Tampere University

Hasan Farooq

MOTION DATA ANALYSIS USING ACCELEROMETER

For aircraft mobility studies and shipping industry

Master of Science Thesis
Information Technology and Communication Sciences
Examiners: Prof. Elena-Simona Lohan
Jani Käppi
November 2022

ABSTRACT

Hasan Farooq: Motion Data Analysis using Accelerometer
Master of Science Thesis
Tampere University
Information Technology, Robotics and Artificial Intelligence
November 2022

There are millions of shipments in transit every day. Some of the shipments are important enough that they are needed to be tracked and monitored during their journey. There are various methods to track the shipments, e.g., by scanning the bar-codes, tracking the courier, tracking the vehicle, and tracking the actual shipment. This thesis concerns an asset tracking device, which is attached to the shipment at the origin, then the shipment is tracked using this asset tracking device during transit, and finally, the device is detached from the shipment once it reaches its destination. The device is an IoT-based hardware equipped with multiple sensors (accelerometer, thermometer, barometer, and hygrometer), communication modules, a micro-controller, and a battery.

It is sometimes required to attach the asset tracking device in a powered-on state to the shipment long before it leaves the origin warehouse. The device needs to consume as little power as possible in this idle state to have enough battery power left to track the journey when the shipment actually leaves the warehouse. The proposed solution in this thesis uses accelerometer data to detect any motion. This information is used to keep the device in a low-power mode as long as there is no motion. The device starts to operate in the normal mode once it detects movement; hence it leaves the origin warehouse with more battery capacity, which enables it to track the journey better.

Secondly, shipments can be mishandled during transit and damaged upon arrival. This thesis proposes an algorithm to detect and report undesirable shocks that can potentially break the asset. Corrective actions can be taken beforehand if the mishandling is detected as soon as it occurs, reducing the time and the associated monetary costs incurred upon arrival of a broken shipment.

Finally, to enable the use of air cargo, the asset tracking device needs to have an autonomous flight mode in which the cellular modem must be turned off to comply with aviation regulations. A method is proposed to automatically detect the plane take-off using acceleration and air pressure data which triggers the flight mode autonomously in the asset tracking device.

Keywords: motion detection, accelerometers, asset tracking, shipment visibility, vector arithmetic, quaternions, real-time signal processing

PREFACE

Words can never be enough to acknowledge the support and motivation I have received from my family, especially my mother during my studies. Her support, goodwill, and prayers always motivated me to continue working hard and making me the person I am today. I am forever indebted to my family for their continuous love and support.

This thesis was commissioned by HERE Europe B.V. sivuliike suomessa, and I am grateful to my managers at the company, Matias Mikkola, Tuula Jakovuori, and Olli Seppälä who always provided me with a relaxed and productive work environment that enabled me to focus on my studies. It is a pleasure to extend my gratitude to my thesis supervisors, Jani Käppi and Elena-Simona Lohan for their guidance in overcoming technical challenges and academic writing.

Tampere, 29th November 2022

Hasan Farooq

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
BT	Bluetooth
EASA	European Union Aviation Safety Agency
EKF	Extended Kalman Filter
ETA	Estimated Time of Arrival
FIFO	First In First Out
GNSS	Global Navigation Satellite System
HAR	Human Activity Recognition
HDT	Heavy Duty Truck
IATA	International Air Transport Association
ICT	Information and Communications Technology
IoT	Internet of Things
LDT	Light Duty Truck
LSB	Least Significant Bit
MEMS	Micro-Electro-Mechanical Systems
NASA	National Aeronautics and Space Administration
NFC	Near Field Communication
PCB	Printed Circuit Board
REST	Representational State Transfer
RMS	Root Mean Squared
RTCA	Radio Technical Commission for Aeronautics
SIM	Subscriber Identity Module
WiFi	Wireless Fidelity

1. INTRODUCTION

1.1 Background

Shipping is the process of moving goods from the origin to the destination. Without modern shipping methods, the dynamics of the contemporary world cannot be sustained. This fact remains evident in [1] which describes the shipping industry's evolution, modern shape, and future. One of the articles [2] mentions the 2021 shipping crisis due to COVID-19, which had a vast impact on many industries, including agriculture, clothing, manufacturing, e-commerce, and many others. It signifies the importance of the shipping industry in the modern world.

The shipments can be routed through sea, land, air, or any combination of these. Each method of shipping has its own usage, pros, and cons. Airplanes are used when a shipment is expected to travel quickly over long distances. Otherwise, a cheap alternative is to use ships to take packages over long distances if time is not of the essence. Railways and Heavy Duty Trucks (HDT) are used for long-distance road transport, and Light Duty Trucks (LDT) are used for short-distance road transport, and end-customer delivery [3].

Any specific package is usually transported over more than one kind of vehicle. For example, when a package leaves the manufacturing factory, it is shipped to a warehouse by a road vehicle. If it is intended to be sent to another country, then it can be transported to a port where it is placed on a ship. When the ship anchors at the destination port, the package is then loaded onto a road vehicle, e.g., a truck or a cargo train, and finally to its destination. Airplanes can also replace ships to transport goods quickly where it is required.

Logistics is the process of acquisition, storage, transportation, and readiness of goods to be consumed at their final destination. In other words, the management of the flow of things from the point of origin to the point of final destination can be called logistics. Logistics involves the control of the supply chain and delivery of goods to the consumers in the current business world and especially in the manufacturing industry.

There are several questions that logistics tries to answer that include and are not limited to

- What is the significance of required goods?

- How can the required goods be acquired?
- When are the required goods needed to perform a particular operation?
- What is to be done with the produced goods? Where are produced goods stored?
- When are those shipped?
- Who are those shipped to?
- What is the required deadline for delivering the goods?
- How can it all be managed?

Logistics is deeply embedded into the shipping industry. Logistics provides the tools to manage the shipments and provides visibility over them. The global expansion of the shipping industry has led to more and more complex logistics operations. Logistics operations are responsible for the optimization of the ever-increasing physical process of shipping goods. It may sound simple when a truck is taking goods across 2000 Km, whereas optimizing the route of the truck can be a real challenge in practice. The optimization can, for example, include but is not limited to choosing the best route for 2000 Km by taking into consideration all the pick up, drop off, pit stop, layover, vehicle checkup, and other locations.

The use of Information and Communications Technology (ICT) in the existing processes can facilitate the processes themselves and improve their efficiency. Let us imagine that each parcel, container, ship, truck, and everything related to the shipping industry is connected to the internet, and real-time information of anything at any time is known. Such a system would answer multitudes of logistics problems and could solve simple to relatively complex issues. The questions that this system can answer may include

- Where is the package that I ordered online?
- Which truck is taking the packages to which destination?
- How long would the Estimated Time of Arrival (ETA) be for any particular package?

The questions that such an imaginary system can answer could be endless. This made-up scenario is possible to be achieved in its entirety by integrating the tools and techniques offered by the Internet of Things (IoT). However, integrating IoT solutions to such problems remains a challenge. The rest of the thesis will discuss one such solution.

1.2 Internet of Things

The term internet of things refers to, according to [4], the use of internet to enable the communication between electronic devices and sensors in order to facilitate our lives. This paradigm extends the internet connectivity to miniature and small things, henceforth making them connected and enabling them to have smart or intelligent features. The IoT-

connected things can transmit data about themselves over the internet to be analyzed and used. Moreover, commands to these miniature and small devices can be executed over the internet. For example, a question such as "Hello, my sensor! What's the humidity of my bedroom?" can be responded to as, "The bedroom humidity is 34.7%"

Making things connected with the internet alone is not enough as it is of very limited or no use. To be able to make use of the internet connectivity, additional features must be implemented to use the connectivity. The intended features may be implemented on devices and the internet or, as it is typically called, client and server, respectively. As far as the example in the above paragraph is concerned, this can include the ability to send the voice commands from the server and then send the commands over to the client device via the internet, and finally, the device should be able to understand and react to the commands. These kinds of scenarios give rise to a typical IoT architecture that is shown in figure 1.1. In a typical IoT architecture, the devices are connected to the internet via a gateway to enable devices to perform intended actions. Actuators and sensors are needed to be attached to the devices to perform intended operations. In many applications, along with the real-time data, users would also be interested in the historical data, so a database is also typically connected to the system. There are many other purposes of the database as well, but we will not go into their details. With the database connected, there needs to be a service that collects the data from the devices, stores it in the database and provides the required data to a user. Similarly, there needs to be a service that records the actions the user wishes to perform on the device and then relays the intended actions to the device via the internet. Such services are also part of a typical IoT system.

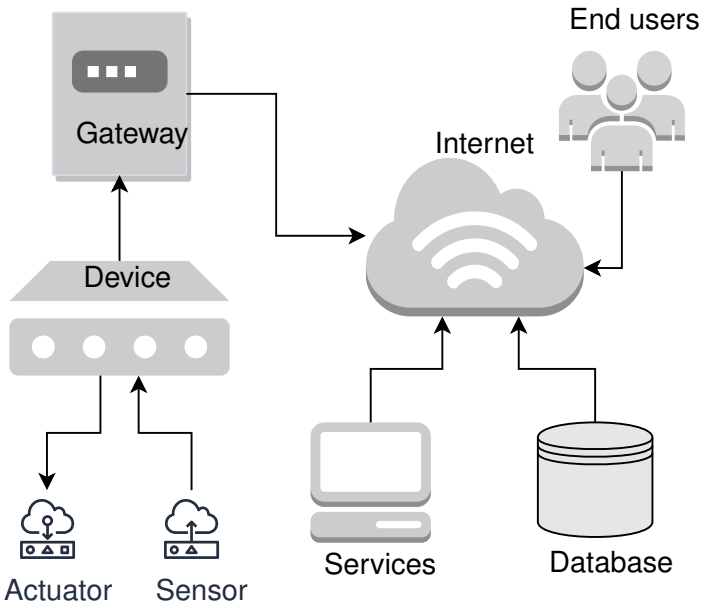


Figure 1.1. *Typical IoT architecture.*

The applications of IoT can be versatile and range from remotely controlled appliances

in one's home to tracking vehicles and moving objects on the other side of the world. However, before the advent of the internet in the field of technology, the paradigm of computer hacking was unknown to most people. Internet enabled hackers to hack into connected computers more easily. While it extends the internet to small devices, IoT also introduces security risks to the connected devices. In fact, securing the IoT devices is a significant blockade in the development and deployment of large-scale IoT. Highlighting the severity of security concerns, someone may want to know themselves when the trash bin of their room needs to be emptied, but no one wants thieves and robbers to have such information, henceforth announcing the precise time when they will be out of their home to get rid of the trash.

As the famous proverb goes, "necessity is the mother of invention" where there are challenges in IoT development, there are ever-increasing solutions to make the IoT systems and devices more secure, scalable, and better. Today, our world offers unprecedented technical development power, and multiple businesses and research groups are trying to find solutions to the same problems. This poses another problem that even though when solutions are public, they are not standardized. Subsequently, there is an incompatibility in communication between IoT devices from different manufacturers [5].

IoT is a relevant technological field that can facilitate the industry of shipment and logistics. As discussed previously, if we can have connected devices, it can simplify the logistics operations to an unparalleled extent by giving visibility to the shipments. IoT can enable fetching information about shipped assets and other cargo details. Moreover, it can also facilitate or even perform small tasks that are required during the shipment. This thesis will discuss the usage of IoT in shipment and logistics.

1.3 Thesis objectives and research questions

HERE supply chain [6] is an end-to-end solution for asset tracking [7] and routing [8]. It includes a cloud platform, back-end services, front-end applications, and hardware tracking devices. The overall solution also takes care of the connectivity of hardware tracking devices with the cloud platform. The back-end services host Representational State Transfer (REST) Application Programming Interfaces (API) on the cloud platform, which is used by front-end applications and hardware tracking devices. A brief architectural overview of the HERE tracking solution is presented in figure 1.2.

Hardware tracking devices ingest data into the cloud while the devices are being transported using different transportation methods which can be airplanes, trucks, trains or any other means of transport. The ingested data is then processed and stored by the back-end services. The user accesses the data available on the cloud using the front-end application. The data includes but is not limited to location, temperature, humidity, pressure, and other system information.

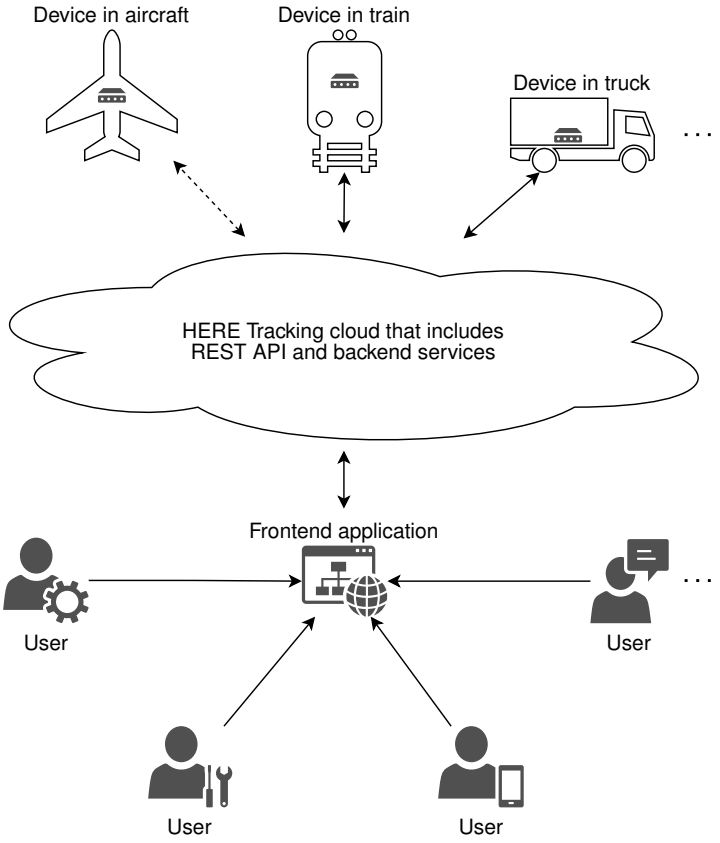


Figure 1.2. *Brief technical overview of the HERE tracking solution.*

Some of the hardware tracking devices are also equipped with an accelerometer that can detect acceleration experienced by the tracking device. This means that the sensor enables prospective motion detection and processing in the tracking device. The objective of this thesis is to add intelligence to the tracking devices by motion detection. The following subsections detail the types of motions, along with research questions, whose detection would add value to the hardware tracking devices.

1.3.1 Movement detection

During the shipment process of an asset from the point of origin to the point of destination, there comes a time when it is not being moved. Sometimes this so-called stationary period can even stretch out to days. This information can be important for the tracking device in terms of implementing better power consumption techniques. For example, if the tracking device is stationary, it does not need to report the duplicate location repeatedly for days or as long as it is stationary. Also, when the tracking device is stationary, it does not even need to fetch the location information from one of its sensors which also consumes power. The tracking device only needs to be fully active when it is not stationary. **One objective of this thesis is to assess if movement can be detected to optimize the power consumption of the hardware tracking device based on this information**. It needs to be assessed whether the pre-installed sensor/s can be used to differentiate

between stationary and moving state of the tracker. If such a differentiation is possible then an algorithm will be proposed which uses sensor data to report the movement within 1 minute of the movement after processing the sampled data.

1.3.2 Shock detection

The packages may be mishandled during the shipment process. It can happen due to human carelessness, or it can be an accident. For certain types of assets, it is a crucial piece of information to know that it has been dropped. Another critical piece of information would be to know the intensity of the shock that asset received when it was dropped. Once the shock intensity is known, and with some historical data correlation, the probability of whether the asset will be shipped broken to the destination can be predicted. **Another objective of this thesis is to study if shocks and their intensities can be detected and reported to the cloud**. This is needed so that it can be known in advance that an asset might be broken when it reaches its destination. With this information, mitigation actions can be started, if needed, as soon as shock information is received. A system should be proposed to fetch the shock intensity in m/s^2 from the installed accelerometer upon experiencing a shock. It also needs to be tested if such a system can report the shocks within the acceptable range of accuracy subjected to the real world testing.

1.3.3 Airplane takeoff detection

The hardware tracking devices need to communicate with a cloud on a need basis for an effective solution. This requires the tracking devices also to have a component responsible for communication that is typically a cellular modem. A cellular modem can provide global connectivity, but one of its drawbacks is that it cannot be used in an aircraft because cellular modems have been reported to be responsible for anomalous events during flights [9]. The use of cellular modems in airborne equipment is restricted with various degrees of strictness across the world, and it can also be completely prohibited [10] in some countries.

International Air Transport Association (IATA) is the regulatory authority for air transport. It states guidelines for the devices that contain cellular modems which are supposed to be transported via aircraft unattended. According to the guidelines of IATA stated in [11], it is required that such devices must employ two independent methods to detect the aircraft takeoff and turn off the RF emitting components available in the device. Moreover, such a device must be certified by DO-160 [12] to ensure that radio emissions are contained within an acceptable range.

Yet another one of the **objectives of this thesis is to assess the possibility of implementing a method to detect airplane takeoff using installed accelerometer**. The

sensor data needs to be collected during aircraft takeoff as a first step. Then, the acceleration patterns during takeoff should be analyzed, and afterwards it should be studied, in the light of sensor behavior during takeoff, whether a method to detect those patterns can be implemented. Anecdotal evidence suggests that aircrafts experience a sustained increase in acceleration while taking off. This thesis will try to answer whether such sustained increase in acceleration can be detected using accelerometer installed in the asset tracking device. Moreover, a method to calculate the liftoff angle of aircraft using the accelerometer data will be studied. It will be tested whether such a system, which detects the sustained acceleration increase and change in pitch angle, is able to distinguish the aircraft takeoff from other activities in a robust manner. IATA guidelines [11] require two independent methods of aircraft takeoff detection, so a similar approach is to be undertaken for the barometer-based takeoff detector. The barometer-based takeoff detection solution will be discussed for the sake of completeness, but it is out of the scope of this thesis. Finally, the proposed algorithms should be implemented and tested. Some of the challenging tasks could be data collection, pattern identification, pattern detection, and implementing the algorithms in compliance with aviation safety regulations.

1.4 Author's contributions

The author's contributions started with this thesis's ideation after evaluating the business needs. The author was also involved in researching and developing the methods to address the objectives listed in 1.3.1 - 1.3.3. The methods include

- Designing and implementing the shock detection and reporting the adequate shock values to the HERE tracking cloud
- Designing the motion detection solution to be able to optimize power use efficiently
- Designing, implementing, simulating, and testing the techniques to detect the aircraft takeoff detection

The author has worked independently on the integration of sensors to the system, data collection of accelerometer and barometer during flights, research and development of accelerometer takeoff detector algorithm, and the commercial testing of the overall system. Furthermore, the author is also responsible for analyzing the results and identifying the improvements and bugs in the implementation.

1.5 System overview and limitations

The target device for this thesis is an IoT based asset tracking device that operates on a battery. The device sends the positioning data, scans, and sensor readings to the HERE asset tracking platform. The transmitted data consists of Global Navigation Satellite System (GNSS), Wireless Fidelity (WiFi), Bluetooth, and 2G/4G cellular network scans. The

device is shown in figure 1.3.

The peripherals of the device include:

- Cellular modem
- WiFi/Bluetooth
- Data storage
- Near Field Communication (NFC) tag
- Sensors
  - Accelerometer
  - Thermometer
  - Hygrometer
  - Barometer
- 10,000 mAh battery

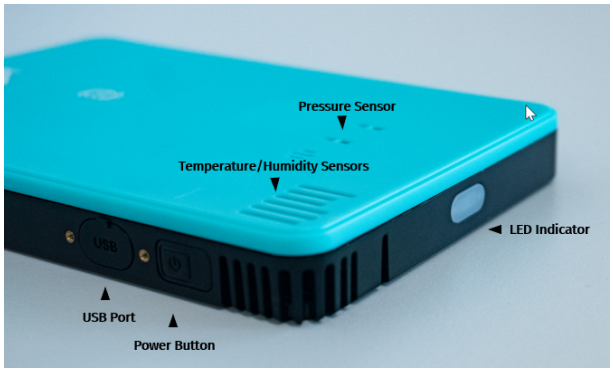


Figure 1.3. *Product appearance of the asset tracking device.*

The device is IP65 dust, and water resistance certified and comes with a pre-installed Subscriber Identity Module (SIM) card in its modem for global cellular connectivity. This device is supposed to be attached to the asset at the beginning of the shipping process and shipped with the asset until the asset arrives at its destination, where it is detached from the asset.

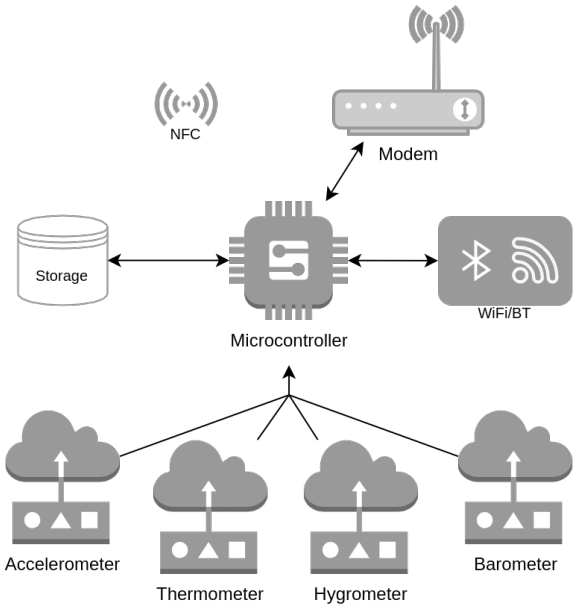


Figure 1.4. *Block diagram of hardware components.*

The thesis objectives must be met within the resources available in the device. The hardware resources available in the device are shown in the figure 1.4 block diagram. It is to be noted here that the current tracking device functionality must not be altered with

the new implementation. The detection algorithms described in section 1.3 can use the available sensors. The relevant sensors may include but are not limited to accelerometer and barometer. New features should be designed in a way that those can be implemented on an embedded hardware device of limited compute and power.

2. LITERATURE REVIEW

2.1 Principles of accelerometers

An accelerometer is a tool designed to detect and measure acceleration from its own reference frame. The internal reference frame of the accelerometers is different from any external reference frame, e.g., reference frame w.r.t earth. Accelerometers are widely used in technological and scientific applications. There are multiple types of MEMS accelerometers based on their working principle that include piezoresistive, piezoelectric, capacitive, tunnel, e.t.c. [13].

MEMS-based accelerometers are prevalent in various products, from mobile phones and tablets to aircraft and missiles. These accelerometers can be multi- or single-axis, and depending on the orientation they are installed in the system, they can detect the acceleration vector that includes magnitude as well as the direction of acceleration. Linear elements of vibrations due to the engine or any rotatory device can also be measured using MEMS accelerometers. Our discussion in this thesis will be limited to MEMS accelerometers.

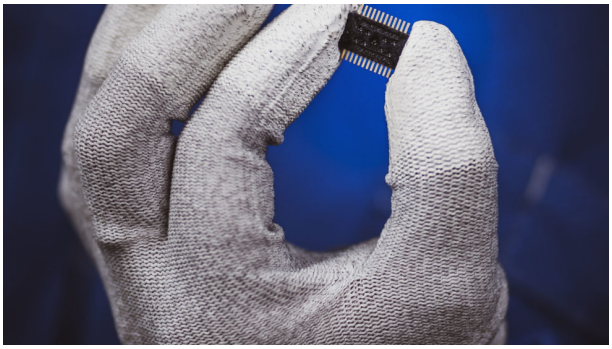


Figure 2.1. *A photograph showing a MEMS sensor [14].*

2.1.1 Parameters of accelerometer

Acceleration is an analog quantity that, in typical MEMS accelerometers, is measured using the differential capacitance principle by a specialized designed structure. The difference in capacitance is then converted into an electrical voltage which is subsequently sampled by Analog to Digital Converter (ADC) [15] and represented as a number. The relation between digital acceleration ($\mathbf{a}_{digital}$) and analog acceleration ($\mathbf{a}_{analog}$) is given by

$$\mathbf{a}_{digital} = \frac{\mathbf{a}_{analog}}{range * 2^{resolution} - 1} \tag{2.1}$$

Where:

- **Resolution:**
  It corresponds to the number of digital bits that are used to provide the acceleration values. Resolution can also be considered as a single step size when increasing or decreasing acceleration. The step size is typically measured in LSB/g in digital accelerometers, which describe how much change in g-force will be represented by 1 Least Significant Bit (LSB). A higher resolution means more accurate acceleration values or, in other words, a smaller LSB/g. The resolution of a sensor is one of the contributing factors to the sensitivity of the accelerometer. The analog counterpart of LSB/g is mV/g. If there are 2 bits to represent an acceleration range of 0.0 g to 1.0 g then the possible values that can be represented are given in table 2.1. In contrast to table 2.1, if there are 3 bits to represent same range of acceleration

Digital bit value	Corresponding analog value
0b00	0.0
0b01	0.33
0b10	0.66
0b11	1.0

Table 2.1. *2 bit representation of values between 0.0 to 1.0.*

  from 0.0g to 1.0g then the possible values to represent are given in table 2.2. It is clear from the comparison of the two tables that a more number of bits result in a smaller LSB/g which results in better accuracy.

Digital bit value	Corresponding analog value
0b000	0.0
0b001	0.142857143
0b010	0.285714286
0b011	0.428571429
0b100	0.571428571
0b101	0.714285714
0b110	0.857142857
0b111	1.0

Table 2.2. *3 bit representation of values between 0.0 to 1.0.*

- **Range:**
  It corresponds to the maximum and the minimum values that an accelerometer can measure. When the range of an accelerometer is small with a low resolution, then the step size, i.e., LSB/g in accelerometer values, would be little. However,

if the range of an accelerometer is wide with a low resolution, then the step size, i.e., LSB/g in accelerometer values, would be relatively bigger. For example, if the range is 16g and the resolution is 16 bits, then LSB/g would be equal to 0.0024414 g, and if the range is 16g with 4bit resolution, then LSB/g would be equal to 1 g.

The range and resolution of an accelerometer together define how accurately an accelerometer is able to provide acceleration values to a system. Another important parameter of an accelerometer is sampling frequency which is measured in hertz (Hz). Sampling is the process of time-series reduction of a continuous signal into a discrete signal. The value of a signal at a specific time is called a sample. Any digital system has a limited sampling frequency (f_s), and the limit is typically chosen as defined by the Nyquist limit i.e.

$$f_s = 2 * f_{max} \tag{2.2}$$

which states that sampling frequency (f_s) should be at least twice the maximum frequency (f_{max}) that is supposed to be obtained from a signal. There might be other relevant parameters to consider while choosing an accelerometer, e.g., power consumption, operating environment, size, etc.

The sensor values change over the lifetime of the sensor. Soldering an accelerometer on a Printed Circuit Board (PCB) usually changes the offsets described in section 2.1.5. It is an important consideration to make while choosing an accelerometer how the aging of the sensor will affect the application in which the sensor is used.

2.1.2 Characteristics of Bosch BMA253

The asset tracker is equipped with Bosch BMA253 which is a tri-axial digital accelerometer. This sensor is able to detect acceleration in the range of -16g to 16g with a resolution of 12 bits. It measures the acceleration from 3 mutually perpendicular axes via a MEMS-based acceleration sensing structure that works according to the differential capacitance principle. It can provide the data with a sampling frequency of 1KHz. Both the range and the sampling frequency are configurable and can be lowered to increase accuracy or decrease power consumption.

One sample of this accelerometer will provide 36 bits of data which is called a frame. These 36 bits include 16-bit data from each x-, y- and z-axis. The sensor can be connected with a micro-controller using SPI or I2C protocol. It incorporates a FIFO buffer that can store 32 samples. When the FIFO is enabled, 32 frames can be read in a burst once the FIFO buffer is completely filled up. It results in a smaller number of transactions but with the cost of a delayed response from the accelerometer as the oldest samples in the FIFO buffer will be read after 32 ms if 1 KHz sampling frequency is being used. The sensor also has an interrupt controller that can generate interrupts for

- **New data:**
  This interrupt is generated after storing the new data sample to the data registers. It is cleared automatically when user starts reading the data.
- **Slope detection:**
  This interrupt uses the difference between two successive acceleration data samples and compares it with a pre-configured threshold. The interrupt is generated when the difference in two successive values is more than the pre-configured threshold. It is cleared when difference gets below than the threshold. This interrupt is configured for each axis of acceleration separately.
- **Tap sensing:**
  Tapping the accelerometer results in positive and negative slopes, or vice versa, in a row. This principle is utilized in the tap sensing interrupt which can be configured by providing slope threshold which can be a measure of how strong tap needs to be detected. This interrupt can also be generated after a double tap event for which a quiet time between the taps is also needed to be configured.
- **Orientation recognition:**
  We will discuss in section 2.1.6 that how the gravitational acceleration **g** is always reported by an acceleration. This interrupt generation mechanism uses the **g** vector to detect change in orientation and report the change by generating an interrupt.
- **Flat detection:**
  This is a special orientation recognition interrupt when the device is not in a horizontally flat orientation. The flat angle can is configurable and can range from 0 to 44.8 degrees.
- **Low-g/high-g detection:**
  The low-g and high-g interrupts are generated when a preconfigured number of samples get below or above preconfigured acceleration values respectively. The interrupt is cleared as soon as the interrupt condition goes false.
- **No-motion detection:**
  The no-motion detection is a kind of slope detection where the difference between consecutive samples for N samples must be smaller than the configured difference or slope threshold. Therefore, it is similar to slope detection but works with a different set of parameters.

2.1.3 Nature of accelerometer data

The tri-axial digital accelerometers, illustrated in figure 2.2, measure the acceleration on three axes and give out the digital values. The values are typically read from the sensor registers in the form of bits. Bosch BMA 253 has 12 bits for each axis that collectively

give tri-axial acceleration values. All of the axis values together give the magnitude and direction of the acceleration vector. The values are updated in the sensor data registers at the sampling frequency. The interrupt generator processes are the sampled values and generate corresponding interrupts if necessary. Acceleration values can be read from the acceleration data registers at the frequency that is required by the application, and it is not necessarily equal to the sampling frequency.

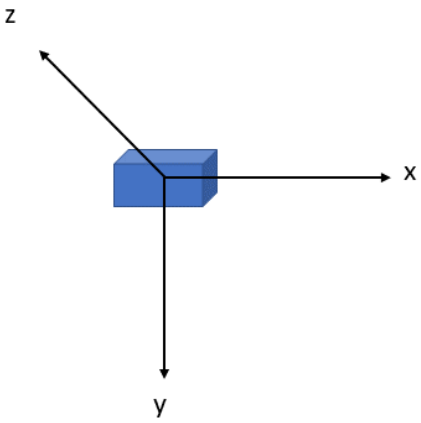


Figure 2.2. *Triaxial accelerometer with orthogonal axes.*

2.1.4 Noise in accelerometer data

Noise is an unwanted part of a signal that does not represent the real process. It is also present in the signals generated by the accelerometers. Noise can be caused by electrical components of accelerometers, e.g., amplifiers, or mechanical components, including the sensing structure of the acceleration. However, irrelevant sources of acceleration in the data can also be considered as noise. For example, if an accelerometer is supposed to detect only a shock to a system that is being transported on an airplane, then the engine vibration of the airplane might be irrelevant for the analysis and may be considered as noise in the acceleration data.

The accelerometers typically come with a built-in low-pass filter that is intended to eliminate the aliasing effect when data is downsampled and read in the applications. This low pass filter is also called an anti-aliasing filter and can be configured to avoid the aliasing of the data read from the sensor. For example, BMA253 can sample the data at 1 KHz which is provided to the interrupt controller and data registers. The data may be read at a much smaller frequency from the data registers, e.g., 10 Hz. The lack of an anti-aliasing filter causes unwanted signal aliasing in the decimated signal; therefore, it is recommended to set up an anti-aliasing filter, e.g., a simple low pass filter, before decimating a higher frequency signal.

2.1.5 Stationary offset in accelerometer data

The accelerometers do not always report the acceleration values accurately. One major issue with the accelerometer data is the stationary offset in its values. Stationary offset describes the error in accelerometer value when it is stationary. Ideally, accelerometer provides 9.8 m/s^2 when it is stationary. Practically, it is rarely the case, and accelerometers provide any other value, e.g., 8.8 m/s^2 or 10.8 m/s^2 than the correct value while they are stationary. There are different methods to calibrate the accelerometer offset. The offset error does not typically change during the lifetime of the accelerometer, and accelerometers are typically calibrated only once during their lifetime.

A method to calibrate the tri-axial accelerometer with the help of a gyroscope is proposed in [16]. A special hardware to get the calibration data is designed and the cost function of a linear sensor model is proposed to be minimized to get the calibration parameters. A method to calibrate the accelerometer without any specialized or laboratory equipment is presented in [17]. It uses only the accelerometer data where a non-linear optimization problem is formulated with the assumption that the magnitude of tri-axial accelerometer data must be one when it is stationary. This non-linear problem is solved using statistical linearization method. [18] discusses a method to use accelerometer, gyroscope, and magnetometer together to calibrate offsets along with other sensor errors.

2.1.6 Gravity vector in accelerometer

The accelerometer measures any external acceleration that it experiences. In other words, the accelerometer measures the acceleration in its own frame of reference. It means that the accelerometer will also report acceleration when it is stationary due to gravity. However, in a free-fall motion accelerometer will report 0 g. It is similar to an accelerating vehicle that increases its speed from 0-100 km/h. When the vehicle accelerates, it experiences acceleration, but when it attains the final speed, the acceleration on it tends to be zero. In tri-axial accelerometers, the gravitational acceleration can be represented on one or more axes depending on the orientation of the accelerometer. The gravitational acceleration vector shifts from one axis to another when the orientation of the accelerometer is changed. Reporting of gravitational acceleration vector and its rotation when an accelerometer is rotated is illustrated in figure 2.3.

2.2 Principles of barometers

A barometer is a tool designed to measure the air pressure in the atmosphere. The air pressure is inversely proportional to the altitude, and it may be given as

$$P_h \propto \frac{1}{h} \tag{2.3}$$

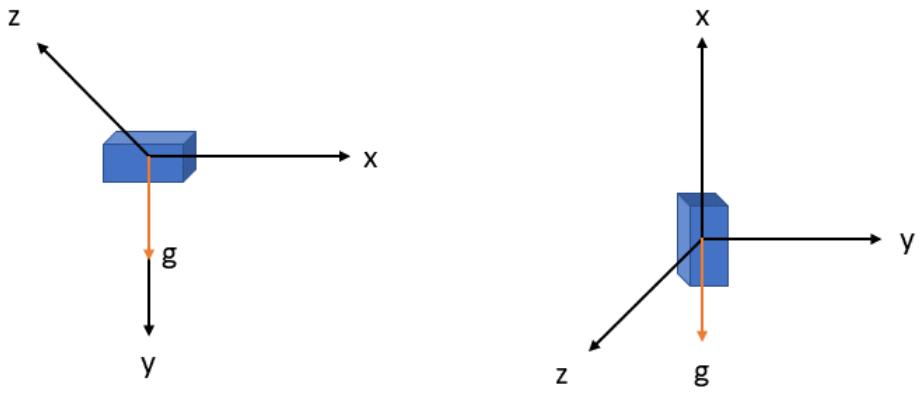


Figure 2.3. *The rotation of gravity vector when a tri-axial accelerometer is rotated clockwise on the z-axis.*

which states that air pressure P_h at height h are inversely proportional to each other. This thesis deals with a barometer that is based on the capacitive sensing principle. In this type of pressure sensing mechanism, one plate of the capacitor is fixed while the other is flexible and prone to deformation due to changes in air pressure. The deformation changes the capacitance between the two plates, which is subsequently measured and converted into an electrical signal by the sensor.

2.2.1 Parameters of barometer

Atmospheric air pressure is a natural phenomenon that inherently makes it an analog quantity. Like other signals, it can be represented digitally after the signal acquisition, sampling, and analog to digital conversion. There are some crucial parameters to consider while choosing a barometer for an application, and some of the parameters are mentioned below:

- Precision: It defines how close to the actual value the sensor can report.
- Sensitivity: How small steps can the sensor take while increasing or decreasing the output value.
- Range: Defines the minimum and maximum limits of the sensor output values.
- Power consumption: It defines the amount of power the sensor consumes. Usually, sensors can be operated in different power modes.
- Size: Chip size can be an important factor when considering the hardware design.

2.2.2 Characteristics of SPL06-007

This asset tracker is equipped with SPL06-007 which is a MEMS-based thermometer and barometer. This sensor works on the differential capacitance principle. It has a MEMS-

based pressure sensing membrane that is responsible for air pressure detection. The sensor offers a range of 300 hPa to 1100 hPa that roughly corresponds to the air pressure at altitudes of 9000 m and -500 m respectively. It has an absolute accuracy of $\pm$1 hPa within its operating range. It outputs the 24-bit values for pressure and temperature each and can be operated in different working modes that affect power consumption. The sensor also incorporates FIFO memory, which can store 32 latest pressure and temperature values. Using FIFO memory allows the host controller to stay in the power-saving mode for a longer period of time to have a lower power consumption.

2.2.3 Nature of barometer data

Atmospheric pressure is reported by the barometer, which is higher at lower altitudes than the air pressure at higher altitudes. The reason is that air has weight. Air at a higher altitude presses down the air beneath it, resulting in higher air pressure at lower altitudes. Air pressure normally ranges from 1000 hPa to 1013.25 hPa on the sea level. Barometers can be used for altitude calculation because the altitude can be estimated with air pressure data. There are various atmospheric pressure models for the estimation of altitude using air pressure. However, air pressure is not constant given an altitude and may vary according to atmospheric and weather conditions. For example, the highest ever air pressure was recorded to be 1084 hPa in Siberia on December 31st, 1968 [19] and the typical air pressure in Siberia is 1033.69 hPa [20]. For this reason, simpler atmospheric pressure models can have large errors.

2.2.4 Noise in barometer data

Barometers are not as prone to noise problems as accelerometers. However, it depends on the application they are used in. The barometer noise is typically measured as a root mean square (RMS) value. The smaller the RMS noise value is, the better sensor it makes. Modern barometers can provide atmospheric pressure readings with a minimal noise component. Bosch Sensortec is a world-leading MEMS sensor manufacturer, and the BMP390 barometer by Bosch has an RMS noise level of 0.02 Pa which corresponds to an accuracy of 1.67 cm when converted into altitude. The SPL06-007 has a corresponding RMS noise level of 0.6 Pa.

2.3 Related research

The authors have proposed and tested the methods to detect physical activities using multiple accelerometers worn on different parts of the body in [21]. All accelerometer were biaxial for which mean energy, frequency-domain entropy and correlation of acceleration data were computed. These features were tested with different classifiers and

the decision tree classifier gave the best accuracy of 82%. The recent advancements in deep learning has also influenced the research on the use of accelerometer data for detection physical activities. In [22], the authors review the recent work that exploited the deep learning based HAR methods. It is claimed in [23] that accelerometer is one of the most promising sensor to detect the physical activity. We will try to use the accelerometer to develop a light-weight algorithm which can merely distinguish between stationary and moving state.

IATA is an association of airlines from around the world. It regulates the airline industry, including safety standards, aviation training, price fixation, technical standards, and many other aspects of the industry. IATA also regulates the use of hardware equipment inside aircraft. For example, it provides guidelines for cellphone usage during flights. Data logging and shipment tracking devices or so-called battery-operated embedded systems can be needed to put on aircraft unattended for various kinds of purposes such as analyzing the bacteria and effect of air filtration in commercial airliner cabins as described in [24][25][26]. IATA regulates the usage of such devices aboard an aircraft [11]. As there are many competitors in the industry, proprietary products are commercially available for these and other kinds of purposes.

A flight detection method is proposed by [27] in which authors discuss a method of detecting airplane takeoff on a system that has a low-cost 3-axis accelerometer LIS302DL by ST. The authors describe the data collection techniques and the analysis of the collected data on a Geneva to London flight. They propose a method based on moving variance to detect flight activity. The results of calculating the normal moving variance vs exponential moving variance are compared in the article, and finally, a thresholding-based flight detection state machine is proposed. The variance of acceleration data can be increased in the conditions where there is vibration, random movements, walking, etc. Along with the inherent drawbacks of the variance-based thresholding approach, the sampling frequency of 50Hz is proposed, and that may consume a significant amount of power in systems with limited battery capacity. The proposed detection method is also discussed in [28].

In [29], the authors discuss a system containing an accelerometer, gyroscope, flight status detection system, radio, and a remote computer. They propose a flight status detection system with three modes, namely, disabled, normal, and airborne modes. An accelerometer and barometer aid the transition between normal and airborne modes. The accelerometer may be used to detect the shifting of the gravity vector from one axis to another during takeoff. The acceleration data might be filtered and then used for different purposes during the detection process. However, the cited reference is missing a complete description of the algorithms. A similar case can be observed in [30][31][32][33].

3. MOTION DETECTION

3.1 Shock detection

When an acceleration sharply increases or plummets, it illustrates a shock. Shocks beyond a respective threshold for everything are dangerous as high acceleration, or deceleration, for that matter, can cause severe damage. As it tries to measure the acceleration experienced on itself, the accelerometer can also be a helpful sensor to detect a shock. Ideally, when it experiences a shock, it reports a high acceleration value. The acceleration values by the accelerometer can be thresholded to get a shock event notification.

BMA253 that is being used in the asset tracking device is a tri-axial MEMS accelerometer, and it can give a complete acceleration vector by measuring the acceleration on its three axes. An accelerometer always reports the gravity vector except when the accelerometer is free falling. So, it is safe to assume that the accelerometer data will also have the gravity vector in normal circumstances.

An algorithm to configure and detect shocks on supported devices is proposed. The asset tracking device collects the telemetry, WiFi & Bluetooth scans, sensor data, and information about sensors when it boots up. It ingests the collected telemetry to the cloud at the configured sending interval. At this point, the cloud knows which sensors are available in the device in order for a user to use them. The user can then define an acceleration threshold, i.e., shock level from the web app specific to the asset tracking device. The web app calls the respective API and stores the sensor configuration in the device-specific storage on the cloud. When the device connects to the cloud next time, then it receives the new sensor configuration, including the acceleration threshold that the user configures. The algorithm is illustrated in figure 3.1.

The asset tracking device starts its shock detection mechanism upon receiving the acceleration threshold by the cloud. It starts sampling data from its accelerometer at 1 KHz which is required by the shock detection mechanism. The data is subsequently passed on to the acceleration data processor, which monitors the incoming acceleration data and compares it with the acceleration threshold that the user configured. The device generates an alarm that a shock event has occurred when enough samples above the user configured threshold are detected. As a consequence of this alarm, telemetry

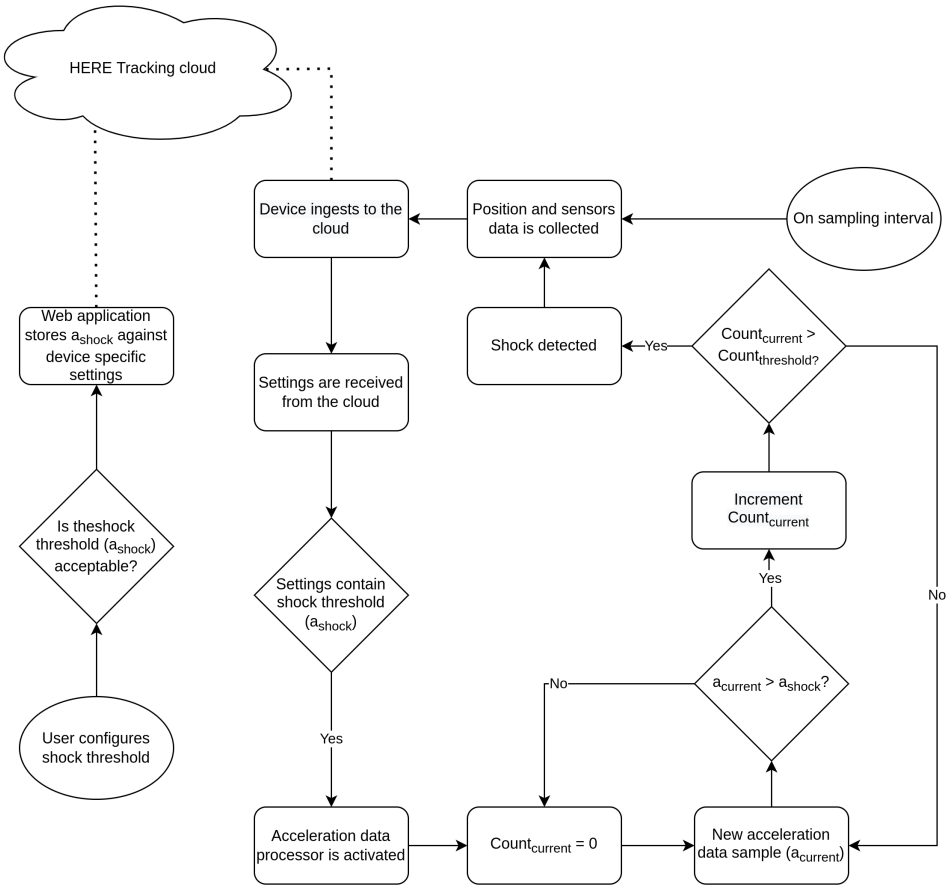


Figure 3.1. *Proposed shock detection architecture for the asset tracking device.*

is collected containing location information and other relevant sensor data and ingested into the cloud. The user is then notified that their asset experienced a shock above the configured acceleration threshold at the location that was reported by the tracking device.

There are known limitations of the current algorithm. A tri-axial accelerometer like BMA253 cannot predict how the gravity vector will be distributed among its three axes, especially in a use case like that of the asset tracking device. The asset tracking device is attached to the asset that is being shipped during its whole journey from origin to destination. The asset experiences normal acceleration patterns as it is turned upside down or moved abruptly in random order during its journey, which makes it impossible to predict the gravity vector in such situations.

In the current system, the acceleration data processor cannot threshold the acceleration magnitude, but it can only take input from the three axes of the accelerometer individually and threshold the values on each axis. The shock vector can either be manifested on a single axis or all three axes. In the former case, the acceleration data processor can correctly threshold the shock value because it is represented on one axis only. However, in the latter case, the acceleration data processor may not be able to generate an alarm because none of the axis values may be over configured threshold individually, but they

collectively, as magnitude, go over the configured threshold. This is a known limitation of the current system.

Also, gravity can enforce the shock vector or be irrelevant to the shock event. The gravity vector enforces the shock vector when the shock vector is parallel with the gravity vector. Furthermore, it will be less relevant when the parallel component of the shock vector to the gravity vector is smaller. The gravity vector will be completely irrelevant when the shock vector is perpendicular to the gravity vector.

The implementation of the acceleration data processor cannot be changed due to power optimization reasons. On the product level, the above-mentioned limitations of the system are not severe enough that they would drive the development of a solution that consumes more power. However, the limitations are identified in the current system, and the next version of hardware may better support the requirements.

3.2 Movement detection

The asset tracker is equipped with multiple sensors. An accelerometer is also installed in the tracker which can be thought of as a sensor that quantifies the movement index. Acceleration to anybody at rest causes motion to that body. The accelerometer quantizes the cause of motion, i.e., acceleration, and can be used to ideally predict the velocity of a body. Therefore it is established that whenever the state of motion changes from rest to moving, the acceleration changes on that body. This change in acceleration can be used to detect if the body is stationary or whether it has experienced any movement.

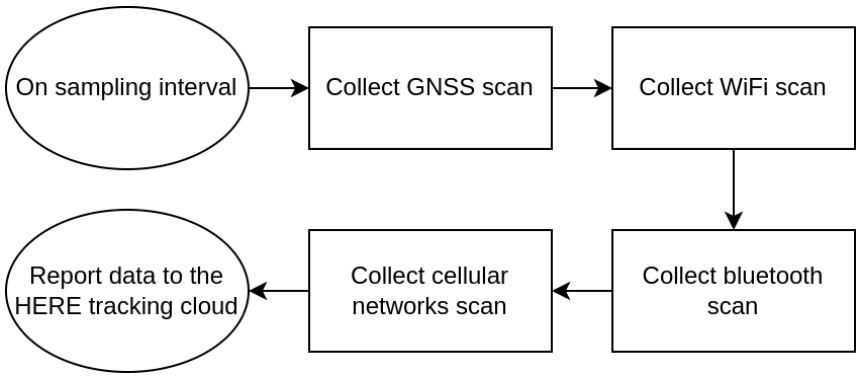


Figure 3.2. *Old reporting architecture of the asset tracking device.*

The main goal of the asset tracking device is to track the assets effectively during their journey. The user of the tracking device is able to configure a sampling interval. At each sampling interval, the tracking device gathers all the required data for estimating the position of the tracking device. This data includes but is not limited to GNSS, WiFi, Bluetooth, and cellular network scans, as shown in figure 3.2. However, when the tracking device has not moved, all such scans would produce pretty much similar position estimates compared to what was estimated when the tracking device was moving last time. Therefore,

as long as the tracking device is stationary, the cached scans can be reported to the user along with their old timestamps and the data stating that the tracking device has not moved, so a position was not collected. The user can only be notified of the new position whenever it changes and hence reducing power consumption.

The installed accelerometer has an interrupt controller as discussed in section 2.1.2. The interrupt controller can also report when the slope between successive acceleration signals goes above a configured threshold for a preset duration. The movement detection can be achieved by configuring the sensor to calculate the difference between consecutive acceleration samples. If the acceleration difference goes beyond a configured threshold and the difference remains over the threshold longer than a preset duration of time, then the sensor notifies that a movement has been detected and consequently, the cached position data is discarded. Upon the interval of collecting the position, when the cached position data is not available then new scans are collected. If the cached position data is available, then the cached data is reported to the cloud. The proposed solution is illustrated in figure 3.3.

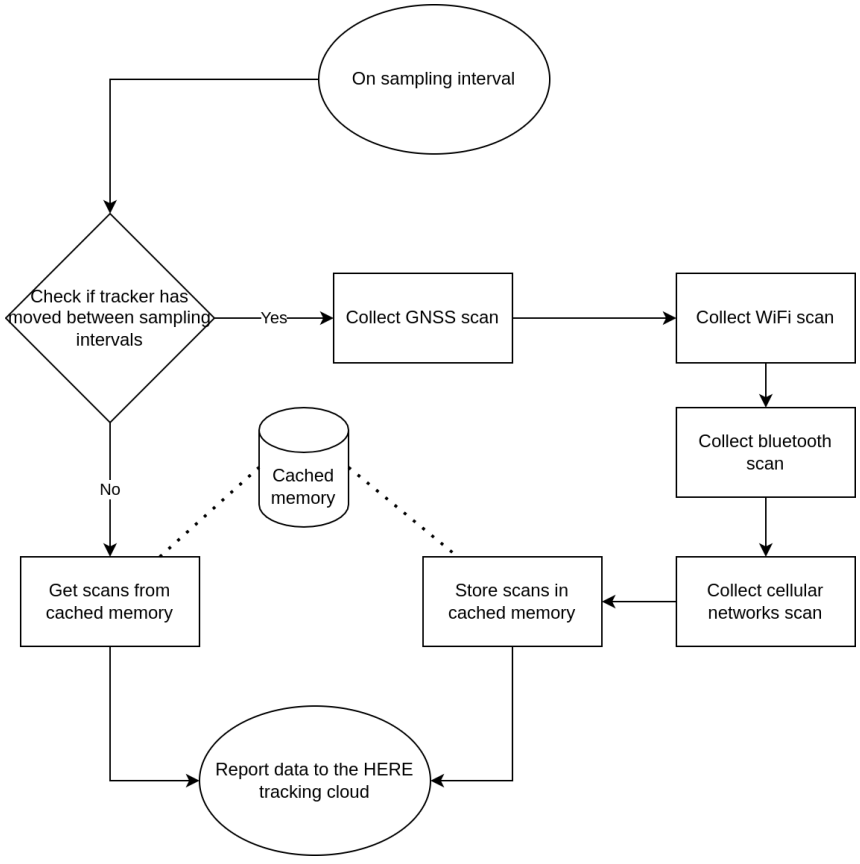


Figure 3.3. *Proposed reporting architecture of the asset tracking device.*

4. BUILDING BLOCKS FOR TAKEOFF DETECTION ALGORITHM

4.1 Data collection and tooling

The accelerometer and barometer data need to be collected to understand what aircraft mobility events look like in terms of the sensor data. It makes collecting accelerometer and barometer data the first step in building the aircraft take-off detection algorithm. The sensor data was collected on commercial and cargo aircrafts to develop the necessary algorithms. Table 4.1 shows the list of flights undertaken for data collection and testing purposes.

Origin	Destination	Aircraft
Helsinki (HEL)	Doha (DOH)	Boeing 787
Doha (DOH)	Islamabad (ISB)	Airbus 350
Islamabad (ISB)	Doha (DOH)	Boeing 777
Doha (DOH)	Helsinki (HEL)	Boeing 787
Helsinki (HEL)	Istanbul (IST)	Airbus A321
Istanbul (IST)	Islamabad (ISB)	Airbus A330
Islamabad (ISB)	Istanbul (IST)	Boeing 777
Istanbul (IST)	Helsinki (HEL)	Airbus A321
Helsinki (HEL)	Paris (BVA)	Boeing 737
Brussels (CRL)	Helsinki (HEL)	Boeing 737
Helsinki (HEL)	Stockholm (ARN)	Airbus A321
Stockholm (ARN)	Malmö (MMX)	ATR 72
Malmö (MMX)	Stockholm (ARN)	ATR 42
Stockholm (ARN)	Helsinki (HEL)	Airbus A320
Frankfurt (FRA)	Birmingham (BHX)	ATR 72
Birmingham (BHX)	Frankfurt (FRA)	ATR 72
Berlin (BER)	Helsinki (HEL)	Embraer 190

Table 4.1. *Flights undertaken for algorithm development and testing.*

Data collection was generally started before taking data collection devices into the aircraft.

The cargo in the cargo hold is not supposed to move while onboard, so to have similar acceleration patterns, the devices were appropriately stowed for the duration of the flight after taking them into the aircraft. After the landing, the devices were picked up and taken out of the aircraft, and data collection was turned off. Moreover, the data was also collected during normal (non-flight) activities, including walking, running, car, tram, and high-speed train rides. It was done to reduce the false detections in the algorithm. The data was collected on two kinds of devices:

1. Using the smartphone:
   At the beginning of the project, the embedded devices where the algorithm was supposed to be deployed were not ready to collect data. Therefore, the data collection was performed using smartphones running Android. A custom-built Android application was used to collect the accelerometer data at $50Hz$ and barometer data at $10Hz$. The data was collected on Google Pixel and Samsung Galaxy S10.
2. Using the embedded device:
   A storage chip was integrated into the system to store the collected data on the asset tracking devices. Then, the accelerometer and barometer data collection features were developed in the asset tracking device, which collected the accelerometer data at $10Hz$ and the barometer data at $1Hz$. Once data collection was stopped, the data was extracted from the devices via a USB connection.

4.2 Airplane acceleration patterns

In this section, one flight will be demonstrated for analysis of acceleration patterns of an aircraft during take-off and landing. The flight departed from Helsinki for Doha on a Boeing 787. Data collection was started in the airport boarding lounge, and the asset tracking device was taken into the aircraft. The data plot is shown in figure 4.1. After entering the aircraft, the data collection device was stowed in the overhead compartment. The walking period can be observed in the graph, marked as 1, where the acceleration values are oscillating, followed by a static period in which acceleration was distributed on the x- and negative z-axis. At this point, the device was stowed in the overhead compartment, and the aircraft was preparing to take off. The engine vibration while turning on can be observed in the graph, followed by taxing on the runway, marked as 2, and finally taking off, marked as 3. During the take-off, the device experienced additional acceleration on the y-axis due to 2 reasons:

- The aircraft was accelerating in the direction of the y-axis.
- During and after liftoff, the gravity vector is also represented on the y-axis until the aircraft reached cruising altitude.

The reduction in the take-off angle as the aircraft continues to increase its altitude after

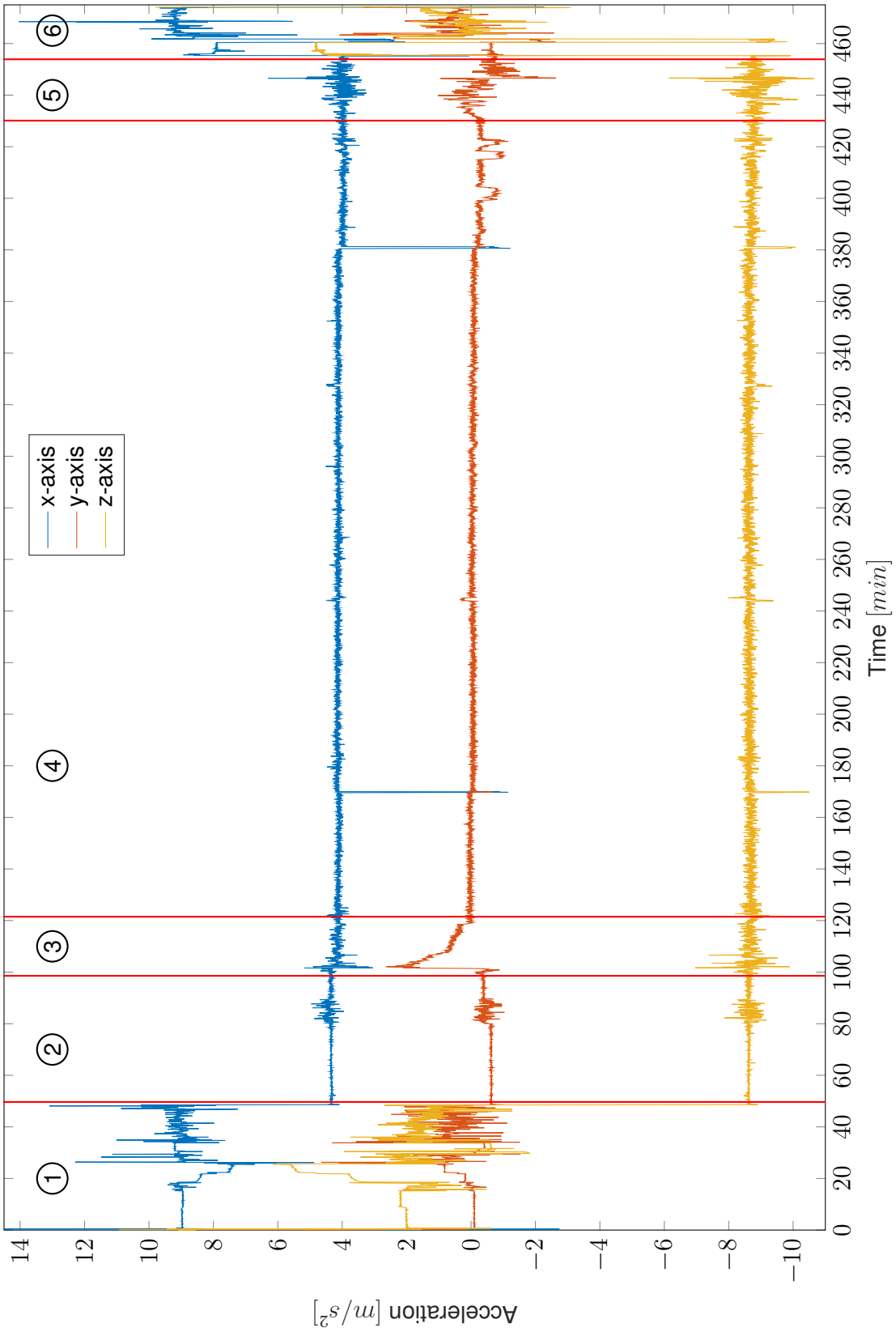


Figure 4.1. *Accelerometer data during the flight from Helsinki (HEL) to Doha (DOH) collected on Boeing 787.*

the liftoff causes the slowly diminishing acceleration on the y-axis. The typical take-off angle of commercial aircrafts is $10-15degrees$ [34], which slowly returns to normal as the aircraft approaches its cruising altitude. The turbulences during the flight can also be seen in the recorded data in the phase marked as 4. A high variation in the data plot of figure 4.1 is due to aircraft landing, and this period is marked as 5. After a brief period when the aircraft completely stopped, the device was first taken out of the overhead compartment and then out of the aircraft. The data collection was stopped after taking the device out of the aircraft, and this last period is marked as 6. The data marked as two and three holds significant importance for developing the take-off detection algorithm. However, the rest of the data is also important as it will help us to avoid false detections. Similar datasets with multiple devices on various flights were collected during the development and testing work as shown in table 4.1. The rest of the datasets will not be described in detail to keep this discussion relatively concise.

4.3 Filtration techniques to reduce noise

Accelerometer data generally contains too much noise. Many parameters introduce noise in the accelerometer data, e.g., acceleration sensing structure, amplifiers, etc. However, noise can be reduced from the accelerometer data by using filters, and in this section, we will take a look at filtration methods that can reduce the noise from accelerometer data.

4.3.1 Moving average filter

The moving average filter is one of the most commonly used filters for digital data filtration. It is easy to understand and use due to its simplicity. This filter is optimal for reducing random noise while keeping the sharp changes to the signal. However, it is not an optimal filter for filtering out the frequency response of a signal. It is computationally lightweight as it averages out the signal over a certain window length n [35]. The filtered data $\bar{x}$ using raw data x is given by:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{4.1}$$

Figure 4.2 shows the moving average filter results on accelerometer data. The raw data was collected with the accelerometer in the backpack while walking. Moving average filtration is performed in this plot with three different rolling window sizes n. The total number of samples collected were 300 at $10Hz$, i.e., $30seconds$, and then filtration is performed for windows size n of 5, 25, and 50. It can be observed from the plot that the smaller the window size n, the filtered data quickly follows the raw data and has a lesser smoothing effect. The opposite behavior can be observed while window size n is

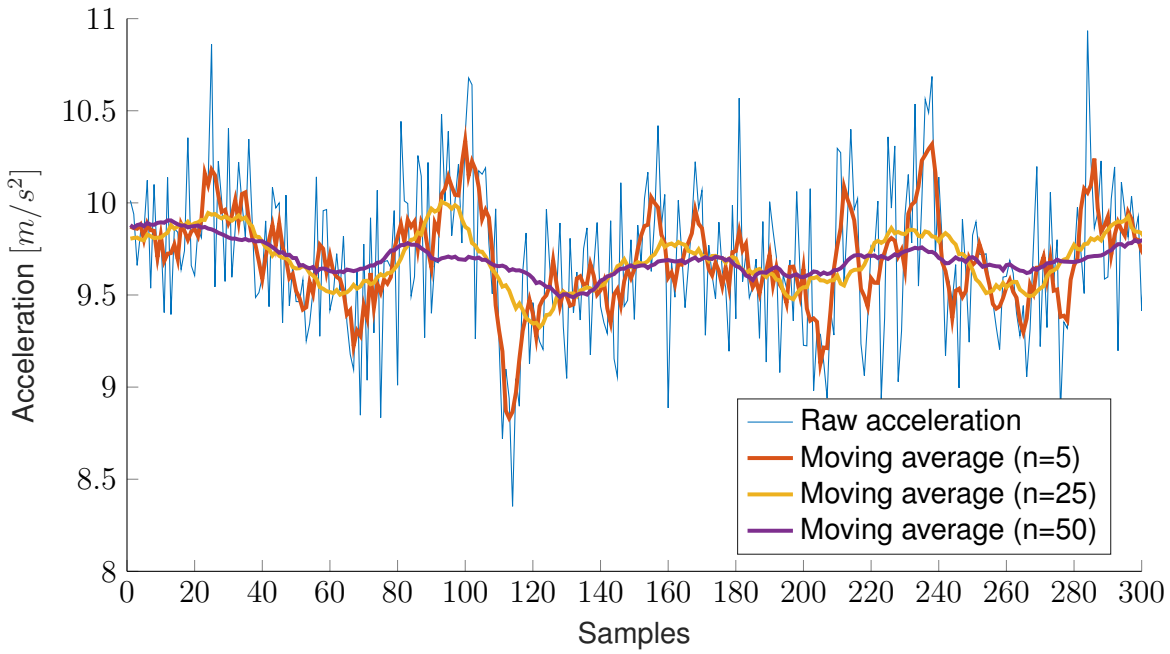


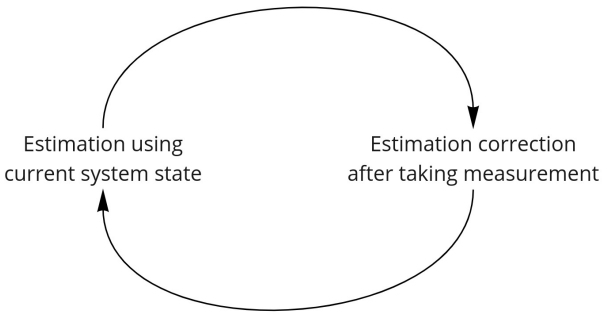
Figure 4.2. *Moving average filter with different window sizes.*

increased. Another observation that can be made is that filtered data lags behind the raw data and increases the window size n also increases the lag and vice versa.

4.3.2 Kalman filter

R.E. Kalman proposed the Kalman filter in 1960 [36], which provides an iterative solution for discrete data filtration. The basic Kalman filter only deals with linear data. However, Extended Kalman Filter (EKF) can also model and filter the data from a non-linear system. This thesis will only deal with the basic Kalman filter as the accelerometer that is being used has a linear response. The Kalman filter estimates the future state of the system using the current state. It can even work when the system cannot be accurately modeled. Figure 4.3 illustrates the iterative Kalman update process. After taking the measurement, the filter estimates the next system state, and then the next measurement is used to correct the estimation. Typically, if the measurement is unreliable, then estimation is weighted more, and if the measurement is more reliable, then the measurement is weighted more. Let us look at how estimation and correction work from the mathematical point of view. Kalman filter tries to estimate the state $x \in \mathbb{R}^n$ by the equations 4.2 - 4.6 [37].

Figure 4.3. *Iterative Kalman process.*



$$\hat{x}_{k+1}^{-} = A_k \hat{x}_k + B u_k \tag{4.2}$$

$\hat{x}^-$ is an a-priori estimate that is expressed by 4.2. $\hat{x}^-$ depends a-posteriori estimate $\hat{x}$ and controlled input to the system $u_k; u \in \mathbb{R}^l$. The relationship between $\hat{x}$ and $\hat{x}^-$ is given by $n \times n$ matrix A. The relationship between u_k and $\hat{x}^-$ is given by $n \times l$ matrix B.

$$P_{k+1}^{-} = A_k P_k A_k^T + Q_k \tag{4.3}$$

P_k^- is the error covariance $\hat{x}^-$ and it is given by expression stated in 4.3. P_k^- depends on process error covariance Q_k, $\hat{x}$, P_k, and A.

$$K_k = \frac{P_k^- H_k^T}{H_k P_k^- H_k^T + R_k} \tag{4.4}$$

Kalman gain K_k dynamically governs the weightage between $\hat{x}^-$ or measurement $z_k; x \in \mathbb{R}^n$. K_k is computed by expression given by 4.4 and it relies on P_k^-, measurement error covariance R_k and the relation between measurement z_k and $\hat{x}^-$ that is given by $m \times n$ matrix H_k.

$$\hat{x}_k = \hat{x}_k^- K(z_k - H_k \hat{x}_k^-) \tag{4.5}$$

$\hat{x}$ is given by expression stated in 4.5 and it uses z_k, K_k, H_k and $\hat{x}^-$.

$$P_k = (I - K_k H_k) P_k^- \tag{4.6}$$

Finally, for the next iteration, P_k is computed by the expression stated in 4.6.

Now we will discuss the application of the Kalman filter on raw acceleration data. Figure 4.4 shows the Kalman filtration result on raw acceleration over 300 samples. It can be observed from the graph that Kalman filtered output is following the raw acceleration without any significant delays. Moreover, it can also be observed that the Kalman filter output is smooth as compared to raw acceleration and moving average filtration output. This graph also compares the results of the moving average filter with the Kalman filter. It can be noticed that moving average filtered output is more susceptible to sudden changes in the raw acceleration. This susceptibility of moving the average filter to sudden changes can be reduced by increasing the window length n of the filter. However, that makes the output of the moving average filter lag behind the raw acceleration, which is undesirable in our application.

Figure 4.5 shows the error covariance P_k of the Kalman filtered output. It can be observed that error covariance P_k is high in the beginning, as expected, and drops down quickly to about 0.12 after 20 samples. Once the error covariance achieves its steady state, the

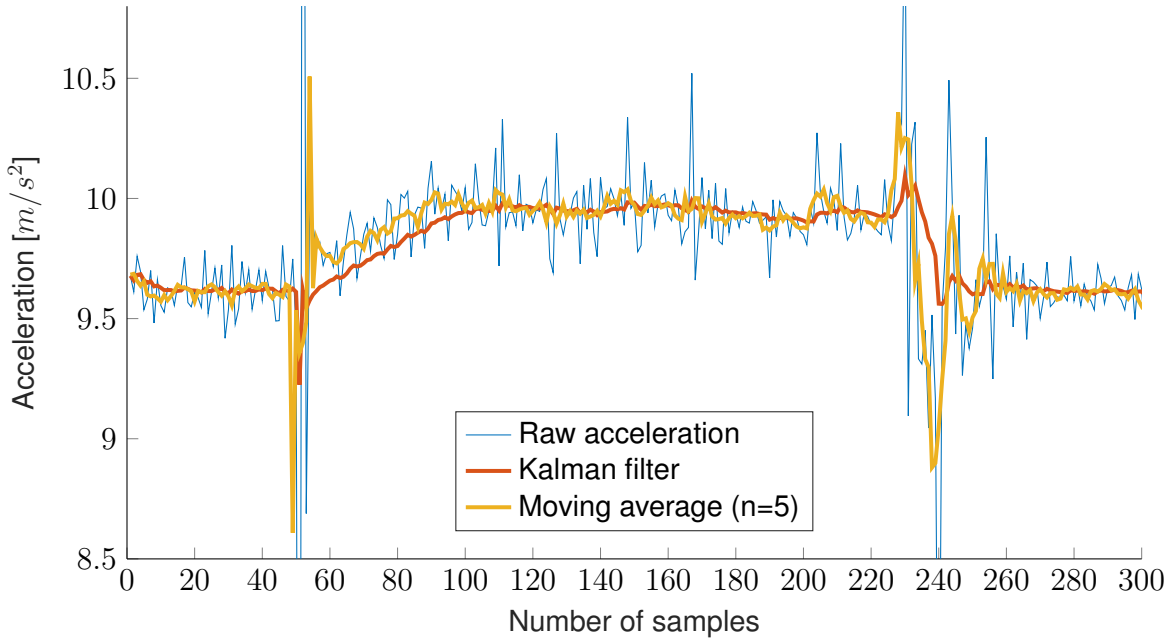


Figure 4.4. *Kalman filter results on raw acceleration data along with moving average filter comparison.*

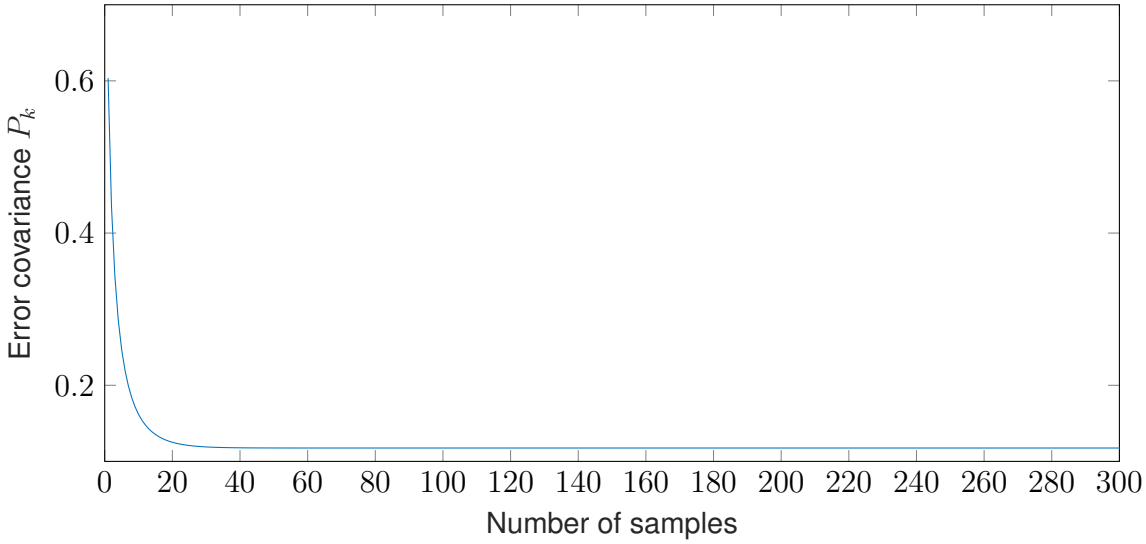


Figure 4.5. *Error covariance P_k (Eq. 4.6) of kalman filter.*

results of the Kalman filter are within a reliable range of error. It is to be noted here that this is the typical behavior of the error covariance of Kalman filters.

4.4 Steady condition assessment using rolling variance

Variance is the numerical description of how much data is spread out across its distribution [38]. It is given by:

$$\sigma_t = \frac{\sum_{i=t}^{t-n} (x_i - \bar{x})}{n-1} \tag{4.7}$$

Where:

σ_t: Variance at time t

$\bar{x}$: Mean of the dataset

n: Total number of elements in the dataset

The concept of rolling variance is relevant for our application for assessing no-motion detection. The total number of elements n over which variance is calculated are kept fixed while calculating the variance. The variance can be calculated on a First In First Out (FIFO) buffer to assess whether the data is static or not.

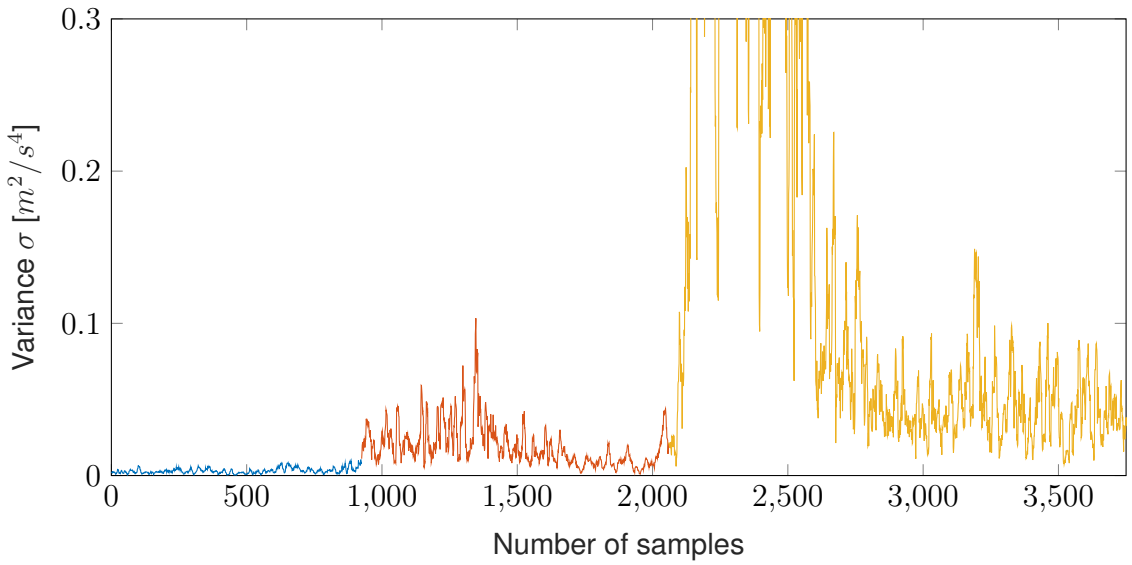


Figure 4.6. *Rolling variance σ of accelerometer data before, during, and after take-off.*

Figure 4.6 shows the rolling variance calculated over window size $n = 10$ of accelerometer data collected on an aircraft. It can be observed from the figure that the variance is low in the beginning, shown in blue when the aircraft was steady on the tarmac taxing. After the aircraft started taxing, the variance is shown in red while the aircraft was still on the tarmac. The variance shown in yellow was recorded at the point when full throttle was turned on just before the take-off, and then the aircraft was lifted off the ground. At this point, it can be noted that variance is higher than when the aircraft was steady on the ground. A pattern of higher variance is experienced throughout the flight on different datasets.

4.5 Pose detection using quaternions

To understand the basics of quaternions, let us visit the vector dot and cross products. The dot product of two vectors $\bar{u}$ and $\bar{v}$ is given by

$$\bar{u}.\bar{v} = |u||v| \cos\theta \tag{4.8}$$

where θ is the angle between the vectors around the axis of rotation. The cross product between the two vectors is given by

$$\bar{u} \times \bar{v} = |u||v| \sin\theta\hat{n} \tag{4.9}$$

where θ is the angle between the vectors and $\hat{n}$ is normal to the both vectors. Therefore cross product always gives out the vector parallel to the axis of rotation. The quaternion between the two vectors can be given by a derivation of [39]

$$q = \begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \hat{n}_x \sin\theta \\ \hat{n}_y \sin\theta \\ \hat{n}_z \sin\theta \end{bmatrix} \tag{4.10}$$

Afterward, the Euler angles of rotation against each axis in xyz order can be computed by

$$m = \begin{bmatrix} q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_z q_z) & 2(q_x q_z - q_w q_y) \\ 2(q_x q_y - q_w q_z) & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z - q_w q_x) & q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \tag{4.11}$$

where i corresponds to number of rows, j corresponds to number of columns, and m_{ij} is an element of the matrix m.

$$\theta_x = \tan^{-1}(\frac{-m_{23}}{m_{33}}) \tag{4.12}$$

$$\theta_y = \sin^{-1}(m_{13}) \tag{4.13}$$

$$\theta_z = \tan^{-1}(\frac{m_{32}}{m_{22}}) \tag{4.14}$$

This way, we can detect the rotation angle on each axis of the accelerometer.

4.6 Atmospheric and aircraft pressurization

The air pressure decreases as altitude is increased. There are many well-defined atmospheric pressure models [40] that give out the altitude as a function of air pressure. However, the atmospheric pressure models predict the air pressure/altitude in the natural atmosphere. In our use case, the aircraft cabin is artificially pressurized [41] in order to make flying a comfortable experience for passengers. The whole cabin was not pressurized in the past, but that is not the case anymore. It gives rise to the problem that the altitude cannot be calculated using barometer data using the atmospheric pressure model alone. The aircraft cabin pressurization model must be known in order to calculate a reasonably reliable altitude from the barometer data. Different aircraft types have different cabin pressurization techniques, which give rise to different cabin pressurization models. Therefore, in our application, the pressurization models cannot be incorporated as it cannot always be known in advance that which kind of aircraft will transport the device.

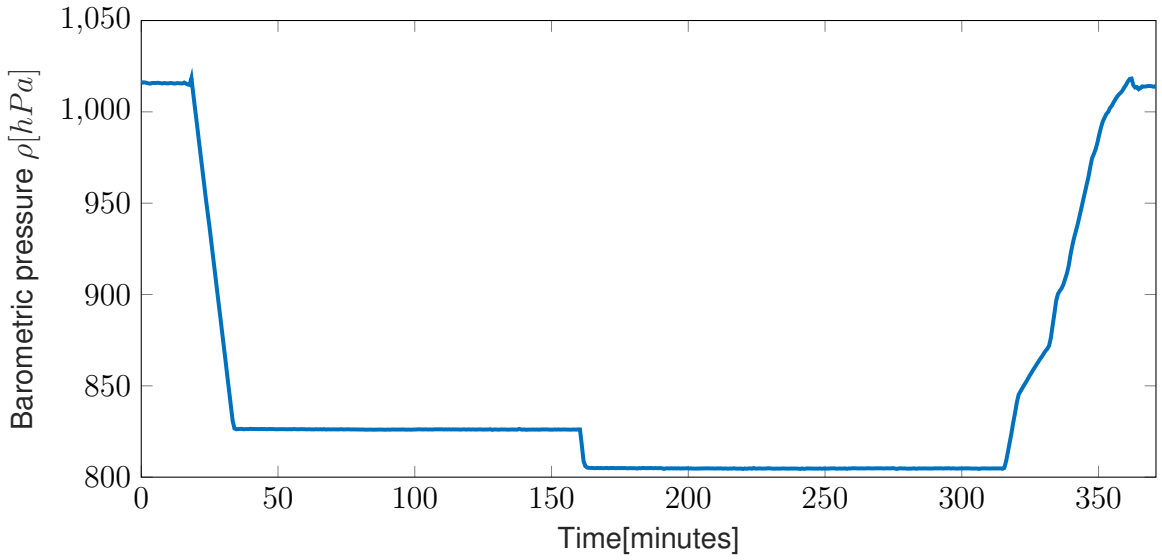


Figure 4.7. *Barometric pressure $\rho[hPa]$ data collected during a flight between Helsinki and Doha on Boeing 787.*

Even with the unavailability of exact altitude, we can detect the increase/decrease in altitude using barometric data. Figure 4.7 shows the pressure data collected by a barometer inside the cabin of a commercial aircraft. The cabin pressurization just before the take-off can be noted in this graph. This pressurization is characterized by a peak in pressure at about 18 minutes. Afterward, the air pressure is decreased as the aircraft gains altitude. The typical cruising altitude of a Boeing 787 is 10,700 meters and outside air pressure at this altitude is around 265 hPa. However, as can be observed in this figure, the pressurized cabin has an air pressure of 800 hPa approximately. As the aircraft starts to descend, the air pressure starts to increase, and when it touches down the ground, de-

pressurization of the aircraft cabin can be observed by a small peak in this pressure plot.

5. AIRCRAFT TAKEOFF DETECTION ALGORITHM

5.1 Accelerometer based detection algorithm

When the airplane is taking off, it experiences a unique pattern of acceleration followed by a change in the aircraft's orientation. As discussed in 2.1.6, the accelerometers always report a gravitational acceleration i.e. $1g$ when present. Since we are using three accelerometers configured in the right angle configuration in our system, we can theoretically calculate the accelerometer's orientation by the gravitational acceleration vector distribution among three axes. Moreover, the overall acceleration value also increases when the airplane takes off. The increase in deceleration is observed when the airplane lands on the ground.

5.1.1 Sustained increase in acceleration

Let us revisit the life cycle of the asset tracking device from origin to destination. The tracking device is attached with the asset to be tracked, and the shipper may use aircraft to ship the asset. The device can be placed in any orientation in an aircraft's cargo hold when the shipper uses air cargo for shipping. The device is installed with a tri-axial accelerometer, so the detection algorithm must be orientation agnostic. It is required because the gravitational vector and additional take-off acceleration can be distributed at any orientation on a different axis of the accelerometer. An algorithm to detect a sustained increase in acceleration and orientation detection is proposed in light of this requirement.

First of all, the magnitude of each sample of acceleration is calculated.

$$|A| = \sqrt{A_x^2 + A_y^2 + A_z^2} \tag{5.1}$$

Equation 5.1 represents the magnitude of acceleration that does not contain any information about the orientation of the acceleration vector, but it contains the intensity of overall acceleration that is experienced from any direction. Afterward, the acceleration magnitude data is filtered using the moving average filtration technique described in section 4.3.1. Equation 4.1 gives the filtered acceleration magnitude data $\bar{x}$ that is used in the further processing.

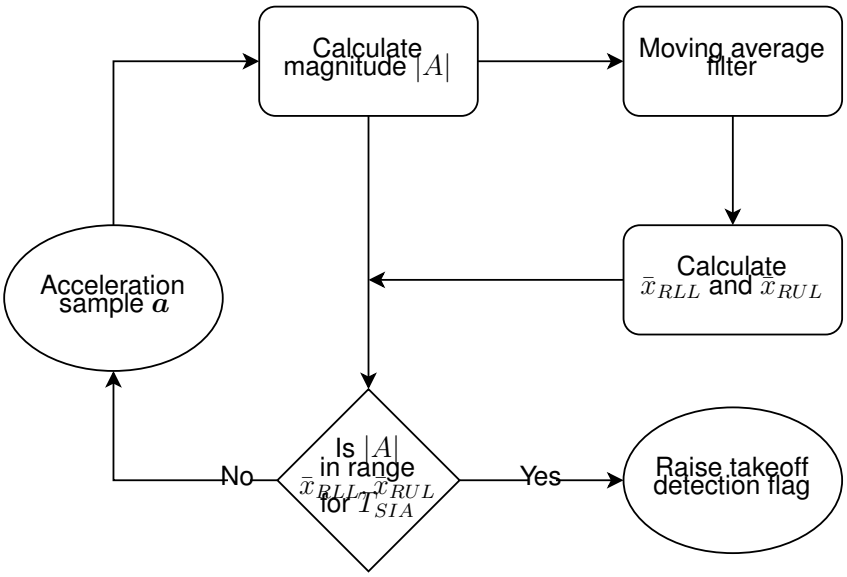


Figure 5.1. *Algorithm for detecting a sustained increase in acceleration.*

Consider a range of values that is higher than $\bar{x}$ and the lower limit of the range is $\bar{x}_{RLL}$ and the upper limit is $\bar{x}_{RUL}$.

$$\bar{x}_{RLL} = \bar{x} + \bar{x}_{LC} \tag{5.2}$$

$$\bar{x}_{RUL} = \bar{x} + \bar{x}_{UC} \tag{5.3}$$

where $\bar{x}_{UC}$ is greater than $\bar{x}_{LC}$ and both are constants of the system.

When the acceleration magnitude $|A|$, given by eq. 5.1, falls within the range of $\bar{x}_{RLL}$ and $\bar{x}_{RUL}$ consecutively for a time period T_{SIA} then the sustained increase in acceleration is detected by the algorithm. The algorithm is illustrated in figure 5.1.

5.1.2 Detecting take-off angle

After the cargo is loaded into an aircraft, aircraft doors are closed from the outside, and the aircraft remains steady in its parking spot for a while when a pre-takeoff briefing is held in the cockpit. Afterward, the aircraft slowly starts taxing before getting permission for take-off. Therefore, it can be safely assumed that the cargo in the aircraft remains steady for a while after being loaded into the cargo hold. This is a crucial assumption from the perspective of the pose detection technique explained in section 4.5. This steady period can provide us with the reference vector that can be used to detect the change in the angle when the aircraft is taking off. It is to be noted here that the change in the aircraft's pitch angle shortly follows the sustained increase in acceleration while the aircraft takes off from the ground. Figure 5.2 illustrates the algorithm to detect the take-off angle of an aircraft.

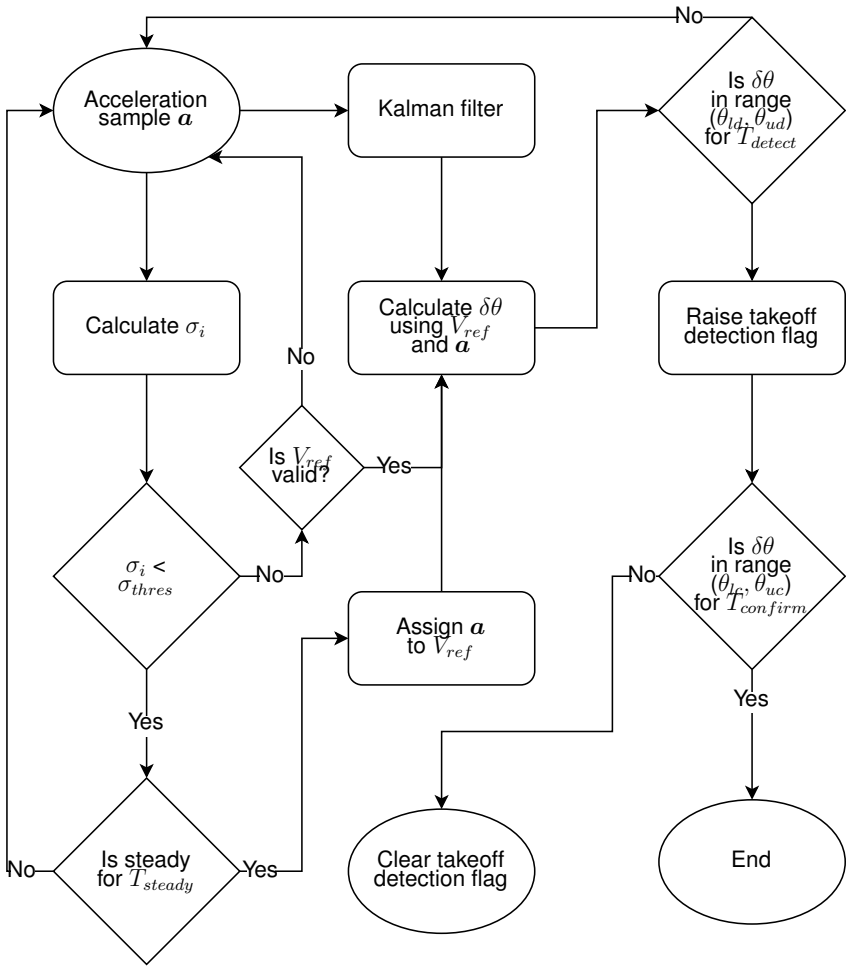


Figure 5.2. *Algorithm for detecting aircraft take-off angle.*

Let us now look at how the reference vector can be determined by calculating steadiness in the motion reported by the accelerometer. The concept of rolling variance was discussed in section 4.4 which explained an observation about rolling variance σ_t in figure 4.6 before aircraft takes off. The rolling variance is thresholded to get the steady state acceleration vector so that when rolling variance σ_t remains under σ_{thres} consecutively for a time period T_{steady} then the current acceleration sample $\boldsymbol{a}$ is selected as the reference vector u.

Once a reference vector is fixated, an algorithm to detect the change in orientation is triggered. The raw acceleration samples are filtered using the Kalman filter, and the change in pose is detected as explained in section 4.5. Equations 4.12-4.14 are used to detect the change in takeoff angle $\delta\theta$. When the change in takeoff angle $\delta\theta$ remains within the range $(\theta_{ld}, \theta_{ud})$ continuously for a time period T_{detect} then takeoff detection is triggered by this block.

Once this block has detected take-off, the take-off angle must remain within the range $(\theta_{lc}, \theta_{uc})$ for a time period $T_{confirm}$ or otherwise, take-off detection is canceled. This is done to avoid false positives as the take-off angle in an actual aircraft take-off slowly

returns back to zero. In case of an actual takeoff, if the time taken for change in takeoff angle $\delta\theta$ to get out of range $(\theta_{lc}, \theta_{uc})$ is T_{cruise} then $T_{confirm}$ must be shorter than T_{cruise}.

It was observed during the development of the takeoff detection algorithm that detecting the sustained increase in acceleration alone cannot detect aircraft takeoffs in a robust manner. Such detection alone might be sufficient for a high accuracy of true positives but fails to detect true negatives robustly. Therefore, an algorithm to calculate and detect the change in pitch angle was incorporated in the system. It was also observed that filtered data through a Kalman filter provided a better estimate of pitch angle in a real world scenario which necessitated the incorporation of Kalman filter in the calculation of pitch angle.

5.2 Barometer based detection algorithm

The barometer reports a decrease in air pressure while the airplane is taking off. This decrease in air pressure is reported by barometer even with the pressurized cabin. The cabin pressurization does not entirely halt the decrease in air pressure but somewhat limits it and slows it down. At cruising altitude, the air pressure reported by the barometer remains constant as long as the altitude is constant. Air pressure starts to rise when the aircraft is landing, and it becomes constant on the tarmac after landing is completed.

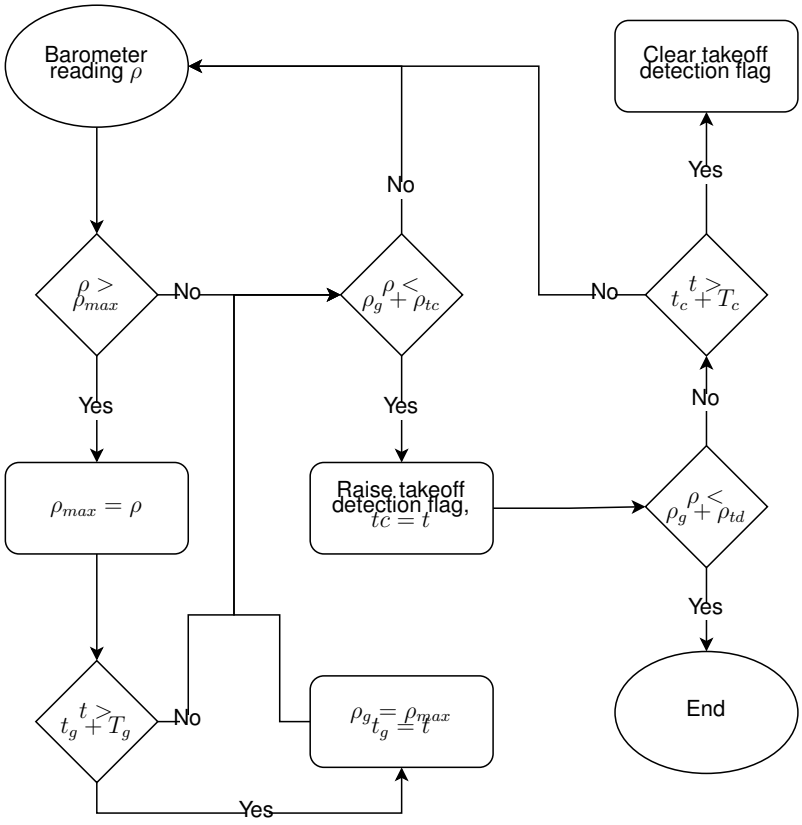


Figure 5.3. *Algorithm for detecting the increase in altitude of an aircraft.*

Let us now discuss the algorithm to detect aircraft take-off which is illustrated in 5.3. The initial barometer reading ρ_i is assumed to be ground-level air pressure ρ_g. The ground level air pressure ρ_g is chosen as the minimum barometer reading ρ in a time interval T_g. When the barometric reading gets below ground level air pressure ρ_g by take-off detection pressure ρ_{td}, then the take-off detection flag is raised. In an actual take-off, the barometric reading keeps dropping while the aircraft is taking off, i.e., as the altitude of the aircraft increases. Therefore, once the take-off detection flag has been raised, the barometric reading ρ also needs to keep decreasing until take-off confirmation pressure ρ_{tc} is attained within a time limit of T_c. However, if the barometric reading gets above ρ_{td} during the interval T_c or if it does not get below ρ_{tc} within the time limit T_c, then the take-off detection flag is cleared.

5.3 Interoperability of detection algorithms

We have discussed multiple detection algorithms earlier in this chapter. Let us now look at how these algorithms can work together. The algorithms discussed in 5.1.1 and 5.1.2 use accelerometer data to detect takeoff while algorithm discussed in 5.2 uses barometer to detect the takeoff of an aircraft. According to regulations defined in [11], the battery-powered cargo tracking devices must have two independent methods to detect aircraft take-off. It means that barometer and accelerometer algorithms cannot depend on each other when these two sensors are supposed to provide independent means for the take-off detection.

A sustained increase in the acceleration can be observed from the accelerometer data shortly before and after the aircraft lifts off the ground. While the aircraft is lifted off the ground, the change in pitch angle is also observed. It can be safely assumed that these two events, i.e., sustained high acceleration and change in pitch angle, happen within a short time period. This time period can be specified as T_{inter}. The barometer also reports the start in dropping air pressure simultaneously, but this information cannot be used in the overall algorithm because the barometer-based algorithm must remain utterly independent of the accelerometer-based algorithm.

The accelerometer-based detection algorithm comprises of two parts, and both must raise detection flags within a time interval of T_{inter} so that the accelerometer-based algorithm can detect take-off. Both accelerometer- and barometer-based algorithms can independently enable the tracking device's flight mode, i.e., turn off radios. In other words, both detectors have an 'OR' logic for their interoperability. When any detector raises its take-off detection flag, the device is put into flight mode. Afterward, when the other detector raises its take-off detection flag, it does not affect the current state of the flight mode, and the device continues to be in the flight mode.

6. ALGORITHM TESTING AND RESULTS

6.1 Motion detection

The testing procedure and results for algorithms proposed in chapter 3 are described in this section.

6.1.1 Movement detection

The asset tracking device ingests the GNSS position and other scans at the minimum interval of one minute. Therefore, it in important to detect the movement at once per minute when the device is moving. The algorithm proposed in section 3.2 is deployed in the asset tracker and a test round was performed.

The test round consisted of various activities concerning device being stationary or moving. The activities which were tested are listed below:

- **Walking:**
  The device was put in a backpack and tested during a walk from Tampere university, Hervanta campus to Suolijärvi sauna and back. The walk lasted 31 minutes and the movement was also detected 31 times.
- **Bus:**
  The device was taken to a bus ride starting from Opiskelija C to HoitoKoti which lasted 31 minutes. The movement was detected in 30 minutes where 1 minute was classified as stationary during the bus ride.
- **Tram:**
  The movement detection algorithm was also tested in Tampere's tram which is a special case because the algorithm under test relies on the differential acceleration data and the modern tram offers a fairly steady ride. The test was started on a ride between Tuulensuu and Hervantajärvi B which lasted 28 minutes. The movement was detected in 16 out of 28 minutes.
- **Stationary:**
  The device was put stationary on a rigid desk for 90 minutes and it detected movement once during the testing period.

The test results explained above are compiled in the form of confusion matrix in figure 6.1. The overall accuracy of correct detection was found to be 92.2%

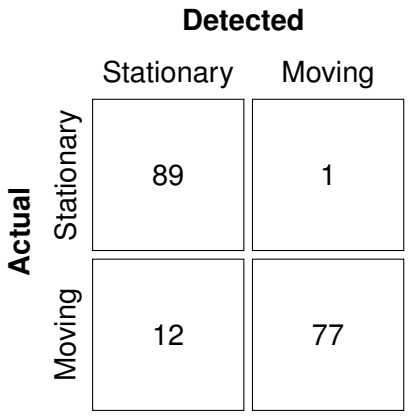


Figure 6.1. *Movement detection confusion matrix for 180 samples.*

6.1.2 Shock detection

The shock detection testing was held in a real-world environment. The device was attached to a string and dropped from different heights and the detected shock intensity was reported. The test setup is shown in the figure 6.2.

Figure 6.2. *The device attached to a string during test.*

The shock detection algorithm was configured to report the detected acceleration values above 2g in this test and the device was dropped from different heights with a string attached to the device to prevent the damage to the device. Each drop height was repeatedly tested five times in a row. A summary of test result is described in table 6.1.

Height	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
0.5m	3.25g	3.27g	3.36g	3.94g	3.02g
1.0m	4.86g	4.68g	4.73g	5.09g	4.45g
1.5m	6.21g	6.15g	6.35g	5.93g	6.57g

Table 6.1. *Drop test results for shock reporting*

6.2 Airplane takeoff detection

The testing procedure and results for algorithms proposed in chapter 5 are described in this section. First of all, aircraft takeoff data was needed to be collected. This data collection was done at higher frequencies during the early development phase of this thesis. The take-off algorithm was later optimized to use a lower frequency in order to have less run-time computations and hence longer battery life of the asset tracking device. The accelerometer and barometer data were initially collected over several flights at 50Hz and 10Hz respectively. The data was then analyzed, and offline algorithm development work was started. Once the algorithm produced results within acceptable error margins, it was deployed to the tracking device. Then the algorithm performance on a live tracking device was assessed during some random activities, two flights, and a high-speed train. Once there was enough confidence in the operability of the take-off detection algorithm, twenty tests were made to assess the reliability and robustness of the deployed algorithms. In all twenty tests, no false negatives were reported and all take-off events were correctly detected within the acceptable margin of error. The test results were documented and later used to analyze the compliance requirements to put the tracking devices unattended in aircraft.

The tracking device has a cellular modem that is not allowed to be functional aboard an aircraft during a flight, and this restriction was also taken into account during the development and testing of the algorithms. A particular radio off mode was developed on the tracking device in which all radios, including cellular modem, WLAN, and Bluetooth modules, were powered down at all times. The take-off detection was logged in the device logs to enable later analysis of the working of the algorithm.

6.2.1 Algorithm test run for tuning

The algorithm was deployed to the asset tracking device after gaining enough confidence in its operability. Initial algorithm testing was done using this deployment on the asset tracking device on the following trips:

- Flight from Helsinki (HEL) to Paris Beauvais (BVA) airport on Boeing 737-800

- High-speed train from Paris Nord to Brussels Midi
- Flight from Charleroi (CRL) to Helsinki (HEL) on Boeing 737-800

The asset tracking device was running the initial implementation of the take-off detection algorithm in the above flights and train ride. Real-time algorithm results were logged in the device logs. Moreover, the data was collected during the flights in the radio-off mode to further tune and improve the algorithm. The data collection system was developed so that the data that is provided to accelerometer- and barometer-based take-off detectors is also logged separately. An offline data simulator was also developed that consumed the sensor data, ran it through the algorithms, and formulated results from the simulation. The simulator output provided the timestamps of each algorithm's take-off detections, which were used for analysis purposes. The analysis included the comparison of the timestamps of take-off detections from different flights between simulated and real-time results.

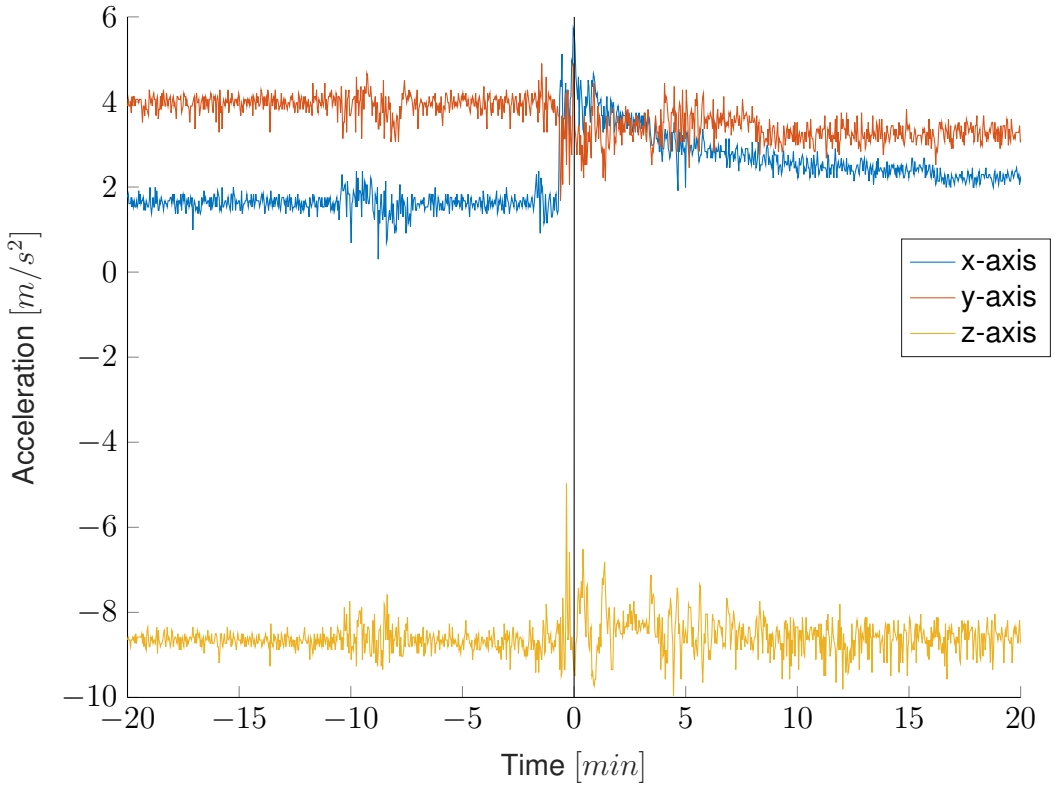


Figure 6.3. *Acceleration $[m/s^2]$ data while Boeing 737 is taking off from Helsinki (HEL) airport.*

Figure 6.3 shows the accelerometer data from the flight from Helsinki to Paris. The time at zero minutes corresponds to the time when the acceleration-based take-off detector raised the take-off detection flag. The plot shows the data from twenty minutes before the take-off to twenty minutes after. The acceleration data is shown in m/s^2. It is to be noted here that $1g$ is equivalent to $9.80665m/s^2$. The acceleration was stable during the period before the aircraft was lifted off the ground, and the magnitude of the acceleration is

expected to be around $9.80665 m/s^2$ during this time. It was the period when the tracking device was placed in the overhead compartment of the Boeing 737-800, and the aircraft was on the tarmac. As the time approached zero minutes, the engines were turned on at full throttle, resulting in increased acceleration. Around this time, i.e., zero minutes, it can be observed in the plot that the acceleration vector was shifted from the y-axis to the x-axis. The acceleration on the y-axis was decreased while increasing the acceleration vector representation on the x-axis. This was the point when the aircraft was changing its pitch angle and getting lifted off the ground, and after a few seconds, the acceleration-based take-off detector raised the take-off detection flag, which is shown by a vertical black line in the plot.

The take-off angle slowly returns to zero as the aircraft approaches cruising altitude once the aircraft has lifted off the ground. It is interesting to note that this phenomenon is reflected in the accelerometer data from the above flight, where the acceleration vector was slowly shifted back towards the y-axis, i.e., the original vector position before take-off. The acceleration on the y-axis started to slowly increase back, while it showed the opposite behavior on the x-axis, where it started to slowly decrease.

The z-axis of the accelerometer was perpendicular to the axis of rotation in this flight; hence it did not have any significant effect due to the change in pitch angle of the aircraft while taking off. It can be observed from the plot of the z-axis that before the aircraft lifted off the ground, it experienced a significant amount of vibration. This vibration was due to mainly two factors:

- The aircraft itself heavily vibrates when its engines are turned on at full throttle while it is on the tarmac.
- The accelerometer sensor is especially prone to the vibration noise from the environment and usually reports amplified vibration.

The increase in variance can also be observed on all axes when the time period of before and after take-off is compared. This is also a characteristic of most aircraft that there is more vibration when the aircraft is flying as compared to when it is on the tarmac, and the aircraft vibrates the most while taking off.

Figure 6.4 shows the barometric pressure plot spanning over ten minutes during aircraft take-off. This data was collected from the same flight as presented in figure 6.3. It can be observed from the figure 6.4 that the air pressure remained constant before the aircraft took off, i.e., air pressure at Helsinki Vantaa airport. A peak in the air pressure can be observed between negative 2 and negative 1 minutes. This happened due to the cabin pressurization of the Boeing 737 before the lift-off. The aircraft started to take off right after the cabin was pressurized, which decreased the air pressure of the pressurized internal cabin. The algorithm explained in section 5.2 detected this decrease in the air

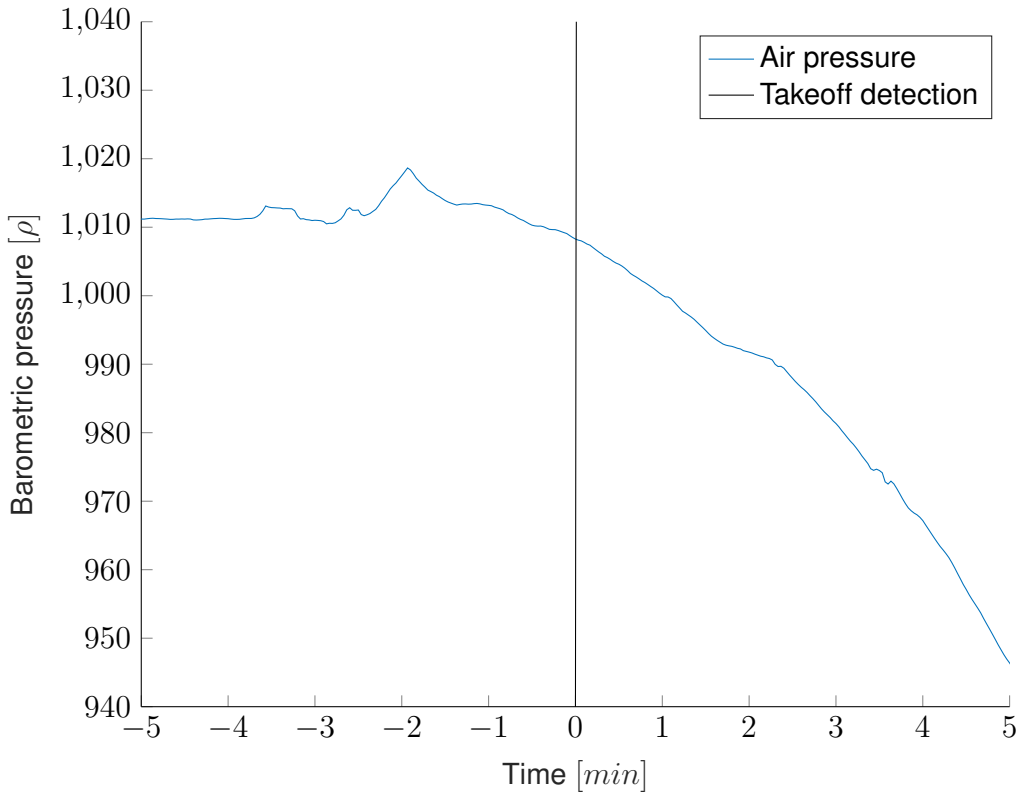


Figure 6.4. *Barometric pressure* $[\rho]$ *while Boeing 737 is taking off from Helsinki (HEL) airport.*

pressure as a take-off which is shown by a vertical black line in figure 6.4.

Figure 6.5 shows the air pressure measured inside the cabin of a high-speed train during a journey from Paris Nord to Brussels Midi. The train's speed went up to $300Kmph$ during the journey. Barometer-based take-off detector looks for the decrease in the air pressure in general, and simply put, the air pressure needs to be decreased significantly in a short period to trigger the take-off detection flag. However, the air pressure also needs to decrease continuously; otherwise, the barometer-based take-off detection algorithm clears the flag. During this high-speed train ride, the air pressure changed abruptly, and at one point, the change in air pressure pattern matched the conditions required for raising the take-off detection flag. However, since the algorithm also looks for the continuous decrease in the air pressure and the air pressure during train ride changes abruptly, therefore, air pressure also increased at times which canceled the take-off detection by the barometer-based take-off detection algorithm.

6.2.2 Final algorithm testing

The algorithms were tested with forty test cases in this phase. Twenty datasets included the aircraft takeoff data and the rest of the datasets contained normal activities. Once the algorithm was tuned to perform predictably well in a field test, four test flights were

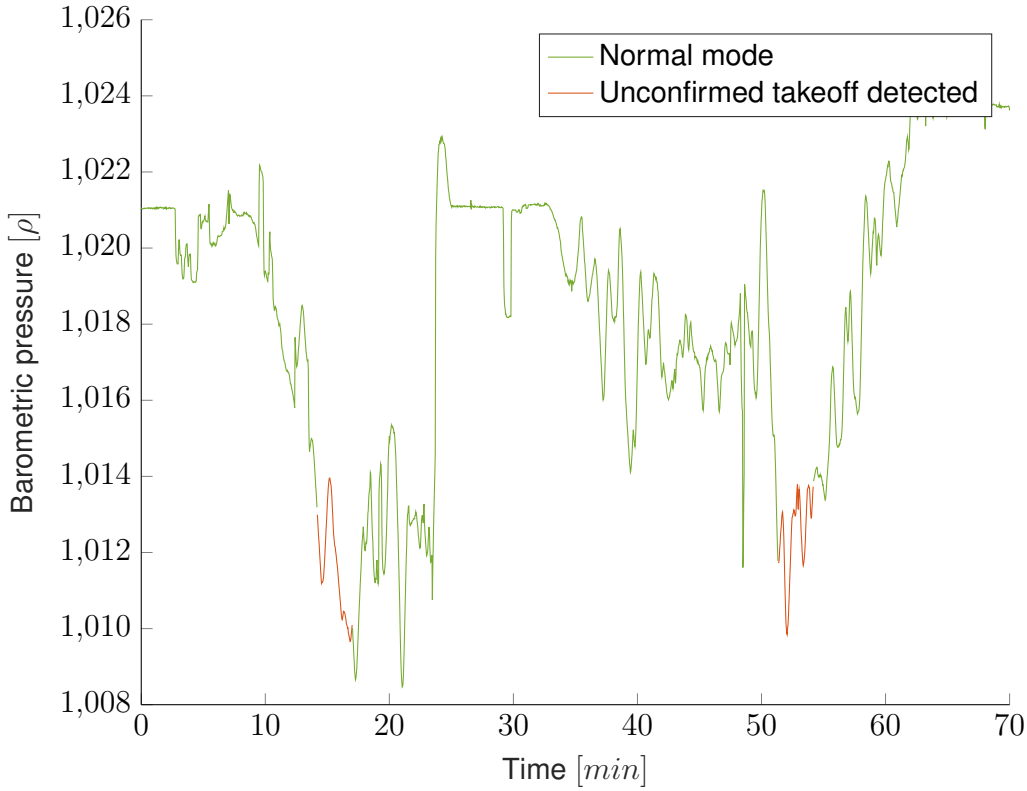


Figure 6.5. *Barometric pressure $[\rho]$ in a high-speed train from Paris Nord to Brussels Midi.*

scheduled, and in each test flight, five devices were taken aboard to asses the algorithm performance, which makes a total of twenty test cases. All radios of devices under test were powered off during the testing procedure to comply with aviation safety regulations. The devices stored logs and sensor data internally for further algorithm development, analysis, and simulation purposes. This section will present a summary of one test case from each flight. All of the test cases generally consisted of the following events, which are also highlighted in the table 6.2 against the timestamps of respective events from each flight:

- Powering on the device under test
- Turning the aircraft's engines on
- Aircraft getting reverse pulled from hangar, especially when it is a large aircraft like of Boeing 777
- Aircraft taxing on the tarmac
- Aircraft engines maxed out at full throttle
- Aircraft lifting off the ground
- The asset tracking device detecting the lift-off

Table 6.2 contains the data from four test cases where the takeoff event was present in

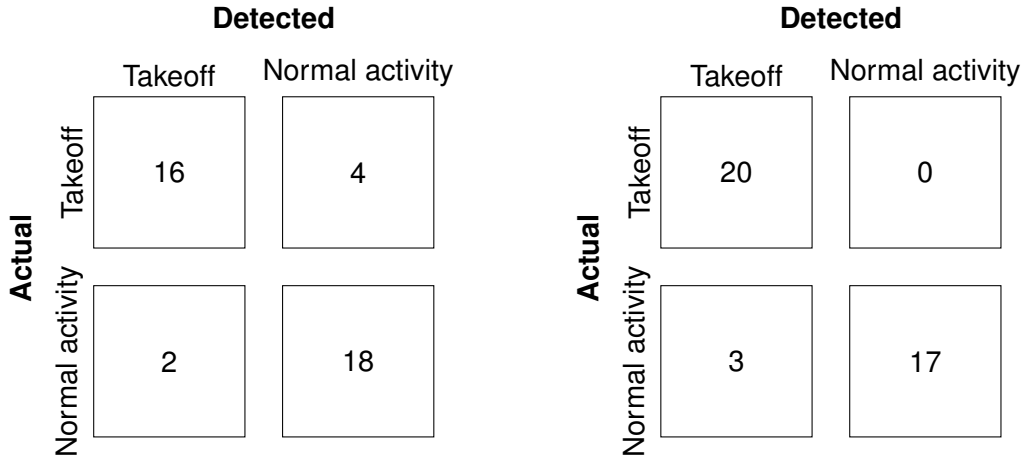
Event	Flight 1*	Flight 2*	Flight 3**	Flight 4**
DUT Powered on	09:53	23:22	01:32	11:50
Engines on	10:19	23:45	01:56	12:12
Reverse	10:34	NA	NA	12:19
Taxi start	10:38	23:48	02:58	12:22
Full throttle	10:50	23:53	02:02	12:27
Lift off	10:50	23:53	02:03	12:28
Detection	10:50	23:53	02:03	12:28

Table 6.2. *Time of the events during test flights. Date: 10.01.2022* and 11.01.2022***

the dataset. The data was labelled with different events while data collection and the detection time is shown in the table for the four flights. Each flight used a different aircraft to ensure that the flight mode algorithm could be tested with various aircraft. The following aircraft were used to perform the testing:

- Flight 1: Airbus A321-231
- Flight 2: ATR72-200
- Flight 3: ATR42-300
- Flight 4: Airbus A320-214

The takeoff was detected in all test flights data. However, the data from normal activities is also important to evaluate the algorithm performance. Twenty datasets of normal activities were collected which included no takeoffs. The algorithm was evaluated against these cases as well and the individual confusion matrices of accelerometer- and barometer-based takeoff detectors are provided in figure 6.6



(a) *Accelerometer based takeoff detector* ***(b)*** *Barometer based takeoff detector*

Figure 6.6. *Confusion matrices of takeoff algorithms.*

The devices in different test cases were placed in different locations in the aircraft, in-

cluding overhead compartments, cargo hold, cockpit, etc. The results showed that the placement of the device in the aircraft does not affect the algorithm performance. The takeoff was detected within the acceptable margin of error in all twenty test cases. The algorithm development targeted to detect the takeoff within a minute of actual aircraft liftoff. Table 6.2 shows the timestamps of actual liftoff and algorithm detection of one device per flight.

No false negatives were detected during this round of testing due to the interoperability principles discussed in section 5.3. However, the overall algorithm is found to be prone to false positives. The algorithm can report the false take-off detections due to mainly two factors. The first cause is to artificially have enough change in the air pressure that satisfies all the takeoff requirements of the barometer-based take-off detection algorithm e.g. while quickly ascending some steep hill. The second cause is relatively more complicated and it is caused by the offsets in the accelerometer data. The accelerometer axis reports a different acceleration value when axes are not offset compensated as discussed earlier. This generates false data for the algorithm in some devices so that algorithm detects increased acceleration when the device is rotated. If the rotation also follows the detection rules for the change in pitch angle then the algorithm reports a false take-off detection i.e. false positive. The false positives are harmful for the use case of the asset tracking device in the way that false detections can put device offline until the device gets out of the flight mode.

7. CONCLUSION

This thesis aimed to study the use of the accelerometer in the shipping industry and aircraft mobility. The shipping industry is a multi-billion dollar industry where millions of parcels are shipped every day. This thesis aimed to streamline and improve shipping processes by employing modern technology in the shipping processes. Sensor data processing techniques were implemented for various use cases, which were deployed on an IoT device in this thesis. The IoT device is attached to the asset to be shipped, and it reports various kinds of events along the journey. There were mainly three use cases that were covered by this thesis.

One use case was to detect when the asset has experienced a shock that can potentially break the asset entirely or partially. An algorithm was proposed in section 3.1 based on the idea of thresholding the accelerometer values. This algorithm generates an event for the stakeholders, and the stakeholders can know in advance whether to expect a broken asset at the destination and take mitigatory measures ahead of time. The exact accuracy of reported shock values is unknown and cannot be found out without a specialized lab equipment. This is due to the fact that a shock of a deterministic g-force cannot be applied to the device in the real-world conditions, and without specialized equipment. However, the shock value reported by the device is observed to be generally inline with the applied shock intensity. The exact accuracy might found out with the help of laboratory testing in the future if required.

Another use case was to distinguish between device's state of motion from state of rest. From the journey tracking point of view, stakeholders are usually not interested in the regular position updates when the asset is not moving. An algorithm was proposed in section 3.2 which detects whether the device is moving. The algorithm resulted in an 92.2% accuracy for movement detection. Based on this information, the position updates and power consumption can be optimized in the device.

The final use case was to detect the takeoff of the aircrafts. The aviation industry heavily regulates the usage of radio communication in airplanes. A relevant regulation by IATA states that the cellular modems must be powered off during the flight. For the unattended devices in the aircraft, the regulations state that the devices need to have an automated algorithm to turn off the cellular communication when the aircraft is in the air. This thesis

proposes an algorithm that is in line with these regulations.

The IATA regulation states that the unattended system aboard the aircraft must have two independent algorithms to trigger the takeoff detection. The proposed algorithms are described in chapter 5. The algorithm to detect takeoff using acceleration data uses the principle of detecting the sustained increase in acceleration and change in pitch angle. Section 5.3 specifies how the independent algorithms based on accelerometer and barometer inter-operate with each other. When any algorithm raises its takeoff detection flag, the asset tracking device is put to flight mode. In other words, the device can be put to flight mode independently by any algorithm, which complies with the IATA regulations. The overall takeoff detection algorithm produces the accuracy of 100% to detect takeoffs where the individual performance of the algorithms can still be improved. The overall positive detection, including true positives and true negatives, accuracy was found to be 92.5$\%$ which means the algorithms may also be improved in terms of true negative detections as well.

The accelerometer data processing has its challenges which are discussed in detail in section 2.1. Noise and offset in the accelerometer data are two leading causes of producing anomalous data, which is consequently propagated to algorithm results as well. Noise filtration of accelerometer data can be studied in detail to improve the algorithms described in the thesis. The limited available compute power in the embedded device is a challenging factor while employing higher order filters for the noise filtration from the accelerometer data. Moreover, methods for offset correction for the individual axis of the accelerometer is another area where this system can use improvement. In an ideal world with ideal accelerometer data, velocity can be computed by taking the integral of acceleration, which can be an important piece of information for analyzing aircraft takeoff velocities. If a database of geo-fences of airports can be developed, then by having an additional condition that the device must be in the vicinity of an airport before raising the take-off detection flag can reduce the false positives from the take-off detection system. If the system constraints can allow the deployment of more computationally expensive algorithms then artificial intelligence and machine learning based solutions can also be considered. However, it is not feasible to deploy any computationally complex algorithms in embedded IoT system that was targeted by this thesis which has a limited compute power and strict requirements of long-lasting battery life. The next generation of hardware may be able to handle AI-based algorithms better. Lastly, the optimal values for the proposed algorithm parameters can be learned using the statistical data analysis and machine learning algorithms. Doing so will ideally result in a superior quality algorithms and result in a better confusion matrix.

This thesis targeted algorithm development on an asset tracking device. These algorithms added more intelligence to the device by making it more 'motion aware'. Shipping companies can ship assets and can use any means of transport to ship the assets from

the origin to the destination. One transportation method is air cargo, where the asset tracking device could not operate due to aviation regulations. This limitation reduced the use of the tracking device to non-air cargo shipments which also resulted in extra work of carefully analyzing the intended route of the shipment before sending it. The work in this thesis proposed an algorithm for aircraft takeoff detection, which can be deployed in the asset tracking devices to place them with the assets unattended in the cargo hold of aircraft, thus removing a significant blockade in many use cases of the asset tracking.

REFERENCES

[1] Casaca, A. 2021 World of Shipping Portugal: An International Research Conference on Maritime Affairs Editorial. *Journal of Shipping and Trade* 6 (2021).

[2] Park, O. Shipping crisis disrupts industry. (2021): *The Corn & Soybean Digest*.

[3] Inkinen, T. and Hämäläinen, E. Reviewing Truck Logistics: Solutions for Achieving Low Emission Road Freight Transport. (2020).

[4] Kumar, S., Tiwari, P. and Zymbler, M. Internet of Things is a revolutionary approach for future technology enhancement: a review. *Journal of Big Data* (2019).

[5] Bouhaï, N. and Saleh, I. *Internet of Things : Evolutions and Innovations*. ISTE Ltd and John Wiley & Sons Inc., Nov. 29, 2017.

[6] *HERE supply chain*. URL: https://www.here.com/solutions/supply-chain (visited on 11/22/2022).

[7] *HERE asset tracking*. URL: https://www.here.com/products/asset-tracking (visited on 11/22/2022).

[8] *HERE shipment routing*. URL: https://www.here.com/platform/routing (visited on 11/22/2022).

[9] Interference from Electronic Devices. (). URL: https://www.boeing.com/commercial/aeromagazine/aero_10/interfere_textonly.html (visited on 11/22/2022).

[10] *Prohibition on airborne operation of cellular telephones*. Code of Federal Regulations. URL: https://www.ecfr.gov/current/title-47/chapter-I/subchapter-B/part-22/subpart-H/section-22.925 (visited on 11/22/2022).

[11] *2021 Guidance Document – Battery Powered Cargo Tracking Devices / Data Loggers*. Version 1. IATA. 2021. URL: https://www.iata.org/contentassets/6d7404d9ccca4e4e9c4ce146e4a2acb1/lithium-battery-guidance-document-2021-for-pharma-en.pdf (visited on 11/22/2022).

[12] *DO-160, Environmental Conditions and Test Procedures for Airborne Equipment*. RTCA. URL: https://do160.org/ (visited on 11/22/2022).

[13] Niu, W., Fang, L., Xu, L., Li, X., Huo, R., Guo, D. and Qi, Z. Summary of Research Status and Application of MEMS Accelerometers. (2018).

[14] *A MEMS sensor the size of the tip of your little finger can achieve atomic-level measurement accuracy*. URL: https://muratafinland.com/a-mems-sensor-the-size-of-the-tip-of-your-little-finger-can-achieve-atomic-level-measurement-accuracy/?lang=en (visited on 11/22/2022).

[15] Ohnhäuser, F. *Analog-digital converters for industrial applications including an introduction to digital-analog converters*. Springer, 2015. ISBN: 978-3-662-47019-0.

[16] Bernal-Polo, P. and Martínez-Barberá, H. Triaxial Sensor Calibration: A Prototype for Accelerometer and Gyroscope Calibration. (2017).

[17] Glueck, M., Oshinubi, D., Schopp, P. and Manoli, Y. Real-Time Autocalibration of MEMS Accelerometers. (2014).

[18] Zhang, Z.-Q. Two-Step Calibration Methods for Miniature Inertial and Magnetic Sensor Units. (2015).

[19] Hordon, R. M. Siberian (Asiatic) High. *Encyclopedia of World Climatology*. Ed. by J. E. Oliver. Dordrecht: Springer Netherlands, 2005, pp. 656–657. ISBN: 978-1-4020-3266-0. DOI: 10.1007/1-4020-3266-8_185. URL: https://doi.org/10.1007/1-4020-3266-8_185.

[20] Ghanghermeh, A., Roshan, G. and Shahkooeei, E. Evaluation of the effect of Siberia's high pressure extension on daily minimum temperature changes in Iran. (2015): *Modeling Earth Systems and Environment*.

[21] Bao, L. and Intille, S. S. Activity Recognition from User-Annotated Acceleration Data. (2004).

[22] Wang, J., Chena, Y., Haoc, S., Penga, X. and Hu, L. Deep Learning for Sensor-based Activity Recognition: A Survey. (2017).

[23] Mimouna, A. and Khalifa, A. B. A Survey of Human Action Recognition using Accelerometer Data. (2021).

[24] Liu, M., Liu, J., Ren, J., Liu, L., Chen, R. and Li, Y. Bacterial community in commercial airliner cabins in China. *International Journal of Environmental Health Research 2020* 30.3 (2019), pp. 284–295.

[25] Schuchardt, S., Koch, W. and Rosenberger, W. Cabin air quality – Quantitative comparison of volatile air contaminants at different flight phases during 177 commercial flights. *Building and Environment* 148 (2019), pp. 498–507.

[26] Rosenberger, W. Effect of charcoal equipped HEPA filters on cabin air quality in aircraft. A case study including smell event related in-flight measurements. *Building and Environment* 143 (2018), pp. 358–365.

[27] Tawk, Y., Jovanovic, A., Tomé, P., Leclère, J., Botteron, C., Farine, P.-A., Riem-Vis, R. and Spaeth, B. A New Movement Recognition Technique for Flight Mode Detection. *International Journal of Vehicular Technology* 2013 (2013).

[28] Riem-Vis, R., Späth, B., Tome, P., Tawk, Y., Jovanovic, A., Leclere, J. and Botteron, C. Automatic flight mode. 2013. URL: https://patents.google.com/patent/WO2013044399A1/en (visited on 11/22/2021).

[29] Lemmon, A. N., Skaaksrud, O.-P. and Jacobs, J. R. System and method for management of wireless devices aboard an aircraft. 2015. URL: https://pdfstore.patentorder.com/pdf/us/001/us9095001.pdf.

[30] Sheynblat, L. and Scibert, C. A. Altitude-dependent power management. 2012. URL: https://pdfstore.patentorder.com/pdf/us/108/us8195108.pdf.

[31] Tanabe, S. and Morita, H. Mobile electronic device, control method, and non-transitory storage medium. 2018. URL: https://pdfstore.patentorder.com/pdf/us/324/us9992324.pdf.

[32] Carlson, C. R., Pisharody, G., Khanna, R. and Kumar, A. G. Managing transmissions for a wireless sensor network during air transport. 2021. URL: https://pdfstore.patentorder.com/pdf/us/505/us10897505.pdf.

[33] Turon, M. A. and Liccardo, D. S. Real-time aircraft status detection system and method. 2016. URL: https://pdfstore.patentorder.com/pdf/us/683/us9279683.pdf.

[34] Aero 2009 Q2. (). URL: https://www.boeing.com/commercial/aeromagazine/articles/qtr_02_09/index.html (visited on 11/22/2022).

[35] Smith, S. W. *The Scientist & Engineer's Guide to Digital Signal Processing*. California Technical Publishing. ISBN: 0-9660176-7-6. URL: https://www.analog.com/en/education/education-library/scientist_engineers_guide.html.

[36] Kalman, R. A New Approach to Linear Filtering and Prediction Problems. (1960).

[37] Welch, G. and Bishop, G. An Introduction to the Kalman Filter. (2006).

[38] Frey, B. B. *The SAGE Encyclopedia of Educational Research, Measurement, and Evaluation*. Thousand Oaks: SAGE Publications, Incorporated. ISBN: 1506326153.

[39] Henderson, D. Euler Angles, Quaternions and Transformation Matrices. (1977). URL: https://ntrs.nasa.gov/api/citations/19770024290/downloads/19770024290.pdf (visited on 11/22/2022).

[40] Lente, G. and Ősz, K. Barometric formulas: various derivations and comparisons to environmentally relevant observations. (2020). URL: https://link.springer.com/article/10.1007/s40828-020-0111-6 (visited on 11/22/2022).

[41] Bagshaw, M. Commercial aircraft cabin altitude. (2007).