

The advantages of MEMS accelerometers over conventional piezoelectric sensors have led to an increased use of these sensors in condition monitoring systems. These sensors are compact, lightweight, suitable for low power and inexpensive.

The first objective of this master thesis is to study the applicability of consumer grade microelectromechanical (MEMS) based accelerometers when measuring machine vibrations. The second objective is to study the stochastic noises related to MEMS- accelerometer sensors and the last objective is to evaluate the ability of a single board computer such as the Raspberry Pi to process information from MEMS sensors. This is done by comparing the noise characteristic of several MEMS- accelerometers. This performance comparison is done firstly, by comparing their stochastic noise behaviour and secondly, by comparing their performance when they detect oscillatory motion. The tests were carried out on an elevator and on a wheel loader machine. To ensure that all sensors operate under similar conditions and results are comparable, a base was built where the sensors under test were mounted. The sensors were connected to a Raspberry Pi 3b via the I2C bus. The piezoelectric sensor was used together with the MEMS sensors as a reference sensor. Robot operating system (ROS) used to control the robots was used to read and store the sensor data. The stored data were further processed and analyzed in Matlab / Simulink by using methods such as Allan variance, noise spectral density and power spectral density. For oscillation analysis, data preprocessing methods for MEMS accelerometers were investigated. The vibration analysis was performed in both time and frequency domain.

The results showed that MEMS accelerometers contain some noise that might affect vibration results, therefore the preprocessing of data is crucial. Additionally, It was seen that random noises such as Velocity Random Walk (VRW), Bias Instability (BI) and Acceleration Random Walk (ARW) are all common disturbances in a MEMS accelerometer and need to be taken into consideration. Based on vibration analysis results, it was found that the output of the MEMS accelerometers was similar to that of the piezoelectric sensor when considering harmonic frequencies within a given range, but there was a clear difference in terms of amplitudes. Also MEMS accelerometers sensors showed to work well with a raspberryPi since they use a very low power.

Keywords: **MEMS**,**validation**,**vibration**,**stochastic**,**sensor**

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

TIIVISTELMÄ

MEMS-kiihtyvyysanturien edut perinteisiin pietsosähköisiin antureihin verrattuna on johtanut lisääntyneeseen näiden anturien käyttöön kunnonvalvontajärjestelmissä. Nämä anturit ovat pienikokoisia, keveitä ja käyttävät vähän tehoa sekä ovat edullisia.

Tämän diplomityön ensisijainena tavoitteena on tutkia kuluttaja tason kuuluvien mikroelektromekaanisten (MEMS) kiihtyvyysanturien soveltuvuutta koneen värähtelyiden mittaamisessa. Toisena tavoitteena on tutkia MEMS-kiihtyvyysantureitteen liittyviä stokastisia kohinoita ja viimeisenä tavoitteena on arvioida yhden piirilevyn tietokoneen kuten Raspberry Pi:n kykyä käsitellä MEMS-antureista saatavaa tietoa sekä validoida näiden ulostuloja. Tietojen validointia tutkittiin ensinnäkin suorittamalla stokastinen kohina-analyysi ja toiseksi vertaamalla niiden suorituskykyä, silloin kun ne havaitsevat värähtelyliikettä. Käyttökohteena testeissä oli hissi ja pyöräkuormaaja. Jotta kaikki anturit toimisivat samanlaisissa olosuhteissa ja tulokset vertailukelpoisia, rakennettiin alusta, mihin tutkittavat anturit asennettiin. Anturit yhdistettiin Raspberry Pi 3b malliin I2C-väylän kautta. Pietsosähköistä anturia käytettiin yhdessä MEMS-antureiden kanssa referenssianturina. Robottien ohjauksessa käytettyä käyttöjärjestelmää (ROS) käytettiin lukemaan ja tallentamaan anturidata. Tietojen käsittely ja analyysi tehtiin Matlab / Simulinkillä käyttämällä menetelmiä, kuten Allan-varianssi ja meluspektritiheys ja tehospektritiheys. Värähtelyanalyysia varten tutkittiin MEMS-kiihtyvyysantureiden datan esikäsittelymenetelmiä. Värähtelyanalyysi tehtiin sekä aika-, että taajuustasossa.

Tuloksista nähtiin, MEMS-kiihtyvyysantureiden datassa sisälsivät häiriöitä, jotka saattavat vaikuttaa värähtelyn tuloksiin, joten datan esikäsittely on tosi tärkeä. Lisäksi, tulokset näyttivät että satunnaiset kohinat, kuten Velocity Random Walk (VRW), Bias Instability (BI) sekä Acceleration Random Walk (ARW) ovat kuitenkin yleisiä häiriöitä MEMS-kiihtyvyysantureissa ja niiden on otettava huomioon käsittelyssä. Värähtelyanalyysin perusteella havaittiin, että MEMS-kiihtyvyysanturin ulostulo on samanlaista kuin pietsosähköisen anturin, kun tarkastellaan taajuuden harmonisia alueita tietyllä alueella, mutta niiden amplitudissa oli kuitenkin selvä ero.

Avainsanat: **MEMS**,**validointi**,**värähtely**,**stokastinen**,**sensori**

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

PREFACE

This thesis was carried out at Tampere University (TUNI) in the Faculty of engineering and natural sciences at the laboratory of Hydraulics and Automation.

I would like to declare my sincere gratitude to Prof. Kalevi Huhtala for supervising, guiding and supporting as well as offering valuable comments during this thesis process. Gratitude is granted also to Prof. Reza Ghabcheloo for his important comments and for working with Prof. Kalevi Huhtala as examiners of this work. Many thanks to the staff in heavy lab for their support in technical issues. I am also grateful to colleagues Miika, Eemeli and Simo for their presence and fruitful conversation during this thesis process. Additionally, thanks are accorded to former colleague and friend Dr. Janne Koivumäki for his bothersome motivation.

I would like to thank all my family members and especially my parents Anastasie and Robert for their endless love and support all the time. Thanks to my sister Mariette for helping with grammatical issues in this work. Special thanks are accorded to my very close relatives Pirkko and Kari Repo for their financial and spiritual support throughout the course of my studies.This thesis is dedicated to them.

Tampere, 15th March 2020

Jonathan Shabulinzenze

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

$\{0\}$	inertial frame
$\{B\}$	body frame
${}^{B}\mathbf{a}^{*}$	accelerometer sensor output in body frame
${}^{0}\hat{\mathbf{R}}_{B}$	rotation matrix from body frame to inertial frame
${}^{0}\hat{\mathbf{a}}_{v}$	linear acceleration vector in inertial frame
${}^{0}\hat{\mathbf{p}}_{v}$	linear displacement vector in inertial frame
${}^{0}\hat{\mathbf{v}}_{v}$	linear velocity vector in inertial frame
$\hat{\mathbf{a}}$	acceleration vector
AANN	Auto-associative Neural Network
ADC	Analog-to-digital Converter
ANN	Artificial Neural Network
ARW	Acceleration Random Walk
ASV	Algorithmic Sensor Validation
AV	Allan Variance
BI	Bias Instability
DKF	Discrete Kalman Filter
DOF	Degree Of Freedom
EKF	Extended Kalman Filter
$\mathbf{F}$	force
FFT	Fast Fourier Transform
g	gravitational acceleration
GM	Gauss-Markov
GPS	Global Positioning System
HSV	Heuristic Sensor Validation
I2C	Inter-Integrated Circuit
IMU	Inertial Measurement Unit
ISO	International Organization for Standardization
KPCA	Kernel Principal Component Analysis
LPC	Linear Predictive Coding
m	mass
MATLAB	a multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks
MEMS	Micro-Electro-Mechanical Systems
MLP	Multilayer Perceptron
PCA	Principal Component Analysis
PSD	Power Spectral Density
RMS	Root Mean Square
ROS	Robot Operating System
RSS	Root Sum Squared

SI system	international system of units (Système international d'unités in French)
SPI	Serial Peripheral Interface
STD	standard deviation
SVM	Support Vectot Machine
TUNI	Tampere Universities
VRW	Velocity Random Walk

1 INTRODUCTION

Inertial sensors play an important role in our modern electonic devices. These sensors are made of miniaturized capacitive Micro-electro-mechanical system (MEMS) which allow them to have a small size, use low power and to have high level of functionality in terms of application[1]. Additionally, due to the improved reliability and reduced cost, these sensors are gaining an increased popularity in machine condition monitoring. For vibration monitoring application capacitive MEMS accelerometers are being used in the replacement of the traditional vibration monitoring transducers that are based on piezo-electric technology. The advantages of using capacitive MEMS accelerometers in stead of piezoelectric accelerometers are not only due to their low price, but also due to their ability to be easily integrated to existing industrial IoT platform with extremely power consumption as well as reduced cabling requirements. Accelerometers are categorized in terms of grades as consumer grade, automotive grade, industrial grade, tactical grade an navigation grade according to [2][3],where the consumer grade is the lowest and the most inexpensive. These sensors can be found in the market at a price less that 10 €, making them to be widely available for for all users.

Even though capacitive MEMS accelerometers are becoming popular in the application mentioned above, they are still affected by wide noises caused by different sources such as temperature, pressure, magnetic field electric field and other sources. All these noises may lead to a poor and unreliable performance in application that demand high accuracy.

In order to achieve a better quality of performance, MEMS inertial sensors needs to be calibrated at a high level. This calibration is mainly performed by the sensor manufacturer and intend to give the sensor characteristics as well as remove or compensate static or deterministic errors. However the stochastic noises of the sensors still have a big influence in the overall performance of the these sensors. However, since the consumer grade sensor are low cost and mostly designed mostly for hobbyist, the level of calibration is very poor and barely done. In this case the user may need to perform this by his own, causing a lot of extra work.

1.1 Related work and earlier research on MEMS sensor application

MEMS- sensor applications have been a study of concern in recent time as different methodologies have been developed to analyse and study the performance of these sensors. In his work, René [4] evaluated low cost MEMS- accelerometers and investigated inertial algorithms for dead reckoning application in railway environment. He used a method of assigning different weights and criteria for evaluation. His method achieved good results in sensor comparison as a 10% error was achieved while estimating the displacement error on his test platform. Similar work have been done by Göekem Secer[5] in which analyzed the deterministic and stochastic error modeling of inertial sensors and magnetometers. Daniel R.Greenheck[6] studied the design and characterization of low cost MEMS-IMU clusters for precision navigation. In his work, he developed a prototype that incorporate several low cost MEMS-IMUs on a single circuit board. Other work related to analysing and modeling error related to MEMS- inertial senors can be found in [7][8][9][10][11].

Other work have been done by Quinchia Alex et al[12] that compared different error modeling of MEMS sensors applied to GPS/INS integrated systems. In their work different stochastic error model for the measurement noise components of MEMS IMU sensor was derived from experimental data using the autoregressive, wavelet de-noising and Allan variance methods as well as their combination. MEMS-based inertial sensors have found usages in application such as robotics and hydraulic manipulators. Honkakorpi[13] studied MEMS-based motion state estimation and control of hydraulic manipulators. As results, he found that the combination of rigid body motion kinematics with an understanding of efficient, yet straightforward signal processing methods, low-cost and relatively low resolution components can be used in creating innovating solutions in hydraulic manipulator motion sensing. Related to condition monitoring applications, a vibration monitoring of rotating machine using MEMS accelerometer have been done by Chaudhury et al[14], where they propose a basic design for the development of a low cost MEMS accelerometer based vibration sensor by the integration a basic sensor and the intelligent of the vibration analysis methods. Other work analysing the suitability of MEMS accelerometer for condition was done by Albarbar et al[15] using an experimental approach. A alternative work related to the suitability of low-cost MEMS accelerometer when working as vibration monitor was studied in [16], where a market review on potential sensors was done in order to see if low cost MEMS sensors can meet requirements assigned.

1.2 Objectives and requirements of this thesis

In order to insure the reliability of data produced by MEMS- inertial sensors, the source of noises needs to be investigated and a models that can optimally correct these noises needs to be done. The performance level requirements of MEMS sensors vary depending on the application area as well as accuracy characteristics. In many cases, the price of the sensor have a significant contribution while making the choice of the sensor.

In most of the cases sensor manufacturer provide all information related to the product in the data sheet (or spec sheet). However, the information provided is not always complete and it might be difficult for the user to understand what is being stated in the data sheet. Thus, it is critical to have a perfect understanding on what is provided and how to validate that information in order to make decisions that feat well the application on which the sensor will be used.

This thesis have two main objectives:

- The first objective is to study the applicability of MEMS accelerometers when applied for machine vibration monitoring.
- The second objective is to evaluate the performance of a raspberry Pi to process data from several MEMS sensors.
- Finally the validity of data from the MEMS sensors is studied.

To better understand the differences, three type of experiments was performed. Firstly, the static data of sensors was acquired while no motion is applied to the sensors, secondly, the data was acquired on moving elevator and thirdly, the data was acquired on a vibrating wheel loader machine base.

As requirement, sensors to be used was set to be from consumer grade, of a price less that 50 €. It was set that these sensors must be of fairly high accuracy and less noise. The frequency range to be measured by these sensors was set to be up to 300 Hz. More importantly, these sensors must compatible with low power systems of course must have the capability to be integrated into IoT platforms.

1.3 Research methods and limitations

Since MEMS sensor output is affected by different noises, the understanding of the noise source is extremely important in order to validate the data from these sensors. The noise from MEMS- sensors can be categorized as deterministic and stochastic noise. Deterministic noise are not time dependent and are mainly caused by a sensor static bias,

non-orthogonality and axis misalignment. These error parameters can be defined while performing the laboratory calibration and the used in the error compensation algorithms, error in scale factor and temperature effect [17]. Fortunately, these errors can be corrected by utilizing the error compensation models after calibration. For higher grade sensors, this calibration is usually done by the sensor manufacturer, however, for consumer grade sensors the calibration might not be specified properly. Stochastic errors are mostly random noises and time dependent. These noises cannot be corrected during calibration. The errors will be studied more in this work. To analyze these errors, Allan variance and power spectral density (PSD) methods [18][19][9]have been utilized. By knowing the noise characteristics of sensors, different methods can be used to compensate these and reduce their influence in the overall output data. Methods such as Kalman filter, [20] and Gauss-Markov and autoregressive processes[12] can be utilized to model the stochastic errors. Other methods such as moving average[21], wavelet de-noising[22] and median filter can be applied to remove noises and to smooth MEMS sensor data. These methods will be discussed more in details in chapter 3. The stochastic analysis conducted in this work can be applied to other sensors in MEMS- IMU unit but for the purpose of simplicity this thesis focuses only in accelerometer application. Additionally, since the deterministic errors are not the main focus in this work and the application experiment have been done in laboratory environment, where temperature was constant, It was assumed that the effect of temperature change in MEMS sensor is minimal.

The research conducted in this thesis is divided in time domain and frequency domain analysis.

- In time domain analysis, the comparison of sensor noise have been studied. Sensor noises related to static motion was compared. Bias variation of these sensor was analyzed using the most widely used Allan variance algorithm.
- In frequency domain analysis, Fast Fourier Transform (FFT) and the power spectral density (PSD) as well as spectrogram analysis were utilized.

Other details on the methods and methodologies applied for each experiment is explained in chapter 3

1.4 Thesis structure and contributions

The rest of this work is structured as follow: Chapter 2 contains the literature background of MEMS accelerometer functionality as well as terminologies related to noises in MEMS sensors.

Chapter 3 contains the derivation of methods used in this thesis. A literature review related to data validation techniques was discussed and methods used in this work are derived more in details.

Chapter 4 explains how the experiment setup was constructed and the characteristics of sensors used in this work.In this chapter it is different method used in data acquisition.

In chapter 5, the results obtained from experiments conducted in this work are presented along with a discussion on relevant findings.

Chapter 6 expresses what was done in a compact form and points out the major findings of this work.

Additional information is provided in the Appendix after Chapter 6.

2 BACKGROUND

2.1 Background and Applications of MEMS sensors

In it's simple way of operation, an accelerometer consist of a mass, also called proof mass, supported by a spring as it is illustrated in figure 2.1 From the figure, Newton's second law can be formulated as:

$$m\ddot{x}_m = F_s - mg \tag{2.1}$$

For the spring with natural length l_0 ,the relationship between the force and extension d is:

$$F_s = kd, \tag{2.2}$$

where the correlation between the displacement is:

$$x_b - (l_o + d) = x_m \tag{2.3}$$

Taking the double derivative of this equation and substituting it into equation 2.1,this produce

$$\ddot{x}_b - \ddot{d} = \frac{1}{m}(kd + mg) \tag{2.4}$$

The quantity desired to measure is the acceleration $a = \ddot{x}_b$ and the relative displacement of the proof mass. However, it was assumed that $\ddot{d} = 0$ in steady state, but in reality there would be a dumping element that will increase the friction and stop the mass from oscillating. Because of this, a term $-B\dot{x}_m$ is added to the right hand side of equation 2.1. The relative displacement of the prof mass is linearly related to the acceleration according to equation 2.5.

$$d = \frac{m}{k}(a + g) \tag{2.5}$$

The displacement is measured and scaled by the factor of k/m such that the output of the sensor is $a^* = a + g$ in m/s^2

If the accelerometer is placed on a completely horizontal table, the measured acceleration would be $a^* = 0 + g$ in the upward direction, because only the gravitational force mg is

acting on the sensor according to Newton law as shown in equation 2.1. However, in many applications, the accelerometer output is referred as proper or inertial acceleration. Accelerometers measures acceleration typically in a single axis. However, three dimentianal accelerometer can be done by arranging three similar accelerometers such that their sensitive axes are aligned orthogonally. The triaxial accelerometer output is the components of vector $^{B}a^{*}$ measured in body frame $\{B\}$.[23]
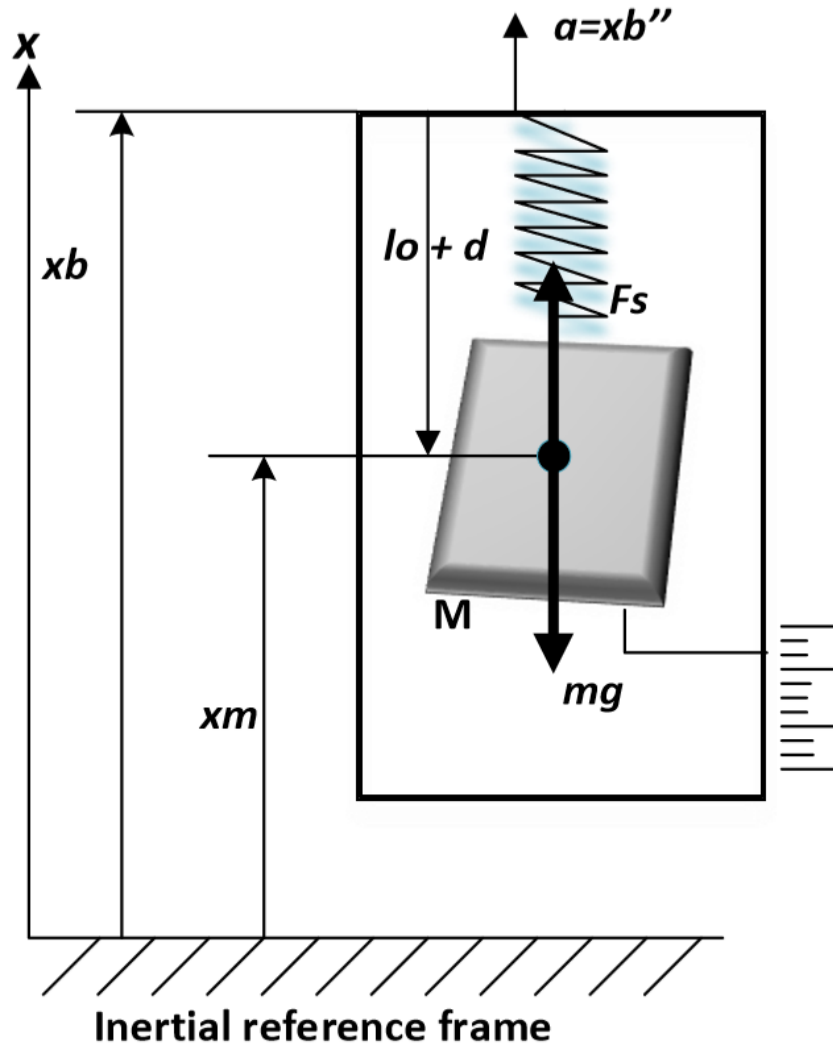
Figure 2.1. *Simple Accelerometer working principle and it's elements [23]*

In inertial navigation applications, the estimate of the vehicle motion is frequently done in inertial frame $\{0\}$ rather than in body frame $\{B\}$. However, moving systems will experience both acceleration due to gravity and acceleration due to motion when MEMS sensor are being used. In applications where only the acceleration due to motion is needed will require to transform the sensed acceleration to inertial frame, or in other word the

acceleration due gravity need to be removed from acceleration data. In navigation systems, this can be done by applying complex multiple sensor fusion algorithms such as Kalman filter. Equation 2.6 show a simplified method how to remove gravity from acceleration data.

$$^{0}\hat{\mathbf{a}}_{v} =^{0}\hat{\mathbf{R}}_{B}^{B}\mathbf{a}^{*} -^{0}\mathbf{g}, \tag{2.6}$$

where $^{0}\hat{\mathbf{R}}_{B}$ is the rotation matrix from body frame to inertial frame. Other approach to remove the gravity from accelerometer data is using a high pass filter that let pass high frequency components and attenuate constant components as well as slowly varying components such as gravity.

By assuming that $^{0}\hat{\mathbf{R}}_{B}$ and g are known or using simple filtering methods, one can integrate the acceleration in order to get the velocity as in Equation 2.7

$$^{0}\hat{\mathbf{v}}_{v}\left(t\right) = \int \hat{\mathbf{a}}_{v}\left(t\right) dt \tag{2.7}$$

and further integrate the velocity in order to get the displacement as in Equation 2.8 .

$$^{0}\hat{\mathbf{p}}_{v}\left(t\right) = \int \hat{\mathbf{v}}_{v}\left(t\right) dt \tag{2.8}$$

The sensing principal of micromechanical accelerometers can be grouped according to [24] to piezoresistive sensing, capacitive sensing and piezoelectric sensing. The piezoresistive sensing is based on piezoresistors that are integrated onto the spring in the sens that the piezoresistor resistance changes when subjected to acceleration induced stress. In this case the acceleration can be obtained by measuring the change in the resistance.This sensing method is known to be robust and simple to implement, its suffers for a poor noise and power performance[24] Piezoelectric sensing mechanism is based on a charge polarization of piezoelectric materials due to the strain that is caused by the inertial force. The simplest configuration of this sensing is that the proof mass is attached to a piezoelectric plate that acts as a spring. This plate generates current that is proportional to the change in acceleration. Some of the drawback of this type of sensing is that the sensor can only measure the change in acceleration and cannot measure constant acceleration such as gravity. This mechanism is used mostly in macroscopic sensors [24]. The capacitive sensing mechanism is based on detecting small changes in capacitance due relative movement of the proof mass and the frame. This mechanism is the most widely found in MEMS sensors and are at this time the most odopted since they are less expensive, perform well i terms of noise and power consumption.In this work capasitive MEMS sensors are the main fucus. However, a piezoelectric based accelerometer is used mostly for comparison purposes. More details on these sensing methodologies have been

clearly opened in [24].

In applications related to attitude and heading reference systems (AHRS) parameters used in filtering algorithms are modeled with stochastic information derived from Allan Variance curves. In this case the choice of accelerometer is highly dependent on the application. However accelerometer needs to be calibrated in order to output the useful information. The purpose of the sensor calibration is to find the gain or scale factor and the bias/offset as well as their temperature dependency, the cross-axis sensitivity or misalignment in case of 3D accelerometers.[23] When an accelerometer is in the static position, it senses the acceleration due to gravity in the downward direction. This acceleration is due to the material in the Earth beneath us and the distance from the Earth's center. Nonetheless, the earth is not a perfect sphere, which means that points in the equatorial region are further from the center than points in polar area. As is stated in [23], the gravitational acceleration can be approximated by $g \approx 9.780327(1+0.0053024sin^2\phi-0.0000058sin^22\phi)-0.000003086h$, where ϕ is the angle a latitude and h is the height above sea level.

2.2 MEMS- Accelerometer Error Model

Due of their imperfectness, when it comes to errors, MEMS- IMUs contain different error that originate from different sources. In order to get rid of those error, an understanding of their source is very important, after that a mathematical modeling of these error need to be done. Based on the error model of the IMU sensor, a compensation model is required in order to eliminate or reduce errors containing IMU data. In this section the error model of an accelerometer and gyroscope is formulated and a compensation model is studied. Equation 2.9 [17][25][26] illustrates the overall error model of MEME accelerometers. The error parameters are explained bellow:

$$\begin{bmatrix} \hat{a}_x \\ \hat{a}_y \\ \hat{a}_z \end{bmatrix} = \begin{bmatrix} 1+S_x+\delta S_x & M_{xy} & M_{xz} \\ M_{yx} & 1+S_y+\delta S_y & M_{yz} \\ M_{zx} & M_{zy} & 1+S_z+\delta S_z \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} B_x+\delta B_x \\ B_y+\delta B_y \\ B_z+\delta B_z \end{bmatrix} + \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \quad (2.9)$$

where a:acceleromenter and gyroscope output signal before transformation to real physical value.

$\hat{a}$: accelerometer and gyroscope output after transformation

$S, \delta S$: scale factor error and scale factor error instability

$B, \delta B$: bias/drift and bias/drift instability

B_g: g-dependent bias error coefficient

n: wight/random sensor noise

M: non-orthogonality/ misalignment errors.

MEMS errors are categorized into deterministic and stochastic errors. Deterministic sensor errors includes the sensor bias, the scale factor error and the misalignment errors. As stated earlier in section 2 the bias error define the sensor output when no input is applied. Sensor bias may include different term such as:

- fixed terms

- temperature variation

- turn-on to turn-on variation

- in-run variation

The scale factor error is the error due to the ratio of change in the output signal to the change in input signal of the IMU sensor. Similar to bias error, the scale factor error contain fixed terms, temperature and other error terms due to asymmetry and non-linearity[17].

Misalignment errors are due non-orthogonality of axis of sensitivity in IMU sensors. This is mostly caused by the mechanical manufacturing failure. The correlation between IMU axis due to misalignment can seen in above equations.

In order to eliminate the deterministic errors in sensor data, full laboratory calibration of the sensor is required. I general this calibration is performed by the sensor manufacturer. However, for own purposes, the user can be interested in performing the calibration. Different algorithms can be used to calibrate the IMU sensors. The basic approach for IMU sensor calibration is to use the so called "Six Position Direct Method", or "Six Position Weighted Least Squares Method " as shown in [26]. For more investigation of IMU sensor calibration and error compensation,refer to [25, p. 253][27][28][29][30]. Since MEMS sensors are pre-calibrated by the manufacturer, all deterministic errors are assumed constant and only the stochastic errors will be analyzed in order to see the difference of low-cost MEMS sensors compared to a high cost sensor.

As it was mentioned in section 2, stochastic errors are random errors that appear in IMU data due to random variation of bias and scale factor over time. In the stochastic error, there are high and low frequency noises that affect the data. Random changes in bias and scale factor errors are low frequency components of the stochastic errors. However, noise

due to measurement produce high frequency component of stochastic errors. According to [25][17] the cause of stochastic noise is due flicker noise in electronics and effect caused by interference of the signal.

Bias instability (flicker noise) appear due to change of bias in time. In order to analyze the characteristics of stochastic errors, different methods can be utilized. The Allan variance, power spectral density and autocorrelation methods are the most widely used for this purpose [18]. The Allan variance is studied more in details later. The result of Allan variance gives a way to start modeling the stochastic error and their compensation method. In recent studied Kalman filter [31] have been the state of the art for modeling and analyzing stochastic noises In this thesis the Kalman filter algorithm is introduced and the working of this method is applied to models MEMS- accelerometer stochastic errors.

3 MEMS ACCELEROMETER STOCKASTIC ERROR MODELING

3.1 Allan Variance Background

Also known as $two-sample variance$, the Allan variance $AVAR$ was developed $David W. Allan$ in 1966 to measure the frequency stability in clocks, oscillators and amplifiers in time domain.However,it can also be used to identify different kind of noise source that are present in different measurement instruments. such as gyroscope and accelerometer sensors. The effectiveness of Allan variance is seem when the data is plotted in a logarithmic scale, where different errors including in the data can be distinguished by investigating the varying slop on the plot. [32]. According to the author of [18], the Allan variance is simply a method of representing root mean square (RMS) random drift error as a function of averaging time.

3.1.1 Allan variance cluster sampling techniques

A sampling technique for data analysis of ring laser gyroscope was introduced by Tehrani et all[33] in 1983. This technique have been widely used in inertial sensor stochastic error analysis[34]. In order to analyse IMu data with Allan variance approach, data consisting of N data points is acquired at sample time τ_0. This data is further divided in M groups that contain n successive data points as it can been seen in Figure 3.1,
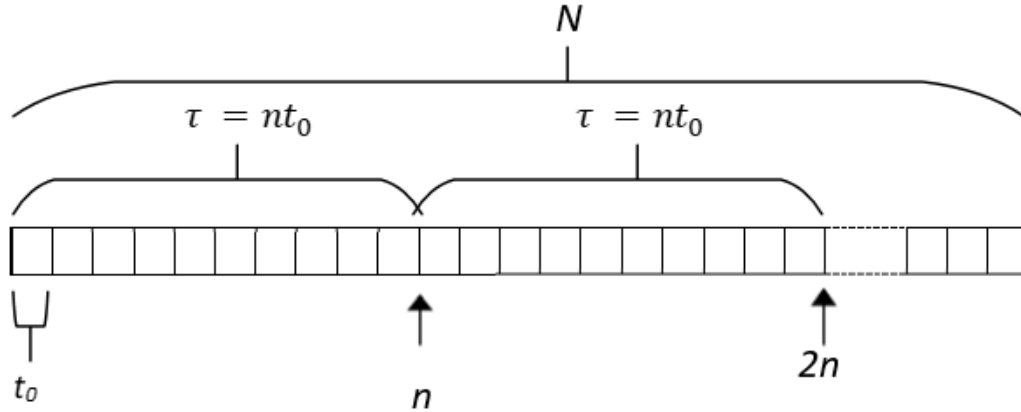
Figure 3.1. *Allan variance algorithm scheme structure[11]*

where each group covers the time period $\tau = n\tau_0$. In this case τ is referred as cluster time, which is the length of the period over which windows the averaging was done. The average of each cluster is referred as cluster sample[34]. Different ways of performing the cluster sampling have been a subject of research in recent time. The simplest methods use the technique overlapping and the non-overlapping sampling for Allan variance analysis. Here bellow is shortly the difference between overlapping and non-overlapping sampling technique of Allan variance.

- Non-overlapping Allan variance

  In the non-overlapping, two sample variance are taken as fundamental measure of frequency stability. The time averaging is computed on either fractional frequency or phase measurements, similarly, the Allan deviation is derived for inertial sensors based on both typed of measurements. For accelerometer measurements the velocity data is utilized for Allan Variance measurements[34]

- Overlapping Allan variance

  In overlapping sample technique, all possible combinations of data from a certain distance are performed. This have an advantage in increasing the effective number of degree of freedom and in improving the confidence of the estimation but in expense of the computation time according to [34]. More improvement on cluster sampling techniques for Allan variance analysis is studied in [35][36], where the fully overlapping cluster sampling have been extended to not fully overlapping, which improve the computation time. Li et all[35] introduces the total variance and the modified total variance techniques for computing Allan variance. Yadav et. all [36], introduce a fast parallel algorithm that improves the fully overlapping Allan variance in terms

of computation speed.

3.1.2 Power Spectral Density

Power spectral density (PSD) is one of most powerful tools used to characterize data and performing stochastic modeling. It is considered as the most commonly used technique to represent the spectral decomposition of time series data. Moreover, PSD is also better suited to analyzing periodic and non-periodic signals than other methods according to [37][18]. PSD analysis can be used in the same purposes as Allan variance. The basic relationship for stationary process between the two-sided PSD $S(\omega)$ and the covariance $K(\tau)$ is expressed as a Fourier transform pairs in equations 3.1 and 3.2bellow:

$$S(\omega) = \int_{-\infty}^{\infty} e^{-j\omega\tau} K(\tau) d\tau \tag{3.1}$$

and

$$K(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{j\omega\tau} S(\omega) d\omega \tag{3.2}$$

As stated in [18],the transfer function form of the stochastic model can be estimated from the PSD of the output data. In case of linear systems, the output PSD is a product of the input PSD and the magnitude squared of system transfer function. If the state space methods are used, the PSD matrices of the input and output are related to the system transfer function matrix by Equation 3.3 bellow:

$$S_{yy}(\omega) = H(j\omega)S_{xx}(\omega)H^{*T}(j\omega), \tag{3.3}$$

where

- H: transfer function of the system
- H^{*T}: complex conjugate transpose of H
- S_{yy}: output PSD
- S_{xx}: input PSD

In the case of white noise input, the output PSD gives directly the system transfer function according to[18].

3.1.3 Allan Variance Formulation

Suppose that there are N consecutive data points in the measurements and each data point having sample time of t_0. A group of n consecutive data points $\tau_0, 2\tau_0, ...n\tau_0$ such that $n < N/2$ can be former considering that each member of the group is a cluster and is associated with the cluster time τ equal to nt_0 as it can be depicted in figure 3.1. If the instantaneous output rate of inertial sensor is $\Omega(t)$[10], the cluster average can be defined as

$$\bar{\Omega}_k(\tau) = \frac{1}{\tau} \int_{t_k}^{t_k+\tau} \Omega(t)\, dt, \tag{3.4}$$

where $\bar{\Omega(t)}$ is the cluster average of the output rate for a cluster from kth data point to n data point. The subsequent cluster average can be defined as

$$\bar{\Omega}_{next}(\tau) = \frac{1}{\tau} \int_{t_{k+1}}^{t_{k+1}+\tau} \Omega(t)\, dt, \tag{3.5}$$

where $t_{k+1} = t_k + T$. The difference between clusters can be formed by performing the average operation for each consecutive cluster as shown in equation 3.6.

$$\xi_{k+1,k} = \bar{\Omega}_{next}(\tau) - \bar{\Omega}_k(\tau) \tag{3.6}$$

The Allan variance of length T can be defined as in Equation 3.7 below

$$\sigma^2(\tau) = \frac{1}{2(N-2n)} \sum_{k=1}^{N-2n} \left[\bar{\Omega}_{next}(\tau) - \bar{\Omega}_k(\tau)\right]^2, \tag{3.7}$$

where it can be noted that any finite number of data points N, can form a finite number of clusters of fixed length τ.However, the variance $\sigma^2(\tau)$ is an estimate value and it's quality depends in the number of clusters that can be formed. In the case MEMS accelerometer sensors, the Allan variance is formed in terms of velocity by integrating the acceleration data according to Equation 3.8.

$$\theta(t) = \int^t \Omega(t)\, dt. \tag{3.8}$$

As stated in [10],the lower integration limit is not specified because only the velocity are employed in the definitions.

The measurements of velocity from IMU sensor can be provided in discrete time given by $t = kt_0$, where $k = 1, 2, 3, ..., N$ This notation can be simplified as $\theta_k = \theta(kt_0)$, and

therefor Equation 3.8 can be simplified to Equation 3.4

$$\bar{\Omega}_k\left(\tau\right) = \frac{\theta_{k+n} - \theta_k}{\tau} \qquad (3.9)$$

and Equation 3.5 to Equation 3.10

$$\bar{\Omega}_{next}\left(\tau\right) = \frac{\theta_{k+2n} - \theta_{k+n}}{\tau}. \qquad (3.10)$$

The Allan variance can be computed from Equation 3.7 as

$$\sigma^2\left(\tau\right) = \frac{1}{2\tau^2\left(N - 2n\right)} \sum_{k=1}^{N-2n} \left(\theta_{k+2n} - 2\theta_{k+n} + \theta_k\right)^2 \qquad (3.11)$$

The relationship between the Allan variance $\sigma^2\left(\tau\right)$ and the two-sided power spectral density (PSD) of random noise parameters in the original data set can be formulated according to [18] as

$$\sigma^2\left(\tau\right) = 4 \int_0^\infty S_\Omega\left(f\right) \frac{\sin^4 \pi f \tau}{\left(\pi f \tau\right)^2} \, df, \qquad (3.12)$$

where $S_\Omega\left(f\right)$ is the power spectral density of the random process $S_\Omega\left(\tau\right)$

The power spectral density of any physically meaningful random process can be substituted in the integral, and an expression of the Allan variance $\sigma^2\left(\tau\right)$ as a function of cluster length can be identified.

A log-log plot of the square root of the Allan variance $\sigma^2\left(\tau\right)$ with respect to time τ provides a means of identifying and quantifying various noise terms that exist in the MEMS sensor data[11]

3.1.4 Determination of sensor noise parameters using Allan Variance

The key attribute of the method is that it allows for a finer, easier characterization and identification of error source and their contribution to the overall noise statistics. In this study focuses 3 stochastic noises namely:

- velocity random walk
- acceleration random walk
- bias instability
- Exponentially correlated

Additionally,quantization noise, rate ramp, sinusoidal noise and exponentially correlated or ($Markov$) noises can be identified through the same Allan variance method. Note that any number of random noise components may occur in sensor data depending mostly on the type of sensor and and frequently on the environment where the sensor is being used. If the noise sources are statistically independent, then the overall Allan variance can be taken as the sum of the squares of each noise source according to [11][10].

1. **Angle/Velocity Random Walk (A/VRW)** This noise can be refered as the additive white noise of the output from MEMS sensors.As stated in [18], the main source of this noise is the spontaneously emitted photons that are always present in the output data. The velocity random walk is the high frequency noise that have correlation time that is much shorter than the sample time can contribute to the accelerometer velocity random walk. The noises are characterized by a white noise spectrum on the IMU output rate. Most of these noise source can be eliminated by improving the design. The associated rate PSD is:

$$S_\Omega\left(f\right) = N^2, \tag{3.13}$$

where N is the angle or velocity random walk coefficient. By substituting Equation 3.13 in Equation 3.7, the integration yields:

$$\sigma^2\left(\tau\right) = \frac{N^2}{\tau} \tag{3.14}$$

From the Figure 3.2 it can depicted that the log-log plot of $\sigma\left(\tau\right)$ versus τ has a slope equal to $-1/2$. The numerical value of N can be obtained directly by reading the slope line at $\tau = 1$
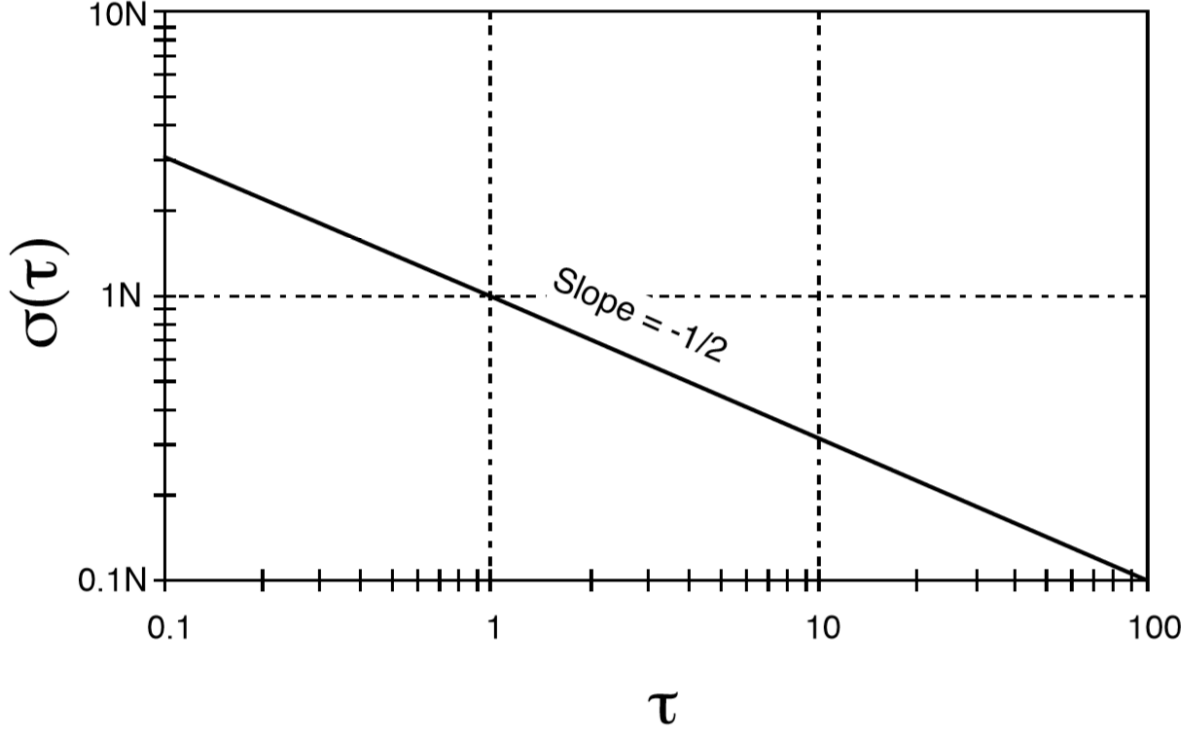
Figure 3.2. *plot of angle random walk[18]*

2. **Bias instability (BI)**

   This noise is due to electronic or other components that are succeptible to random flickering. The instability shows up as bias fluctuation in the data because of its low frequency nature. The associated rate PSD is given as:

   $$S_\Omega(f) = \begin{cases} \left(\frac{B^2}{2\pi}\right)\frac{1}{f}, & \text{if } f \leq f_0. \\ 0, & \text{otherwise.} \end{cases} \tag{3.15}$$

   where B is the bias instability coefficient and f_0 is the cutoff frequency.

   Again by substituting the Equation 3.15 into Equation 3.7 the integration yields:

   $$\sigma^2(\tau) = \frac{2B^2}{\pi}\left[ln2 - \frac{\sin^3 x}{2x^2}(\sin x + 3x\cos x) + Ci(2x) - Ci(4x)\right], \tag{3.16}$$

   where x is $\pi f_0\tau$ and Ci is the cosine-integral function as stated in [18] This can be illustrated in Figure 3.3 as the log-log plot of the Equation 3.16 .It can be seen that the Allan variance for bias instability reaches a plateau (the highest)for τ that is much longer than the inverse cutoff frequency. In this case by examining the flat region of the plot, the limit of the bias instability and the cutoff frequency of the underlying flicker noise can be estimated [18]
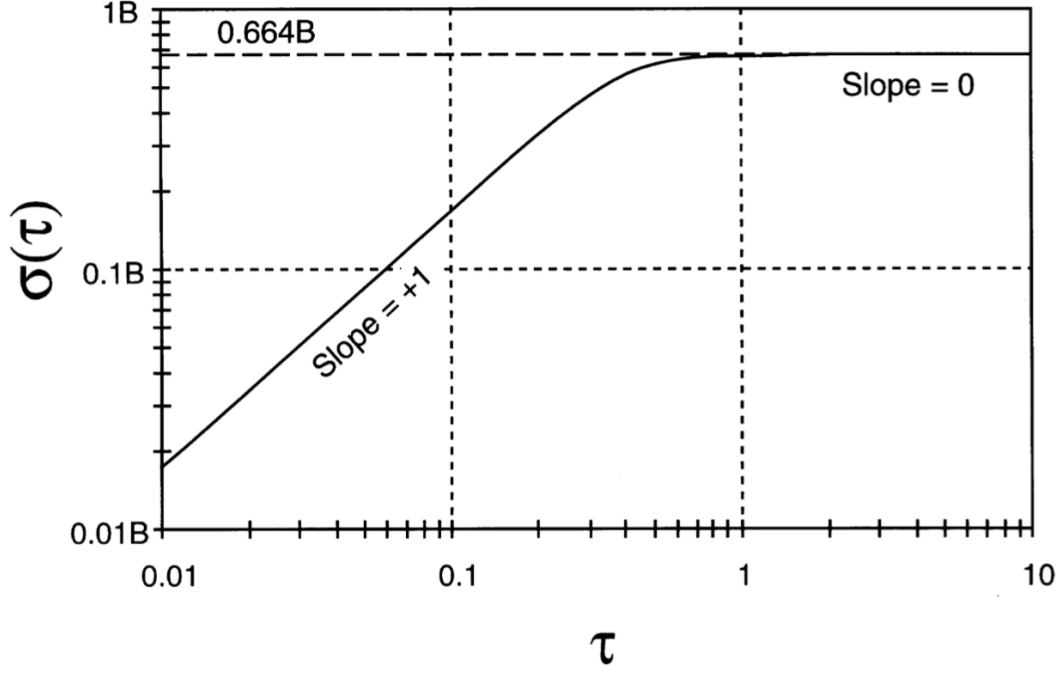
Figure 3.3. *Bias instability for $f_0 = 1$ [18]*

3. **Acceleration/Rate Random Walk (ARW)** The rate random walk is the random process of the uncertainty origin in the data. It is possibly a limiting case of an exponentially correlated noise with a very long correlation time as stated in [18]. The associated rate PSD of rate random walk is:

$$S_\Omega(f) = \left(\frac{K}{2\pi}\right)^2 \frac{1}{f^2},$$ (3.17)

where K denotes the rate random walk coefficient.

By substituting equation $Eq.4.18$ in to equation $Eq.4.9$ and performing the integration this yields:

$$\sigma^2(\tau) = \frac{K^2\tau}{3}$$ (3.18)

The rate random walk is presented by a slope of $1/2$ on a log-log plot of $\sigma(\tau)$ versus τ. Figure 3.4 illustrates the rate random walk, where the magnitude of the noise can be read off the slop line at $\tau = 3$.
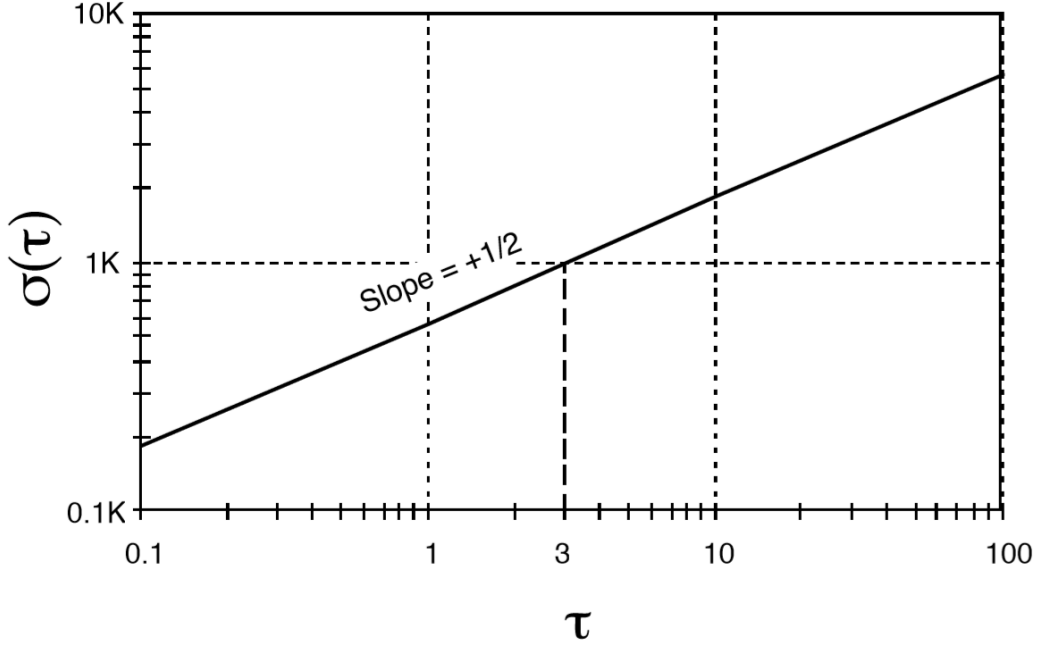
Figure 3.4. *Rate random walk plot[18]*

4. **Markov or Exponentially correlated noise**

The Markov noise is characterized by an exponentially decaying function with a finite correlation time. Its rate PSD is given by[18]

$$S_\Omega(f) = \frac{(q_c T_c)^2}{1 + (2\pi f T_c)^2},\tag{3.19}$$

where q_c and T_c are the noise amplitude and the correlation time respectively. Substituting the above equation in $Eq.4.9$ gives:

$$\sigma^2(\tau) = \frac{(q_c T_c)^2}{\tau}\left[1 - \frac{T_c}{2\tau}\left(3 - 4e^{-\frac{\tau}{T_c}} + e^{-\frac{2\tau}{T_c}}\right)\right].\tag{3.20}$$

The log-log plot of the Equation above can be seen in the figure Figure3.5. Various limits of this equation can be examined for τ that is much longer than the correlation time as it can be seen in Equation 3.20

$$\sigma^2(\tau) \Rightarrow \begin{cases} \frac{(q_c T - c)^2}{\tau}, & \text{for } \tau \gg T_c. \\ \frac{q_c^2}{3}\tau, & \text{for } \tau \ll T_c. \end{cases}\tag{3.21}$$

It can be noted that for $\tau \gg T_c$ the Allan variance in Equation id is the angle/velocity random walk, where $N = q_c T_c$ is angle/velocity random walk coefficient. However for τ much smaller than the correlation time, Equation 3.20 gives the
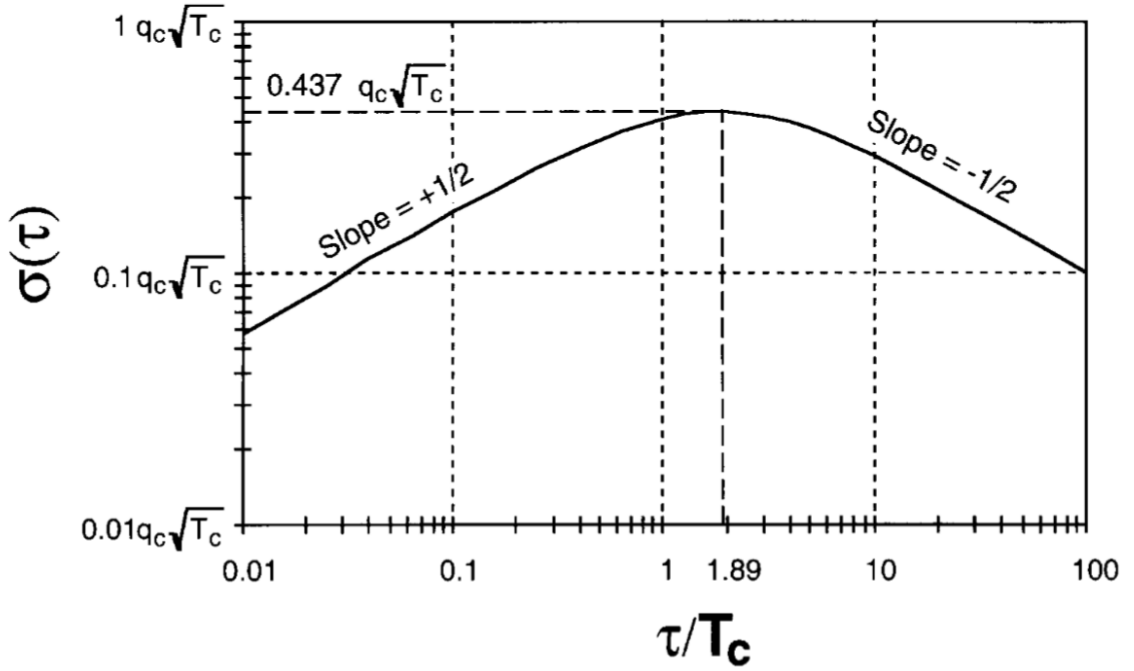
Figure 3.5. *Markov (correlated) noise plot[18]*

Allan variance for rate random walk [18]

From the peak of Figure 3.5 the parameters of the fist order Gauss-Markov process in Equation 3.22[18] can be retrieved as: $T_c = \tau_{peak}/1.89$ and $q_c = \frac{\sigma(\tau_{peak})}{0.437\sqrt{T_c}}$

$$\dot{x}(t) = -\frac{1}{T_c}x(t) + q_c n(t) \tag{3.22}$$

Figure 3.6 shows the overall plot of all noises defined above.Generally, any number of random process noise can be present in IMU dataset, in that sense this plot the typical plot that is seen in for the Allan variance. As it can be seen in the figure, different noises terms appears in different noise region of τ, which helps in identifying different random processes existing that in the data. By assuming that all the random noises are statistically independent, the the Allan variance at any given cluster time τ is the sum of all Allan variances caused by individual random process at the same time. In this case, in order to estimate the amplitude of a given random noise in any region of τ will requires a knowledge of the amplitudes of the random noises in the same region of τ[18].

$$\sigma_{tot}^2(\tau) = \sigma_{ARW}^2(\tau) + \sigma_{quant}^2(\tau) + \sigma_{BiasInst}^2(\tau) + ... \tag{3.23}$$
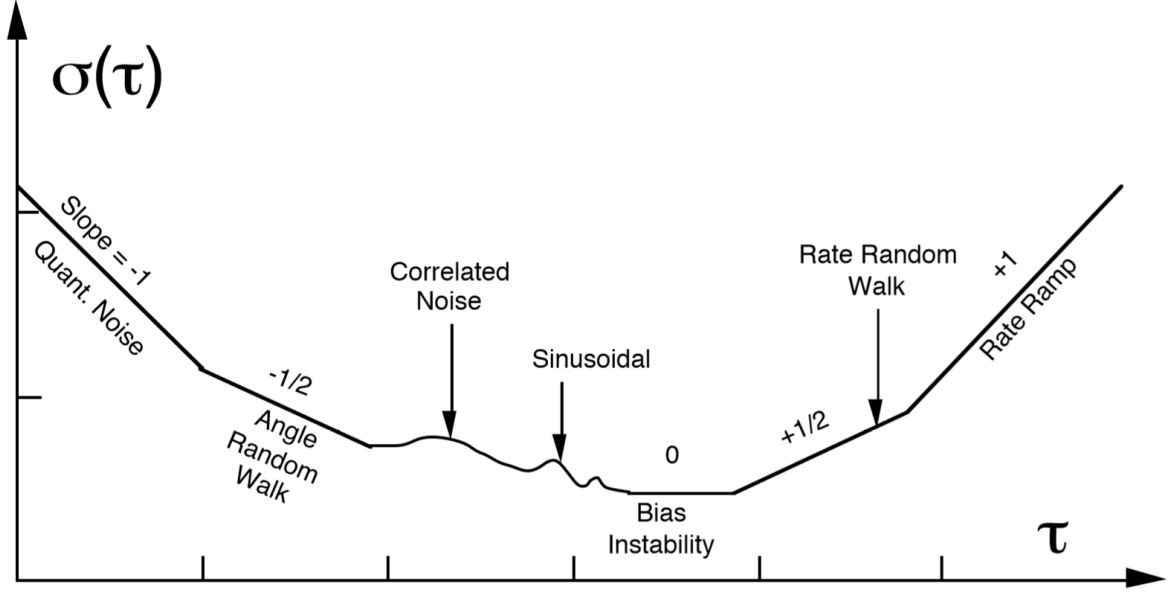
Figure 3.6. *Overall plot of Allan variance analysis[18]*

3.1.5 Allan Variance Estimation Accuracy

As stated in [10], there would exist a gradual transition between the different Allan standard deviation slopes. As consequence a certain amount of noise or harsh would appear in the plot curve due to the uncertainty of the measured Allan variance.

Having any finite data set, a finite number of clusters can be formed. The Allan variance of any noise term can be estimated using the total number of clusters of a given length that can be created. However, the Allan variance estimation accuracy at any given τ depends on the number of independent clusters within the data set. While estimating $\sigma(\tau)$, the percentage error σ for clusters containing K data points from a data set of N points is given by [18].

$$\sigma = \frac{1}{\sqrt{2\left(\frac{N}{K} - 1\right)}} \tag{3.24}$$

From Equation 4.29, it can be noted that the estimation errors in regions of short (long) τ are small (large) as the number of independent clusters in these regions is large(small). This equation is very useful in designing a test to observe a particular noise of certain characteristics to within a given accuracy. For instance, in order to verify the existence of random process within a characteristic time of $24h$ in a data set within an error of 25% one can set $\sigma = 0.25$ in Equation 4.29 and get:

$$K_{max} = \frac{N}{9} \tag{3.25}$$

Because the suspected characteristic time is $24h$, clusters of the same length are created. In this case, the total test length that is needed for such a test is $24 \times 9 = 216h$. [18]

3.2 Gauss-Markov Autocorrelation

Various methods for stochastic modeling of inertial sensor error have been discussed in[26] namely Random constant model, Random Walk Model, Gauss-Markov model, Autoregressive Model and Allan variance. In this thesis, first-order Gauss-Markov model (GM) will be used along with the Kalman filter to model the MEMS-accelerometer flicker noises. More information on these methods can be found in [38][39]. For better understanding of stochastic models, some terminologies have been defined in[38][26]:

- **Continous time signals**: signals described by an analytical function of time

- **Discrete time signals**: signals that have values only at discrete time instants or in other words signal generated by sampling the continous time signal

- **Stationary stochastic process**: process whose joint probability distribution as well as its mean and variance does not change when shifted in time or space.

- **Autocorrelation function**: expected value of a product of random signal with a time-shifted version of itself.

As stated in [38], GM random process is one of the critical methods for modeling stochastic errors. Its benefit is that it can represent large number of physical processes with a reasonably high accuracy and its implementation being relatively simple. Gelb et al.[38] defined a stationary Gaussian process that has an exponentially decaying autocorrelation as the first-order GM process. The autocorrelation function can be given as:

$$R(\tau) = E(x(t) \cdot x(t + \tau)) = \sigma^2 e^{\frac{-|\tau|}{T_c}}, \tag{3.26}$$

where τ, T_c and σ^2 denote the time shift, correlation time and the noise variance at $\tau = 0$ respectively. An other important characteristic of GM process is it's ability to represent the bounded uncertainty, meaning that any correlation coefficient at any time shift is less or equal to the correlation coefficient at zero time shift $R(\tau) \leq R(0)$[26][38].

The ideal first order GM autocorrelation function is illustrated in figure 3.7 below.
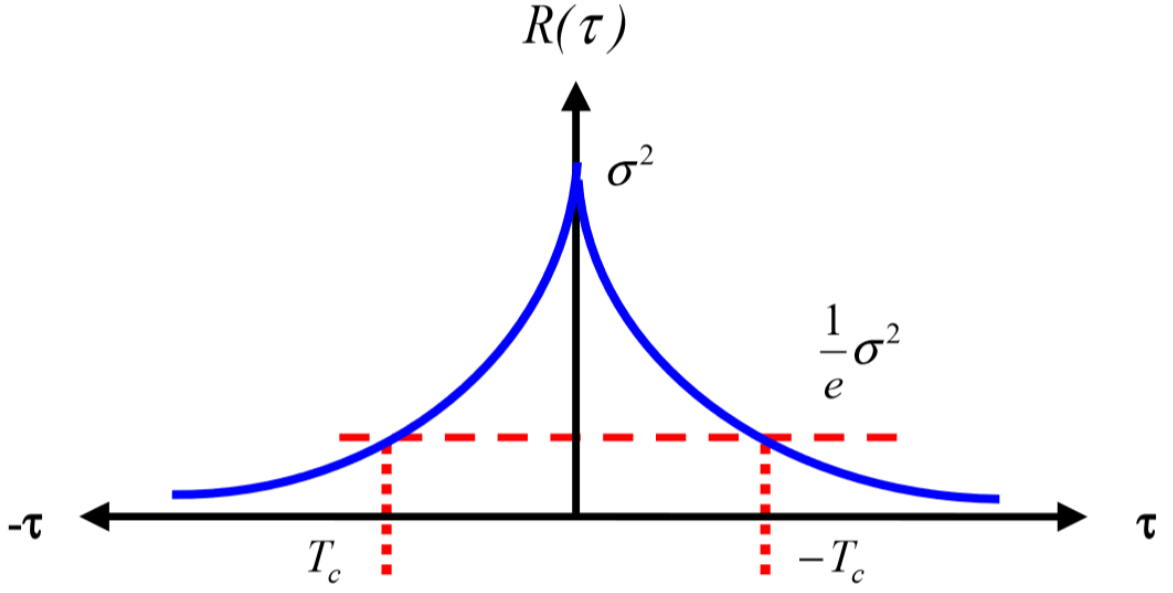
Figure 3.7. *first order Gauss-Markov process autocorrelation function[26]*

The discrete time form Of fist-order GM process is defined in [39][26] as:

$$x_k = e^{\frac{-dt}{T_c}} x_{k-1} + w_k \tag{3.27}$$

The associated variances are given as:

$$\sigma_{x_k}^2 = \frac{\sigma_{w_k}^2}{1 - e^{\frac{-2dt_k}{T_c}}} \tag{3.28}$$

and

$$\sigma_{w_k}^2 = \sigma_{x_k}^2 \left(1 - e^{\frac{-2dt_k}{T_c}}\right), \tag{3.29}$$

where σ^2 is the variance and w normally distributed driven random noise.

Higher order GM random process have been studied in[26], but for simplicity, only the first order model will be implemented. Equation 3.27 and the driven noise variance in Equations 3.28 and 3.29 can be used as error compensation model in sctochastic modeling of MEMS accelerometer. In this study, the bias instability will be modeled as First order Gaus-Markov to tune the Kalman filter in stochastic error compensation. This is studied more in details in the next section.

3.3 Kalman Filtering for Stochastic error tracking and compesation

Kalman filter (KF) was introduced by Rudolf Emil Kalman[20] in 1960 as a recursive solution to discrete data linear filtering problem. Since that time the Kalman filter have

been the subject to extensive research and application due to its advances in digital computing. Kalman filter is widely used in areas such as autonomous, aerodynamics, maritime and many others. [40]. More literature on background of Kalman filtering can be found in [41][42]. An algorithmic based introduction to discrete Kalman Filter (DKF) and extended Kalman filter (EKF) is provided in[43]. KF states for linear processes and EKF states for non-linear processes.

In next subsection a mathematical formulation of Kalman Filter estimation.

3.3.1 The process to be estimated

The general purpose of the Kalman filter is to try to estimate the state variable $\mathbf{x} \in \Re^n$ of the discrete-time process that is controlled by the linear stochastic difference equation with measurement $\mathbf{z} \in \Re^m$ [43]

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k\mathbf{w}_{k-1} \tag{3.30}$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \tag{3.31}$$

where $\mathbf{k}$ denotes the time step and the random variable $\mathbf{w}_k$ and $\mathbf{v}_k$ denote the process and measurement noise, respectively. In many applications these noises are assumed to be independent to each other, white and normally distributed as shown in equation below:

$$p(\mathbf{w}) \sim N(0, \mathbf{Q}) \tag{3.32}$$

$$p(\mathbf{v}) \sim N(0, \mathbf{R}), \tag{3.33}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are the process and measurement noise covariance matrices. Practically, these noise covariance matrices might change at each time step or measurement. At this stage, it is assumed that they remain constant. Other matrices in the equation above are denote[43]:

- The $n \times n$ matrix $\mathbf{A}$ denotes the state matrix. This relates the state variables at the previous time step $k-1$ to the current time step k in the absence of either driving function or process noise. It is assumed here, that matrix $\mathbf{A}$ remain constant, but in practice this might change with each time step.

- The $n \times l$ matrix $\mathbf{B}$ denotes the control matrix. This relates the optional control inputs $\mathbf{u} \in \Re^l$ to the state variables x
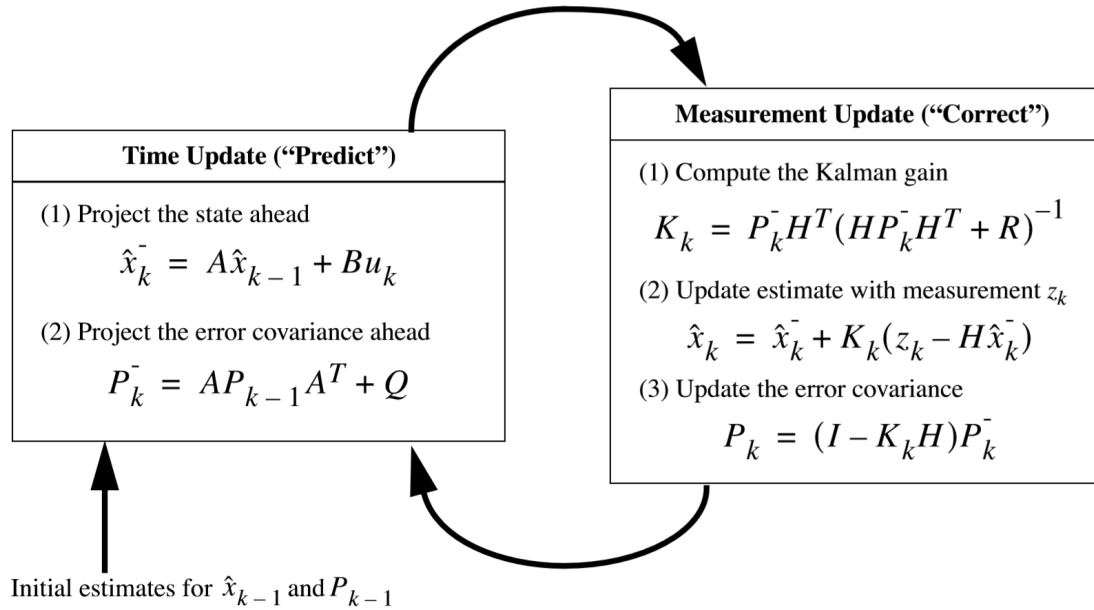
Figure 3.8. *Kalman filter operation principle[43]*

- The $m \times n$ matrix **H** denotes the measurement matrix. This relates the state variables to the measurement variables $\mathbf{z}_k$. This is assumed to remain constant here, but in practice it might change at each time step.

Due to its name, Kalman filter is thought of as a filter like regular filters. However, in reality Kalman filter is an optimal estimator, for instance in case where the process and measurement noise are both zero-mean Gaussian noise.

In its operation, Kalman filter estimates the process state at some time and obtain the feedback in the form of measurements. The measurements are full of noises due do the imperfectness of the sensors. Because of that, Kalman filter equations are formed in to *time* update and *measurement* update equations. The time update equations project forward the current state and error covariance estimates in order to obtain the *apriori* estimate for the next time step. The measurement update equations do the feedback operation, which means that they incorporate new measurement into the *apriori* estimate in order to improve the estimate, which is called *aposteriori* estimate. In other words the time update equations are thought as *predictor* and the measurement update equations as *corrector* equations[43].The complete picture of Kalman filter operation is illustrated in the figure bellow.

For further investigation on Kalman filter equation derivation take a look at [43][23]. Other great work and application of Kalman filtering have been done in [44][45][46].

3.3.2 Stochastic error compensation

In the previous section, it was mentioned that the bias instability can be be modeled using the first order Gauss-Markov process. I this part the working of Kalman filter to model the bias instability will be analysed. To do so, Equation 3.27 can be developed to model the bias instability for three accelerometers that include in IMU unit as it was presented in [17].

$$\underbrace{\begin{bmatrix} \delta B_x \\ \delta B_y \\ \delta B_z \end{bmatrix}}_{x_k} = \underbrace{\begin{bmatrix} e^{\frac{-dt}{Tcx}} & 0 & 0 \\ 0 & e^{\frac{-dt}{Tcy}} & 0 \\ 0 & 0 & e^{\frac{-dt}{Tcz}} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \delta B_{x_{k-1}} \\ \delta B_{y_{k-1}} \\ \delta B_{z_{k-1}} \end{bmatrix}}_{x_{k-1}} + \underbrace{\begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}}_{w_{k-1}}, \tag{3.34}$$

where A: State matrix, δB: bias instability at time k, δB_{k-1}: bias instability at time $k-1$, Tc: correlation time, and dt: sampling time Equation 3.34 and 3.35 shows the model used in Kalman filter algorithm.

$$\underbrace{\begin{bmatrix} \delta B_{xo} \\ \delta B_{yo} \\ \delta B_{xo} \end{bmatrix}}_{z_k} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} \delta B_{x_k} \\ \delta B_{y_k} \\ \delta B_{z_k} \end{bmatrix}}_{x_k} + \underbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}}_{v_k}, \tag{3.35}$$

where z_k the output, C is the measurement matrix and n_k is the measurement error.

4 EXPERIMENT SETUP

In this section, the characteristics of sensors used in this studies are presented. Relevant information regarding sensors were retrieved from the data sheet of each sensor. This helps to better analyse the differences between these sensors as well as evaluate their noise performance.

4.1 Choosing the correct MEMS accelerometer for the application

The choice of the correct sensor to be applied may become a big problem in many cases. In some application this requires the consideration of several aspects and in sometimes very conflicting parameters. Sensor performance is declared in most of the time by using different terminologies that are important to know. Below is summarized some terminology that are used in MEMS- sensors[24].

- **Noise:** This the random fluctuations at the sensor output when there is no input signal.

- **Sensitivity:** This is the ratio of a small change in electrical signal to a small change in physical signal.

- **Resolution:** This is the minimum detectable signal change. In digital sensors, this is considered as the smallest bit change.

- **Dynamic range:** This is considered as the span of physical input which may be converted to electrical signal.

- **Accuracy:** Sensor accuracy is the largest expected error between the actual and the ideal output signals. The inaccuracy is mostly due to the change in the sensor characteristics over time, changes in temperature, initial offsets as well as nonlinearity.

- **Stability:** Sensor stability tells how constant the output is in the constant conditions.

- **Repeatability:** Sensor repeatability refer the to it's ability of to give the same

output data when the conditions are identical.

- **Reliability:** This is the probability to have failure in the sensor operation.

- **Drift:** Sensor drift is the change of output over time. Zero drift is the change of output over time when there is no input present or the input is constant.

- **Cost:** The cost of the sensor may come as the most important characteristics of everyone but engineers[24].

- **Random Walk** A random noise is always present in measurement if a sensor measures a constant signal. This is the equivalent of the integral of white noise in the accelerometer output. It is characterized by a slope of $1/2$ and is contributed by random fluctuations in the signal with correlation time much shorter that sample rate. It's value can be read for $T = 1s$ and it directly represent the noise density in $mg/\sqrt{Hz}$ or $m/s^2/\sqrt{Hz}$ according to [47].A low noise density value is desired when the low amplitude signals are of interest

- **Bias instability or in-run bias** This is also called in-run bias stability. The initial bias of IMU changes over time when the IMU is powered on. Bias instability is often caused by temperature change and mechanical stress on the IMU system. It is represented by the flat portion at the bottom of the curve in Allan variance analysis. This is explained more in Section 3. The values in that section is the in-run bias instability of a accelerometer and indicates the minimum bias that can be estimated.

- **Scale factor** The scale factor is the relation between the input and output. For example if the input is 100%, then the expected output is 100%.In many of cases the actual output is the result of linear effect, where the output is proportional to the input but scaled. For instance if the input is $10m/s^2$, but the is a 2% scale factor error,the output measurement would be $10.2m/s^2$.

- **Non-Orthogonality or Misalignment** In the case of 3-dimensional sensors where the axis are orthogonal to each other, the mounting is not always perfect and contains some errors. This leads to correlation between sensor individual sensors. For example one axis of an accelerometer is pointing perfectly upward and the sensor is on placed on perfectly horizontal level. The accelerometer on this axis is measuring the gravity force. If the other two axis were perfectly orthogonal, they will not experience any effect of the gravity, however if there is a non-orthogonality, the other axis will have a portion of gravity in the output. This can lead to correlation between sensors axes, especially when sensors are combined together to form a compact set such as $6 - DOF$ or $9 - DOF$- Imu sensors.The misalignment can occur also between the sensor set and the enclosure. The misalignment can be minimized

using careful manufacturing and sensor calibration

- **Bandwidth:** This the range of frequencies that can be measured by the sensor. This is related to the sensor's response time.

Other important parameters for characterizing the sensor to be used are the accuracy, dynamic range, noise, resolution as well as the stability. However, the system level specifications may define the product performance. according to [24], the biggest difference between the consumer grade and automotive or aerospace level sensors may sometimes be a single parameter such as guaranteed stability of a certain temperature range.

4.2 Specification of sensor used in this work

Relevant characteristics of sensors related to vibration monitoring were of concern while making the choice of sensors. Those characteristics are such as: acceleration range, noise density, frequency bandwidth and resolution. However, the choice of these characteristics can vary according to the application area. Table 4.2 summarizes the characteristics of the sensors used. More information is can be found in Appendix A.1 as well as in sensor data sheets.

Sensors used in this work			
Module Name	**Manufacturer**	**Seller**	**Price €**
MTI-300	Xsens	Xsens	2500
LSM9DS1	ST Microelectronics	Ozzmaker	22.20
MPU-9250	Invensense	Amazon	12.99
BNO055	Bosch	Amazon	47.07
LSM303DLHC	ST Microelectronics	Amazon	7.99
MTN-1100	Monitran	Monitran	250

Table 4.1. *Sensor used in this thesis*

Sensors used in this work						
Name	range max (g)	Sensitivity (LSB/g)	Bandwidth (Hz)	Noise density ($\mu g/\sqrt{Hz}$)	A/D resolution (bits)	0-g level offset (mg)
MTI-300	±15	-	375	150	16	
LSM9DS1	±16	1366	404	-	16	±90
MPU-9250	±16	384	260	300	16	±60XY,±90 Z
BNO055	±16	1000	1000	190	14	±150
LSM303DLHC	±16	83	2688	220	16	±60
MTN-1100	±80	100 (mV/g)	10k	-	Analog	-

Table 4.2. *Sensor used in this thesis*

It is important to note that MEMS sensor units used in this work are all multipurpose sensors since, which means that several sensors are gathered together to form one unit. All these sensor used contain an $3DOF$ Accelerometer, $3DOF$ Gyroscope and a $3DOF$ Magnetometer as well as an internal thermal sensor. Since the use case of this study was about vibration monitoring, only the accelerometer data was found more informative for this motion. Therefore only Accelerometer was used the whole study. However, the same methodologies used can be applied to characterise gyroscope mounted on the same sensor chip. In vibration monitoring applications, transducers characteristics such as operation range, frequency bandwidth and sensor resolution and of course sensor price play an important role while making while choosing the sensor to use. A big difference is to be noted while comparing piezoelectonic sensors and MEMS-based sensors in this usecase. Even though piezoelectonic sensors are known to be quite good in terms of sensing range higher frequency ranges, there still exist many limitation for using these type of sensors and that is key factor for considering the utilization of MEMS-based sensor in condition monitoring. MEMS-accelerometers have relatively low operation range and low frequency bandwidth compared to piezoelectronic sensor as it can be noted in Table 4.2. However, one must not that the choice of these characteristics depend well upon the application. For instance what is the maximun vibration and what frequency range is to be measured. The maximun acceleration range of these low cost sensors used in this thesis is $\pm 16g$ or $\pm 157 m/s^2$ which is more that enough for this purpose low vibration motion. The maximun frequency to measure is expected to be less than $200Hz$. MEMS-accelerometers are not limited only to $\pm 16g$, because higher ranges and higher frequency bandwidth even up to $5kHz$ can be found already in the market.

The reason why MEMS-sensors produce lower frequency bandwidth is due to relatively high noises in the data. To clarify, internaly MEMS-sensor are sampled with high frequency clock when converting analog signal to digital (ADC). After that the signal is passed through low and high frequency filters in order to filter out noises before it can be sent to digital registers before it can be sent to serial buses such as I2C and SPI. Figure 4.1 shows an example digital block of ST Microelectronic LSM9DS1 Accelerometer and Gyroscope sensors.
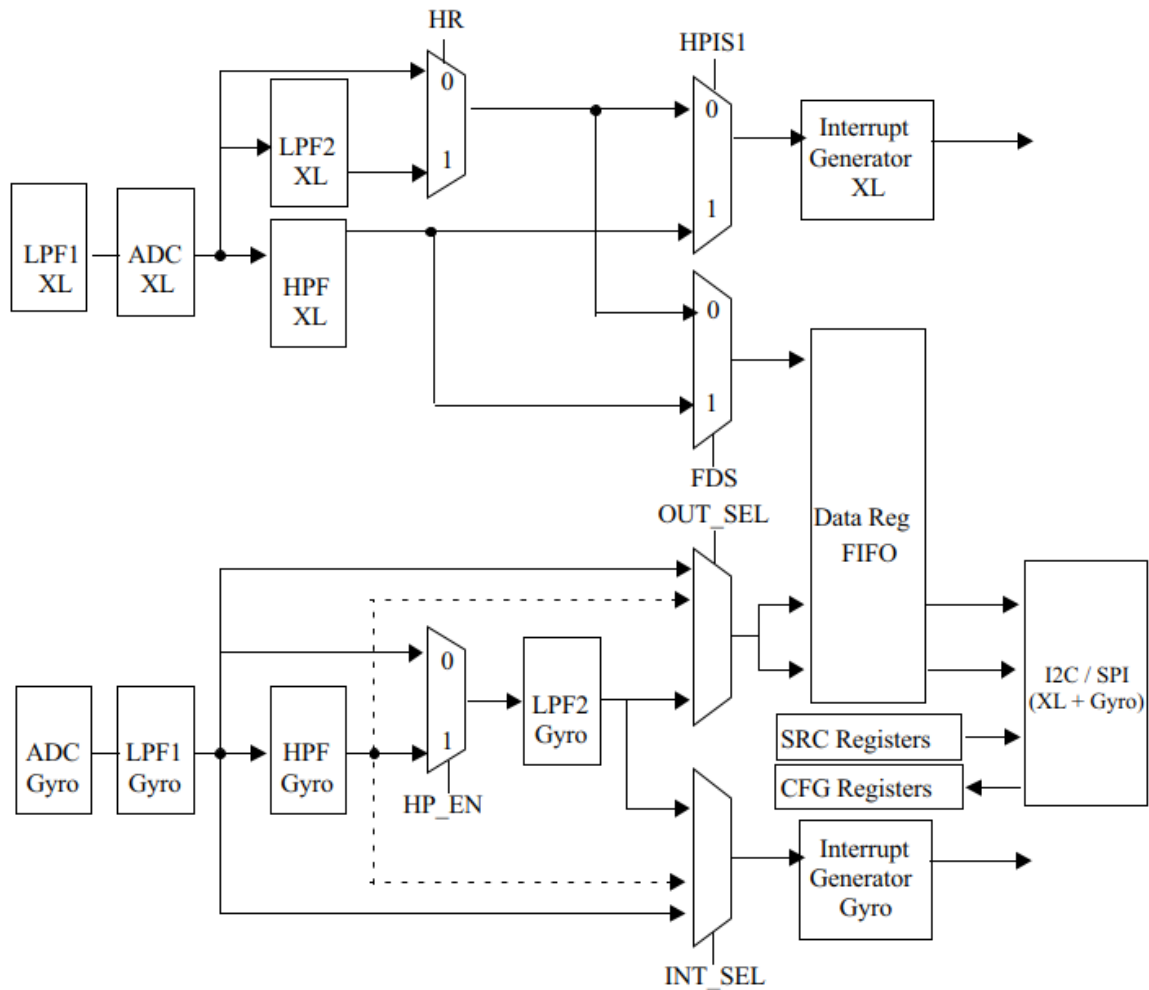
Figure 4.1. *LSM9DS1 Accelerometer and Gyroscope digital block diagram [48]*

4.3 Data Acquisition System

The system used to get data from sensors is presented below. The components of hardware and software utilized are explained more in details. Additionally, the methods utilized to analyze all sensor data are presented in this section.

4.3.1 Hardware

In order to acquire the data from different MEMS-Accelerometers, a simple aluminium plate of size $40 \times 27 \times 5$ cm was done. All sensors were placed on that plate at different location such that the axes direction matches for all sensors. The sensors were screwed on the plate with a stainless screw, where a small plastic tube was placed between the plate and the sensor to isolate metallic parts. Figure 4.2 shows all the sensors mounted on the plate.

All small sensors were connected to a raspberry Pi3 model B+ [49] via a breadboard.
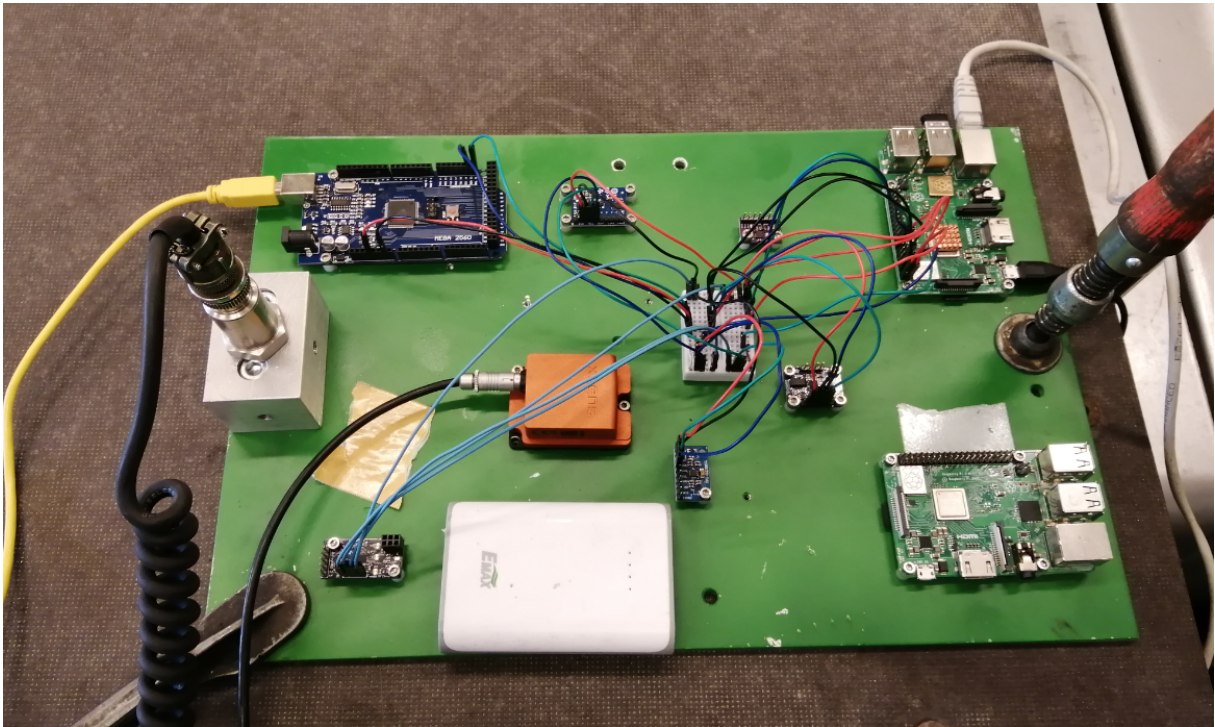
Figure 4.2. *Sensor comparison platform*

This allowed all sensors run at the same time and measure the same phenomenon. The arduino in the mounted on the plate was not used in this purpose.

4.3.2 Robot Operating System (ROS)

Robot operating system (ROS) is a collection of software frameworks for robot software. ROS provides services such as hardware abstraction, low-level device control, implementation of functionalities, the ability perform communication between different devices as well as package management functionality.[50] The benefit of ROS is that it is open source and running on linux systems.In its new features ROS can be integrated to run in real time. However, this functionality was not used in this thesis. The integration of ROS and MATLAB gives the ability to have communication between windows and linux operating system.

The basic concept of ROS is illustrated in the figure below.

As it can be seen in the Figure 4.3 above, ROS processes are represented as nodes that are connected by topics.Trough topics, nodes can send messages to one another, make service call or provide services to other nodes. Nodes can set or retrieve shared data from a communal database called parameter server. For the communication to work, ROS nodes must be registered to the ROS Master Node. The task of ROS master node is to establish
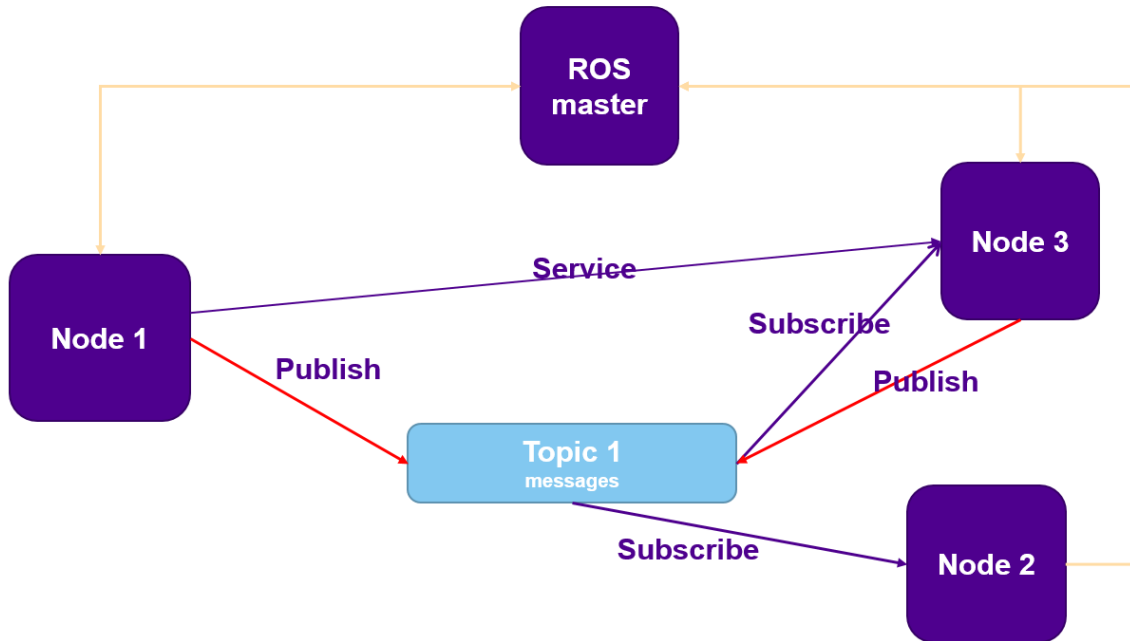
Figure 4.3. *ROS working concept*

the communication between nodes and control the parameter server updates.[50] Some useful commands used in ROS are listed bellow:

- **roscore**: to establish the ros master
- **rostopic**: when combined with other keywords such as list, show, echo, one can get the state of a topic in case.
- **roslaunch**: ROS tool that can be used to start several nodes at the same time
- **rosrun**: ROS tool that is used to start a single node.
- **rosbag**: This tool provide useful functionalities such as recording and replaying the data in the file format called rosbags.
- **rviz**: A powerfull tool for performing the visualization in real time.

In this thesis, ROS was used as working environment in order to have all sensors running on same platform. To put all together, firstly, ROS was installed on raspberry pi3 model B+ running Ubuntu Mate as operating system and on linux laptop running ubuntu 16 lts.

The overview of how the hardware connection was done can be seen in the Figure 4.4 below. Due to clock stretching issues with raspberry raspberry Pi, BNO055 sensor was not able to work properly on i2c bus, and therefore this sensor was required to use SPI bus connection. In order to read data from sensor vias ROS environment, For every sensor, a ROS node was created. The existing open sources C codes for each sensor were
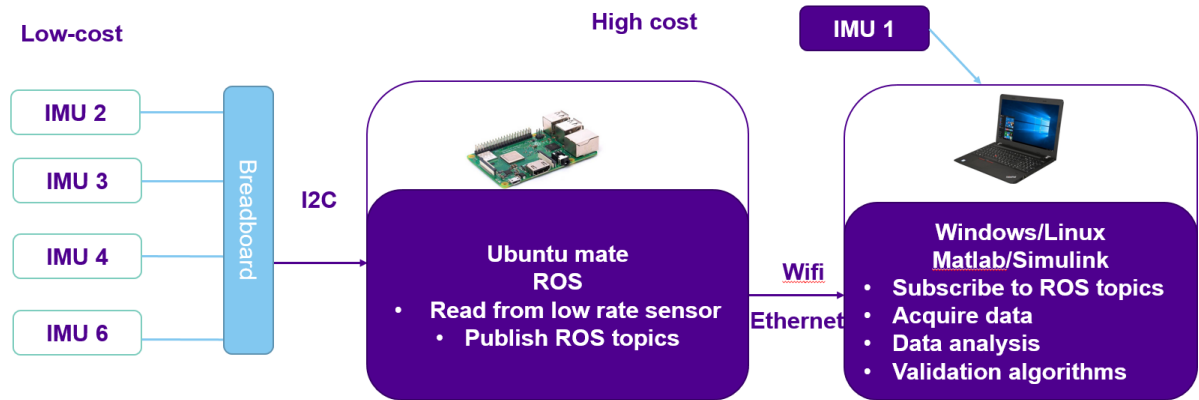
Figure 4.4. *Data acquisition overview*
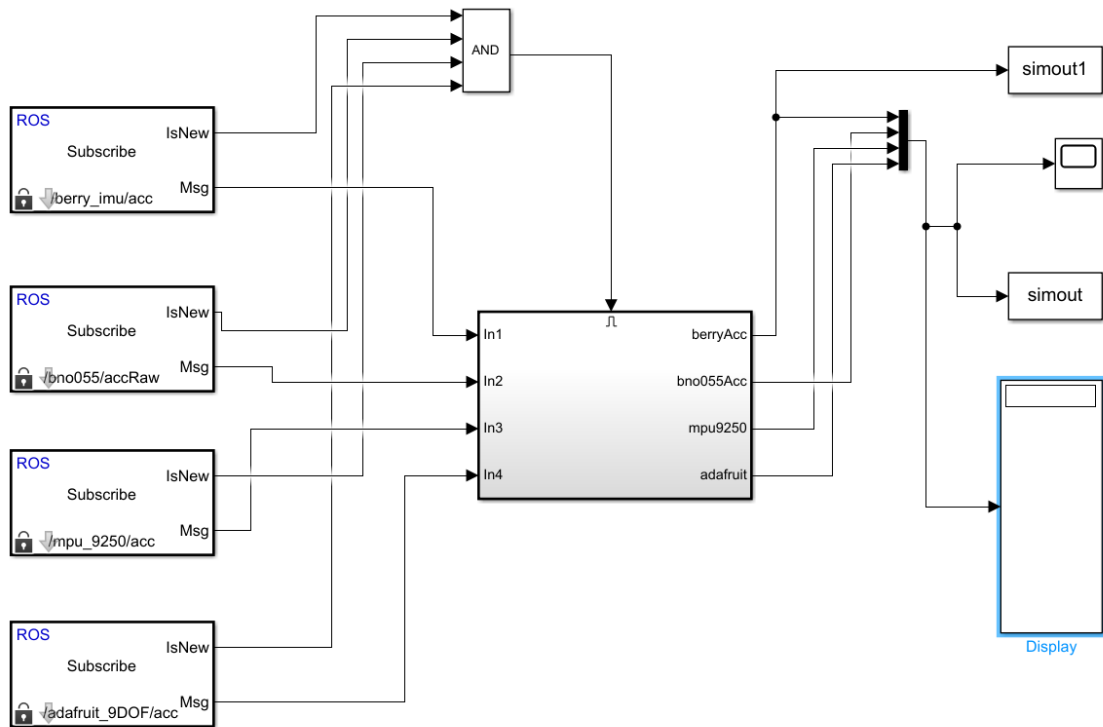
Figure 4.5. *MATLAB Simulink data acquisition block*

used to create a ROS driver for each sensor exept for the MTI-300 where the driver was already provided by the sensor manufacturer. After creating the drivers ROS nodes were made in which every sensor publish all Acceleration, Gyroscope and Magnetometer data through topics. Note that only the Accelerometer topics were used in this study since only the acceleration information was needed. After all this, data was recorded using $rosbag-$command and visualized in real time using ROS functionalities such as $rviz$. For further information on these command, see [50].

During data acquisition, data was acquired from ROS environment by using he command $rosbag - record$. I was also possible to aquires that data from an other environment such sus MATLAB, since MATLAB/Simulink support ROS functionalities. Figure 4.5 illustrate a Simulink block that were used to acquired MEMS sensor data from ROS topics published from a raspberry Pi.

4.4 Data Analysis methodologies

The importance of data analysis is crucial especially when working this sensor data. It is common that data produced by sensors contain noises and other information that is irrelevant to what is being studied. In order to have reliable results, data must be processed by utilising different methodologies depending on what is the goal to be achieved. In this thesis data from sensors was processed using simple and commonly used methods. As the purpose is to work with vibration data, the first thing to do was to remove the constant value from all sensor data produced by MEMS- accelerometers.The effect of constant value can be high while integrating accelerometer data. For piezoelectric sensors, the constant value was very small due to it's nature as an analog sensor. However, there noises in analog sensors might be high depending on the data acquisition system. Removing the constant component from data can be done applying a high pass filter or removing the mean from sample data. In this thesis both methods was used as it will be seen in Section 5 High pass filter remove unwanted low frequency noises from data and low pass filter remove unwanted high frequency noises. However, the choice of the filter design is also important in order to achieve certain results. Additionally, the filter performance highly depends on the sampling sampling rate of data being filtered. In this work Matlab $designfilt$ tools was used to design filters and to process data.

4.4.1 Vibration analysis using MEMS accelerometer

MEMS sensors are well known for applications involving inertial navigation and positioning, dead reckoning and other applications as it was discussed in Section 2.1. However, in order to apply MEMS based accelerometer as vibration monitoring, data from these

Simple 60 Hz Sine Wave
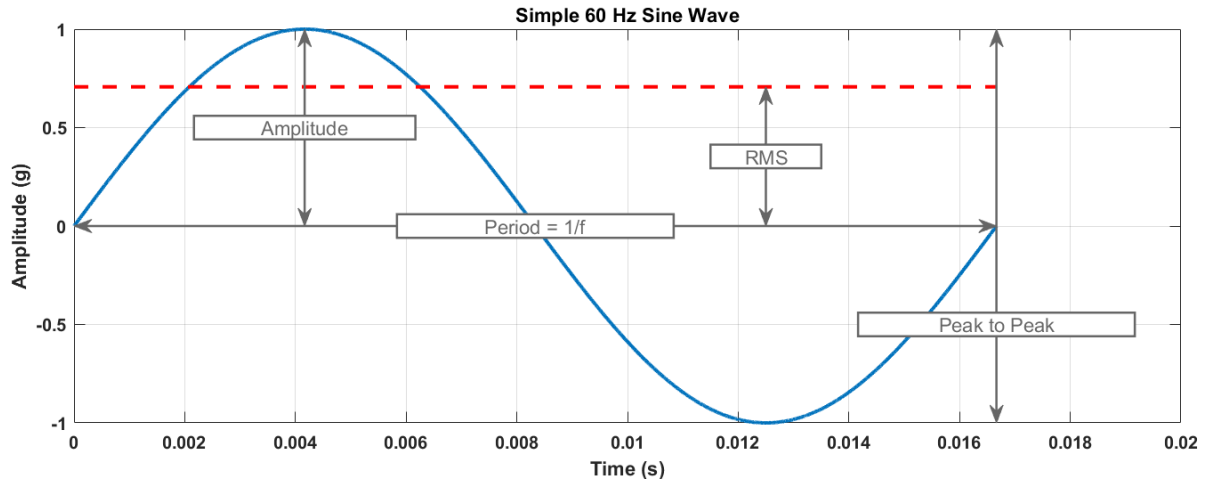
Figure 4.6. *Structure of a simple 60 Hz sine wave principle[51]*

sensors must be preprocessed such that it reflect the vibration phenomenon. This pre-processing included the filtering of low/high frequency noises according to the system requirement. An important thing to do at first is to filter out constant acceleration due to the gravity. The simplest approach is high pass filtering the acceleration data. The vibration representation in terms of velocity and displacement may be crucial especially when working with low frequency ranges. This can be done by integrating the accelerom-eter data for velocity or double integrating this to get the displacement accordingly, as it was illustrated in inertial equations showed in Chapter 2.

4.4.2 Time Domain Analysis

The time domain analysis of sensor data comprises of time history data visualization in xy-plot. In vibration analysis some parameters such as peak-to-peak, peak amplitude and root mean square (RMS) are often the variable of interest.Figure 4.6 shows a simple time domain plot of a 60 Hz sine wave. The most important vibration parameters are shown in this figure. As stated in [51], the peak and peak-to-peak does not take time duration of the signal into consideration. However these parameters are useful when the displacement information and severity is of concern.On the other hand, the root mean square parameter takes into consideration the time duration of the signal. This value is the most useful in time domain analysis as it is related to the energy content of the vibration signal. In other term it gives the information on the destructive capability of the vibration.

Vibration analysis in time domain using MEMS-accelerometers can be done after pre-processing the acceleration data. Figure 4.7shows a simple schema of time domain vibra-tion analysis using MEMS- accelerometer.
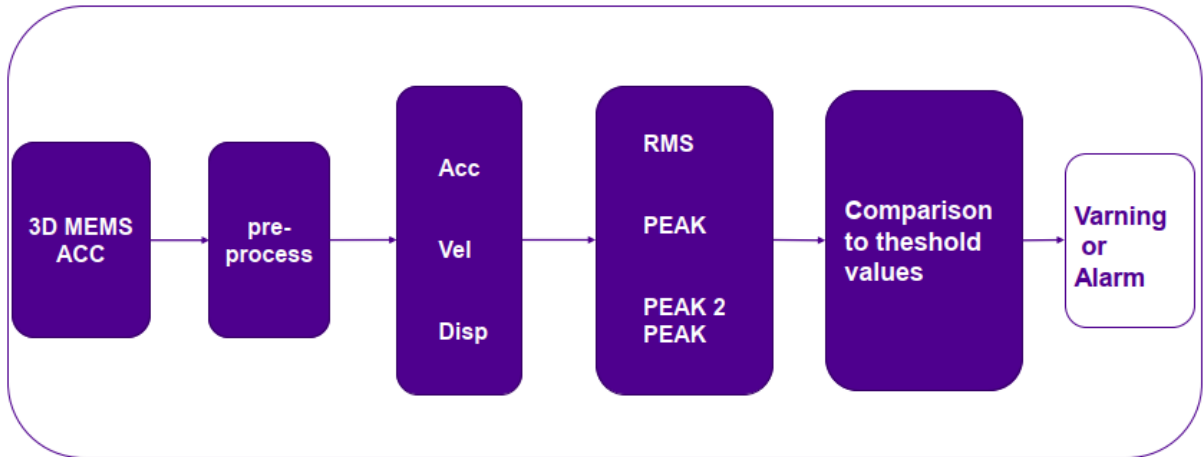
Figure 4.7. *MEMS- accelerometer vibration analysis schema in time domain*

4.4.3 Frequency domain analysis

The idea behind vibration frequency analysis is to determine the rate or frequency of os-
cillation of a certain pattern about an equilibrium point. By vibration frequency, one refer
to the number of times a complete motion cycle occurs during a period of one second[51].
In frequency analysis Fast Fourier Transform (FFT) or Discrete Fourier Transform (DFT)
is the most commonly used method for vibration analysis. An introduction to discrete
Fourier Analysis theory can be found in [52]. Basically, Fourier analysis or spectrum
analysis is a method used to deconstruct a signal to its individual sine wave components.
As results amplitude of the signal versus frequency can be be studied in more details.
This gives more understanding on the signal content as well as their effect in terms of
harmonics.

- **Fast Fourier Transform (FFT)**

  A fast Fourier transform is an algorithm that is widely used to compute the discrete
  Fourier transform (DFT) of a sequence of data. The main idea behind this is
  that FFT converts a signal from its original time domain to a frequency domain
  and vice versa by decomposing the this sequence of values into different frequency
  components[53].

  The general form of DFT can be seen in Equation 4.4.3

  $$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}, k = 0, ..., N-1, \tag{4.1}$$

  where $e^{i2\pi/N}$ is the Nth root of unity. More mathematical background of FFT and
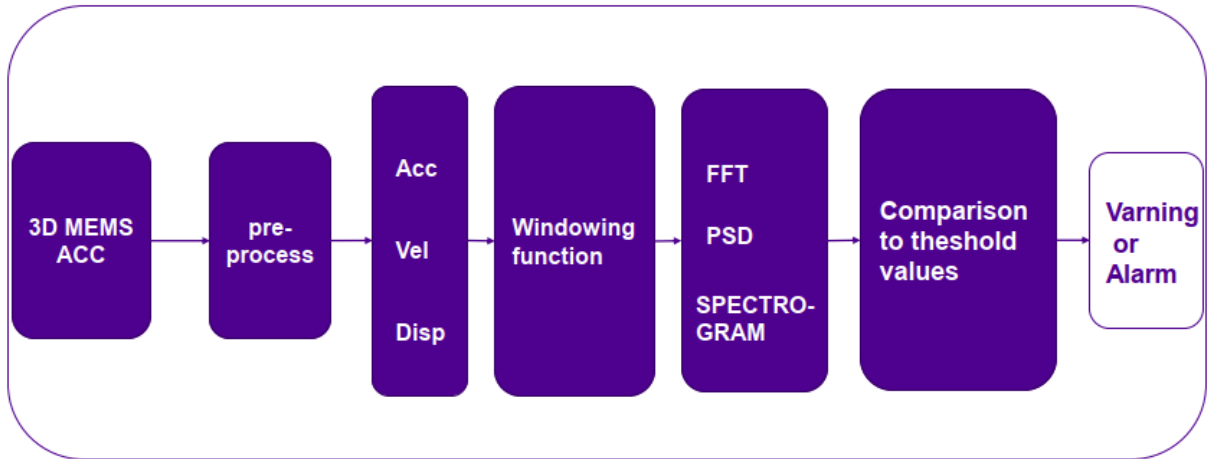
Figure 4.8. *MEMS- accelerometer vibration analysis schema in frequency domain*

DFT can be found in[54]

- **Power Spectral Density (PSD)**

  While FFT is good at analysing vibration when the data contains a finite number of dominant frequency components,PSD method is used to characterise the random frequency. PSD is calculated by computing each frequency bin in FFT by its complex conjugate. The amplitude of PSD is the normalized to the frequency bin in order to get rid of frequency on bin width. This is useful especially when one need to compare signals of different length[51]. More details on PSD was explained in Section 2.3

- **Spectrogram** A spectrogram is an other powerful method used in frequency domain to analyse the change for frequency with respect to time. This more important especially if the vibration is being analysed in an environment that may change over time. This change may be caused by environmental issues such as temperature and humidity[51]. Other information on spectrogram analysis can be found in [55]

While using MEMS-accelerometers as the vibration monitoring, same method used in time domain to pre-process the acceleration data can be applied. However, before applying the Fourier transform, data must go through a window function in order to reduce spectral leakage and frequency attenuation problems. Figure 4.8shows a simple schema of MEMS-accelerometer vibration analysis in frequency domain.

4.4.4 Characterization of vibration motion

Since MEMS accelerometers are capable of capturing information of a particle in motion, spectral analysis can be used to characterize this motion as it was defined previously. In

machine condition monitoring application the form of vibration motion is predominantly linear. Hence, the direction of this motion is crucial. Figure 4.9 show the spectral analysis of a vibration profile. Each moving equipment or machine produces a vibration profile as this Figure show. In the figure it can be seen that the profile has both broadband and narrow-band frequency components. The natural frequency is the frequency at which the vibrating device vibrates when no damping force is imposed. This frequency is also known as fundamental frequency,Eigen frequency or first harmonic. Harmonics are higher frequencies that are superimposed onto the fundamental frequency.The frequencies are formed of a integer multiple of fundamental frequency. As an example, given a natural frequency of $12Hz$, the 2nd and 3rd harmonics would be $24Hz$ and $36Hz$. In rotating machines, the malfunctioning of bearing system and other faults can be detected from the presence of high order harmonics in the vibration profile.

One of the main focus of this work was to compare the performance of low cost sensors when monitoring the vibration of wheel loader machine base. This machine is powered by a diesel engine with four cylinders. All measurements were done when the machine is running at idle. To calculate the natural frequency of a rotating engine, the Equation 4.2 can be used.

$$F_{natural} = \frac{speed(rpm)}{60} \tag{4.2}$$

A four cylinder engine running at idling speed of $800rpm$ will be characterized by a dominant frequency of $26.667Hz$. This is equivalent to twice the crank shaft speed. In this type of engine two ignitions are performed during one engine revolution. In the vibration profile produced by FFT, the characteristic frequencies will appear at 26.67 and $53.33Hz$ corresponding to the frequency of combustion and frequency of strike for exhaust and intake valves respectively. Further investigation on diesel engine vibration have been studied in [56], where biodiesel and petrodiesel fuel blends were used. A basic understanding on how a four-stroke engine works can be found in [57]
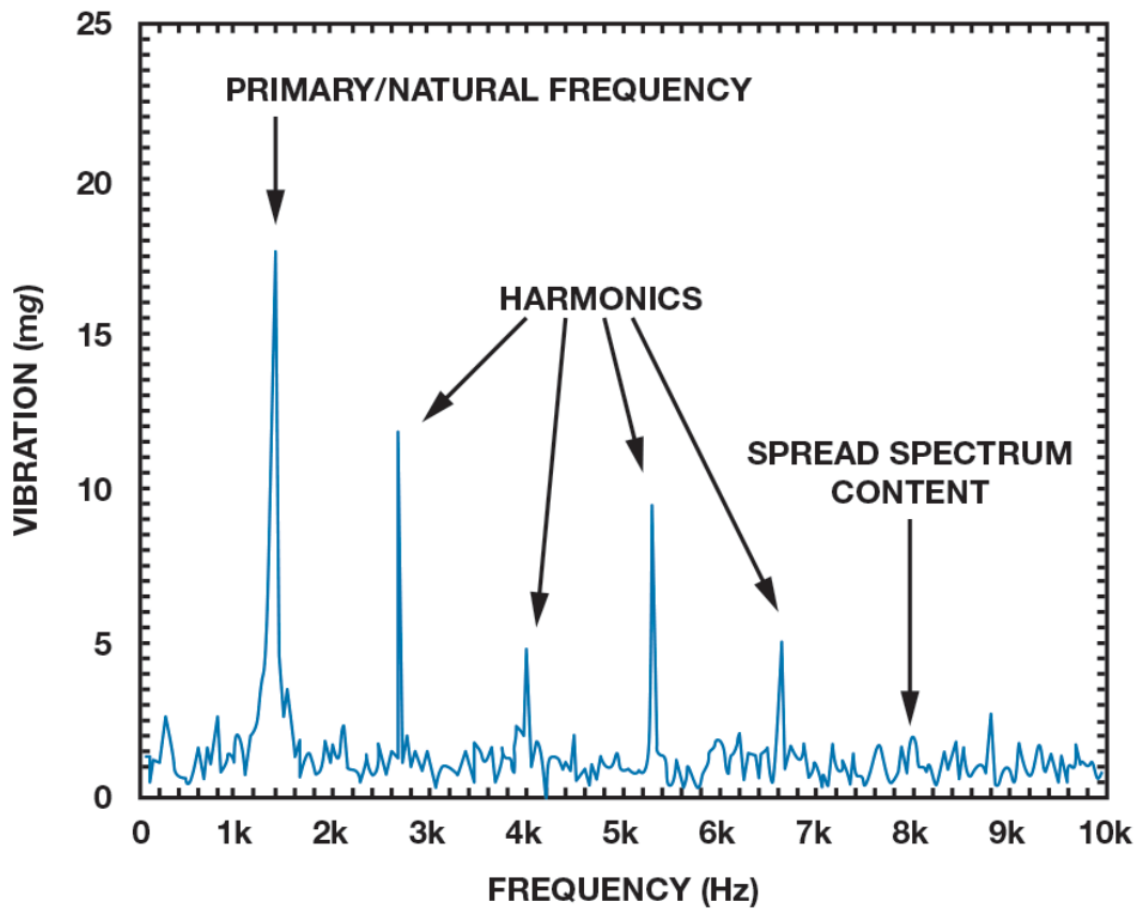
Figure 4.9. *Vibration frequency harmonics[1]*

5 EXPERIMENT RESULTS

5.1 Test Results of Sensor Static Measurements

In this section, the experiments performed on static data are presented. These experiments included the analysis of noises presented in sensor data when sensors are in stable state.

5.1.1 Sensor Noise comparison on stationary data

The purpose of this phase was to illustrate the differences in sensor data while no input is applied to the sensors. This was done by acquiring data while the comparison platform is lying on horizontal table.

- **Procedure:** The data was acquired for 20 seconds while no motion is applied to the comparison platform. The static bias was removed by subtracting the mean (dc component) from the data set.

- **Results:** minimum, maximum, peak to peak, standard deviation and static bias.

Table 5.1 shows the results from the 20 minutes data set, where the minimum, maximum, peak to peak, standard deviation and static bias was retrieved for all axis in all sensors.

Figure 5.1 illustrates the noises plotted from all sensors while the comparison platform is placed on a horizontal table. From the figures it can be seen that the noises are highly dependent on the sampling rate of the sensor. By looking only at the figures, the difference in performance is not obvious, but there is a clear difference in terms of amplitudes. This is illustrated in Figure 5.1f, which shows high spikes in the stationary data. These spikes are outliers and are a symbol of a malfunctioning LSM303D sensor. While plotting the data from this sensor, it was discovered that the acceleration values are frequently zero on x and y axes, which was different to other sensors on the same platform. The origin of this error may stem from a failure during the soldering process or in other case defective sensor installation. Nonetheless, by solely narrowing our focus on the time history of the stationary data, we cannot get sufficient information about the sensor's performance. To extend this analysis, Table 5.1 illustrating the numerical values of stationary data was

(a) MTN-1100 noise

(b) Xsens Mti-300 noise

(c) lsm9ds1 noise

(d) bno055 noise

(e) mpu9250 noise

(f) lsm303dlhc noise
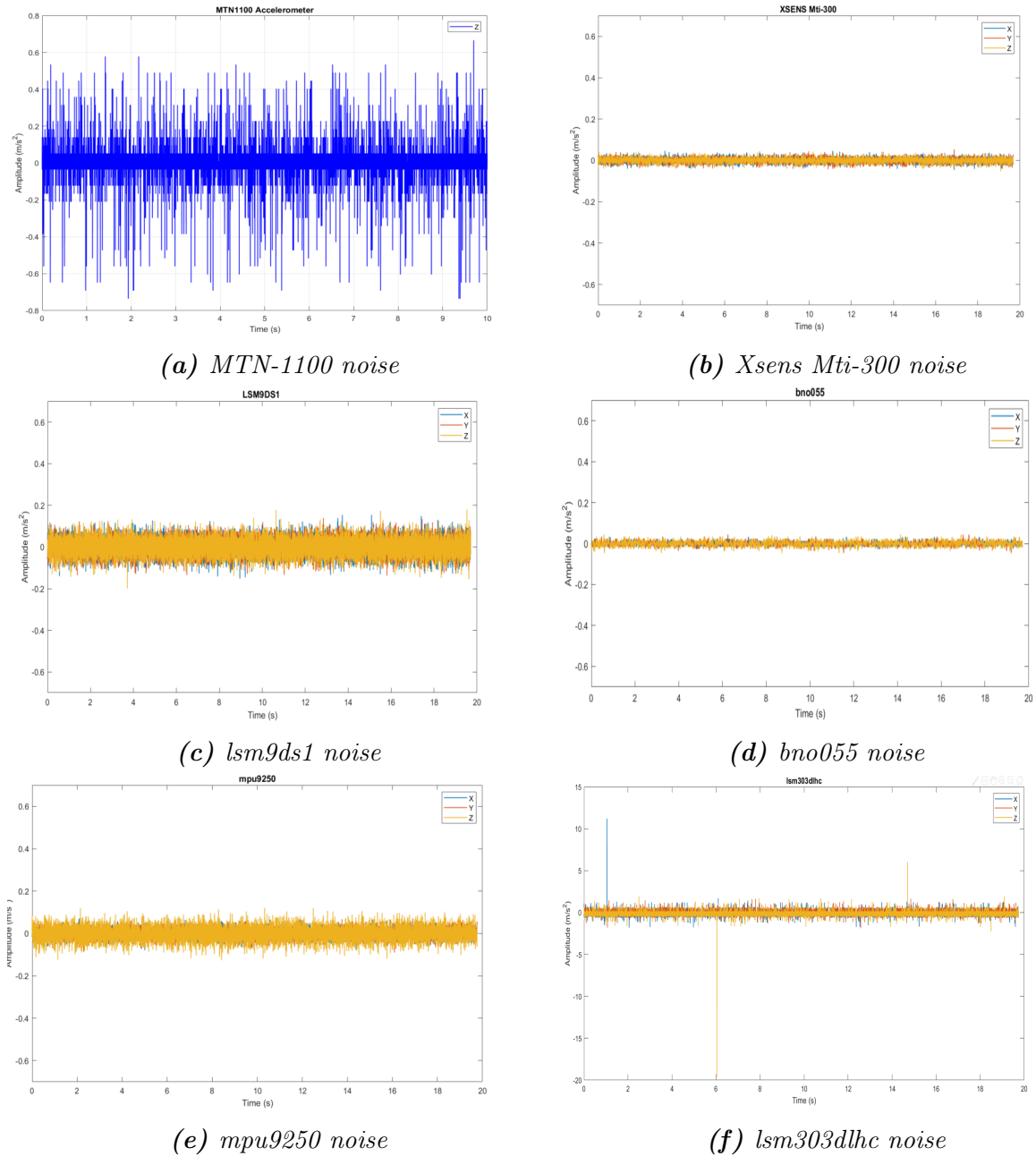
Figure 5.1. *Stationary data of piezoelectric accelerometer vs MEMS-Accelerometers*

created. In this table minimun, maximun, peak-to-peak, standard deviation and the static bias of the sensor were retrieved from all sensors. The table indicates that the LSM303D sensor shows higher values compared to other sensors. Looking at the standard deviation of the data it is found that the the piezoelectric sensor taken as reference has the lowest deviation of $0.0001 m/s^2$.

In order to classify the performance of the MEMS-accelerometers with respect to the piezoelectric accelerometer (MTN-1100), the ratio of standard deviations was utilized. This was achieved by dividing the standard deviation (std) of MEMS-accelerometer with that of the piezoelectric sensor. As this sensor is a single-axis sensor, this comparison

could only be done for z-axis. The same methodology can be used also for other axes. From the table it can be seen that MTI-300 give the lowest result of 116 which means a better performance compared to other MEMS-accelerometer.The high value of LSM303 is due to the existence of outliers/spikes in the data as it is seen in Figure 5.1f. BNO055 is the second in performance giving a ratio of 140. The reason for MTI-300 and BNO055 to have better performance is due to better built-in signal conditioning and higher calibration level of these sensors.

Accelerometer data output [m/s^2]							
metric	axis	MTN-1100	MTI-300	MPU-9250	LSM9DS1	BNO055	LSM303DLHC
	X	-	-0.0460	-0.0784	-0.1516	-0.0273	-15.30
min	Y	-	-0.0437	-0.0899	-0.1540	-0.0460	-1.70
	Z	-0.7358	-0.0525	-0.1265	-0.1985	-0.0464	-16.50
	X	-	0.0461	0.0709	0.1539	0.0327	1.90
max	Y	-	0.0537	0.0722	0.1394	0.0440	1.80
	Z	0.5559	0.0426	0.1199	0.1797	0.0436	2.0
	X	-	0.0921	0.1494	0.3056	0.0600	17.20
p2p	Y	-	0.0974	0.1621	0.2934	0.0900	3.50
	Z	1.4017	0.0951	0.2464	0.3782	0.0900	18.50
	X	-	0.0116	0.0189	0.036	0.0097	0.20
std	Y	-	0.0120	0.0189	0.0349	0.0112	0.20
	Z	0.0788	0.0116	0.0318	0.0383	0.0140	0.30
	X	-	-0.0537	0.0515	-0.2064	0.0173	0
static bias	Y	-	0.0631	0.0472	-0.1918	-0.2740	0
	Z	0	9.8230	9.9171	9.9118	9.7664	9.90
std-ratio	Z	-	116	318	383	140	3000

Table 5.1. *Stationary data output*

The time history data analysis performed above does not give enough information on the sensor characteristics, bringing the need to analyse the performance in terms of noises. The measured noise characteristic of these sensors were compared to the values in data sheets provided by the sensor manufacturer. To clarify the characteristic difference of all sensors, additional information were provided. This information includes the sample rate, bandwidth and the total noise of the sensor.

- **Procedure:** Same data acquired for 20 seconds while no motion is applied on the platform used. The static bias was removed by subtracting the mean (dc component) from the data.

- **Results:** Noise density of sensor was computed in time and frequency domain, additionally, the data rate and total noise of the sensors were analyzed. Additional analysis were done using statistical methodologies in order to make a comparison to the piezoelectric sensor.

In this section, the influence of software utilized to the measurement is also discussed.

Firstly, the noise density of the sensor was computed and validated using the values obtained from each sensor data sheet. The noise density was computed in time domain using the equation 5.1 below

$$ND = \frac{\sigma}{\sqrt{f_{3dB}}}, \tag{5.1}$$

where the σ and f_{3dB} are the signal standard deviation and the frequency bandwidth respectively. In this case the bandwidth from the sensor data sheet was used in order to make comparison to the values from the data sheet. To get the total noise, the noise density was multiplied by the square root of bandwidth. Table 5.2 illustrates the results from this analysis. As results it was noticed that the noise density was different from all axis of each sensor, whereas in the data sheet it is not specified on which axis the computation was done. In this case the value from the data sheet is assumed to hold for all axes. In LSM9DS1 data sheet the noise density is not specified at all, which gives no chance for comparison of measured and data sheet value.

It can be seen that the noise density of MTI-300 is lower than other sensors, which may explain the high cost of this sensor, however still the choice of using a high cost sensor for vibration monitoring was further studied and will be clarified later.

From the Table 5.2, it can be noticed that the measured noise densities are lower than the ones in the data sheet, except for LSM303DLHC. In some cases the noise density of Z-axis is significantly higher than X and Y-axes, which can be caused by different sources such as axis misalignment issues. Additionally, it must be noticed that the noise density is independent on the sampling frequency of the sensor, but somewhat depends on the bandwidth which is consistent with Equation 5.1.

Other analysis done here was the output data rate. In the table, the data rate indicated in the data sheet is computed and compared to the data rate at which the data was acquired. This gave a chance to see whether a single board computer such as Raspberry Pi could handle fast samples coming from sensor.

As it is seen in table 5.2, the measured output data rate was lower than the data rates in the that was configured in the sensor register. The reason for this to happen is probably not related to the sensor performance itself, but the performance of acquisition system. Raspberry Pi may not be able to pull data from sensor registers at the same speed there are written due to the I2C-bus clock. This is because several sensors are connected on same I2C-bus, which make the bus become busy.

Accelerometer noise density							
	noise density$\mu g/\sqrt{Hz}$				data rate$[Hz]$		band width$[Hz]$	noise$[\mu g]$
Sensor	data sheet	measured			data sheet	measured		
		X	Y	Z				
MTN-1100				803.77				8033.7
MTI-300	88.8	60.9716	63.1657	61.1195	400	399.8	375	1719.6
MPU-9250	300	89.8376	89.6624	151.0355	1000	740	460	6434.2
LSM9DS1	-	210.0844	202.9551	222.6298	952	1012	408	4241.7
BNO055	150	125.2382	144.0657	180.5340	100	171	62.5	1185.8
LSM303DLHC	220	841.20	733.0	1099.0	1344	1260	672	5703

Table 5.2. *Sensor noise density and bandwidth*

Other reason is due to the software used at which ROS messages are read from all sensors and published in ROS topics at same time as explained in chapter 4. In order to work on I2C-bus BNO055 sensor was used only in the fused mood, which was able to give data only up to 100 Hz. According to the sensor data sheet[58], BNO055 can sample data up to 5kHz and reach a bandwidth of 1kHz, but this experiment this was not achieved. Since I2C-bus on BNO055 uses a method called clock stretching in order to get ready to send data faster to I2C master device (in this case the Raspberry Pi). This functionality is however not supported by Raspberry Pi I2C hardware. To get data from this sensor, a software based I2C bus was used at first, but this did not improve the performance. At the end an SPI bus was used to read this sensor data.

For all sensors, it was rather impossible to achieve the sampling rate that was configured in the register. The higher rate was achieved by LSM303D sensor reaching 1.26kHz while the configuration was set at 1.344kHz according to the sensor data sheet. An alternative solution for this is to use more powerful acquisition system such as Texas Instrument data acquisition boards.

5.1.2 Allan Variance test results

The analysis done above was purely deterministic, and though does not give information on how the noise performance in long time period. To do so, the stochastic noise analysis must be done in order to understand more the the noises affected by the MEMS sensors. As it was explained in section 3. Allan variance method was used as the main noise analysis method together with the power spectral density. The objective of this analysis was firstly to understand the noise behaviour of low cost MEMS- accelerometer sensors compared to high cost sensor, secondly to define the color of the noises present in the sensor data. To compute the Allan Variance of MEMS- accelerometers, a set of procedure is summarized below. 3.

- **Procedure:** Data was acquired for 14h while the comparison platform is on an horizontal table and no input is applied. Static bias bias and constant value was removed by subtracting the mean from the data. By doing this, we insure that the remaining data is composed only with wight Gaussian noise. Additionally the effect of other errors such as scale factor and misalignment are minimized. Allan variance was computed using the method presented chapter 3.1.1.

- **Results:** Allan variance(Allan deviation)

The Allan Variance results of XYZ-axes for all sensors are illustrated in figures 5.2 It can be noted that the shape of these plots is similar to the shape the overall Allan variance plot in figure 3.6 presented in chapter 3. This indicates the existence of random noises in the stationary data. Additionally, the figures shows that the noises present are mainly velocity random walk (VRW) bias instability (BI) and acceleration random walk (ARW).Surely, Other noise components such as acceleration ramp and quantization noise might exist, but for simplicity, in this analysis, only VRW, BI, and ARW were the main focus in this work.

In order to obtain the noise parameters, the relationship between the Allan variance and the two-sided power spectral density was utilized as it was shown in equation 3.12

In Figure 5.2d it is summarized how noise parameters were identified. For instance the velocity random walk coefficient (N) was identified by drawing a straight line of the slop $-1/2$ in the log-log plot of the Allan deviation. This line is moved such that it coincide with the straight section of the Allan deviation plot. VRW can be read at the point of intersection between the line and Allan deviation plot at the time average τ equal 1 second. The same procedure was done to other noise parameters, but with different slopes. Table 5.3 summarizes the slope and time constant at which the parameter is read.

MEMS- accelerometer noises			
Noises	Coefficient	slope	value
VRW	N	$\frac{-1}{2}$	$\sigma(1)$
BI	B	0	$\frac{\sigma(f_0)}{0.664}$
ARW	K	$\frac{1}{2}$	$\sigma(3)$

Table 5.3. *MEMS accelerometer noise coefficients*

Velocity random walk coefficient N

To compare the velocity random walk noise of all sensors, the velocity random walk coefficient (N) was extracted from the log-log plot of Allan deviation shown in figure 5.2.

It can be seen from the figures that VRW is the most dominant at noise in short cluster time, while the ARW is dominant at more longer cluster time. Table **??** shows the results of VRW coefficients (N) estimated for all sensors at a cluster time equal to one hour. The estimation accuracy was computed using the Equation 3.24 presented in chapter 3.In this test the maximum data points of 100 points in a cluster was used. In this case the accuracy is in order of 0.3%.The results from table 5.4 shows that VRW varies from sensor type to an other. Additionally it was seen that all sensors experience a higher VRW on its z-axis than on x and y-axes. This was recognized more for Xsens Mti sensor. LSM303 performs worst in all sensor as it is seen also in the Allan deviation curves in figures 5.2.

further analysis

Velocity random walk (VRW)										
Sensor	Xsens Mti		MPU-9250		LSM9DS1		BNO055		LSM303		Unit
	N	e	N	e	N	e	N	e	N	e	
X	0.0833	± 0.00026	0.1277	±0.0004	0.0825	±0.00025	0.0743	±0.00023	0.7554	±0.0024	$[m/s/h^{1/2}]$
Y	0.0852	± 0.00026	0.1274	±0.0004	0.0723	±0.00022	0.0840	±0.00026	0.8139	±0.0026	$[m/s/h^{1/2}]$
Z	1.19599	± 0.0062	0.2565	±0.0008	0.0977	±0.0003	0.1032	±0.00032	0.8776	±0.0028	$[m/s/h^{1/2}]$

Table 5.4. *Accelerometer velocity random walk*

Bias instability coefficient B

The bias instability also known as pink noise or flicker noise can be defined from PSD as stated in equation 3.16. This equation was simplified to Equation 5.2 for the time τ much longer than the cut-off frequency f_0.

$$\sigma^2(\tau) = \frac{2B^2}{\pi} ln(2), \tag{5.2}$$

which is a line with a slope equal to 0 as it is seen in figure 5.2d. Table 5.5 shows the results of the bias instability test and the corresponding accuracy for each MEMS-accelerometer.

Bias Instability										
Sensor	Xsens Mti		MPU-9250		LSM9DS1		BNO055		LSM303		Unit
	B	e	B	e	BI	e	B	e	B	e	
X	1.4273	±0.0045	4.4881	±0.0141	1.7619	±0.0055	0.8039	±0.0025	2.6671	±0.0084	$[m/s/h]$
Y	1.9474	±0.0061	4.4140	±0.0139	1.6712	±0.0053	2.9657	±0.0093	4.2955	±0.0135	$[m/s/h]$
Z	2.7729	±0.0087	4.2605	±0.0134	2.3084	±0.0073	3.9984	±0.0126	6.7863	±0.0157	$[m/s/h]$

Table 5.5. *Accelerometer Bias instability*

The results shows that the bias instability is less dominant in Xsens Mti and more effective in MPU-9250. It can be seen again that the performance of this high cost sensor is better

than low cost sensors since the accuracy of this sensor is away greater than low cost sensors.

Acceleration random walk coefficient K

Sensor	Acceleration random walk										Unit
	Xsens Mti		MPU-9250		LSM9DS1		BNO055		LSM303		
	K	e	K	e	K	e	K	e	K	e	
X	4.052	±0.0128	4.9431	±0.0156	2.1911	±0.0069	1.866	±0.0053	3.306	±0.0104	$[m/s/h^{3/2}]$
Y	1.5527	±0.0049	5.1879	± 0.0163	2.8486	±0.0090	12.7052	±0.040	6.7863	±0.0454	$[m/s/h^{3/2}]$
Z	8.3062	±0.0262	8.8999	±0.0375	4.8661	±0.0153	2.3612	±0.0074	14.4080	±0.0454	$[m/s/h^{3/2}]$

Table 5.6. *Accelerometer Acceleration Random Walk*

The overall results of this analysis showed deeply that there was larger VRW and ARW in the sensor LSM303 than other sensors. Other low cost sensors showed low VRW if compared to Xsens Mti sensor. LSM9DS1 showed to have less BI if compared to all other sensors. According to this results, it is worthy to say that even low cost sensors can perform better than high cost sensor in terms of stockastic noises, however, there are several other functionalities that make them to be different.
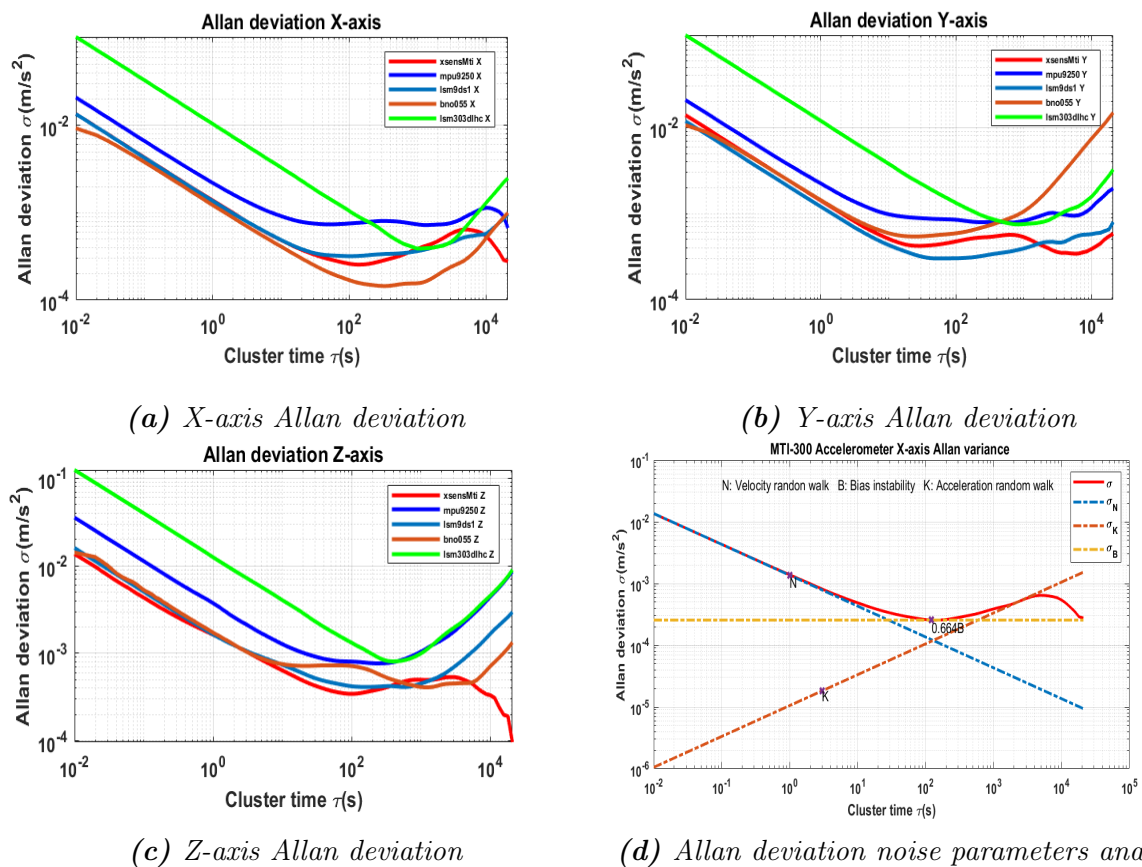

(a) *X-axis Allan deviation*

(b) *Y-axis Allan deviation*

(c) *Z-axis Allan deviation*

(d) *Allan deviation noise parameters analysis*

Figure 5.2. *Allan deviation plots of 14h data set for MEMS- accelerometers*

5.1.3 Gauss-Markov Autocorrelation and Kalman filter test Results

In this section the results on stochastic error modeling using Kalman filter are presented. As it was stated in Section 3.2 that the bias instability of MEMS error can be modeled using the first order Gauss-Markov process and Kalman Filter. The purpose of this test was to found out if this methodology is applicable to MEMS sensors used in this study. In order to process to this goal firstly, it was verified visually that noises in MEMS sensor are white Gaussian by plotting the noise histograms of each sensor. Secondly to get the bias instability from measurement noise, a simple running average filter was used to filter out high frequency noises. Figure 5.3 shows MEMS sensor white noise and noise histogram for X axis.

- **Procedure**: In order to compute the bias instability, the static data acquired for 14 hours was used. Firstly, the static bias was removed by removing the constant component from the data set, secondly, the outliers in the data was removed from the data set and thirdly, the data was filtered using a moving average filter.

- **Results**: White noise and histogram plot was visualised as shown in Figure 5.3. The bias instability was computed and compared to estimated bias instability using the first-order Gauss-Markov process and Kalman filter model.

In order to model the bias instability using Gauss-Markov process in Kalman filter model shown in Equations 3.34 and 3.35, the auto-correlation function of the bias instability was computed using Matlab function *xcorr*. Allan variance plots analysed earlier was used to verify the presence bias instability or flicker noise. It was seen from the plots in Figure 5.2 that bias instability spans in a section from a cluster time equal to 10-2000 seconds. This means that in this section bias instability can be modeled as a first order Gauss-Markov process. The results of the auto-correlation analysis of bias instability is shown in Figure 5.4. From the figure, it can be seen that MEMS sensors have different correlation times. The shapes of auto-correlation of MEMS sensor were compared to the ideal function in Figure 3.7 such that if shape looks similar, it means that the noises are well correlated and therefore the first order Gauss markov process can be used to model this error. From the figure, it was seen that noise correlation of LSM303 X and Y axes as well as BNO055 X and Z ases are highly un-correlated, therefore Gaus-Markov proces cannot be applied. Using the information above, time parameter T_c and noise standard deviation σ was retrieved from the auto-correlation analysis. The variance value σ^2 was taken as the peak value autocorrelation results and the correlation time was computed by finding the time corresponding to σ^2/e as it was done in[26]. Table 5.7 shows the

correlation time and the standard deviation of bias instability for MEMS-accelerometers.
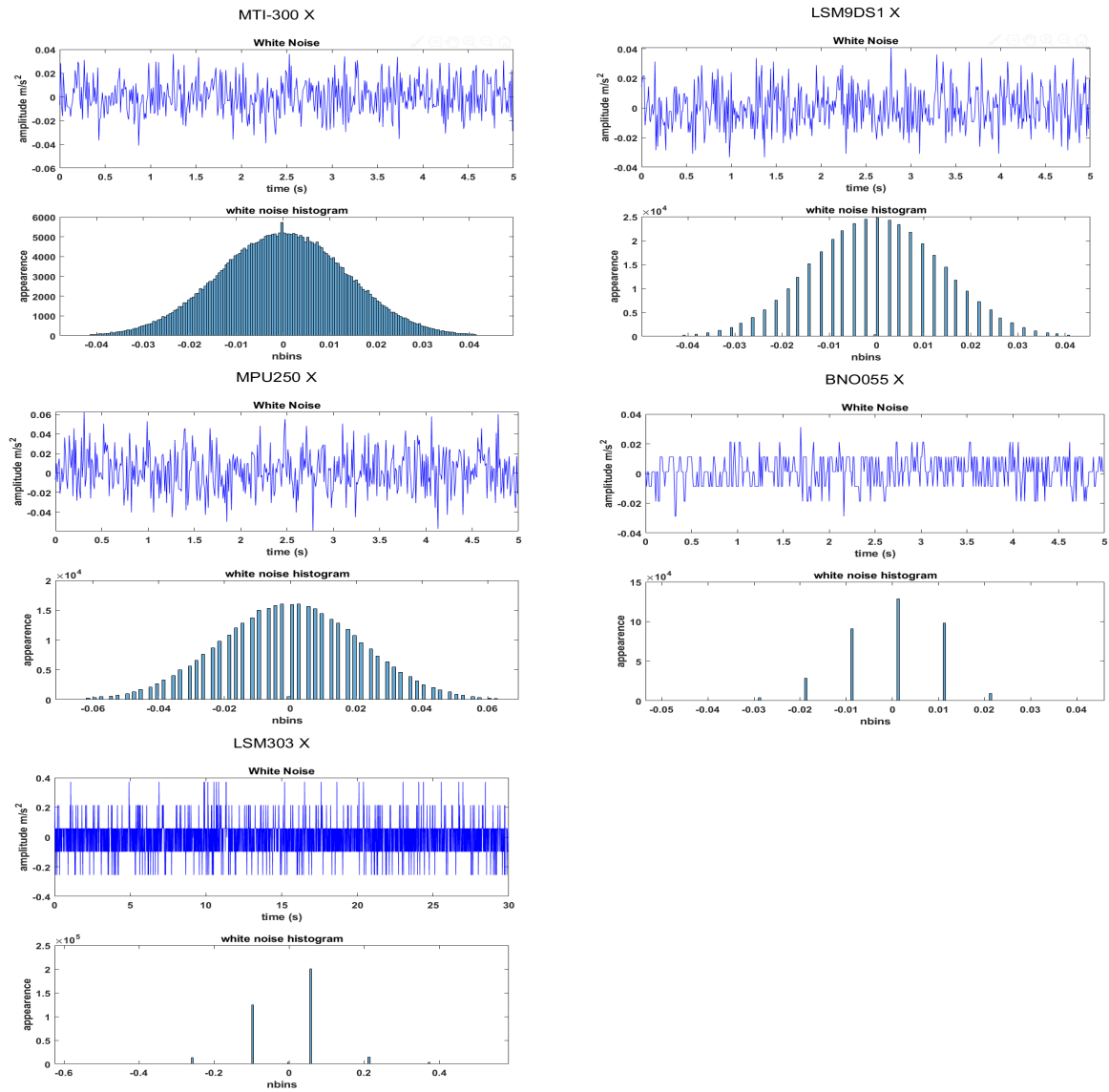
Figure 5.3. *MEMS white noise histogram and bias instability*

MEMS accelerometer BI correlation time and variance										
Sensor	**Xsens Mti**		**MPU-9250**		**LSM9DS1**		**BNO055**		**LSM303**		Unit
	T_c	σ	T_c	σ	T_c	σ	T_c	σ	T_c	σ	
X	69	0.0015	88	0.0027	69	0.0015	71	0.0013	58	0.0104	$s - m/s^2$
Y	97	0.0018	1445	0.0032	86	0.0015	20163	0.0027	59	0.0122	$s - m/s^2$
Z	75	0.0018	93	0.0018	96	0.0018	439	0.0018	63	0.0018	$s - m/s^2$

Table 5.7. *Bias instability correlation time and standard deviation*
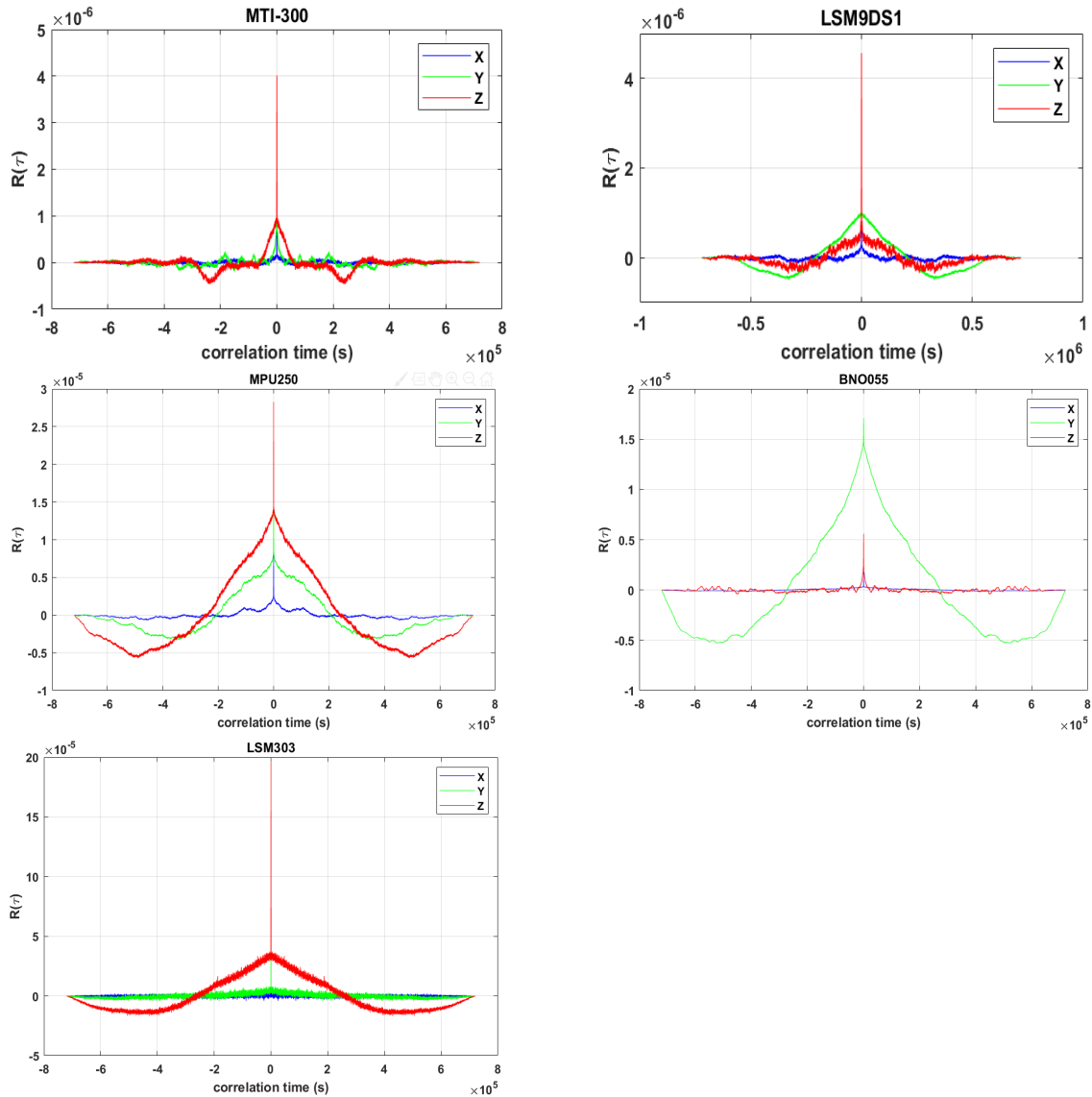
Figure 5.4. *MEMS Bias instability correlation time and Variance*

Other parameter used to initialized the Kalman filter was the sensor noise uncertainty, which is the standard deviation of the sensor noise. To compute this, the measured noise spectral density shown in Table 5.2 was used. The noise standard deviation σ can be found by multiplying the noise density by square root of the sampling rate used to acquire the data as $\sigma = ND \times \sqrt{Fs}$ The results of bias instability estimation with Kalman filter are presented in Figure 5.5. For the sake of simplicity, only the x-asis results is presented. The results showed that the estimated bias instability track well the measured bias instability in MTI-300, LSM9DS1 and MPU250. The performance was really poor for BNO055 and LSM303. This could be probably due to the non Gaussian noise present in the data, as it can be reflected in the histograms presented in figure 5.3. The root-mean-square error (RMSE) between then the measured and estimated bias instability can be seen in Table 5.8, where LSM9DS1 showed the best results of $5.7 \times 10^{-4} m/s^2$.

RMSE of real and estimated bias instability (m/s^2)				
sensor	**MTI-300**	**LSM9DS1**	**MPU-9250**	**BNO055**	**LSM303DLHC**
RMSE	7.1×10^{-4}	5.7×10^{-4}	11×10^{-4}	20×10^{-4}	28×10^{-4}

Table 5.8. *Acceleration bias instability RMSE*
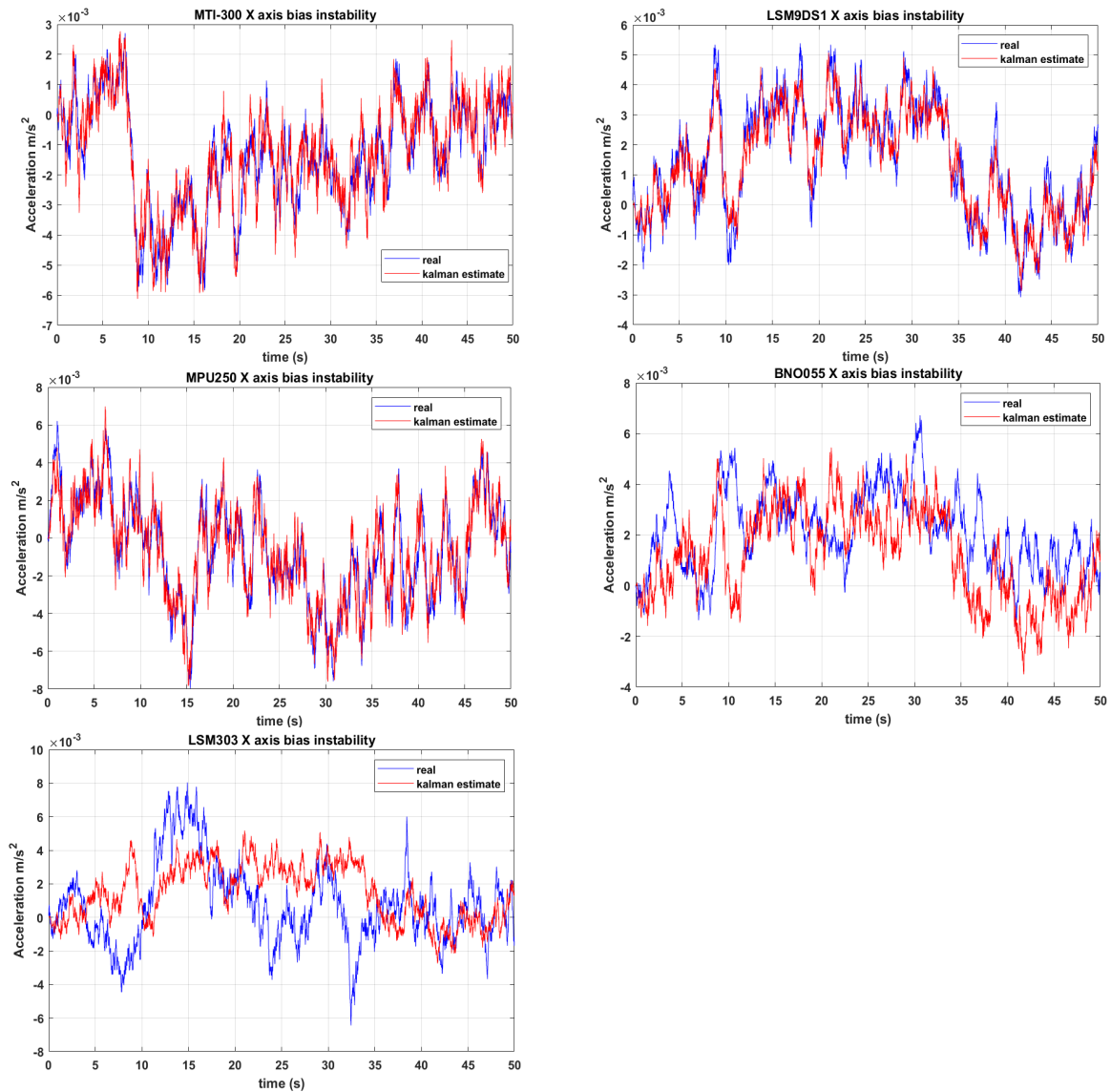
Figure 5.5. *MEMS-accelerometer measured and Kalman filter estimated Bias instability*

5.2 Test Results of Elevator vibration

In this phase, data from all sensors is presented. The main purpose this test was to evaluate the performance between different level MEMS accelerometers and a piezolectric accelerometer while measuring an elevator car vibration. Other objectives were to understand the behavior of these sensors while performing in low frequencies motion. To achieve this goal, a set of procedure were done as explained below:

- **procedure:** The data was acquired while the comparison platform is on the elevator floor. The elevator was moved from floor 0 to floor 2. In the data it is included the part when the door closes before movement and the part when the door opens after the elevator has stopped. Before doing the analysis, the data preprocessed using filtering tool. These methods include the removing of the static bias and applying a moving average and high pass filters to remove measurement noises, high frequency noises and the effect constant gravity acceleration.

- **results:** The plots from all sensors are studied in time and frequency domain. The results from MEMS accelerometers are compared to piezo accelerometer.

To analyse the data, firstly the data was filtered with a 10- step moving average filter, and secondly a high pass filter with the cut-off frequency of $2Hz$ to remove the slow movement noises. Thirdly, a low pass filter with a cut-off frequency of $50Hz$ was used. This ensure the analysis of frequency between $2 - 50Hz$. Figure 5.6 illustrates the raw unfiltered signal of each sensor Z- axis. In analysis only downward motion (Z- axis component) were studied. It can be seen that MEMS-accelerometers sensors share the same shape, but piezolectric sensor is different. This is due to the fact that MEMS- accelerometer can sens the acceleration due to gravity that is pointing downward. It can also be seen that all sensors produces quite much noise that can have impacts in the data analysis.More noises are detected in sensors with high sampling rate.To reduce the effects of these noises, data was low pass filtered using the filter described above. To illustrate the noises in the data, signal-to-noise ratio method. This will be explained more in detail in this section.

After high pass filtering the data, the shape of MEMS- accelerometers look nearly the same as piezolectric sensor, this can be seen in figure 5.7. It can be seen from the figures that all sensors were able to capture the motion of elevator car. From left to right, highest peaks appear when the elevator starts to move and when the elevator stops. The motion of the elevator was sensed by the user notably at the peaks represented in the figure. This could simply indicate that this elevator produce vibration detectable by the user or in other way that this elevator in not a modern one. In the figures it can be seen that the highest peaks that each sensor have sensed from the elevator motion is different in terms of magnitudes.

As it was seen from the static data of the sensor, again LSM303 sensor showed not work properly in sensing the elevator motion. This is seen in bottom right of Figure **??** the presence noise in the acceleration data. As result, this sensor cannot be used for this purpose and tho it was excluded from comparison in this phase. From these time domain figures shown above, it can be seen that the peaks of MEMS- accelerometers sensors are nearly the same in magnitude with a deviation from piezolectric sensor in the order of
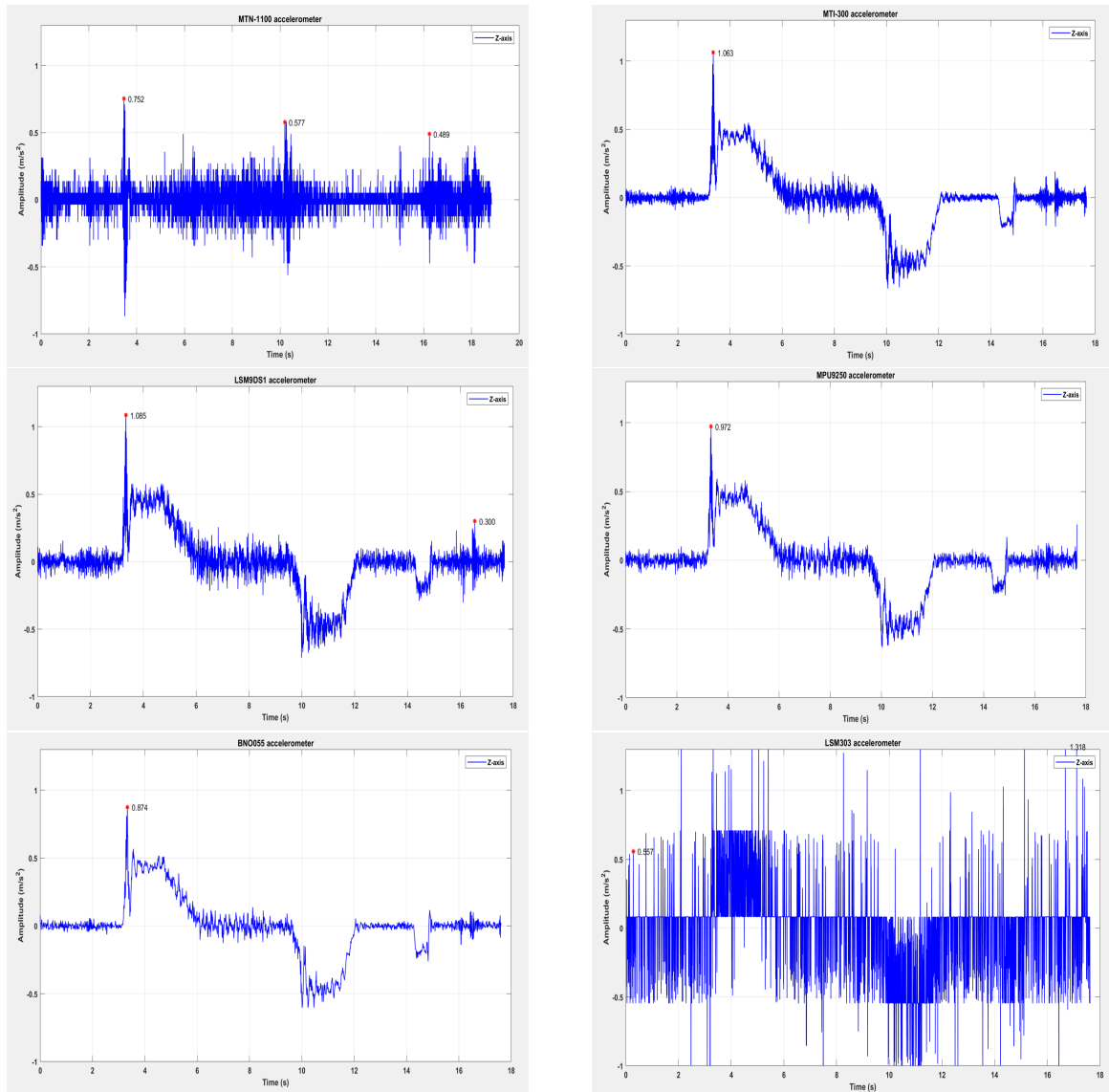
Figure 5.6. *Elevator raw unfiltered data of all sensors*

$0.3m/s^2$ of acceleration. However, this does not give much information because the sensors output the data at different rates. As there are relatively more noises in the piezolectric sensor than in MEMS sensor as it can be seen in the figure.

Frequency domain analysis were done study the vibration characteristics of the elevator. Firstly, the signal-to-noise (SNR) ratio of the sensor signal was studied. This was done by utilizing available Matlab functions for spectrogram analysis. A Kaiser window function of size equal to the length of the data and a beta parameter equal to 38 was applied.

The signal-to-noise ratio of the sensor data before and after filtering is summarized in the table 5.9 below.

It can be seen that the noise present in the data was somewhat reduced in all sensors except for LSM303 sensor, where the noises have increased.

signal-to-noise ratio (SNR) (dB)						
Name	MTN-1100	MTI-300	LSM9DS1	MPU-9250	BNO055	LSM303DLHC
SNR before	-8.6	-6.7	-7.0	-8.9	-6.6	-5.2
SNR after	-6.4	-3.5	-5.7	-3.9	-4.5	-5.9

Table 5.9. *Acceleration signal to noise ratio before and after filtering*
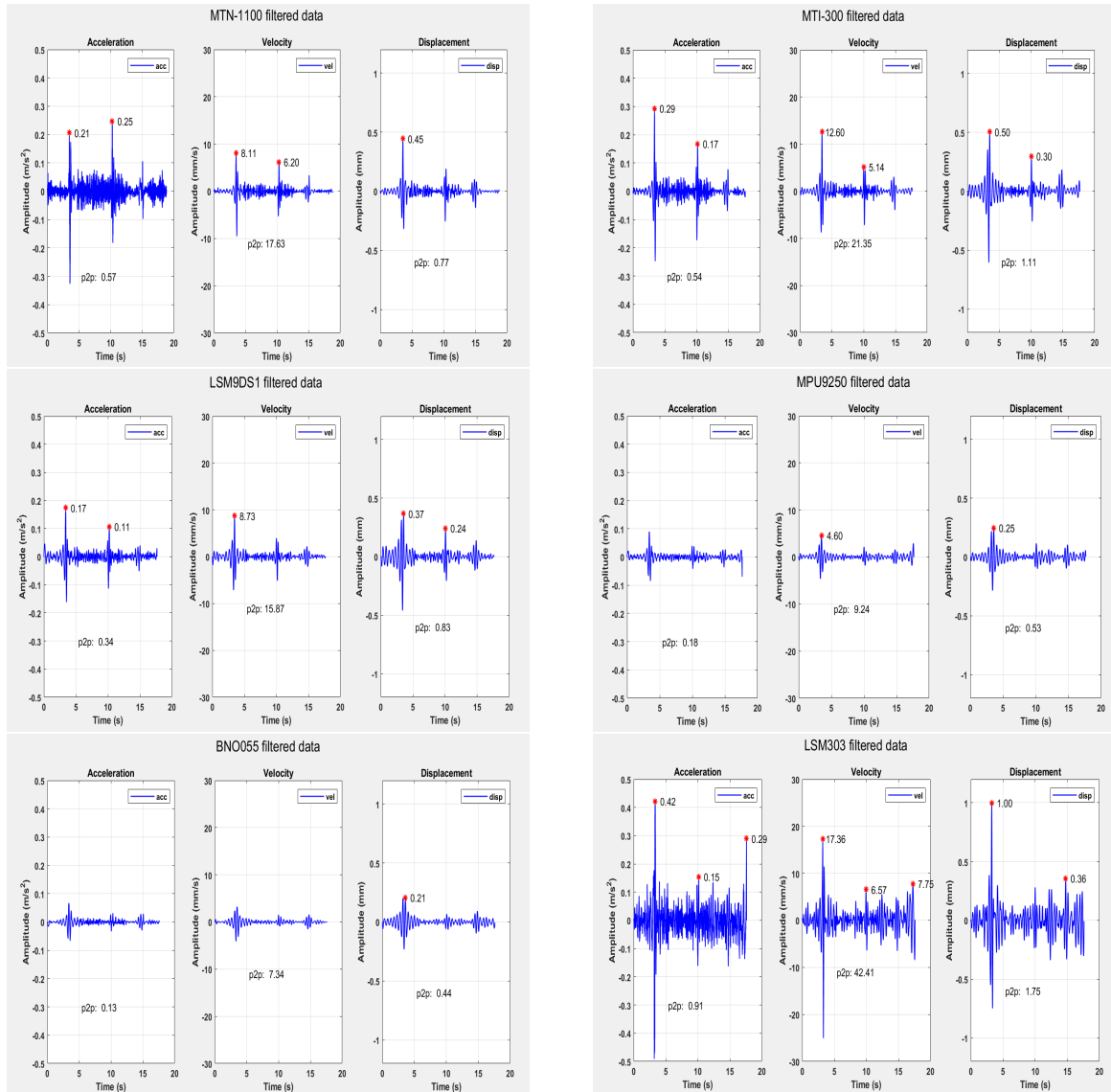
Figure 5.7. *Acceleration, velocity and displacement of elevator vibration*

Fast Fourier Transform (FFT) and Power Spectral Density of filtered data was analyzed in order to have more frequency clue of the elevator vibration while in motion. In figure 5.8 it is shown the FFT, PSD and SNR plots of each sensor. It can be noted that LSM303 is not well suited for comparison since it produced very noisy data. However, all other MEMS- accelerometer show a peak at 4.6 and $48.7Hz$. Looking at the piezolectric sensor plot, it can be seen that there are the presence of other frequencies harmonics in the data, which are clearly an from unknown noise source. For clarification, it is noticed
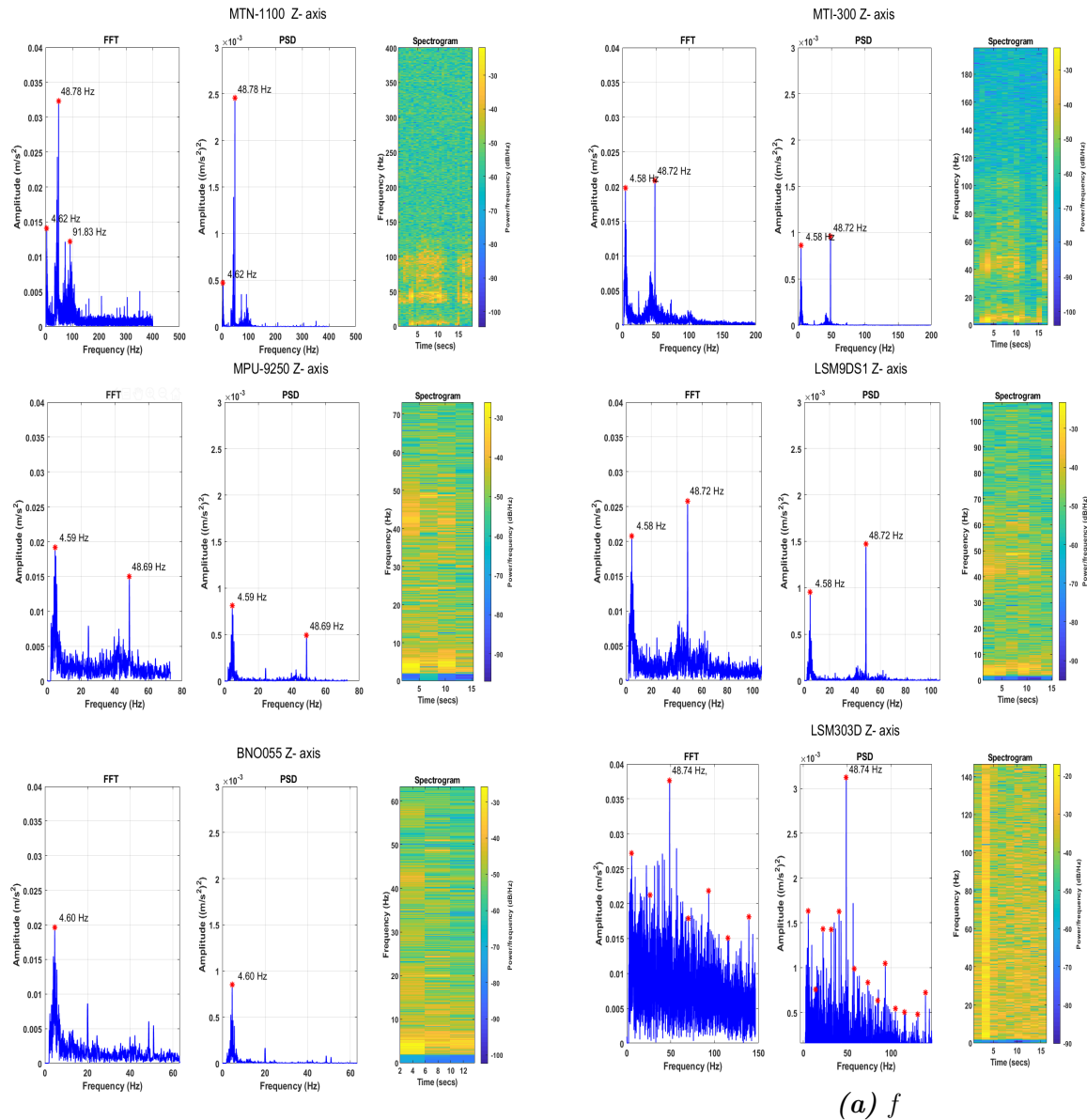
Figure 5.8. *Accelerometer FFT and PSD analysis on elevator*

that this study is mainly related to the sensor comparison and not focused in the overall elevator ride quality analysis as this is an other topic. Nonetheless, in order to analyse performance of an elevator an international standard ISO-18738-1 is available and must be applied. This standard defines well how the elevator ride quality should be analysed in more precise way.

5.3 Test Results of M12 Wheel Loader Vibration

The second use case studied in this thesis was vibration analysis of a wheel loader machine. The purpose of this test was to study the performance of different level MEMS-accelerometers compared to a piezolectric sensor while measuring the vibration of a machine. For this purpose, M12 wheel loader machine of the laboratory of Hydraulics and

Automation was utilized. Figure, 5.9 shows the comparison platform attached on the base of M12 machine.

- **procedure:** While attached to the base of M12, the engine was started and left running at idle. The speed of this engine at idle is 800 rpm as it was mentioned before in section 4. Data was acquired from MEMS- accelerometers sensors using *rosbag record* - command for 20 seconds. For piezoelectroninc sensor, data was acquired in three steps in order to get data for all X, Y and $Z-$axes. To do so, the sensor was removed from the platform and then attached in different direction according to what direction the vibration is being measured. Data was acquired in several sets in different days. As all data sets was showing the same results, this report is based on one data set and not the combination. In this phase, data was prepared in similar way as it was done for elevator case, A high pass filter was used to filter out lower frequencies and a moving average filter was used to remove high frequency noises.

- **results:** Again the performance of sensors is compared in time and frequency domain. The behavior of different level MEMS- accelerometer is compared to the piezolectric accelerometer in all 3- axes in order to determine difference behavior.

Time domain signal of all sensors is plotted after removing high frequency measurement noises with the moving average filter.
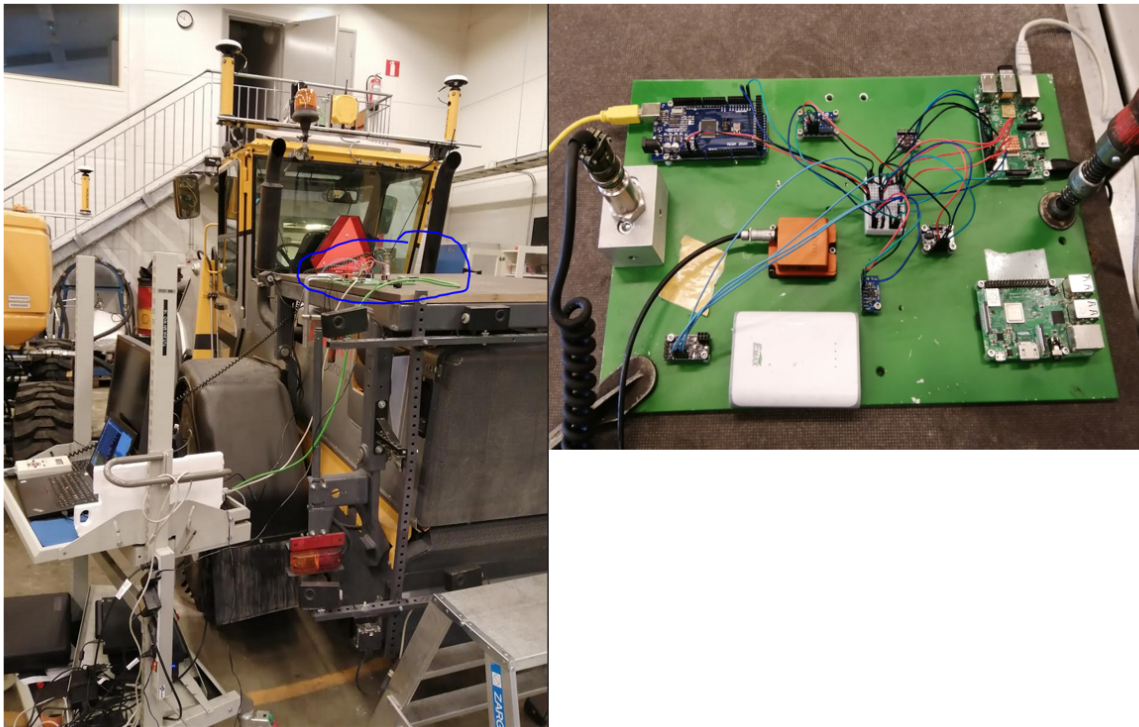
Figure 5.9. *Accelerometer comparison platform attached on base of M12 wheel loader*
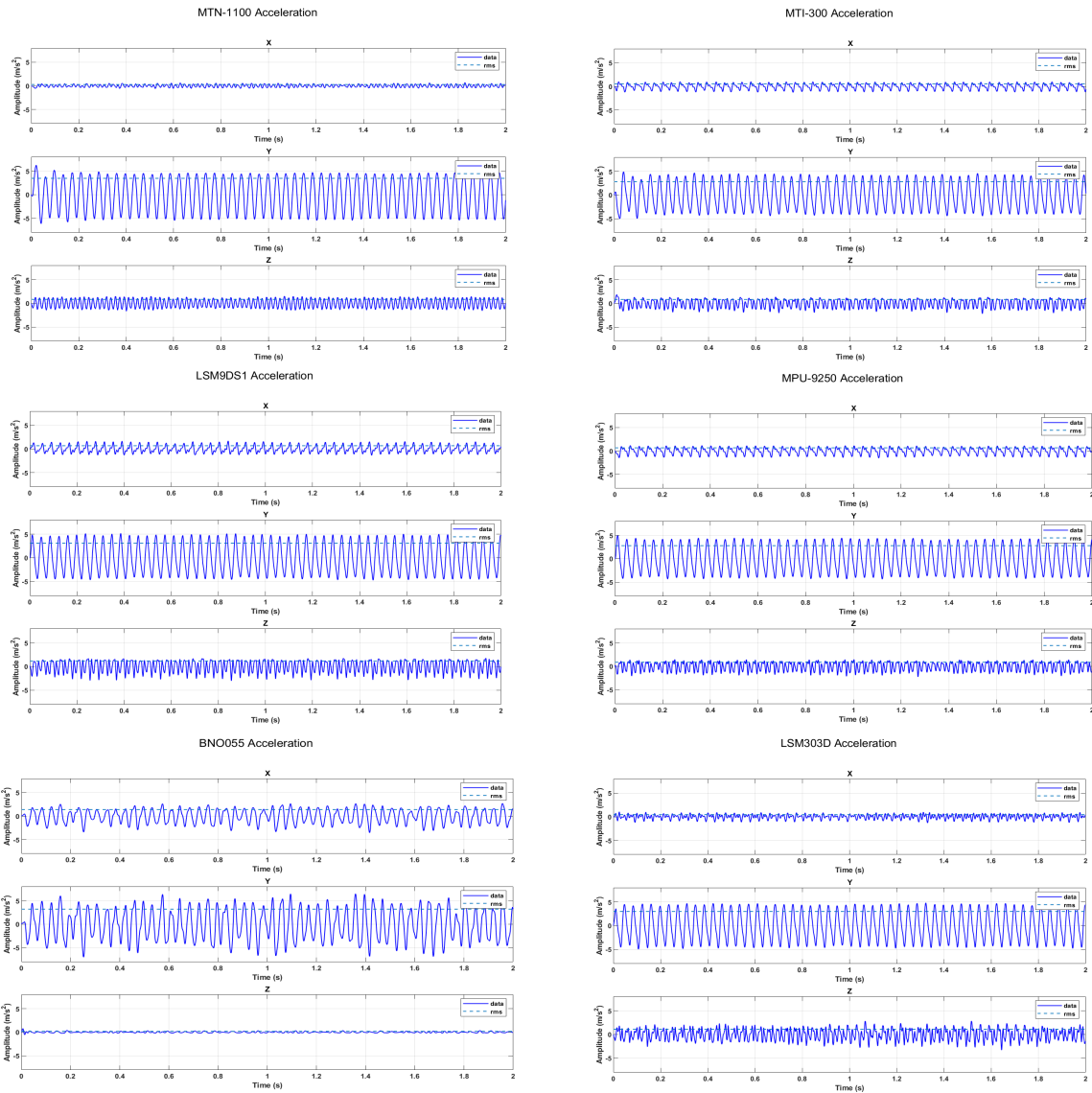
Figure 5.10. *A 2 second time history of acceleration data for all sensors after moving average filter*

Removing measurement noise in the data is crucial especially when time domain analysis is being done.In this study, it was seen that noises in data of vibrating machine are significantly lower compared to the noises in elevator case. This can seen in table 5.10, where signal-to-noise ratio of the vibrating machine is illustrated for all sensors. Again filtering the data have reduced the SNR in all sensors except for LSM303 X-axis. Based on the signal-to-noise ration in the table 5.10 it can be seen that $Y-$ axis is less noisier than X and $Z-$ axes. This is studied more in time history data in all sensors.

Figure 5.10 shows time data of all sensors for all 3 axes. By viewing the figures, it is not straightforward to identify the differences in the figure plot in $Y-$ axis, except for BNO055 sensor, which is different from others.

signal-to-noise ratio (SNR) (dB)																	
sensor	MTN-1100			MTI-300			MPU-9250			LSM9DS1			BNO055			LSM303		
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
Before	-0.7	20.1	6.3	8.9	18.1	-1.5	14.0	23.9	-4.0	9.6	20.6	-0.2	4.0	4.2	-3.8	-0.6	15.8	-6.4
After	4.8	28.3	12.3	12.6	20.6	0.1	17.5	25.8	-1.2	14.8	23.4	1.7	6.5	6.7	-2.5	-1.9	20.0	-3.1

Table 5.10. *Sensor signal-to-noise ratio of wheel loader before and after filtering*

Further analysis was done in time domain, by analysing the vibration in terms of root mean square and peak to peak of acceleration signal. The purpose of this study was to give the overall vibration severity comparison for all sensors in this use case. For each sensor, data of all axes were combined together using the root sum squared method as in Equation 5.3 in order to account for the total vibration.

$$A = \sqrt{X^2 + Y^2 + Z^2} \tag{5.3}$$

Additionally, the standard deviation and power of vibration signal was computed for further comparison. Table 5.11 summarizes the results of this test for acceleration data.

M12 vibration acceleration (m/s^2)				
name	rms	p2p	std	power
MTN-1100	3.55	5.39	1.35	12.65
MTI-300	2.98	4.92	1.17	8.91
LSM9DS1	3.35	5.33	1.26	11.23
MPU-9250	2.98	4.63	1.18	8.91
BNO055	3.50	8.72	1.67	12.29
LSM303DLHC	3.21	5.90	1.31	10.31

Table 5.11. *Acceleration quantitative results*

In some cases when the vibration is being analysed, the vibration in terms of velocity and displacement tell more about the severity in of vibration. Table 5.12 summarizes the vibration of M12 machine base in terms of velocity and displacement.

As time domain analysis does not give all information about the data, frequency domain analysis may be more indicative especially in machine condition monitoring. For the purpose of this study, frequency domain analysis was performed using same methodologies as used in the elevator case. Fast Fourier Transform and power spectral density was performed along with spectrogram for all sensor. In order to reduce ambiguity, the Fourier transform was performed before signal filtration. Figures 5.11, 5.12 and 5.13 show the

M12 vibration velocity (mm/s)					M12 vibration displacement (μm)				
Name	rms	p2p	std	power	Name	rms	p2p	std	power
MTN-1100	20.70	31.33	8.75	428.59	MTN-1100	122.8	188.8	52.3	15080
MTI-300	17.24	28.36	6.68	297.36	MTI-300	103.2	190.0	41.9	10654
LSM9DS1	19.05	30.09	7.16	362.99	LSM9DS1	112.8	186.8	44.2	12746
MPU-9250	17.09	25.23	6.38	292.19	MPU-9250	101.3	168.3	39.9	10280
BNO055	21.58	53.82	10.90	465.8	BNO055	151.6	442.9	85.2	22990
LSM303DLHC	18.16	30.50	6.81	329.89	LSM303DLHC	108.1	187.3	42.4	11699

Table 5.12. *Velocity and Displacement quantitative results*

frequency domain analysis of all sensors for all axes. FFT, PSD and a spectrogram plots are showed. As the machine running speed was kept constant at idle, the fundamental frequency was calculated according to Equation 4.2. In this case, a four cylinder engine running at $800 rpm$ will produce the fundamental frequency at $(800/60) \times 2 = 26.67 Hz$.

It can be seen from Figure 5.12 that the fundamental frequency is present with high amplitude in FFT plot and with high energy in the PSD.

In figure 5.11 it is shown the vibration in the longitudinal axis of the engine rotation speed. In this axis, it was noticed that there is clearly the presence of the second harmonic in MTN-1100 used as reference sensor. Additionally, it can be seen that there is high frequency noises present in the data. The source of this noise could be probably caused by the acquisition system made from the picoscope. MEMS sensors showed three dominant frequencies of the engine, except BNO055 which is driven by low and high frequency noises from external source. Additionally,it was seen also that LSM303 sensor whose performance was poor in the elevator case was able to track well the vibration signal. In X-axis it can be seen that there are presence 2-order harmonics.

Figure 5.12 shows the vibration in Y-axis, which is the transversal axis to the engine rotation direction. The frequency components present in this axis were analysed. As results, the natural frequency component was reflected in all sensors with with high energy in the PSD. The second harmonic was represented accordingly, but with a low energy compared to the first harmonic. Low frequency noises were detected in BNO055 as it was seen previously for X-axis.
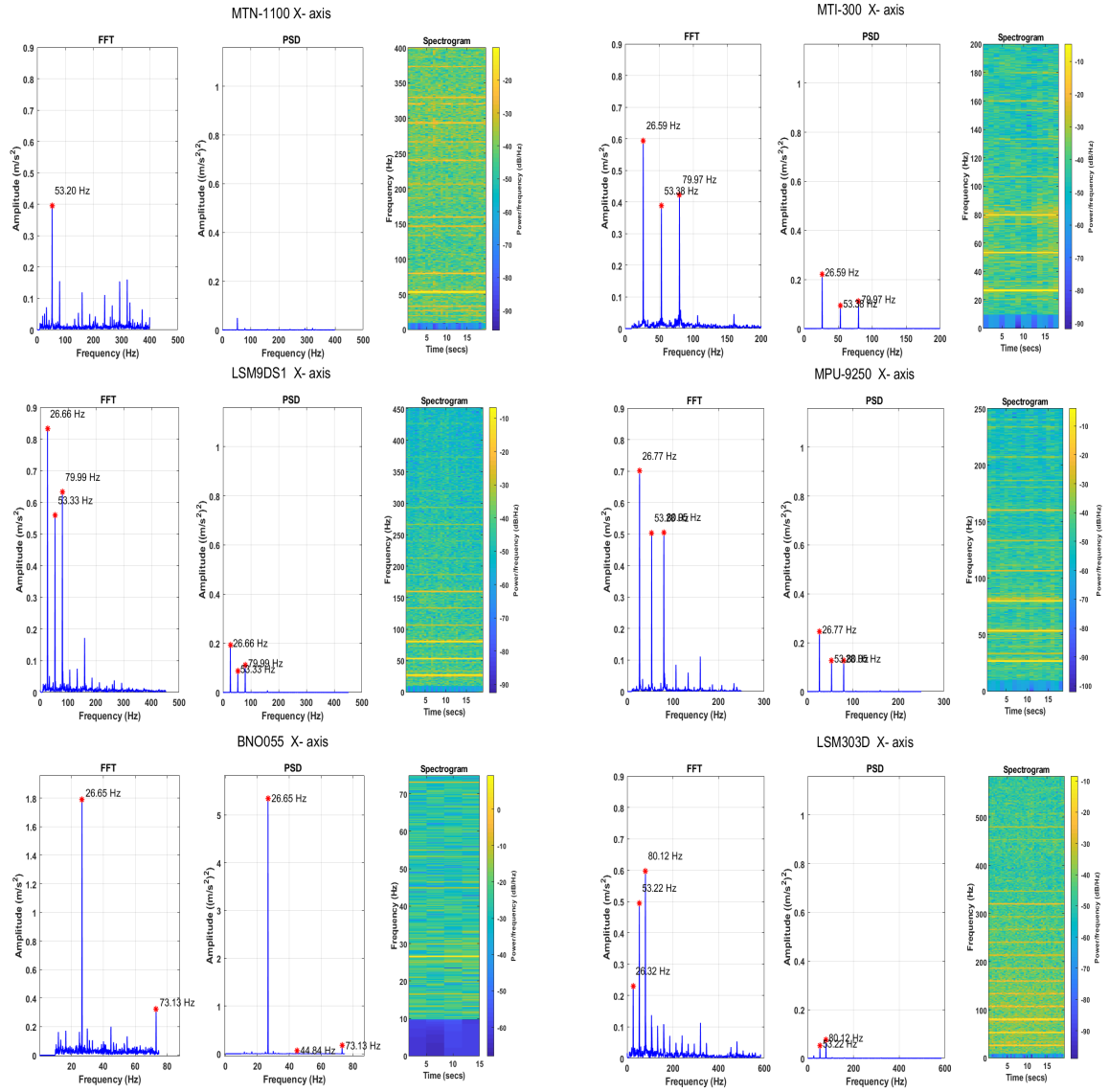
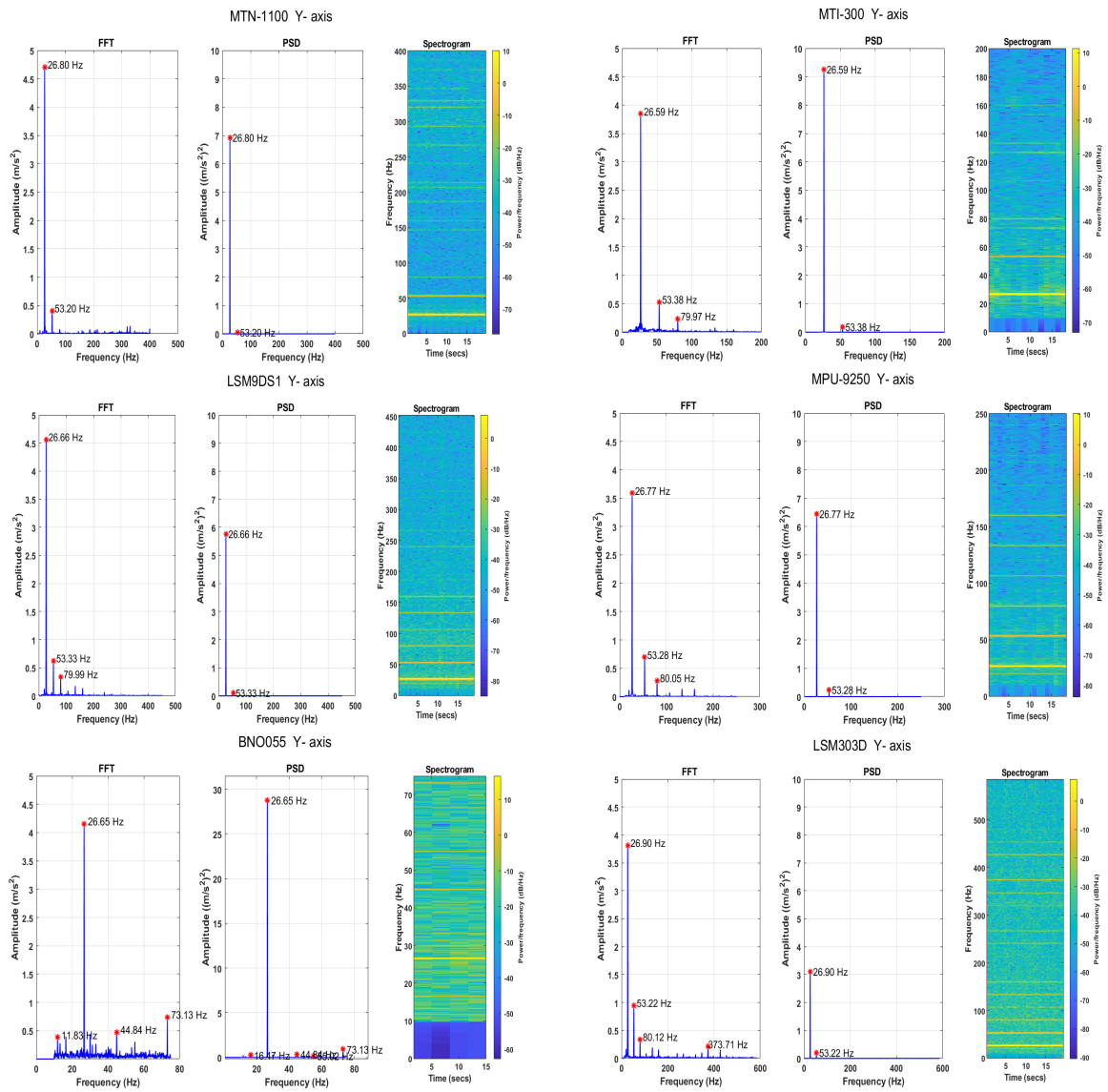**Figure 5.11.** *Accelerometer X- axis frequency analysis on wheel loader*

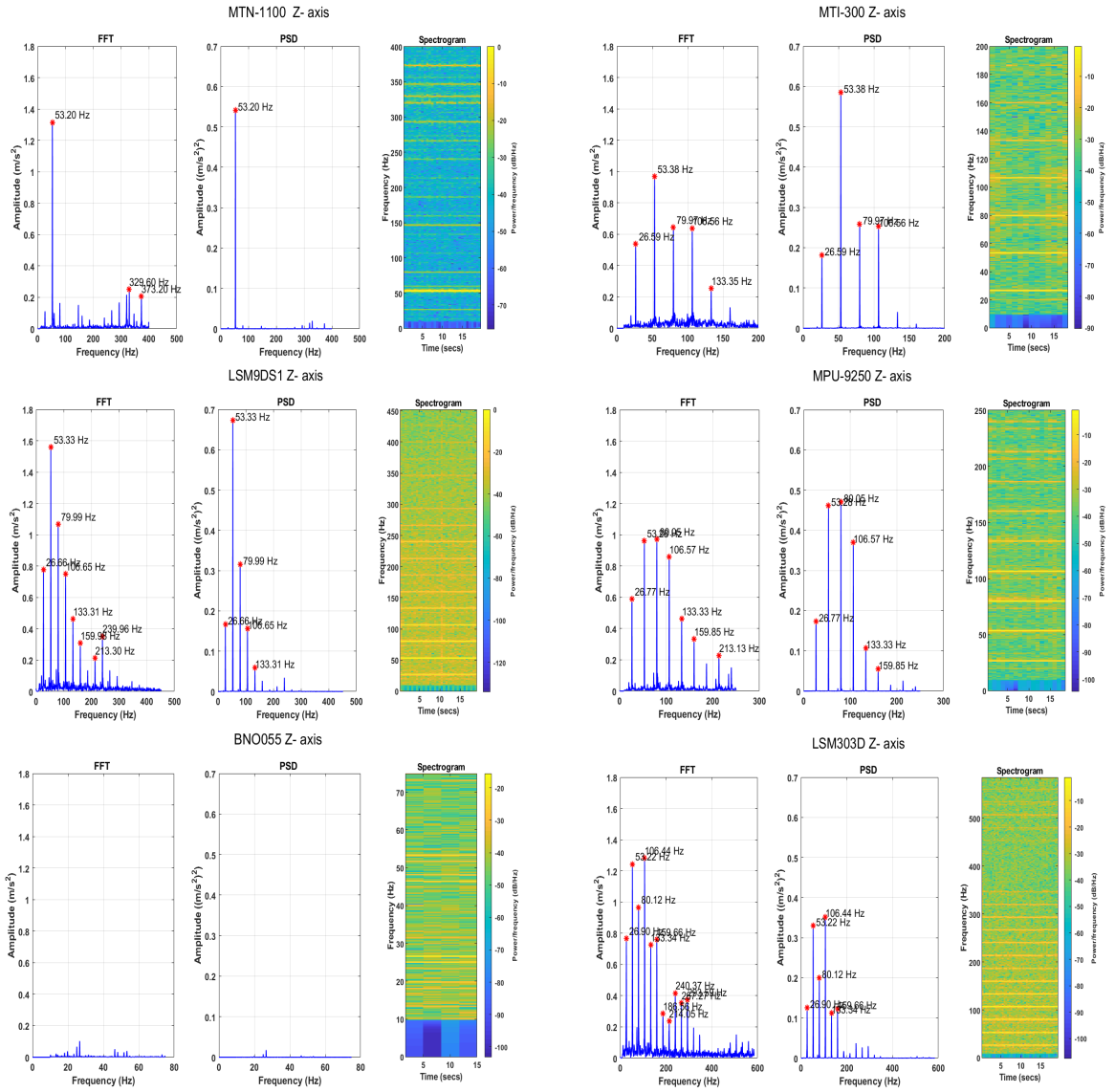***Figure 5.12.*** *Accelerometer Y- axis frequency analysis on wheel loader*

Figure 5.13. *Accelerometer Z- axis frequency analysis on wheel loader*

5.4 Discussion on results

The application of MEMS based accelerometer sensors instead of traditional piezoelectric sensors in machine condition monitoring have been a subject of concern for many users. However, choosing the sensor that meets the requirement is not a simple task especially when sensor price is a critical aspect. Several other aspects that may influence the choice of sensor including price are application environment, acquisition system and sensor performance characteristics. Accelerometer characteristics such as frequency bandwidth, noise density, analog-to-digital resolution, resonance frequency and temperature dependency are mostly taken into consideration when choosing the right sensor. MEMS based accelerometers have gained wide interest due to their functionalities that defers from traditional piezoelectric sensors. Stochastic noise analysis of MEMS sensors may give information that are useful when planning an building algorithm for inertial applications such as inertial navigation, when these information are not provided in the technical data sheets. The validation of sensor noise were performed by studying the most common stochastic noises in MEMS sensors namely: velocity random walk, bias instability and acceleration random walk. It was seen that these noises are present in all sensors at some levels. However, there was a clear difference of noise between different sensor axes. Other results showed that the measured noise densities of sensors were lower than ones ones expressed in the sensor data sheets, which is good thing for the application.

Raspberry Pi is one of the cheapest devices that can be used as Edge device due to it's multipurpose functionalities. This devices can give several opportunities to integrate sensors in IoT loop when it is used as part of the acquisition system. The benefit of using Raspberry Pi in this case is that in addition to data acquisition, some Edge analytics can be done at same time while sensor data is being read. However, using this single board computer have some drawbacks due to the limitation of Raspberry Pi itself. Moreover, it must be noted that the software used have also some disadvantages that may cause the raspberry Pi performance to be slow when several sensors are running at same time. As results, the maximum rates at which data was read from sensor registers and published to ROS topics were different from the output data rate mentioned in the data sheet. Additionally, it was seen that the time interval between samples was not accurate. This is also due to the fact that neither ROS nor raspberry Pi has real time functionalities. To get reliable results in data analysis, all data must be resampled in order to make the sampling time interval to be constant for all data set. As advantage, data resampling reduces problem of aliasing. The signal to noise ratio was one of the methods used to evaluate the noise characteristics of sensors. The results showed that the ratio of noise is highly dependent on the quality of sensor. This is also due to the different level of

calibration and the signal conditioning between sensors.

The performance of sensors when working in low vibration motion (elevator case) showed good results for all sensors except for LSM303, which contained a lot of outliers in the data. As result, this this sensor cannot be trusted in this application case. Despite this malfunction of LSM303 sensor in low motion application, this sensor worked well in detecting the wheel loader vibration. Among all MEMS sensors, Xsens Mti-300 showed to work the best. According to the experiments, it was found that Bosch BNO055 was not suitable for working with raspberry Pi due to the clock stretching issues. Due to that, higher sampling rates could not be achieved.

The results from frequency domain analysis showed that the vibration frequencies present in the data were restricted to the sampling rate and bandwidth characteristics of each sensor. Moreover, it was found a slight disagreement in the signal amplitudes for all sensors. This is due to the fact that MEMS sensors have different built-in vibration damping factor that may affects the signal amplitudes.

The overall results from experiment in this work showed that a simple low cost vibration monitoring system can be build using a single board computer such as raspberry Pi. Nonetheless, the choice of sensor to be utilized will depend not only on the physical characteristics of the sensor, but also on the budget limit and application environment. Some sensors used in this work are well suitable when using for experiment purposes in lab environments. The applicability of these sensors in industrial environment is not possible since they do not complete the requirements and standards of this application area. However, since in recent year there have been a large interest in MEMS sensor application in machine condition monitoring applications, different manufacturer have offered good solution that are suitable in industrial applications for fairly low prices. Sensor manufacturers such as Analog Devices, ST Microelectronics and Bosch offers already good solutions for MEMS accelerometers that could fit well in critical environments.

6 CONCLUSION AND FUTURE WORK

In this thesis the applicability of MEMS- based accelerometer sensors for machine vibration monitoring was studied, Additionally, the stochastic behavior of these MEMS accelerometers was analysed. The capability of a raspberry Pi to process this sensor data was evaluated respectively. The study was conducted firstly by analyzing the static data from sensor output and secondly by measuring the vibration of an elevator car and a wheel loader machine. The analysis was done by comparing several MEMS accelerometer sensor output.To have more critical results, a piezoelectric based sensor was used as reference in order to enhance the comparison process. To study the noise characteristics of MEMS- accelerometers the combination of power spectral density and Allan variance methods was utilized. As result it was seen that Allan Variance analysis is powerful in analysing these stochastic noises. Allan variance plots showed that these MEMS sensors are highly dominated by Velocity Random walk, Bias Instability and Acceleration Random Walk. The stochastic noise modeling using the first-order Gauss-Markov process and the Kalman filter functionalities were also tested giving positive results. Other analysis related to sensor performance such as noise density and signal-to-noise ratio in the sensor data was studied. The signal-to-noise ratio analysis showed that noises are more effective in signal that contain low vibration motion compared to high vibration motion.

Vibration monitoring system based on MEMS- accelerometer sensors might be beneficial in several applications. However, the capability of each single sensor must be analyzed before the application. This is due to the fact that in some applications the requirement are high, so that using the low level sensors would not an option. Nevertheless, it must be put in mind that building a cheap acquisition system is not always a simple task, since not all MEMS sensors that are capable to work with single board computers or low cost acquisition system. Due to the nature and characteristics, their application in condition monitoring systems have been a study of interest for many. However, in industrial applications, one need to be more precise while making the choice of sensors. In future work, a wide investigation of sensor market in industrial application point of view could be beneficial and for the purpose of EDGE computing, all analysis will be done on the Raspberry Pi at same time while the sensor is reading the data and then the

analysed results could be monitored in cloud through a data base.

REFERENCES

[1] M. Looney. An introduction to MEMS vibration monitoring. In: *Analog Dialogue* 48.06 (2014), 1–3.

[2] C. Murphy. Choosing the Most Suitable MEMS Accelerometer for Your Application—Part. In: *50 Maximize the Run Time in Automotive Battery Stacks Even as Cells Age* (2017), 5.

[3] VectorNav. *Inertial sensor categories*. URL: https://www.vectornav.com/support/library/imu-and-ins (visited on 10/04/2019).

[4] R. Rail-Ip. Evaluation of low-cost MEMS accelerometers and investigation of inertial algorithms for dead reckoning in railway environment. PhD thesis. Citeseer, 2014.

[5] G. Seçer. Deterministic and stochastic error modeling of inertial sensors and magnetometers. PhD thesis. bilkent university, 2012.

[6] D. R. Greenheck. Design and characterization of a low cost MEMS IMU cluster for precision navigation. In: (2015).

[7] J. M. Barrett. Analyzing and modeling low-cost MEMS IMUs for use in an inertial navigation system. In: (2014).

[8] A. R. Spielvogel and L. L. Whitcomb. Adaptive estimation of measurement bias in six degree of freedom inertial measurement units: Theory and preliminary simulation evaluation. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Sept. 2017, 5880–5885. DOI: 10.1109/IROS.2017.8206480.

[9] R. Bhardwaj, V. Kumar, and N. Kumar. Allan variance the stability analysis algorithm for MEMS based inertial sensors stochastic error. In: *2015 International Conference and Workshop on Computing and Communication (IEMCON)*. Oct. 2015, 1–5. DOI: 10.1109/IEMCON.2015.7344524.

[10] N. El-Sheimy, H. Hou, and X. Niu. Analysis and modeling of inertial sensors using Allan variance. In: *IEEE Transactions on instrumentation and measurement* 57.1 (2007), 140–149.

[11] A. Hussen and I. Jleta. Low Cost Inertial Sensors Modeling Using Allan Variance. In: *International Journal of Computer, Electrical, Automation, Control and Information Engineering* 9.5 (2015), 1237–1242.

[12] A. Quinchia, G. Falco, E. Falletti, F. Dovis, and C. Ferrer. A comparison between different error modeling of MEMS applied to GPS/INS integrated systems. In: *Sensors* 13.8 (2013), 9549–9588.

[13] J. Honkakorpi. MEMS-based motion state estimation and control of hydraulic manipulators. In: *Tampereen teknillinen yliopisto. Julkaisu-Tampere University of Technology. Publication; 1219* (2014).

[14] S. B. Chaudhury, M. Sengupta, and K. Mukherjee. Vibration monitoring of rotating machines using MEMS accelerometer. In: *International journal of scientific engineering and research* 2.9 (2014), 5–11.

[15] A. Albarbar, S. Mekid, A. Starr, and R. Pietruszkiewicz. Suitability of MEMS accelerometers for condition monitoring: An experimental study. In: *Sensors* 8.2 (2008), 784–799.

[16] J. Junnola. *The suitability of low-cost measurement systems for rolling element bearing vibration monitoring.* 2017.

[17] D. Unsal and K. Demirbas. Estimation of deterministic and stochastic IMU error parameters. In: *Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium.* IEEE. 2012, 862–868.

[18] IEEE Standard Specification Format Guide and Test Procedure for Linear,Single-Axis, Nongyroscopic Accelerometers Corrigendum 1: Changes to Annex K and Annex L. In: *IEEE Std 1293-1998/Cor 1-2008 (Corrigendum to IEEE Std 1293-1998)* (Nov. 2008), 1–13. DOI: 10.1109/IEEESTD.2008.4690904.

[19] D. W. Allan and J. A. Barnes. A modified Allan variance with increased oscillator characterization ability. In: *Proceedings of the 35th Annual Frequency Control Symposium.* Vol. 5. 1981, 470–475.

[20] R. E. Kalman. A new approach to linear filtering and prediction problems. In: *Journal of basic Engineering* 82.1 (1960), 35–45.

[21] G. G. Redhyka, D. Setiawan, and D. Soetraprawata. Embedded sensor fusion and moving-average filter for Inertial Measurement Unit (IMU) on the microcontroller-based stabilized platform. In: *2015 International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT).* IEEE. 2015, 72–77.

[22] N. El-Sheimy, S. Nassar, and A. Noureldin. Wavelet de-noising for IMU alignment. In: *IEEE Aerospace and Electronic Systems Magazine* 19.10 (2004), 32–39.

[23] C. Peter. *Robotics, Vision and Control Fundamentals Algorithms in Matlan.* Springer, 2017. ISBN: 978-3-319-54412-0. DOI: 10.1007/978-3-319-54413-7.

[24] V. Kaajakari et al. Practical MEMS: Design of microsystems, accelerometers, gyroscopes, RF MEMS, optical MEMS, and microfluidic systems. In: *Las Vegas, NV: Small Gear Publishing* (2009).

[25] D. Titterton, J. L. Weston, and J. Weston. *Strapdown inertial navigation technology.* Vol. 17. IET, 2004.

[26] M. El-Diasty and S. Pagiatakis. Calibration and stochastic modelling of inertial navigation sensor errors. In: *Journal of Global Positioning Systems* 7.2 (2008), 170–182.

[27] I. Skog and P. Händel. Calibration of a MEMS inertial measurement unit. In: *XVII IMEKO world congress*. 2006, 1–6.

[28] A. Kim and M. Golnaraghi. Initial calibration of an inertial measurement unit using an optical position tracking system. In: *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No. 04CH37556)*. IEEE. 2004, 96–101.

[29] H. Guang-Lin, T. Si-Qian, S. Qiang, and Z. Pian. Research on Calibration and Parameter Compensation of MEMS Inertial Sensors Based on Error Analysis. In: *2012 Fifth International Symposium on Computational Intelligence and Design*. Vol. 1. Oct. 2012, 325–329. DOI: `10.1109/ISCID.2012.89`.

[30] Z. F. Syed, P. Aggarwal, C. Goodall, X. Niu, and N. El-Sheimy. A new multi-position calibration method for MEMS inertial navigation systems. In: *Measurement Science and Technology* 18.7 (2007), 1897.

[31] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. In: *Journal of Basic Engineering* 82 (1960). cited By 29839. DOI: `10.1115/1.3662552`. URL: `http://dx.doi.org/10.1115/1.3662552`.

[32] C. C. M. Naranjo. Analysis and modeling of MEMS based inertial sensors. In: *Kungliga Tekniska Kgskolan, Stockholm, School of Electrical Engineering* (2008).

[33] M. M. Tehrani. Ring laser gyro data analysis with cluster sampling technique. In: *Fiber Optic and Laser Sensors I*. Vol. 412. International Society for Optics and Photonics. 1983, 207–220.

[34] R. Bhardwaj, V. Kumar, and N. Kumar. Allan variance the stability analysis algorithm for MEMS based inertial sensors stochastic error. In: *2015 International Conference and Workshop on Computing and Communication (IEMCON)*. IEEE. 2015, 1–5.

[35] J. Li and J. Fang. Not fully overlapping Allan variance and total variance for inertial sensor stochastic error analysis. In: *IEEE Transactions on Instrumentation and Measurement* 62.10 (2013), 2659–2672.

[36] S. M. Yadav, S. K. Shastri, G. B. Chakravarthi, V. Kumar, A. D. Rao, and V. K. Agrawal. A Fast, Parallel Algorithm for Fully Overlapped Allan Variance and Total Variance for Analysis and Modeling of Noise in Inertial Sensors. In: *IEEE Sensors Letters* 2 (2018), 1–4.

[37] I. Board. Ieee standard specification format guide and test procedure for single-axis interferometric fiber optic gyros. In: *IEEE Std* (1998), 952–1997.

[38] A. Gelb. *Applied optimal estimation*. MIT press, 1974.

[39] M. S. Grewal, L. R. Weill, and A. P. Andrews. *Global positioning systems, inertial navigation, and integration*. John Wiley & Sons, 2007.

[40] H. Hou. *Modeling inertial sensors errors using Allan variance*. University of Calgary, Department of Geomatics Engineering, 2004.

[41] H. W. Sorenson. Least-squares estimation: from Gauss to Kalman. In: *IEEE spectrum* 7.7 (1970), 63–68.

[42] P. S. Maybeck. *Stochastic models, estimation, and control*. Vol. 3. Academic press, 1982.

[43] G. Welch, G. Bishop, et al. An introduction to the Kalman filter. In: (1995).

[44] Q. Li, R. Li, K. Ji, and W. Dai. Kalman Filter and Its Application. In: *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*. Nov. 2015, 74–77. DOI: `10.1109/ICINIS.2015.35`.

[45] P. J. Hargrave. A tutorial introduction to Kalman filtering. In: *IEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments*. Feb. 1989, 1/1–1/6.

[46] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. MIT press, 2005.

[47] Xsens. *MTi 100-series Product Specification*. URL: `https://www.xsens.com/products/mti-100-series/` (visited on 09/21/2019).

[48] STMicroelectronics. *LSM9DS1 Datasheet*. URL: `https://www.digikey.fi/product-detail/en/stmicroelectronics/LSM9DS1` (visited on 12/21/2019).

[49] *Raspberry Pi specs*. URL: `https://en.wikipedia.org/wiki/Raspberry_Pi` (visited on 10/03/2019).

[50] *Robot Operating System*. URL: `https://www.ros.org/about-ros/` (visited on 10/01/2019).

[51] *Vibration Analysis*. Dec. 11, 2019. URL: `https://blog.endaq.com/vibration-analysis-fft-psd-and-spectrogram` (visited on 12/11/2018).

[52] W. Briggs and V. Henson. *The DFT: An Owners' Manual for the Discrete Fourier Transform*. Other Titles in Applied Mathematics. Society for Industrial and Applied Mathematics, 1995. ISBN: 9780898713428. URL: `https://books.google.fi/books?id=coq49%5C_LRURUC`.

[53] M. Heideman, D. Johnson, and C. Burrus. Gauss and the history of the fast Fourier transform. In: *IEEE ASSP Magazine* 1.4 (1984), 14–21.

[54] P. Heckbert. Fourier transforms and the fast Fourier transform (FFT) Algorithm. In: *Computer Graphics* 2 (1995), 15–463.

[55] J. L. Flanagan. Speech synthesis. In: *Speech Analysis Synthesis and Perception*. Springer, 1972, 204–276.

[56] A. Taghizadeh-Alisaraei, B. Ghobadian, T. Tavakoli-Hashjin, and S. S. Mohtasebi. Vibration analysis of a diesel engine using biodiesel and petrodiesel fuel blends. In: *fuel* 102 (2012), 414–422.

[57] AGMLABS. *A Four stroke engine working principle*. URL: `https://www.youtube.com/watch?v=OGj8OneMjek`.

[58] Bosch. *Bosch BNO055 Intelligent 9-Axis Absolute Sensor*. URL: `https://www.mouser.fi/new/bosch/bosch-bno55-sensor/` (visited on 11/07/2019).

[59] I. Inc. *MPU-9250 Product Specification*. URL: `https://www.digikey.fi/product-detail/en/tdk-invensense/MPU-9250/1428-1019-1-ND/4626450` (visited on 09/21/2019).

[60] Monitran. *mtn/1100 piezoelectric sensors*. URL: `http://monitran.com/Products/Accelerometers` (visited on 01/20/2020).

A APPENDIX

A.1 Sensor datasheet specifications

Figures A.1,A.2,A.3,A.4,A.5,A.6,A.7,A.8 below illustrate the characteristics of the sensors used in this study. Additional information related to each sensor can be found from the manufacturer websites or in reference attached to the figures.

4.6.2   Accelerometers and magnetometer

The MTi 10-series and MTi 100-series use the same accelerometers and magnetometer. The output of the magnetometer is in arbitrary units (a.u.), one a.u. is the magnetic field strength during calibration at Xsens' calibration lab. This is approximately 40 uT. An accelerometer range of 15g is available as well.

Accelerometers/magnetometer: all products: MTi-10, MTi-20, MTi-30, MTi-100, MTi-200, MTi-300, MTi-G-700

Accelerometers/magnetometers specification			MTi 10-series and MTi 100-series	
Acceleration			Typical	Max
	Standard full range	[m/s2]	50	-
	Bias repeatability (1 yr)	[m/s2]	0.03	0.05
	In-run bias stability	[µg]	40	-
	Bandwidth (-3dB)	[Hz]	375	N/A
	Noise density	[µg/√Hz]	80	150
	Non-orthogonality	[deg]	0.05	0.05
	Non-linearity	[% FS]	0.03	0.5
	A/D resolution	[bits]	16	N/A
Magnetic field				
	Full range	[µT]	-	80
	Noise density	[µGauss/√Hz]	200	-
	Non-linearity	[% FS]	0.1	-
	A/D resolution	[bits]	12	N/A

Figure A.1. *Mti-300 accelerometer specification [47]*

InvenSense	MPU-9250 Product Specification	Document Number: PS-MPU-9250A-01 Revision: 1.1 Release Date: 06/20/2016

3.2 Accelerometer Specifications

Typical Operating Circuit of section 4.2, VDD = 2.5V, VDDIO = 2.5V, T_A=25°C, unless otherwise noted.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Full-Scale Range	AFS_SEL=0		±2		g
	AFS_SEL=1		±4		g
	AFS_SEL=2		±8		g
	AFS_SEL=3		±16		g
ADC Word Length	Output in two's complement format		16		bits
Sensitivity Scale Factor	AFS_SEL=0		16,384		LSB/g
	AFS_SEL=1		8,192		LSB/g
	AFS_SEL=2		4,096		LSB/g
	AFS_SEL=3		2,048		LSB/g
Initial Tolerance	Component-Level		±3		%
Sensitivity Change vs. Temperature	-40°C to +85°C AFS_SEL=0 Component-level		±0.026		%/°C
Nonlinearity	Best Fit Straight Line		±0.5		%
Cross-Axis Sensitivity			±2		%
Zero-G Initial Calibration Tolerance	Component-level, X,Y		±60		mg
	Component-level, Z		±80		mg
Zero-G Level Change vs. Temperature	-40°C to +85°C		±1.5		mg/°C
Noise Power Spectral Density	Low noise mode		300		µg/√Hz
Total RMS Noise	DLPFCFG=2 (94Hz)			8	mg-rms
Low Pass Filter Response	Programmable Range	5		260	Hz
Intelligence Function Increment			4		mg/LSB
Accelerometer Startup Time	From Sleep mode		20		ms
	From Cold Start, 1ms V_{DD} ramp		30		ms
Output Data Rate	Low power (duty-cycled)	0.24		500	Hz
	Duty-cycled, over temp		±15		%
	Low noise (active)	4		4000	Hz

Table 2 Accelerometer Specifications

Figure A.2. MPU-9250 accelerometer specification [59]

Figure A.3. *MTN/1100 accelerometer specification[60]*

2 Module specifications

2.1 Sensor characteristics

@ Vdd = 2.2 V, T = 25 °C unless otherwise noted[a]

Table 3. Sensor characteristics

Symbol	Parameter	Test conditions	Min.	Typ.[1]	Max.	Unit
LA_FS	Linear acceleration measurement range			±2		g
				±4		
				±8		
				±16		
M_FS	Magnetic measurement range			±4		gauss
				±8		
				±12		
				±16		
G_FS	Angular rate measurement range			±245		dps
				±500		
				±2000		
LA_So	Linear acceleration sensitivity	Linear acceleration FS = ±2 g		0.061		mg/LSB
		Linear acceleration FS = ±4 g		0.122		
		Linear acceleration FS = ±8 g		0.244		
		Linear acceleration FS = ±16 g		0.732		
M_GN	Magnetic sensitivity	Magnetic FS = ±4 gauss		0.14		mgauss/LSB
		Magnetic FS = ±8 gauss		0.29		
		Magnetic FS = ±12 gauss		0.43		
		Magnetic FS = ±16 gauss		0.58		
G_So	Angular rate sensitivity	Angular rate FS = ±245 dps		8.75		mdps/LSB
		Angular rate FS = ±500 dps		17.50		
		Angular rate FS = ±2000 dps		70		
LA_TyOff	Linear acceleration typical zero-g level offset accuracy[2]	FS = ±8 g		±90		mg
M_TyOff	Zero-gauss level [3]	FS = ±4 gauss		±1		gauss
G_TyOff	Angular rate typical zero-rate level [4]	FS = ±2000 dps		±30		dps
M_DF	Magnetic disturbance field	Zero-gauss offset starts to degrade			50	gauss
Top	Operating temperature range		-40		+85	°C

1. Typical specifications are not guaranteed
2. Typical zero-g level offset value after soldering
3. Typical zero-gauss level value after test and trimming
4. Typical zero rate level offset value after MSL3 preconditioning

a. The product is factory calibrated at 2.2 V. The operational power supply range is from 1.9 V to 3.6 V.

Figure A.4. *LSM9DS1 accelerometer specification [48]*

2   Module specifications

2.1   Sensor characteristics

@ Vdd = 2.5 V, T = 25 °C unless otherwise noted[a].

Table 3. Sensor characteristics

Symbol	Parameter	Test conditions	Min.	Typ.[1]	Max.	Unit
LA_FS	Linear acceleration measurement range[2]	FS bit set to 00		±2		g
		FS bit set to 01		±4		
		FS bit set to 10		±8		
		FS bit set to 11		±16		
M_FS	Magnetic measurement range	GN bits set to 001		±1.3		gauss
		GN bits set to 010		±1.9		
		GN bits set to 011		±2.5		
		GN bits set to 100		±4.0		
		GN bits set to 101		±4.7		
		GN bits set to 110		±5.6		
		GN bits set to 111		±8.1		
LA_So	Linear acceleration sensitivity	FS bit set to 00		1		mg/LSB
		FS bit set to 01		2		
		FS bit set to 10		4		
		FS bit set to 11		12		
M_GN	Magnetic gain setting	GN bits set to 001 (X,Y)		1100		LSB/ gauss
		GN bits set to 001 (Z)		980		
		GN bits set to 010 (X,Y)		855		
		GN bits set to 010 (Z)		760		
		GN bits set to 011 (X,Y)		670		
		GN bits set to 011 (Z)		600		
		GN bits set to 100 (X,Y)		450		
		GN bits set to 100 (Z)		400		
		GN bits set to 101 (X,Y)		400		
		GN bits set to 101 (Z)		355		
		GN bits set to 110 (X,Y)		330		
		GN bits set to 110 (Z)		295		
		GN bits set to 111[2] (X,Y)		230		
		GN bits set to 111[2] (Z)		205		

a.   The product is factory calibrated at 2.5 V. The operational power supply range is from 2.16 V to 3.6 V.

Figure A.5. LSM303 accelerometer specification [48]

LSM303DLHC   **Module specifications**

Table 3. Sensor characteristics (continued)

Symbol	Parameter	Test conditions	Min.	Typ.[1]	Max.	Unit
LA_TCSo	Linear acceleration sensitivity change vs. temperature	FS bit set to 00		±0.01		%/°C
LA_TyOff	Linear acceleration typical Zero-g level offset accuracy[3],[4]	FS bit set to 00		±60		mg
LA_TCOff	Linear acceleration Zero-g level change vs. temperature	Max delta from 25 °C		±0.5		mg/°C
LA_An	Acceleration noise density	FS bit set to 00, normal mode(Table 8.), ODR bit set to 1001		220		ug/$(\sqrt{Hz})$
M_R	Magnetic resolution			2		mgauss
M_CAS	Magnetic cross-axis sensitivity	Cross field = 0.5 gauss H applied = ±3 gauss		±1		%FS/ gauss
M_EF	Maximum exposed field	No permanent effect on sensor performance			10000	gauss
M_DF	Magnetic disturbance field	Sensitivity starts to degrade. Use S/R pulse to restore sensitivity			20	gauss
Top	Operating temperature range		-40		+85	°C

1. Typical specifications are not guaranteed.

2. Verified by wafer level test and measurement of initial offset and sensitivity.

3. Typical Zero-g level offset value after MSL3 preconditioning.

4. Offset can be eliminated by enabling the built-in high-pass filter.

2.2 Temperature sensor characteristics

@ Vdd = 2.5 V, T = 25 °C unless otherwise noted [b].

Table 4. Temperature sensor characteristics

Symbol	Parameter	Test condition	Min.	Typ.[1]	Max.	Unit
TSDr	Temperature sensor output change vs. temperature			8		LSB/°C[2]
TODR	Temperature refresh rate	-		ODR[3]		Hz
Top	Operating temperature range		-40		+85	°C

1. Typical specifications are not guaranteed.

2. 12-bit resolution.

3. For ODR configuration refer to Table 72.

b. The product is factory calibrated at 2.5 V.

Figure A.6. *LSM303 accelerometer specification [48]*

BOSCH	BNO055 Data sheet	Page 13

1.2 Electrical and physical characteristics, measurement performance

Table 0-2: Electrical characteristics BNO055

OPERATING CONDITIONS BNO055						
Parameter	Symbol	Condition	Min	Typ	Max	Unit
Start-Up time	T_{Sup}	From Off to configuration mode		400		ms
POR time	T_{POR}	From Reset to Normal mode		650		ms
Data Rate	DR	s. Par. Fusion Output data rates				
Data rate tolerance 9DOF @100Hz output data rate (if internal oscillator is used)	DR_{tol}			±1		%

OPERATING CONDITIONS ACCELEROMETER						
Parameter	Symbol	Condition	Min	Typ	Max	Units
Acceleration Range	g_{FS2g}	Selectable via serial digital interface		±2		g
	g_{FS4g}			±4		g
	g_{FS8g}			±8		g
	g_{FS16g}			±16		g

OUTPUT SIGNAL ACCELEROMETER (ACCELEROMETER ONLY MODE)						
Parameter	Symbol	Condition	Min	Typ	Max	Units
Sensitivity	S	All g_{FSXg} Values, T_A=25°C		1		LSB/mg
Sensitivity tolerance	S_{tol}	Ta=25°C, g_{FS2g}		±1	±4	%
Sensitivity Temperature Drift	TCS	g_{FS2g}, Nominal V_{DD} supplies, Temp operating conditions		±0.03		%/K
Sensitivity Supply Volt. Drift	S_{VDD}	g_{FS2g}, T_A=25°C, $V_{DD\_min} \le V_{DD} \le V_{DD\_max}$		0.065	0.2	%/V
Zero-g Offset (x,y.z)	Off_{xyz}	g_{FS2g}, T_A=25°C, nominal V_{DD} supplies, over life-time	-150	±80	+150	mg
Zero-g Offset Temperature Drift	TCO	g_{FS2g}, Nominal V_{DD} supplies		±1	+/-3.5	mg/K
Zero-g Offset Supply Volt. Drift	Off_{VDD}	g_{FS2g}, T_A=25°C, $V_{DD\_min} \le V_{DD} \le V_{DD\_max}$		1.5	2.5	mg/V
Bandwidth	bw_8	2nd order filter, bandwidth programmable		8		Hz
	bw_{16}			16		Hz
	bw_{31}			31		Hz
	bw_{63}			63		Hz
	bw_{125}			125		Hz
	bw_{250}			250		Hz
	bw_{500}			500		Hz

Figure A.7. *BNO055 accelerometer specification [58]*

BOSCH	BNO055 Data sheet	Page 14

	bw$_{1000}$			1,000		Hz
Nonlinearity	NL	best fit straight line, g$_{FS2g}$		0.5	2	%FS
Output Noise Density	n$_{rms}$	g$_{FS2g}$, T$_A$=25℃ Nominal V$_{DD}$ supplies Normal mode		150	190	µg/√Hz

MECHANICAL CHARACTERISTICS ACCELEROMETER

Parameter	Symbol	Condition	Min	Typ	Max	Units
Cross Axis Sensitivity	CAS	relative contribution between any two of the three axes		1	2	%
Alignment Error	E$_A$	relative to package outline		0.5	2	°

OPERATING CONDITIONS GYROSCOPE

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Rate Range	R$_{FS125}$	Selectable via serial digital interface		125		°/s
	R$_{FS250}$			250		°/s
	R$_{FS500}$			500		°/s
	R$_{FS1000}$			1,000		°/s
	R$_{FS2000}$			2,000		°/s

OUTPUT SIGNAL GYROSCOPE (GYRO ONLY MODE)

Sensitivity via register Map	S	Ta=25℃		16.0 900		LSB/°/s rad/s
Sensitivity tolerance	S$_{tol}$	Ta=25℃, R$_{FS2000}$	--	±1	±3	%
Sensitivity Change over Temperature	TCS	Nominal V$_{DD}$ supplies -40℃ ≤ T$_A$ ≤ +85℃ R$_{FS2000}$		±0.03	±0.07	%/K
Sensitivity Supply Volt. Drift	S$_{VDD}$	T$_A$=25℃, V$_{DD\_min}$ ≤ V$_{DD}$ ≤ V$_{DD\_max}$		<0.4		%/V
Nonlinearity	NL	best fit straight line R$_{FS1000}$, R$_{FS2000}$		±0.05	±0.2	%FS
Zero-rate Offset	Off Ω$_x$ Ω$_y$ and Ω$_z$	Nominal V$_{DD}$ supplies T$_A$=25℃, Slow and fast offset cancellation off	-3	±1	+3	°/s
Zero-Ω Offset Change over Temperature	TCO	Nominal V$_{DD}$ supplies -40℃ ≤ T$_A$ ≤ +85℃ R$_{FS2000}$		±0.015	±0.03	°/s per K
Zero-Ω Offset Supply Volt. Drift	OffΩ$_{VDD}$	T$_A$=25℃, V$_{DD\_min}$ ≤ V$_{DD}$ ≤ V$_{DD\_max}$		0.1		°/s/V
Output Noise	n$_{rms}$	rms, BW=47Hz (@ 0.014°/s/√Hz)		0.1	0.3	°/s

Figure A.8. *BNO055 accelerometer specification [58]*

A.2 Other noises present in sensor data

- **Quantization Noise**

The quantization noise is errors that are introduced into an analog signal by encoding it in digital form. It is caused by small differences between actual amplitudes of the point being sampled and the bit resolution of the analog-to-digital converter[18][10][32]

Taking the gyroscope data as an example, the angle PSD for such a process [18] is:

$$
\begin{aligned}
S_\theta \left(f \right) &= \tau_0 Q^2 \left(\frac{\sin^2 \left(\pi f \tau_0 \right)}{\left(\pi f \tau_0 \right)^2} \right) \\
&\approx \tau_0 Q^2, \forall f < \frac{1}{2\tau_0},
\end{aligned}
\tag{A.1}
$$

where Q is the quantization noise coefficient.

Theoretically, the limit of Q is equal to $S/\sqrt{12}$, where S is the gyroscope scale factor in case of test that have a fixed and uniform sampling times. The relationship between rate PSD and the angle PSD can be obtained through the equation

$$
S_\Omega \left(2\pi f \right) = \left(2\pi f \right)^2 S_\theta \left(2\pi f \right)
\tag{A.2}
$$

which can be transformed to

$$
\begin{aligned}
S_\Omega \left(f \right) &= \frac{4Q^2}{\tau_0} \sin^2 \left(\pi f \tau_0 \right) \\
&\approx \left(2\pi f \right)^2 \tau_0 Q^2, \forall f < \frac{1}{2\tau_0}
\end{aligned}
\tag{A.3}
$$

By substituting Eq.4.12 into Eq.4.9 and performing the integration the Allan variance yields:

$$
\sigma^2 \left(\tau \right) = \frac{3Q^2}{\tau^2},
\tag{A.4}
$$

which indicates that the quantization noise is represanted by a slope of -1 in a log-log plot of the Allan deviation $\sqrt{\sigma^2 \left(\tau \right)}$, with respect to τ as it is illustrated in Fig.
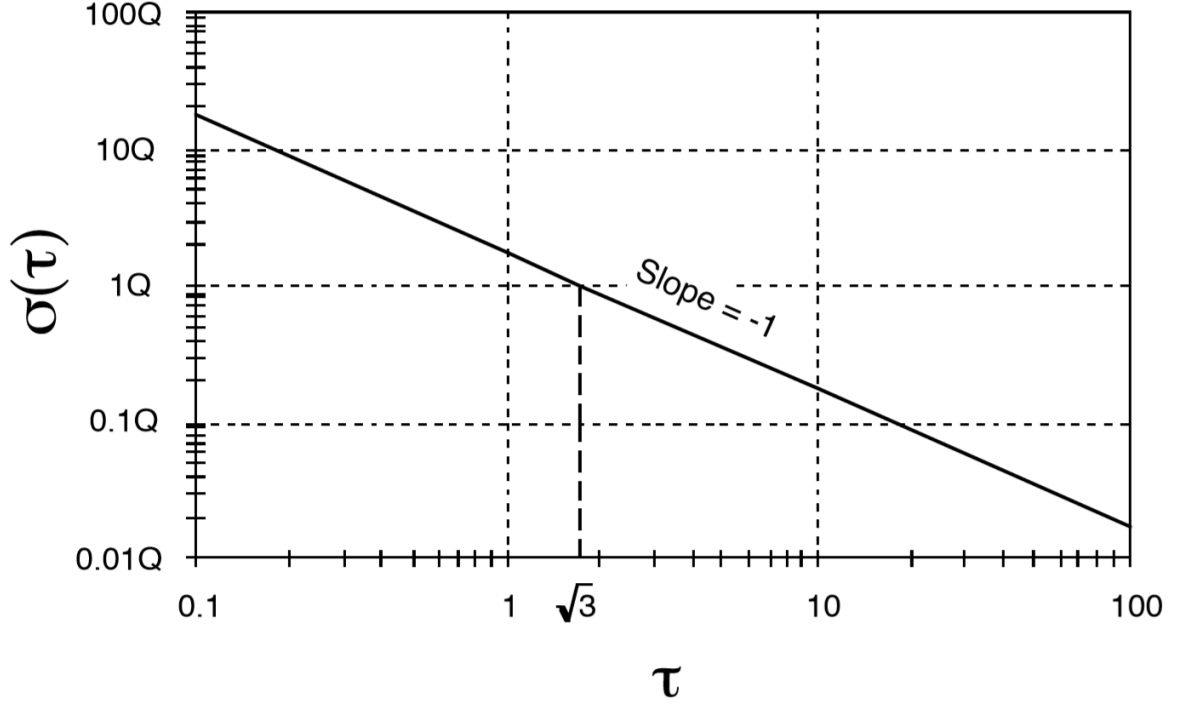
Figure A.9. *Plot for quantization noise [18]*

- **Rate/Acceleration Ramp** The presence of the rate ramp noise in IMU data may indicate a very slow monotonic change of data source intensity that persist over a long time period. However it can be caused also by very small acceleration of the platform in the same direction and persisting over a long time period. Due to fact[18], for a long but finite time interval, the rate ramp noise is rather deterministic than stochastic noise. This given by:

$$\Omega = Rt, \tag{A.5}$$

where R denotes the rate/acceleration ramp coefficient. Performing the operation on the clusters of the data that contains the input given by the equation above fives[18]

$$\sigma^2(\tau) = \frac{R^2 \tau^2}{2}, \tag{A.6}$$

which indicates that rate ramp noise has a slop of $+1$ in the log-log plot of $\sigma(\tau)$ versus τ. Figure $(fig.number)$ shows that the magnitude of the rate ramp R can be obtained from the slop line at $\tau = \sqrt{2}$ This noise is associated with a rate PSD of:

$$S_\Omega(f) = \frac{R^2}{(2\pi f)^2}. \tag{A.7}$$
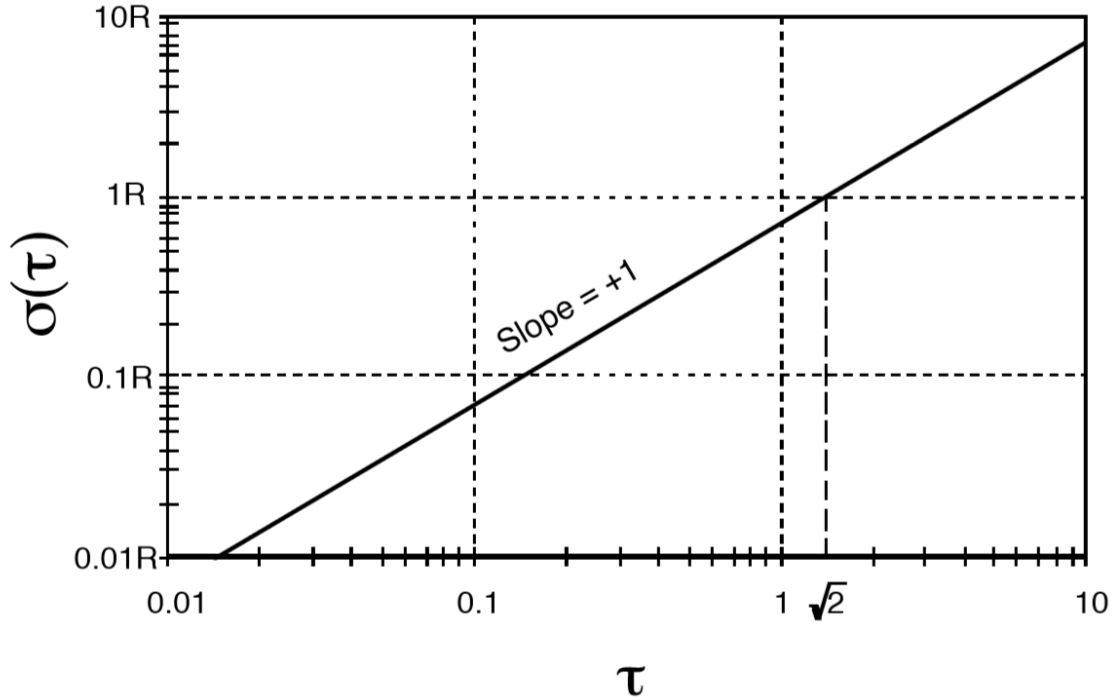
Figure A.10. *Rate ramp plot[18]*

It should be noted that there may be a flicker acceleration noise with $1/f^2$ that can lead to the same Allan variance τ dependence [18]

- **Sinusoidal Noise**

  This noise can be characterized in PSD as one or more distinct low frequencies and can be seen as low motion of the platform caused by periodic environmental changes.The PSD of a single and multiple frequencies are given in Equation (equation number) and (equation number). This later obtained by substituting the Equation(first) in $Equation$4.9 and performing the integration.

$$S_\Omega(f) = \frac{1}{2}\Omega_0^2 \left[\delta(f - f_0) + \delta(f + f_0)\right], \tag{A.8}$$

  where Ω_o is the amplitude, f_0, the fundamental frequency and $\delta(x)$ is the Dirac delta function.

$$\sigma^2(\tau) = \Omega_0^2 \left(\frac{\sin^2 \pi f_0 \tau}{\pi f_o \tau}\right)^2 \tag{A.9}$$

  The log-log plot of the Equation above is shown if figure A.11. As it seen in the figure, in order to identify and estimate these noises in IMU data, several peaks must be analyzed. It also be noted the the amplitudes of consecutive peaks fall off rapidly and may be masked by higher order peaks of other frequencies which make
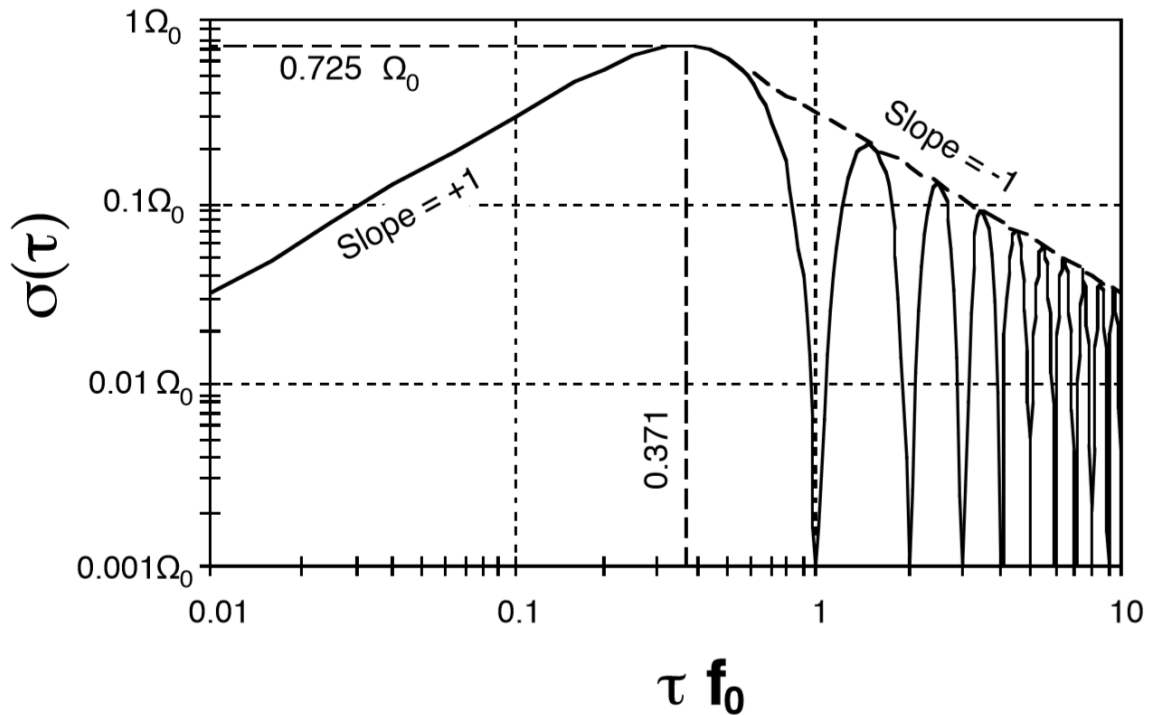
Figure A.11. *Sinusoidal noise plot[18]*

the observation more complex[18].

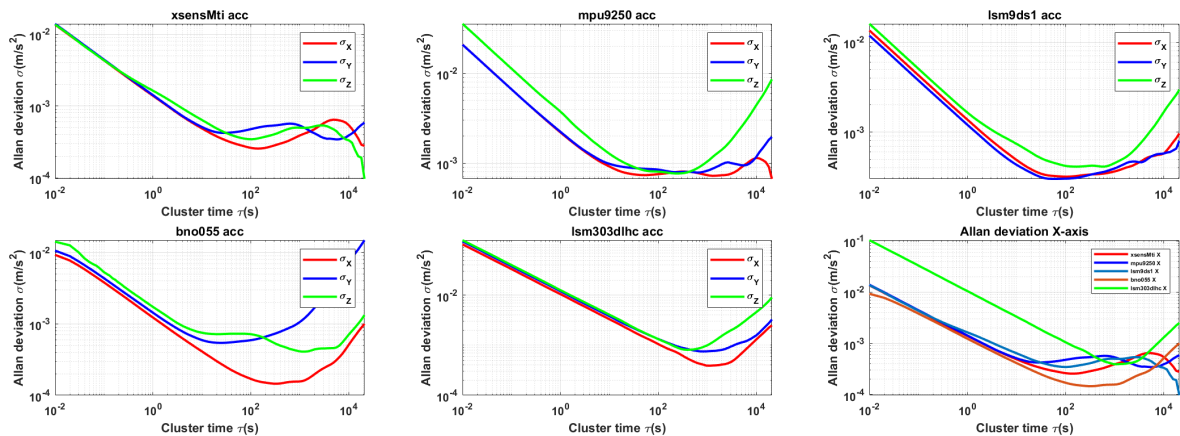## A.3 Allan deviation plot of all sensors

Figure A.12. *Allan variance plot of all sensors*

A.4 Code for Allan Variance calculation

The code for the overlapped Allan Variance calculation

```
function [tau, adev, accur] = overlapped_allan_variance(data,Fs)
    %Function to compute the overlapped allan variance of given dataset
    % based on Matlab code
    % Inputs
```

```matlab
% data:  accelerometer or gyroscocope data
% fs:  Sampling frequency
% outputs:
% Overlapped cluster time, allan deviation and accuracy
t0 = 1/Fs;
theta = cumsum(data, 1)*t0;
maxNumM = 100;
L = size(theta, 1);
maxM = 2.^floor(log2(L/2));
m = logspace(log10(1), log10(maxM), maxNumM).';

m = unique(ceil(m)); % m must be an integer.Remove duplicates
tau = m*t0;

avar = zeros(numel(m), 1);
for i = 1:numel(m)
    mi = m(i);
    avar(i,:) = sum( ...
        (theta(1+2*mi:L) - 2*theta(1+mi:L-mi) + ...
        theta(1:L-2*mi)).^2, 1);
end
avar = avar ./ (2*tau.^2 .* (L - 2*m));
adev = sqrt(avar);
accur = 1/sqrt(2*(L/maxNumM)-1);% accuracy interval
end
```

A.5 m12 velocity and displacement time history
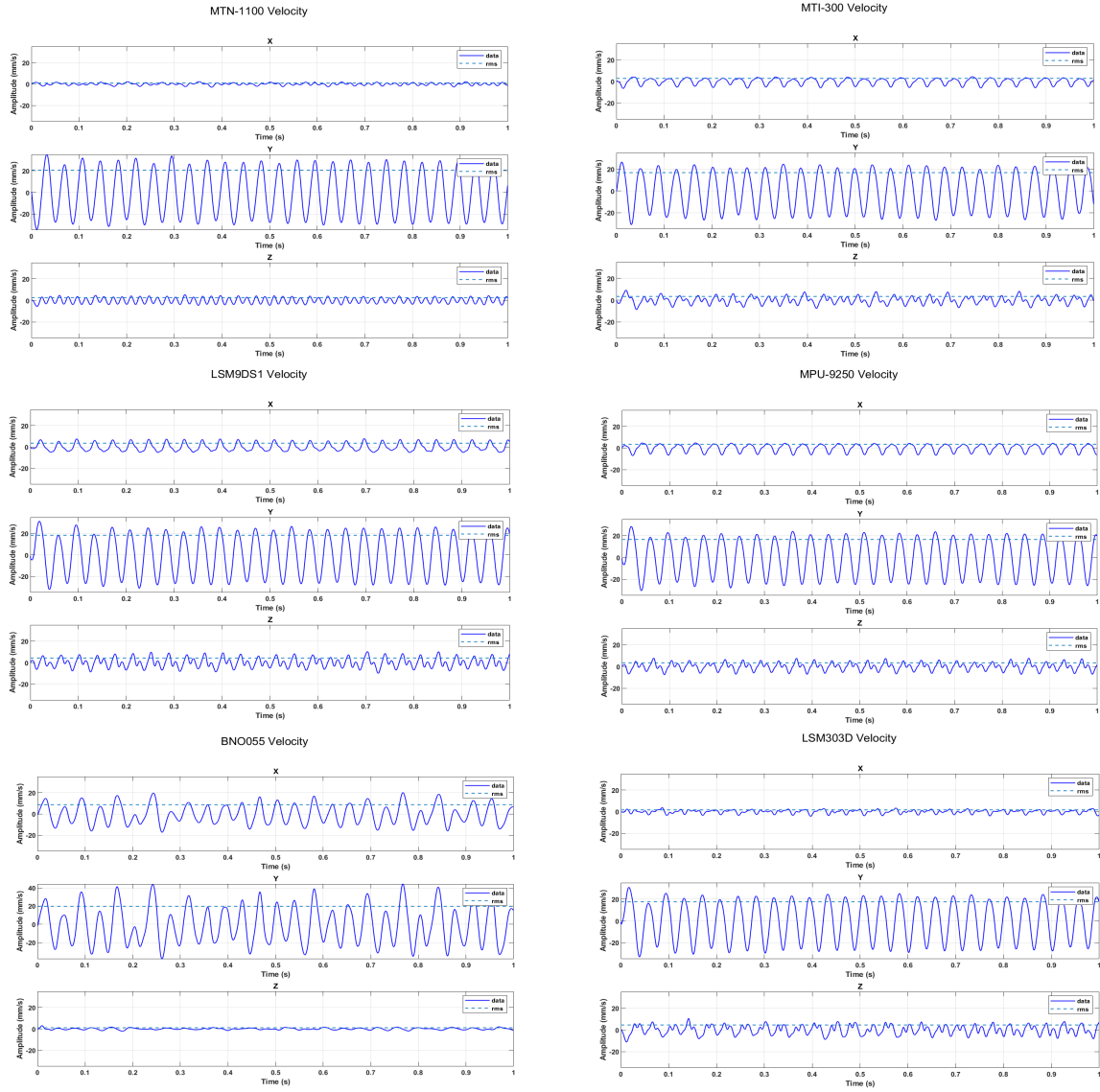
A.5.1 m12 velocity time history

Figure A.13. *A 2 second time history of velocity data for all sensors after moving average filter*

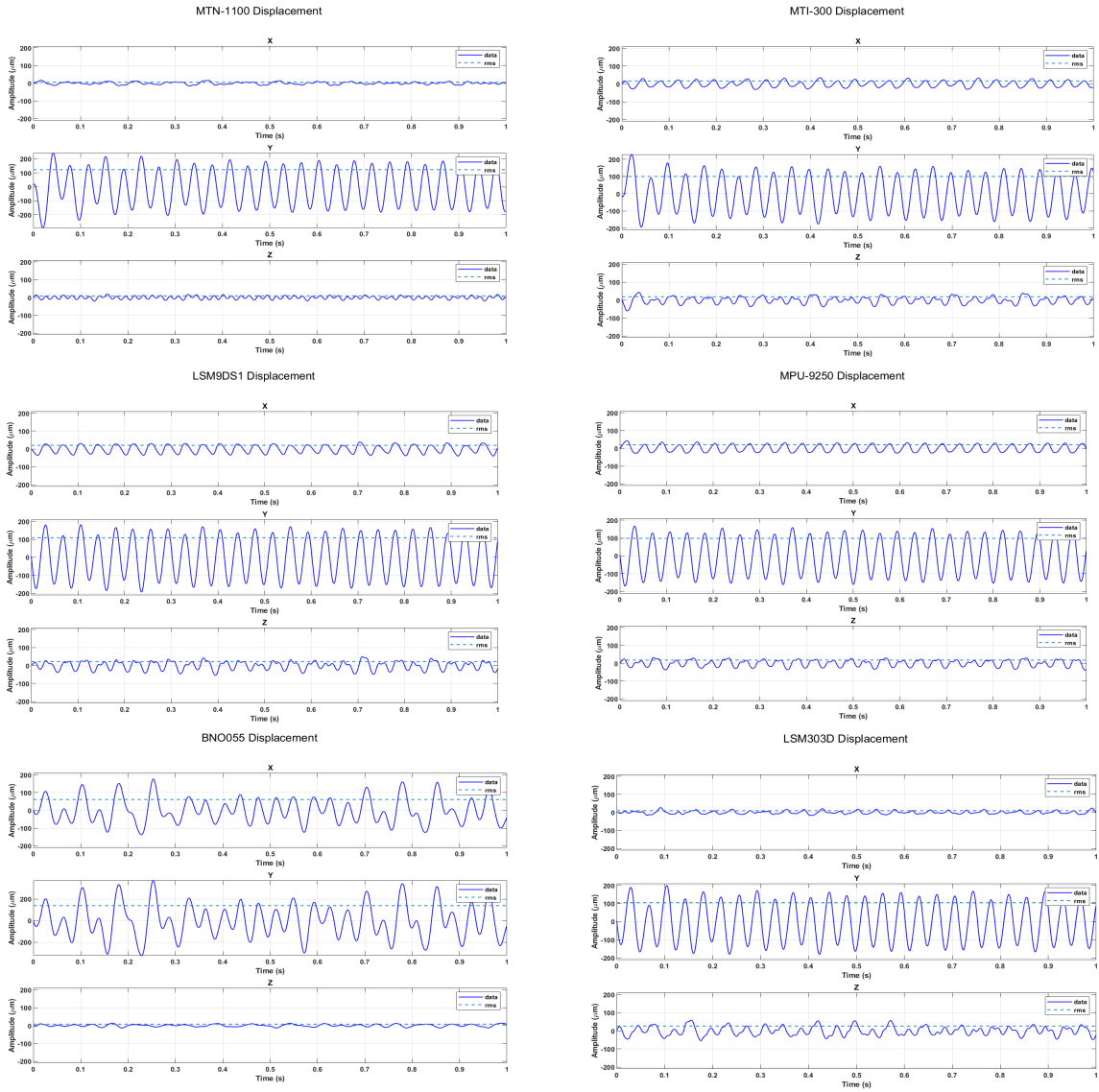# A.5.2 m12 displacement time history

Figure A.14. *A 2 second time history of displacement data for all sensors after moving average filter*