STRIKE ME AND GET MARKED: THE CONCEPTUAL, ANALYTICAL AND SIMULATION MODEL OF AN ADD-ON MODULE TO DETER QUADROTOR STRIKES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

OZAN ŞENYAYLA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

MAY 2020

Approval of the thesis:

STRIKE ME AND GET MARKED: THE CONCEPTUAL, ANALYTICAL AND SIMULATION MODEL OF AN ADD-ON MODULE TO DETER QUADROTOR STRIKES

submitted by **OZAN ŞENYAYLA** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Assoc. Prof. Dr. Erol Şahin
Supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Göktürk Üçoluk
Computer Engineering, METU _____

Assoc. Prof. Dr. Erol Şahin
Computer Engineering, METU _____

Assist. Prof. Dr. Kutluk Bilge Arıkan
Mechanical Engineering, TED University _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Ozan Şenyayla

Signature :

ABSTRACT

STRIKE ME AND GET MARKED: THE CONCEPTUAL, ANALYTICAL AND SIMULATION MODEL OF AN ADD-ON MODULE TO DETER QUADROTOR STRIKES

Şenyayla, Ozan
M.S., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Erol Şahin

MAY 2020, 65 pages

Mini and Micro Unmanned Aerial Vehicle (mUAV) platforms are gaining popularity in low-altitude strike/defense systems due to their low cost, low visibility, and capability to operate as swarms. These platforms, often touted for their low cost and visibility, have come a long way in terms of capabilities, are regularly used to provide eye-in-the-sky support for ground troops, and are on their way towards being deployed as swarms for dangerous missions where the survival rate is low. Despite the hype, these platforms are vulnerable to attacks from the ground and can be brought down easily by snipers.

In this thesis, we propose a "strike me and get marked" (SMAGM) module for quadrotor UAVs that can localize the position of a sniper after being shot. The module is inspired by the defensive behavior of honeybees, where killing a bee releases chemicals that mark the killer as a target. Equipped with such a defense coordination mechanism, bees can deter their enemies since the more bees one kills, the larger the target it becomes, by drawing the wrath of the swarm on itself.

The module, envisioned as a stand-alone unit affixed on the UAV platform, can broadcast the position of the sniper to other forces on the ground or air. Such a competence would deter potential attackers from shooting UAVs since such an action would mark their position as targets. Upon being shot by a bullet, the module takes readings from its Inertial Measurement Unit (IMU) sensor to compute the instantaneous force vector at the time of the impact and uses this information to localize the position of the sniper. Towards this end, this thesis proposed a method to estimate the force vector generated by the impact of the bullet using disturbance estimation methods and then uses the force vector and the altitude of the UAV to localize the position of the sniper. The methods are verified and systematically evaluated on a Quadrotor model simulated in the Gazebo simulator.

Keywords: Swarm, UAV, Quadrotors, Target, Detection

ÖZ

SALDIR BANA VE İŞARETLEN: QUADROTORA YAPILAN SALDIRIYI TESPİT EDEBİLEN BİR MODÜL EKLENTİSİ İÇİN KONSEPT, ANALİTİK VE SİMÜLASYON MODELİ

Şenyayla, Ozan
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Erol Şahin

Mayıs 2020 , 65 sayfa

Sürü zekâsı ile hareket edebilen, düşük maliyetli, düşük görünürlüğe sahip, alçak irtifa, kısa menzilli mini ve mikro insansız hava araçları(İHA), sahip oldukları bu özellikler sebebiyle yeni nesil alçak irtifa savunma/saldırı sistemlerinde daha çok tercih edilmeye başlanmıştır. Bu İHA'lar yerdeki askerlerin gözü olduğu gibi operasyonel olarak riskli bölgelerde sürü halinde kullanılmaya başlanmıştır. Tüm bunlara rağmen, mini İHA'lar yerden gelebilecek tehlikelere karşı savunmasız kalmakta ve nişancılar tarafından kolayca düşürülebilmektedirler.

Bu tezde, döner-kanatlı İHA'lar (ing: quadrotor) için düşürüldükten sonra nişancının yerini tespit edebilen "saldır bana ve işaretlen" (ing: SMAGM) modülü önerilmektedir. Bu çalışmada öldürüldüğünde kimyasal salgılayarak saldırganı işaretleyen bal arılarının savunma davranışından esinlenilmiştir. Bu türde bir savunma koordinasyon mekanizması ile, arı sürüsünün geri kalanı düşmanlarını tespit edebilir ve saldırgan ne kadar çok arı öldürürse sürü için daha belirgin hale gelir.

Bu modül, İHA sisteminden bağımsız olarak çalışabilir ve hedefi tespit ettikten sonra bir süre yerdeki nişancının konum bilgisini yayınlayabilecek olarak tasarlanmıştır. Bu tür bir yetenek ile nişancı, İHA'yı vurarak aslında kendi yerini işaretlemiş olur. İHA sistemi üzerine isabet eden mermi veya saçma parçalarının yarattacağı anlık kuvvet vektörü ataletsel ölçüm birimi (ing: IMU) algılayıcısından alınan veriler ile hesaplanır ve bir dizi filtreden ve değerlendirmeden geçirildikten sonra nişancının yönü tespit edilir. Sonrasında alınan konum ve irtifa verileri ile nişancının kesin yeri tespit edilir. Bu yöntemler Gazebo simülatöründe koşturulan döner-kanatlı İHA modeli üzerinde sınanmış ve doğrulanmıştır.

Anahtar Kelimeler: Sürü, İHA, Drone, Hedef, Tespit

Türk Milletinin Kahraman Evlatlarına,

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisor Assoc.Prof.Dr. Erol Şahin for his vision, encouragement, and support in this thesis. I also express my gratitude to Assist. Prof. Dr. Ali Emre Turgut and Assist. Prof. Dr. Kutluk Bilge Arıkan for their constructive feedback and suggestions. I must also express my appreciation to ROKETSAN Inc. for supporting my academic studies. I would like to thank all my helpful and supportive colleagues and friends in the Avionic Technologies Unit, especially Caner Yiğit, Fatih Yeşilırmak, Dr. Ozan Köroğlu, Necati Çağan and Muhammet Fatih Kalyon.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	
SMAGM	Strike Me And Get Marked
UAV	Unmanned Aerial Vehicle
MCU	Microcontroller Unit
mUAV	Mini and Micro Unmanned Aerial Vehicle
sUAS	Small Unmanned Aircraft Systems
MEDEVAC	Medical Evacuation
C3	Command, Control, and Communication
2D	2 Dimensional
3D	3 Dimensional
ML	Machine Learning
CW	Clockwise
CCW	Counter Clockwise
MEMS	Micro Electro Mechanical Systems
IMU	Inertial Measurement Unit
GPS	Global Positioning System
RF	Radio Frequency
LPF	Low-Pass filter
$\dot{x}$	X Axis Linear Acceleration
$\dot{y}$	Y Axis Linear Acceleration
$\dot{z}$	Z Axis Linear Acceleration
ω	Angular Velocity
$\dot{\omega}$	Angular Acceleration
θ	Roll Angle

ϕ	Pitch Angle
ψ	Yaw Angle
MoI	Moment of Inertia
ToI	Tensor of Inertia
CoM	Center of Mass
CoG	Center of Gravity
DOF	Degrees of Freedom
ROS	Robot Operating System
mt	Meter
ms	Millisecond
s	Second
IDM	Inverse Dynamics Model
RBD	Rigid Body Dynamics
RPM	Revolutions Per Minute
Hz	Hertz
N	Newton

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

Unmanned Aerial Vehicles (UAVs) are aerial vehicles that can operate without an on-board human pilot and come in different sizes. Recently, the use of Mini and micro types of UAVs (mUAVs) have gained popularity in defense applications due to their low visibility, low cost, and capability to operate as swarms[10]. mUAVs are frequently used as *eye-in-the-sky* for surveillance missions in which they relay back the bird's eye of the battleground to the troops on the ground. Operating as swarms, these platforms can cover large regions, or urban environments where the line-of-sight view is limited. Despite these advantages, mUAVs are defenseless to attacks from adversaries on the ground, since they fly at a relatively low altitude with low velocity in comparison to other UAVs and can be easily shot down by snipers.

The primary motivation of this thesis is to provide a "Strike Me And Get Marked " (SMAGM) module for mUAVs to localize the position of a sniper after being shot down. The inspiration for the module comes from honeybee swarms that are known to deter attacks coming from its enemies as a swarm. The honeybees, when they are attacked or killed, are known to mark their enemies by releasing a chemical scent [15] that would attract other members of the swarm to locate and attack the target [16]. Such a swarm strategy would deter a potential attack from its enemies to its tiny individuals member equipped with tiny stings.

The interest in the development and deployment of mUAV swarms on the field has focused on the development of centralized and decentralized coordination among the platforms, to address problems such as "flocking" (moving together towards a com-

mon goal)[17], automatic partitioning[18], and coverage of areas for surveillance[19]. However, carrying out such tasks with adversaries on the ground is not realistic. mUAV platforms are essentially “sitting ducks" for these adversaries and can be shot down easily by snipers. Automatic detection and broadcasting of the position of snipers, by shot-down mUAVs, would make the swarm UAV concept one step closer to deployment on the field. The conceptual development of developing such a module is novel and is the primary contribution of this thesis.

1.2 OBJECTIVE OF THE THESIS

This thesis proposes the conceptual and theoretical proof-of-concept for developing a stand-alone module that can be affixed on mUAVs, which can act as a deterrent to attacks coming from the enemy. Specifically, the module is designed to estimate the position of the sniper using readings from its on-board sensors to compute the instantaneous force vector at the time of the impact and uses this information to localize the position of the sniper. Towards this end, this thesis proposes a method to estimate the force vector generated by the impact of the bullet using disturbance estimation methods and then uses this vector and the altitude of the UAV to localize the position of the sniper. The methods are verified and systematically evaluated on a quadrotor model simulated in the Gazebo simulator[20].

A sample scenario in which the deployment of SMAGM modules on mUAVs would benefit is depicted in Figures 1.1.

- A swarm of quadrotors and ground forces ventures into the enemy territory.
- A sniper shoots down three quadrotors.
- On two of the shot-down quadrotors, the SMAGM modules computes the position of the sniper, starts broadcasting them.
- On one of the shot-down quadrotors, the SMAGM module is damaged, and hence it fails to compute and broadcast the sniper’s position.
- An alert is generated for the rest of the forces causing an update on the task as shown in Figure 1.1-left.

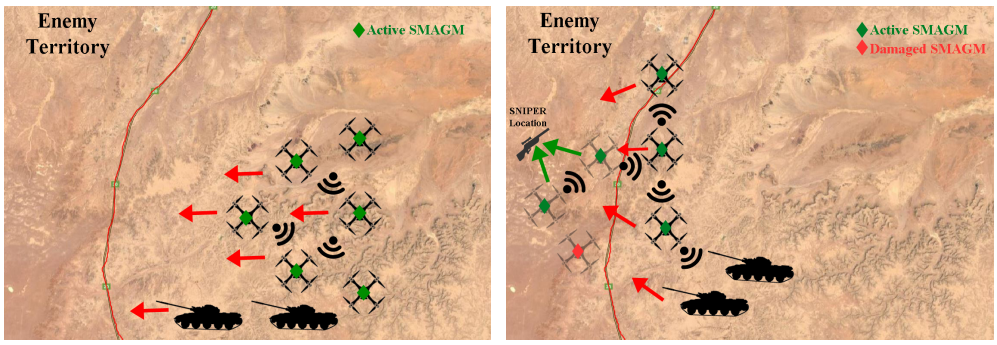


Figure 1.1: Left: A quadrotor swarm along with ground forces, enter the enemy territory. The green diamond shown on the quadrotors indicates active SMAGM modules. Right: Three of the mUAVs are shot down by a sniper. The shot-down quadrotors are shown in gray. The red diamond on the shot-down quadrotor indicates that the SMAGM module is also damaged. The green diamonds on the shot-down quadrotors (shown in gray) indicate that they have successfully computed the location of the sniper and have started to broadcast the information.

1.3 SCOPE OF THE THESIS

In order to achieve its mission, the SMAGM module should be designed as a separate and stand-alone unit in terms of sensing, computation, energy, and communication in a rugged package that can be affixed on the mUAV. Towards this end, the SMAGM module should include:

- an IMU (Inertial Measurement Unit) sensor, to sense the impact force vector at the time of the impact,
- a GNSS/GPS sensor to localize the mUAV at the time of the impact,
- a barometer sensor to estimate the altitude of the mUAV at the time of the impact,
- a magnetometer to compute the attitude of the mUAV with respect to North,
- a wireless tachometer to read the rotor angular velocities at the time of the impact,

- a Microcontroller Unit (MCU) to read the sensor values at the time of the impact and compute the estimated location of the sniper,
- a battery, for energy autonomy, and
- an RF (such as WiFi) Transmitter to broadcast the estimated location of the sniper to others.

1.4 OUTLINE OF THE THESIS

The outline of this thesis is given as follows. In Chapter 2, UAV systems are explained with their definition, basic history, and future expectations. Furthermore, UAV systems are categorized with respect to their capabilities and aerial forms. In addition, quadrotors, mainly focused on UAV type in this thesis, are elaborated with their dynamics. Moreover, the IMU sensor and referenced coordinate frames are detailed in this chapter. Finally, the disturbance estimation concept is examined with analyzing of related studies.

In Chapter 3, the proposed SMAGM module is clarified with their all working flow. Initially, impact force estimation is explained starting from 2D space applications and the proposed approach is systematically verified in the simulation environment. Furthermore, the proposed methodology for impact force estimation is amplified with its sub-processes and verified under different test cases in the simulation environment. Finally, sniper localization from the estimated impact force vector is explained and verified in the same simulation environment with different test scenarios. Broadcasting the estimated location left at a conceptual level.

In Chapter 4, a summary of the work done for the thesis is given with contributions and suggestions for future work.

CHAPTER 2

BACKGROUND INFORMATION

This chapter gives background material about UAVs and especially quadrotors respectively. Furthermore, this chapter gives information about IMU sensors, which is the core sensor of the SMAGM module, and three coordinate frames that are used in the proposed model briefly. Finally, disturbance estimation concept and related studies examined.

2.1 UNMANNED AERIAL VEHICLE SYSTEMS

UAVs are aerial vehicles that can operate predefined autonomous tasks or without an on-board human pilot who controls it remotely using a hand terminal or satellite communication in more advanced ones.

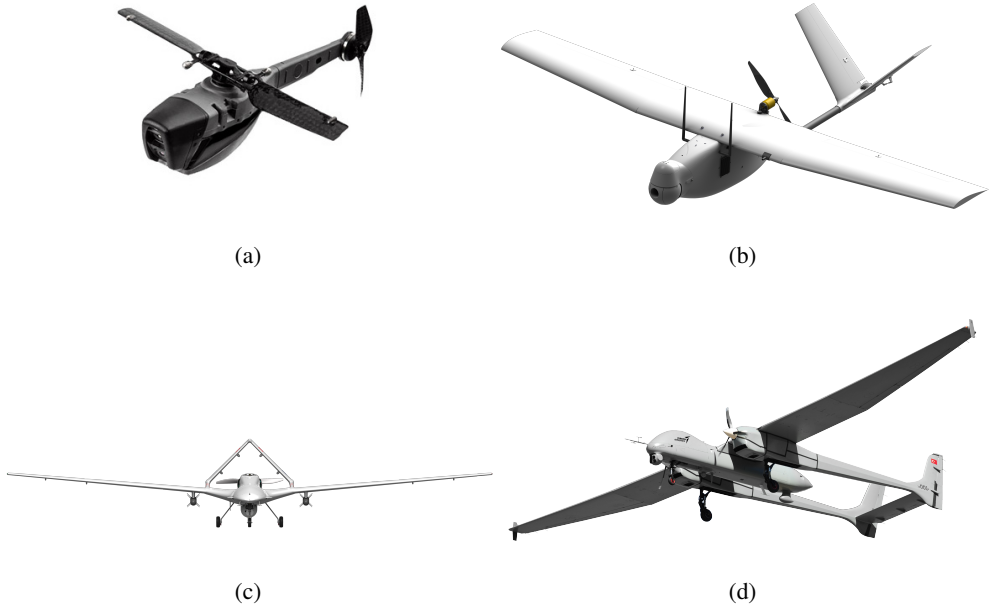
The primitive version of UAVs is balloons these are loaded with explosives in 1849 by the Austrians to attack Venice [21]. It is evident that at the beginning, it was not a commercial or hobby device. From the 1849's to these days, the concept of pilotless aerial vehicles has been changed dramatically. At the earliest ages of UAVs, they were very costly. However, today, they are ubiquitous thanks to advances in power technology, MCU's, Micro-Electro-Mechanical Systems (MEMS), and control methodologies in aerodynamics. These advances also make them cheaper in product and expandable. Today, UAVs are used for both military and civilian benefits, and they are grateful for their popularity to nonmilitary usage.

In this thesis, we focus on its military-based application areas. In these applications, UAVs are mainly used as an *eye-in-the-sky* platform for surveillance in operations

using on-board cameras due to being appropriate for operations both indoor and outdoor. Commonly, these systems are becoming the main operation alternative in fields that are risky for troops.

Used technology in UAV system is nearly related to its payload capacity. In mini to tactical-sized UAV systems, optical cameras and related target detection subsystems are mostly used[22]. Larger UAVs can carry more payloads and may be equipped with Radio Frequency (RF) based sensors and solutions. But they are quite expensive for swarm UAV systems.

UAV platforms are classified into five major categories based on the mass of the system, operating altitude and operating area, as shown in Table 2.1 and in Figure 2.1, Micro UAV, Mini UAV, Tactical UAV, and MALE UAV examples are depicted respectively.



(a) (b)

(c) (d)

Figure 2.1: (a)Micro UAV: Black Hornet PRS 1:3 ratio (taken from [4]), (b)Mini UAV: Bayraktar Mini İHA 1:40 ratio (taken from [5]), (c)Tactical UAV: Bayraktar TB2 1:240 ratio (taken from [6]), (d)MALE UAV: ANKA-Aksungur 1:400 ratio (taken from [7]) examples.

Table 2.1: UAV classification based on mass, operating altitude and area (adopted from [1])

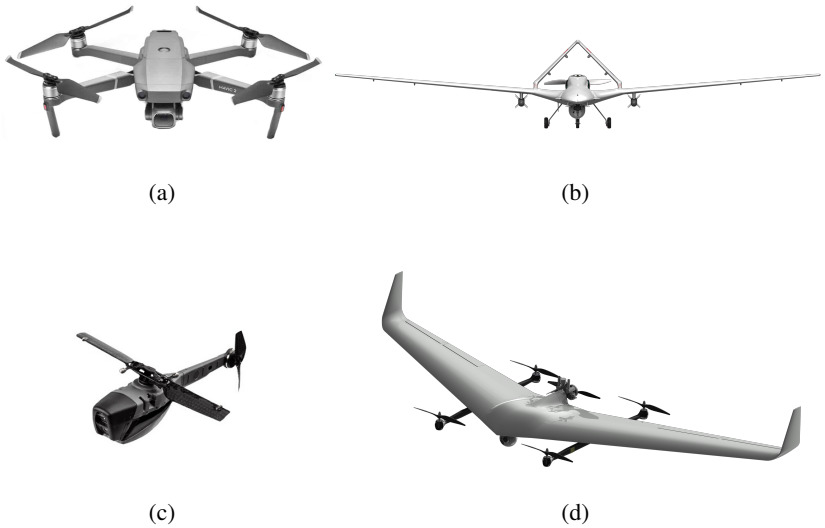
Class	**Mass Range**	**Operating Altitude**	**Operating Area**
Micro	Less than 1 kg	Near-Surface to 150 mt	Local
Mini	1 kg to 15 kg	30 mt to 3 km	Local
Tactical	15 kg to 450 kg	450 mt to 5.5 km	Regional
MALE	450 kg to 13 tons	5.5 km to 18 km	Regional/National
HALE	450 kg to 13 tons	Above 18 km	National/International

UAVs are also classified based on the type of aerial platform used; Multi Rotor, Fixed Wing, Single Rotor Helicopter, and Fixed Wing Hybrid Vertical Take-Off Landing (VTOL). The pros and cons of these types are summarized in Table 2.2.

Table 2.2: UAV classification based on the aerial platform type (adopted from [2])

	Pros	**Cons**
Multi-Rotor	Accessibility Easy to Control VTOL and Hover Flight	Short Flight Duration Small Payload
Fixed-Wing	Long Endurance Fast Flight Speed Large Area Coverage	More Space Required No VTOL/Hovering More Training, Costly
Single-Rotor	VTOL and Hovering Long Endurance More Payload	More Dangerous More Training More Expensive
Fixed-Wing Hybrid	VTOL Long Endurance	Less Hovering Still Very Primitive

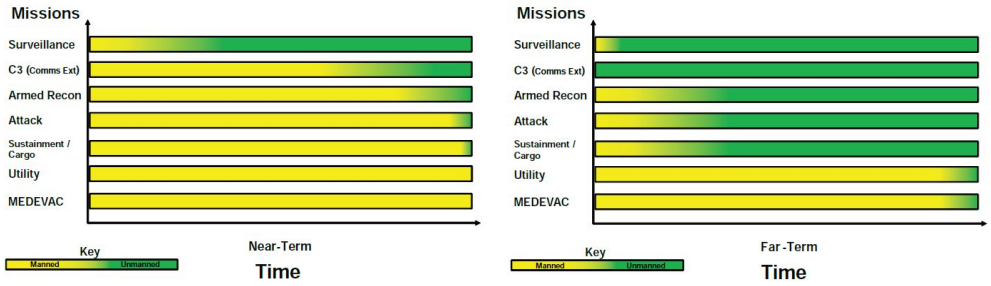
Examples for each platform category are depicted in Figure 2.2.



(a) (b)

(c) (d)

Figure 2.2: (a)Multi-Rotor (taken from[8]), (b)Fixed-Wing (taken from[5]), (c)Single rotor (taken from[4]), (d)Fixed-Wing Hybrid (taken from[9]) UAV examples.



(a) Near-Term (b) Far-Term

Figure 2.3: UAV usage expectations in some fields in near-term(a) and far-term(b) taken from [10]

Soon, for the different tasks, it is expected to be used unmanned systems instead of manned systems. In the literature, these kinds of expectations and visions mostly sourced from defense and military-based publications such as 'Eyes of the Army [10]' that states that the percentage of unmanned systems in surveillance is more than manned systems and this is visualized in Figure 2.3-a. In addition, Command, Control, and Communication(C3) for UAV systems are going to be vital. However,

according to the same resource, in the other tasks such as armed surveillance and strike are still weak. On the other hand, in the far-term that visualized in the right in Figure 2.3-b, most of the missions are expected to handle by unmanned systems except utility and medical evacuation (MEDEVAC) missions.

The recent concept in UAV systems is the swarm intelligence that has appealed to many researchers' attention. So, in the United States, it is announced that the passing of 14 CFR Part 107, which is a federal code of regulations for the commercial use of small unmanned aircraft systems (sUAS) in 2016[23]. It has been observed that some species exist in the brutal nature taking advantage of the power of being swarms, rather than being alone.[24] With the recent advances in artificial intelligence (AI) applications for UAV systems, complete a task as a UAV swarm are getting very common. Especially mUAVs are appropriate aerial platforms for swarm systems due to their high maneuverability, low cost and visibility. Acting as an mUAV swarm that has some advantages such as following[25];

- Swarm members can attack and defense coordinately.
- Information that a member of the swarm is gotten can be shared with the rest.
- Instead of all members have all skills, members of swarm have different skills, and they can protect the others with their powerful skills.
- Even some of the members are lost, the rest can continue to operate.
- Although their communication range is limited, one or more swarm members act as a bridge to extend the operational range.

2.2 QUADROTORS

Quadrotors, also called quadcopters, belong to multi-rotor class UAV systems and named after the 4 rotors placed at the end of the arms. These platforms are built-in micro- to mini sizes as relatively low cost platforms that can provide high maneuverability as well as hovering in place capability. These features allow these platforms to be deployed in eye-in-the-sky missions within regions where adversaries are present.

All quadrotor rotors produce thrust in the upward direction to fly. When all rotors generate equal thrust, depend on the mass of the quadrotor, it hovers, moves up or down in Z-axis. These movements can be achieved by changing the amount of thrust equally. The pose of the quadrotor is represented as the relative position and the orientation of the CoM (center of mass) of the platform in the inertial world coordinate frame. The orientation of aerial platforms is commonly represented as the Yaw-Pitch-Roll rotation model (YPR) with angles ψ, θ, ϕ about Z-Y-X axes as shown in Figure 2.4.

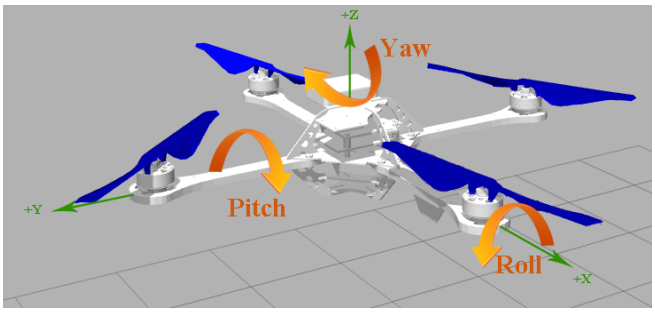
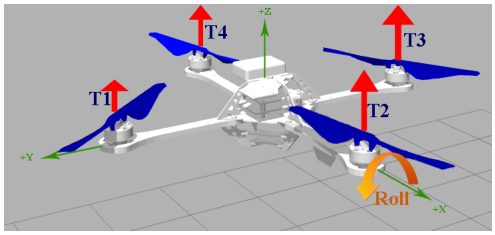
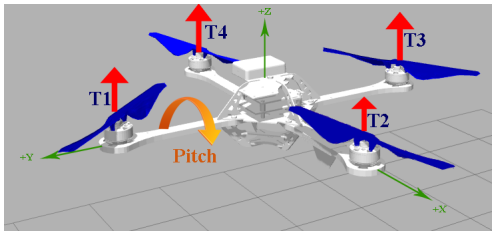


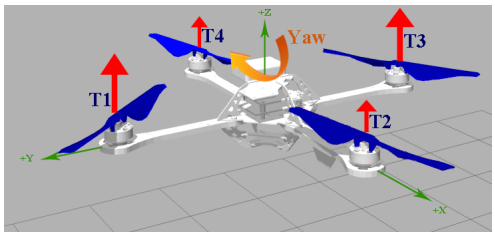
Figure 2.4: Yaw-Pitch-Roll (YPR) representation on a model quadrotor



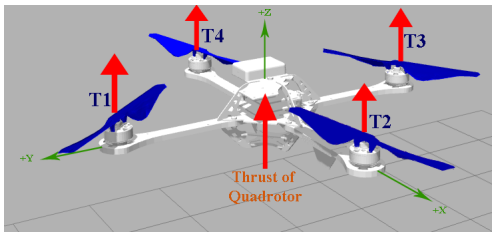
(a) T1>T3 or T3<T1



(b) T2>T4 or T4>T2



(c) T1,T3>T2,T4 or T2,T4>T1,T3



(d) T1+T2+T3+T4 > Weight of Quadrotor

Figure 2.5: (a)Roll, (b)pitch, (c)yaw and (d)vertical movements of the quadrotor generated by different thrust combinations of the rotors. The length of the arrows represent the magnitude of the thrust force vector generated by the corresponding rotor.

In quadrotors, adjacent motors rotate in opposite directions to counteract the moment generated by each other to control the rotation along the Z direction. Figure 2.5 shows how the orientation of the quadrotor changes under different thrust combinations where arrow lengths represent the generated thrust by rotors. The mathematical model of quadrotor dynamics[26] [27] [28] is built on the following assumptions:

- The quadrotor's body is rigid.
- Time latency of power transfer from body to motors is ignored.
- Moment of Inertia (MoI) of the quadrotor propellers is ignored.
- Tensor of Inertia (ToI) of the quadrotor is approximated as MoI of several objects.
- The quadrotor geometrical center of the quadrotor is also assumed to be its center of mass (CoM) and the center of gravity (CoG).

The thrust generated by the rotors depends on parameters such as humidity, air pressure, temperature, the aerodynamics of the propeller, friction, etcetera. Within the context of this thesis, the effects of all these parameters' are encapsulated in the *thrust coefficient* (c_t), which is estimated using the mean of the thrust during hovering. Hence, the thrust force T_i generated by the propeller *i* spinning with angular velocity (ω) is:

$$T_i = c_t \cdot {\omega_i}^2 \tag{2.1}$$

Hence, the total thrust force generated by all the rotors is:

$$T = \sum_{i=1}^{n} T_i \tag{2.2}$$

where n equals to the number of rotors.

The hub moment is the moment of a rotor and the propeller couple is formulated as:

$$H = c_h \cdot {\omega_p}^2$$

where (c_h) is the hub moment coefficient.

The equations of motion for the quadrotor for linear acceleration is given as:

$$m \cdot \ddot{x} = F_{T_x}$$
$$m \cdot \ddot{y} = F_{T_y}$$
$$m \cdot (\ddot{z} - g) = F_{T_z}$$

where F_T refers to total force acting on the particular axis and $\ddot{x}$, $\ddot{y}$ and $\ddot{z}$ are linear accelerations in corresponding axis.

The net moment (M_N) acting on the axis equals to inertia times angular acceleration for each axis as:

$$M_{N_x} = \dot{\omega_x} \cdot I_{xx}$$
$$M_{N_y} = \dot{\omega_y} \cdot I_{yy}$$
$$M_{N_z} = \dot{\omega_z} \cdot I_{zz}$$

where $\dot{\omega}_{x,y,z}$ denotes angular acceleration and $I_{xx,yy,zz}$ denotes the inertia along different axis.

The following equations of motion govern the movement of the quadrotor:

$$\begin{aligned}
\ddot{x} \cdot m &= (cos\phi \cdot sin\theta \cdot cos\psi + sin\phi \cdot sin\psi) \cdot T \\
\ddot{y} \cdot m &= (cos\phi \cdot sin\theta \cdot sin\psi - sin\phi \cdot cos\psi) \cdot T \\
(\ddot{z} - g) \cdot m &= (cos\theta \cdot cos\psi) \cdot T \\
\ddot{\phi}.I_{xx} &= l(T_2 - T_4) - (I_{zz} - I_{yy})\dot{\theta}\dot{\psi} \\
\ddot{\theta}.I_{yy} &= l(T_3 - T_1) - (I_{xx} - I_{zz})\dot{\phi}\dot{\psi} \\
\ddot{\psi}.I_{zz} &= (H_1 + H_3) - (H_2 + H_4) - (I_{yy} - I_{xx})\dot{\phi}\dot{\theta}
\end{aligned} \tag{2.3}$$

where ϕ, θ and ψ denotes Euler Angles, of rigid quadrotor body. T denote the total thrust generated by propellers. T_i denote thrust force and H_i denote hub moment of propeller i.

2.3 INERTIAL MEASUREMENT UNIT

The main sensor of the SMAGM module is the Inertial Measurement Unit (IMU), which measures linear acceleration and angular velocity of a system in motion using multiple gyroscopes and accelerometer sensors[29]. IMUs have broad usage areas such as cars, robots, mobile devices, commercial planes, missiles, rockets, and UAVs. High-accuracy IMUs using fiber optic gyros, ring laser gyros, or pendulous

accelerometers are costly and relatively burdensome. However, with the recent advances in technology, IMUs made using MEMS technology provide high accuracy sensing at an affordable price [30]. An IMU consisting of a perpendicular triad of accelerometers and gyroscopes is shown in Figure 2.6.

Figure 2.6: Multi Wafer hybrid integration: Rover IMU I taken from [11]

In this thesis, the Gazebo[20] simulator is used with a MEMS IMU sensor module with a built-in tunable noise level. The IMU sensor module has 100 Hz frequency and provides linear acceleration and angular velocity along 3 axes.

2.4 COORDINATE FRAMES

Three coordinate frames are used within the analysis: *earth inertial*, *vehicle inertial*, and *vehicle body fixed*. The vehicle body fixed frame is affixed to the CoM. The earth inertial frame is used as a reference to represent the pose of the body frame of the quadrotor. Specifically;

- Aerodynamic forces and moments such generated by propellers are represented within the vehicle body fixed frame.
- Module sensors and on-board sensors, such as the IMU, are measured within the vehicle body fixed frame.
- Global Positioning System (GPS) provides the location of the quadrotor within

the earth inertial frame.

- Newton's equations of motion are calculated within the vehicle body fixed frame.

Figure 2.7 depicts these frames.

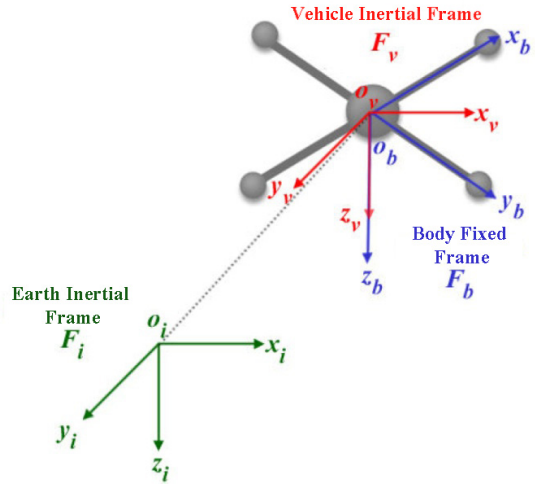


Figure 2.7: Coordinate frames adopted from [12]

2.4.1 Earth Inertial Frame (F_i) with x_i, y_i, z_i

Earth Inertial takes a reference to the earth as a coordinate system, and (X-axis) represents the North, (Y-axis) represents the east and (Z-axis) directed toward the center of the earth.

2.4.2 Vehicle Inertial Frame (F_v) with x_v, y_v, z_v

The CoM of the quadrotor is also the origin of this frame, and the axes are aligned with the axis of the Earth Inertial Frame[26]. This frame is used in sniper localization.

2.4.3 Vehicle Body Fixed Frame (F_b) with x_b, y_b, z_b

Same as inertial vehicle frame, the origin of the fixed body frame is also the center of mass of the quadrotor. However, X, Y, and Z axes are rotating with the body rotation

and along the quadrotor symmetry axis. The X-axis is accepted as the main motion direction, and for planes, it is the nose of the body. Y-axis is imagined as wings in planes, and Z-axis points out the belly[27].

With respect to the Figure 2.7, transformation from the vehicle inertial frame (*v*) to body fixed frame (*b*) is defined as:

$$\mathbf{R}_v^b(\phi,\theta,\psi) = R_x(\phi)R_y(\theta), R_z(\psi)$$

$$\mathbf{R}_v^b(\phi,\theta,\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{bmatrix} \begin{bmatrix} cos\phi & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix} \begin{bmatrix} cos\psi & sin\psi & 0 \\ -sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

$\mathbf{R}_v^b(\phi,\theta,\psi)$ is orthogonal and therefore the inverse of this matrix is equal to its transpose. Therefore, the transformation from the *Vehicle Body Fixed Frame* to *Inertial Frame* is shown in equation 4.5.

$$\mathbf{R}_b^v(\phi,\theta,\psi) = [\mathbf{R}_v^b(\phi,\theta,\psi)]^{-1} = [\mathbf{R}_v^b(\phi,\theta,\psi)]^T \quad (4.5)$$

2.5 DISTURBANCE ESTIMATION

The estimation of the impact force vector impinging upon the mUAV at the instant of a bullet's impact is essential for estimating the direction of the sniper. Such estimation is studied in the literature as *disturbance estimation* with a focus on estimating disturbances generates by external factors, such as sudden gusts of wind in the air[31] [32], or impact forces transmitted to the mUAV during aerial manipulation or transportation. These studies have mostly focused on estimating these disturbances in order to improve the stability of mUAVs by reacting properly to counteract them.

One of these external disturbance force estimation technique works by physically adding force sensors to the system. However, these sensors increase the cost of the system and cut down the manipulator's effective workload. Moreover, inappropriate mechanical integration of these sensors may lead to the entire control system to become unstable[33]. Due to these drawbacks, force estimation methods, such as model-based estimators, have been proposed. Here, "Model" refers to the inverse

dynamics model (IDM) that is derived from the rigid body dynamic (RBD) and identifies the base parameters in the model[34]. However, this model has some drawbacks, such as decreased accuracy due to non-linear effects such as friction. Although there are some drawbacks, supervised learning methods are other techniques that can offer better estimations on modeling non-linear effects[35].

Moreover, there are studies about momentum-based estimators of the external wrench and unmodelled dynamics for control systems such as [36]. In this study, the momentum of the system base is employed to compensate for disturbances effects such as wind. In the proposed model, the disturbance estimator requires both the attitude controller and thrust command that manipulates the propellers, and this model takes the feedback, actually the effect of the given inputs from the aerial vehicle.

Furthermore, in [37], the problem of external disturbance estimation upon aerial vehicles is handled by applying both linear and spin momentum dynamic equations. In addition, in the study on [38], an algorithm based on inertial measurement data and robot dynamics only is proposed in order to estimate wrench and collision. In this study, a closed-loop model that estimates external wrench using the pose, inertial data, motor speeds, and current data from the speed controllers. Moreover, the model needs the commanded control inputs and the torque and force. However, all of these studies, disturbance estimation is done inside the central controller and need control inputs and controller feedback. Then, they are not applicable for a proposed self-contained system that is affixed on top of an aerial vehicle without an extra modification on it.

Study on [39], the Momentum-based external force estimator requires the same inputs to the model with [36]. When the experimental results are examined, the estimation duration is around seconds. In the expected real case for hit by a bullet, the interaction of bullet and vehicle body is expected to be very short. Besides that, after on-board flight controller of aerial vehicle react due to unknown disturbance to stabilize the vehicle. Then, the proposed model has a short time to estimate the impact force. As a result, this study could not be applied to our problem.

Disturbance estimation techniques are applied to both aerial and ground vehicles. In [40], disturbance force estimation upon a car compares the steering wheel angle

with the yaw rate. However, this study does not takes into account collision and delay between steering wheel angle and front-wheel direction. In this study, precise estimation takes approximately 8 seconds that is not applicable for hitting quadrotor due to the possibility of losing rigidity in this duration.

Almost all of these studies have focused on the estimation of relatively mild disturbances that typically last for a relatively long time, then the problem that we address in this thesis. Different from those studies, we assume that the external disturbance acting on the mUAV is impulse-like, and provide catastrophic results on the mUAV. Under such an assumption, the proposed model focuses on the quick-and-dirty estimation of the impact force vector using relatively few readings collected at the time of the impact, since the dynamics model of the mUAV is likely to become invalid shortly after the impact.

CHAPTER 3

STRIKE ME AND GET MARKED

This chapter describes the conceptual and algorithmic design of the SMAGM module. The module, envisioned as a stand-alone unit affixed on the UAV platform, can broadcast the position of the sniper to the rest. Upon being shot by a bullet, the module takes readings from its Inertial Measurement Unit (IMU) sensor to compute the instantaneous force vector at the time of the impact and uses this information to localize the position of the sniper. Towards this end, this thesis proposed a method to estimate the force vector generated by the impact of the bullet using disturbance estimation methods that use the force vector, location and altitude of the UAV, and angle with truth north data to localize the position of the sniper. The methods are verified and systematically evaluated on a Quadrotor model simulated in the Gazebo[20] simulator.

The module consists of three main components:

- Impact Force Estimation: This component uses the Inertial Measurement Unit raw data and Wireless Tachometer sensor data. In this component, raw sensor data is filtered because of the natural existence of noise in signals. The outcome of this component is the estimated impact force vector.
- Snipe Localization: This component uses estimated impact force vector information from impact force estimator and estimates the location of the sniper using GPS, Barometer, and Magnetometer sensor data. The outcome of this component is the exact location of the sniper.
- Broadcasting: This component is provided at the conceptual level to indicate that the position of the sniper is broadcast through a form of RF communication.

SMAGM module estimation process depicted in Figure 3.1.

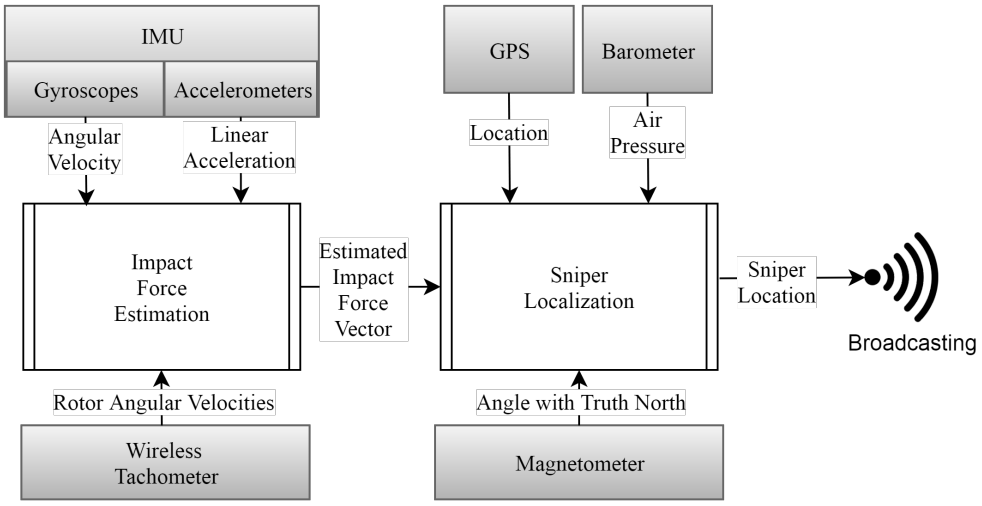


Figure 3.1: SMAGM module overview: from raw sensor data to the estimated sniper location information and broadcasting.

3.1 IMPACT FORCE ESTIMATION

In this section, we will first model the impact force estimation problem in a simple 2D setting and derive the equations for estimation before extending them to 3D to be applied to the quadrotor.

3.1.1 Estimation of Impact Force in 2D

In order to estimate impact force upon a rigid body in 2D space, all forces must be measured in specific periods, and other effects such as gravity, mass, and inertia must be known. With respect to Newton's first and second laws, in each step in a specific period, it is possible to calculate system dynamics such as linear acceleration and angular acceleration. In this calculation, it is possible to take into account an unknown impact force for each dimension, and the impact force or vector sum of forces is estimated. In these computations, IMU sensors are used to measure linear acceleration and angular velocity for each axis. In Figure 3.2, applied forces and system state is depicted.

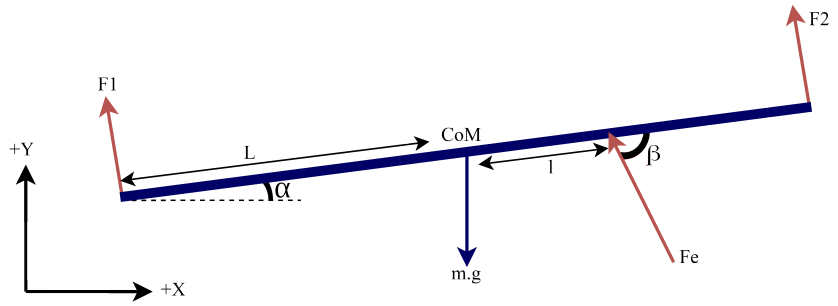


Figure 3.2: Applied forces representation upon 2D rigid body.

In Figure 3.2, F_e represents the unknown impact force, m · g represents gravitational force, F_1 and F_2 represent the thrust generated by rotors. The center of mass of the rigid body is depicted by CoM, and arm length is pointed by *L*. The body has α degree with X-axis. F_e applies force to the rigid body to *l* unit away from the CoM with β degree angle.

It is assumed that we can measure or already knew followings:

- F_1 and F_2: Thrust forces generated by motors,
- L: Distance from CoM to F_1 or F_2,
- α: Angle between the body and the X-axis,
- Linear Accelerations $\ddot{x}$ and $\ddot{y}$ by IMU,
- Angular velocity ω by IMU,
- Inertia of the rigid body I_{xy}, computed experimentally.

The net forces on each axis F_x and F_y can be written as:

$$\begin{aligned} F_x &= F_1 \cdot \sin\alpha + F_2 \cdot \sin\alpha + F_e \cdot \cos(\beta - \alpha) \\ F_y &= F_1 \cdot \cos\alpha + F_2 \cdot \cos\alpha + F_e \cdot \sin(\beta - \alpha) - (m \cdot g) \end{aligned} \tag{1.1}$$

whereas the net moment M_n, about the CoM in the clockwise direction is given by:

$$M_n = F_1 \cdot L - F_2 \cdot L - (F_e \cdot l \cdot \sin\beta) \tag{1.2}$$

The resulting linear accelerations on the body is then computed as:

$$\begin{aligned} \ddot{x} &= \frac{F_1 \cdot \sin\alpha + F_2 \cdot \sin\alpha + F_e \cdot \cos(\beta - \alpha)}{m} \\ \ddot{y} &= \frac{F_1 \cdot \cos\alpha + F_2 \cdot \cos\alpha + F_e \cdot \sin(\beta - \alpha) - (m \cdot g)}{m} \end{aligned} \tag{1.3}$$

Similarly, with $\dot{\omega} = \frac{M_n}{I_{xy}}$, with I_{xy} denoting the rotational inertial of the body, the angular acceleration generated by the moment is given by:

$$\dot{\omega} = \frac{F_1 \cdot L - F_2 \cdot L - (F_e \cdot l \cdot \sin\beta)}{I_{xy}} \tag{1.4}$$

Rearranging these equations in order to compute F_e, l distance and β , we get:

$$\begin{aligned} F_e.\cos(\beta - \alpha) &= (\ddot{x} \cdot m) - (F_1 + F_2) \cdot \sin\alpha \\ F_e \cdot \sin(\beta - \alpha) &= (\ddot{y} + g) \cdot m - (F_1 + F_2) \cdot \cos\alpha \\ F_e \cdot l \cdot \sin\beta &= L \cdot (F_1 - F_2) - (\dot{w} \cdot I_{xy}) \end{aligned} \tag{1.5}$$

Assuming $F_e \neq 0$, we can compute:

$$\tan(\beta - \alpha) = \frac{(\ddot{y} + g) \cdot m - (F_1 + F_2) \cdot \cos\alpha}{(\ddot{x} \cdot m) - (F_1 + F_2) \cdot \sin\alpha}$$

using sensor readings and known constants. The value of the $(\beta - \alpha)$ from its tangent, can be used in computing of $\cos(\beta - \alpha)$ and $\sin(\beta - \alpha)$ in Equation 1.5.

After finding the angle β, following equation gives us the magnitude of the impact force vector F_e.

$$F_e = \frac{(\ddot{y} + g) \cdot m - (F_1 + F_2) \cdot \cos\alpha}{\sin(\beta - \alpha)}$$

Finally, l value is derived as:.

$$l = \frac{L.(F_1 - F_2) - (\dot{w}.I_{xy})}{F_e.\sin\beta}$$

3.1.1.1 Verification of Equations in 2D

The Physics Simulator [41] software, is used to simulated and verify the model that we have derived.

A rigid, homogeneous body in the shape of a rectangle is used with a mass of 1 kg, and inertia of $10kg \cdot m^2$ is used. The simulation environment has 10 ms. resolution and gravity is 9.8 m/s^2. Without any impact force, the system is in balance with two forces on each side of the body with 4.9N. An impact force F_e, -5N in the X-axis and 5N in the Y-axis, is applied as shown in Figure 3.3.

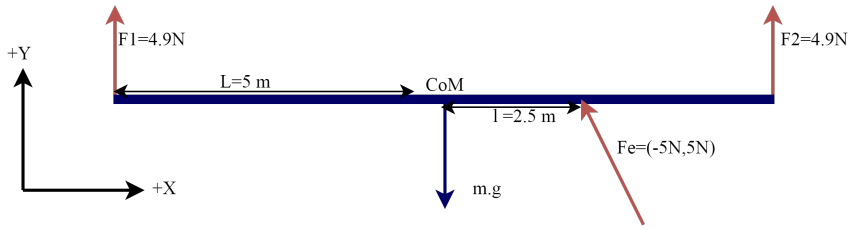


Figure 3.3: Representation of simulation world at $t0$

At each simulation step, linear accelerations ($\dot{x}$, $\dot{y}$) and angular acceleration ($\dot{\omega}$)of the rigid body, rotation of CoM and impact force (F_e) are recorded in both simulation and our model in c++ application. This test took 1 second with 100 simulation steps. Table 3.2 and 3.1 show the a part of recorded outcomes for comparison.

Table 3.2 and Table 3.1, verifies that there are only slight differences due to differences in the step of our model and physics simulator. The comparison between actual and estimated β angle, impact force F_e and l distance are plotted in graphs in Figure 3.4.

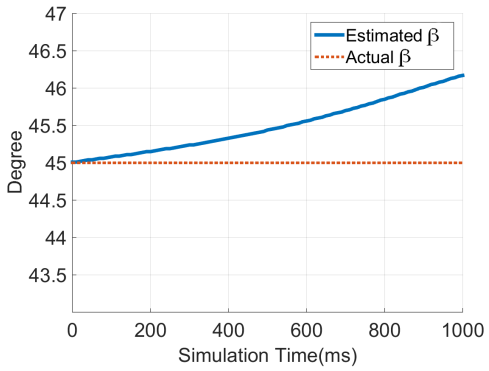
Based on the verification of the 2D model, the impact force F_e, applied to the rigid body in 2D space, is estimated accurately when measurements are noiseless and applied forces, and generated thrusts are measured without delay.

Table 3.1: Moment, angular acceleration ($\dot{\omega}$) ,and rotation of the rigid body comparison between simulation (left) and C++ model (right)

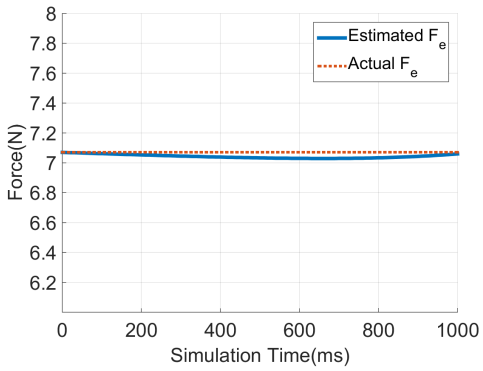
Time	moment	$\dot{w}$	Rotation	moment	$\dot{w}$	Rotation
ms	N-m	rad/s^2	rad	N-m	rad/s^2	rad
0	12.500	1.250	0	12.499	1.249	0.0001
200	12.810	1.281	0.025	12.825	1.282	0.0290
400	13.704	1.370	0.102	13.732	1.373	0.1095
600	15.050	1.505	0.233	15.086	1.508	0.2453
800	16.541	1.654	0.425	16.575	1.657	0.4418
1000	17.585	1.758	0.683	17.597	1.759	0.7047

Table 3.2: Linear accelerations ($\dot{x}$,$\dot{y}$) and impact force (F_e) comparison between simulation (left) and C++ model (right)

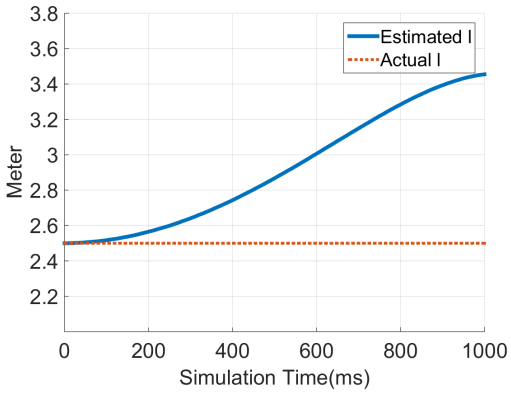
Time	$\dot{x}$	$\dot{y}$	Act. F_e	$\dot{x}$	$\dot{y}$	Est. F_e
ms	m/s^2	m/s^2	N	m/s^2	m/s^2	N
0	-5.0	5.0	7.07	4.999	4.999	7.07
200	-5.246	4.997	7.07	5.258	4.996	7.05
400	-5.994	4.949	7.07	6.019	4.946	7.03
600	-7.264	4.735	7.07	7.302	4.725	7.03
800	-9.040	4.129	7.07	9.089	4.105	7.04
1000	-11.183	2.804	7.07	11.238	2.758	7.06



(a) Estimated vs. actual β



(b) Estimated vs. actual F_e



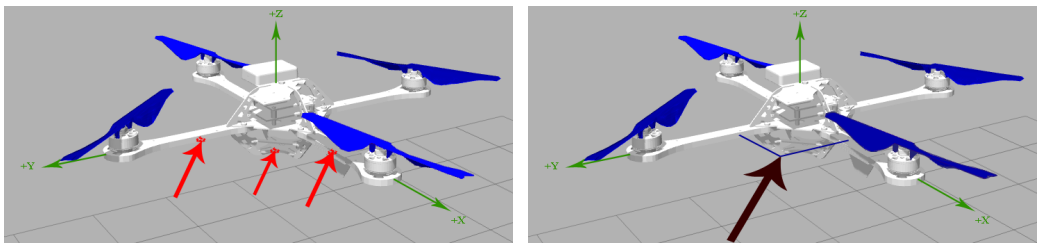
(c) Estimated vs. actual l

Figure 3.4: 2D space β, F_e and l estimation test results

3.1.2 Estimation of Impact Force in 3D on the Quadrotor

In this section, we extend the preliminary analysis of impact force estimation into 3D and apply it to the quadrotor model to estimate the direction and location of the sniper bullet.

Consider a case where the quadrotor is hit by multiple bullets as depicted in Figure 3.5-a. This is possible in real life, where anti-drone systems' bullets shot by buckshot and more than one piece of buckshot hit to the body of the quadrotor. Yet, by computing the resultant force that is shown in Figure 3.5-b, estimating the position of the sniper is possible.



(a) Multiple Force Vectors (b) Resultant Force Vector

Figure 3.5: Multiple and resultant force vectors representations upon a quadrotor model on simulation environment

The impact force vector F_e is represented in 3D as:

$$F_e = \begin{bmatrix} F_{e_x} & F_{e_y} & F_{e_z} \end{bmatrix}^T$$

When external impact vector F_e taken into account on Equation 2.3, the linear accelerations a_x, a_y and a_z are computed as:

$$\begin{aligned} a_x &= \frac{(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \cdot T + F_{e_x}}{m} \\ a_y &= \frac{(\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \cdot T + F_{e_y}}{m} \\ a_z &= \frac{(\cos\phi \cos\theta) \cdot T - (m \cdot g) + F_{e_z}}{m} \end{aligned} \tag{1.6}$$

where T represents the sum of the thrust forces computed in the Equation 2.2.

When external impact vector F_e taken into account on Equation 2.3, the angular accelerations $\ddot{\phi}$, $\ddot{\theta}$ and $\ddot{\psi}$ are computed as:

$$\begin{aligned} \ddot{\phi} &= \frac{L \cdot (T_2 - T_4) - (I_{zz} - I_{yy}) \cdot \dot{\theta}\dot{\psi} + F_{e_z} \cdot l_y}{I_{xx}} \\ \ddot{\theta} &= \frac{L \cdot (T_3 - T_1) - (I_{xx} - I_{zz}) \cdot \dot{\phi}\dot{\psi} + F_{e_z} \cdot l_x}{I_{yy}} \end{aligned} \tag{1.7}$$

$$\ddot{\psi} = \frac{(H_1 + H_3) - (H_2 + H_4) - (I_{yy} - I_{xx}) \cdot \dot{\phi}\dot{\theta} + F_{e_x} \cdot l_y - F_{e_y} \cdot l_x}{I_{zz}}$$

where H_i refers to the hub moment for corresponding rotor i and $\ddot{\phi}$, $\ddot{\theta}$ and $\ddot{\psi}$ represent angular accelerations over axes X, Y and Z respectively and T_i represents the thrust force generated by rotor i.

Here is the key point that, although Equations 1.7 involves the resultant force applied point, Equations 1.6 does not. Therefore, we use the Equations 1.7 in the estimation of impact force in order to be independent from the resultant force applied point.

The impact force estimation process takes the raw IMU readings and rotor angular velocities. It filters the raw IMU data and feeds it into the orientation filter process in order to get Euler Angles. Besides, rotor angular velocities are used for total thrust force estimation. In the core module of the impact force estimation process, Euler Angles, filtered IMU data, and total thrust force is used in the estimation of the raw impact force vector. Finally, the estimated raw impact force vector is applied to linear regression. The final output of this process is the estimated impact force vector.

This flow is shown in Figure 3.6.

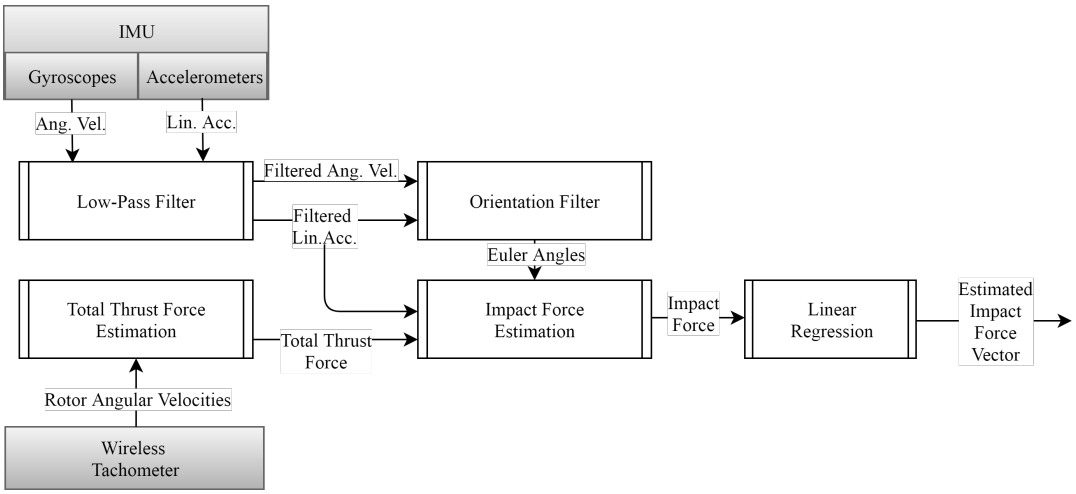


Figure 3.6: Impact force estimation flow

3.1.2.1 Orientation Filter

The sensor readings coming from the IMU sensor needs further preprocessing due to the following issues:

- Sensor measurements are biased and noisy.
- Transformation from the body-frame to the inertial reference frame is required.
- Gyroscope sensors do not measure angular position directly. They only measure angular velocities.
- Accelerometer measurement includes gravitational acceleration.

Estimating the orientation of quadrotor based on the IMU sensor readings has been studied in [13] and [42]. In these studies, an approach that uses accelerometer and gyroscope data in to compute the orientation of a rigid body was used with respect to Figure 3.7, represents the block diagram of the filter where t refers to time, a refers to acceleration, ω refers to angular velocity, and q refers to quaternion.

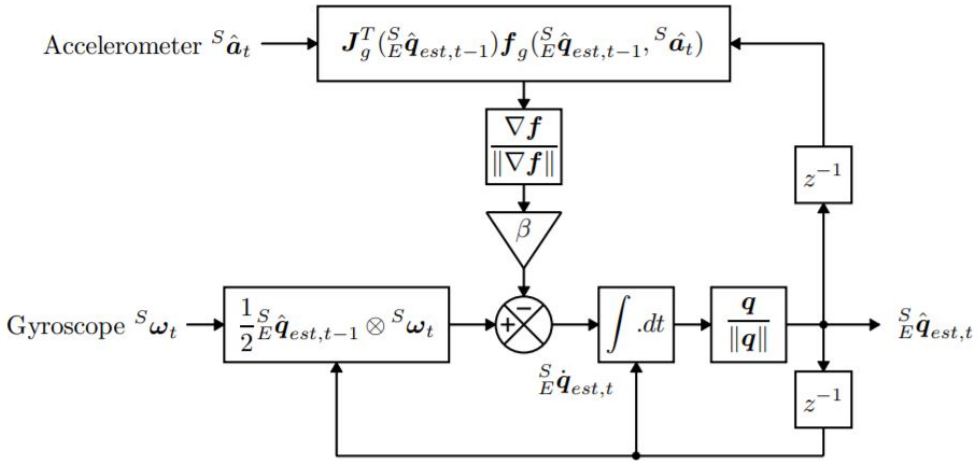


Figure 3.7: Block diagram of orientation computation filter taken from [13].

Typically, orientation filters that are supported with additional sensors such as GPS in Inertial Navigation Systems [30].

3.1.2.2 Low-Pass Filter

The low-pass filter (LPF) is a filter that passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with frequencies higher than the

cutoff frequency. Depending on the application and design, filter parameters are tuned and calibrated.

Since IMU sensor readings are noisy [43],a low-pass filter is designed to filter raw accelerometer data. The filter parameters are determined experimentally. In Figure 3.1.2.2, filtered, and raw acceleration data are shown.

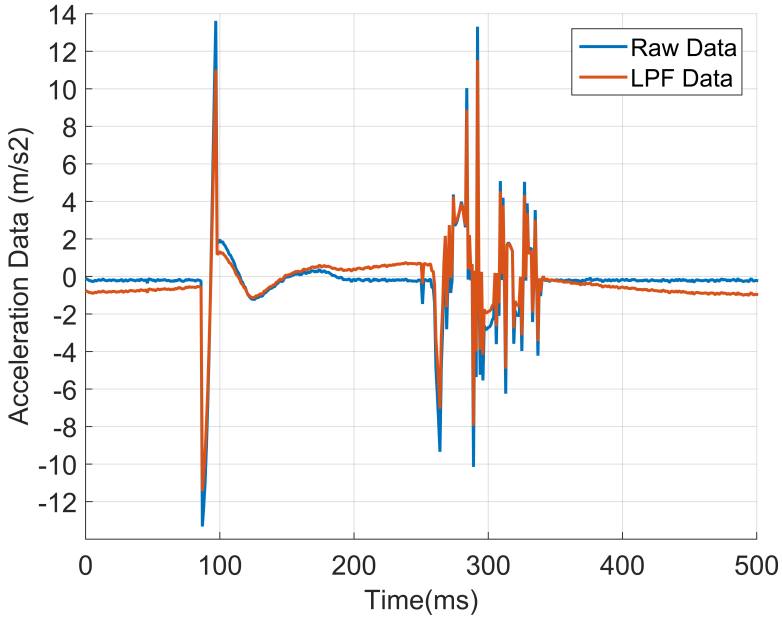


Figure 3.8: Raw vs. Low-Pass filtered acceleration data

3.1.2.3 Thrust Force Computation

Thrust force computation requires three main parameters. The first one is the c_t coefficient, which depends on factors such as humidity, air pressure, air drag, and temperature. The second one is the delay between controllers' command and the attainment of angular velocities of rotors. The last one is to measure the angular velocity of a rotor.

According to equation 2.1, a thrust force coefficient called c_t can be computed when the angular velocity of the rotor and thrust has been known. All propellers generate the same thrust, and the sum of these thrusts equals the weight of the quadrotor during hovering.

We computed the c_t coefficient, during the 1-second hovering of the quadrotor. Within this duration, c_t is computed every 10 ms using equation 1.8.

$$c_t = \frac{\sum_{i=1}^{4} {\omega_i}^2}{m \cdot g} \tag{1.8}$$

Figure 3.9 represents computed c_t vs. mean of computed c_t during 1 second. At the end, last mean value is accepted as c_t throughout flight.

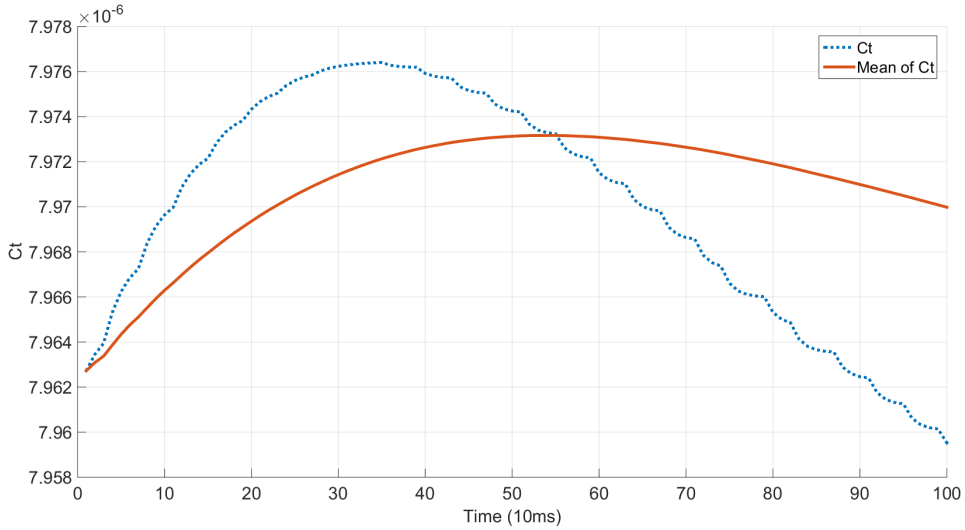


Figure 3.9: Computed c_t represented by blue dotted line and the mean of computed c_t is represented by red line for each 10 ms along 1 ms.

The second core part of computing thrust generated by rotors is the delay between controllers' command to updating angular velocities of rotors. In the computer-based simulations, this delay is close to 0. However, in real-life tests, depending on the wireless tachometer and the quadrotor, this delay is nonzero. The SMAGM module should be able to estimate this delay by taking previous step values of parameters that are experimentally tuned. However, this feature left as future work.

The last core part is computing or measuring the angular velocity of a rotor. In the quadrotors, Electronic Speed Control (ESC) unit is used in order to control rotors' angular velocity. By changing the angular velocities of rotors, the thrust force generated by the rotor is controlled. Therefore, measuring the thrust force of a rotor is nearly related to the angular velocity of the corresponding rotor. According to [14], nearly linear relationship between rpm^2 and thrust generated by rotor is plotted in

Figure 3.10. In order to be self-contained, the SMAGM module is designed as using non-contact tachometers [44] to measure the angular velocity of each rotor.

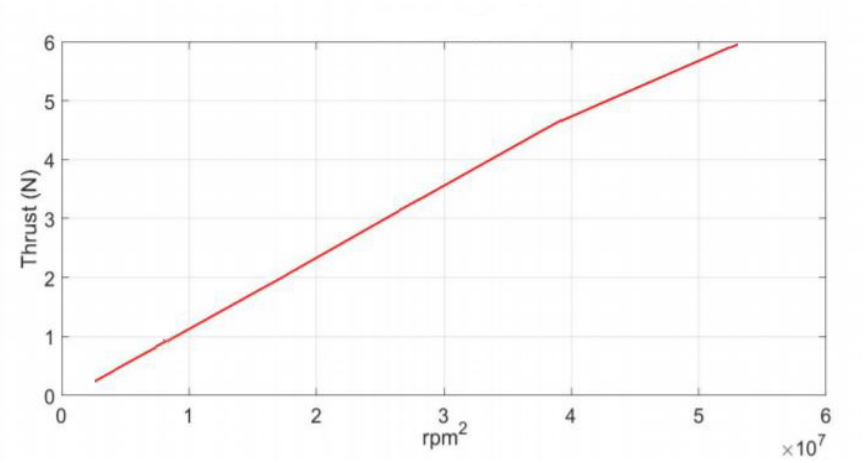


Figure 3.10: Thrust vs. rotation relationship graph adopted from [14].

3.1.2.4 Impact Force Estimation

When impact forces are alone in the left-hand side of the equations on 1.6, we get the following equations.

$$\begin{aligned} F_{e_x} &= a_x \cdot m - (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \cdot T \\ F_{e_y} &= a_y \cdot m - (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \cdot T \\ F_{e_z} &= (a_z + g) \cdot m - (\cos\phi \cos\theta) \cdot T \end{aligned} \tag{1.9}$$

The accuracy of these equations is depended on the accuracy of inputs such as Euler Angles (ϕ, θ, ψ), the total thrust generated by propellers and linear accelerations. Proposed methodologies which aims to increase accuracy of inputs were explained in subsections 3.1.2.1, 3.1.2.2 and 3.1.2.3 respectively.

One possible problem for this estimation is the delay in each measurement. According to the formula above, the total thrust generated by propellers by T, linear accelerations by a, and Euler angles by (ϕ, θ, ψ) are measured in specific periods. In real-life tests, any of measurement might be shifted in the time depend on the sensor delay to tune the system. This means that, after real-life environment tests, in a specific time

estimation for external effect, for example, the total thrust generated by propellers measurement might be taken into account 30 ms, 3 cycles, back. It is possible because the system is not obliged to make estimation immediately and can store previous measurements for later estimations after being shot.

3.1.2.5 Regression

Linear Regression is used to increase accuracy on the estimated impact force. Linear regression estimates a dependent value (y) based on an independent variable (x) based on a set of data pairs by fitting a linear relationship between them, in the form of Equation 1.10. Specifically, given *x* (input for training) and *y* (labels) *A* and *B* are estimated that minimized the prediction error.

$$\mathbf{y} = \mathbf{A} \cdot \mathbf{x} + \mathbf{B} \tag{1.10}$$

In order to collect training data, a script generated random *256* forces and points upon quadrotor. From 30*th* to 2580*th* seconds of simulation, F_{e_x}, F_{e_y} and F_{e_z} forces and applied point (l_x , l_y) which are generated by script and estimated F_{e_x}, F_{e_y} and F_{e_z} are used for training dataset. Using Matlab[3]'s Basic Fitting Tool, the linear regression coefficients and constants are estimated as listed in Table 3.3.

Table 3.3: The parameters computed by Matlab[3] Basic Fitting Tool's linear regression algorithm.

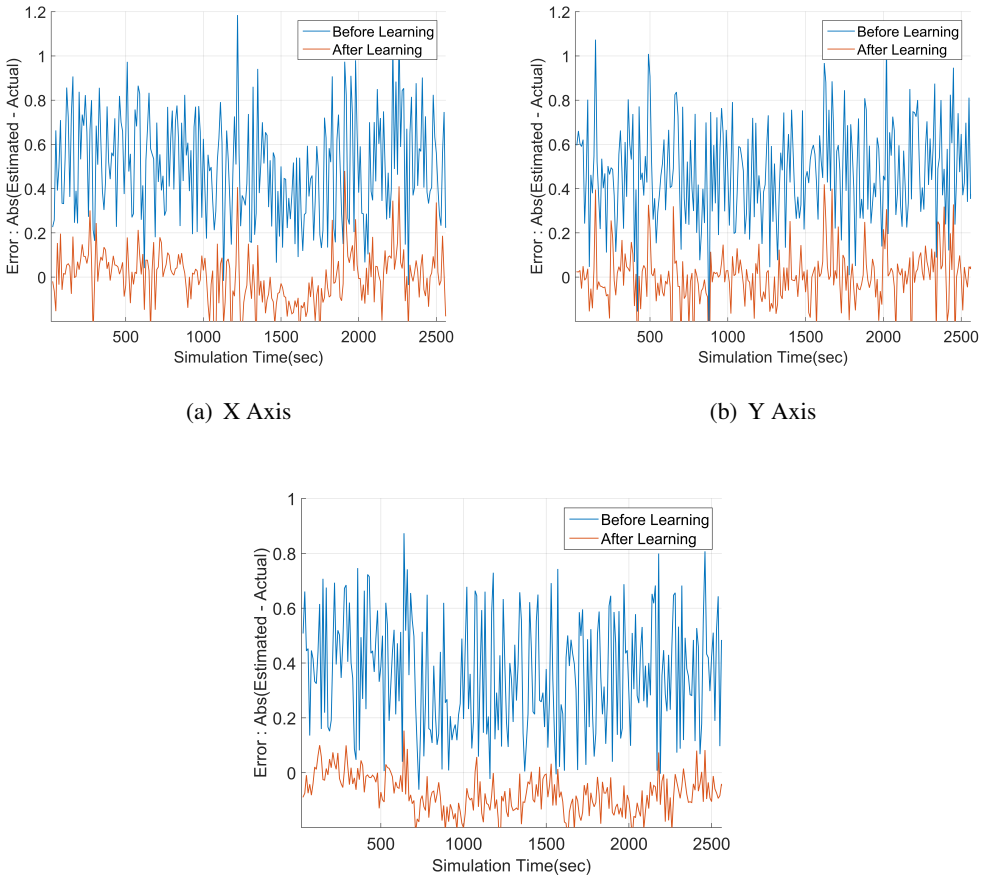
	Coefficient	**Constant**
X Axis	1.0751	0.11629
Y Axis	1.0684	0.13574
Z Axis	1.0747	0.038195

Use linear regression parameters estimated from data; the impact force estimations

are adapted from raw estimations F_i to obtain more accurate estimations F_o.

$$\begin{bmatrix} F_{o_x} \\ F_{o_y} \\ F_{o_z} \end{bmatrix} = \begin{bmatrix} F_{i_x} \\ F_{i_y} \\ F_{i_z} \end{bmatrix} \cdot \begin{bmatrix} 1.0751 \\ 1.0684 \\ 1.0747 \end{bmatrix} + \begin{bmatrix} 0.11629 \\ 0.13574 \\ 0.03819 \end{bmatrix} \tag{1.11}$$

Figures 3.11 and Table 3.4 plot the error for impact force estimation error for each axis, before and after the use of the linear regression.



(a) X Axis

(b) Y Axis

(c) Z Axis

Figure 3.11: Enhancement by linear regression for each axis

Table 3.4: Error median/range comparison between before and after linear regression

	Error Median	**Error Range**
Before Regression	0.4987	1.2210
After Regression	0.0126	0.8188

3.1.2.6 Impact Force Assessment

The SMAGM module estimates the impact forces in real-time, and when the estimated impact force exceeds a threshold value, a strike is detected. The threshold value is set 10 times upper/lower bound of the mean value of the estimated impact force and continuously updated. Figure 3.12 shows the update of threshold values and estimated impact force.

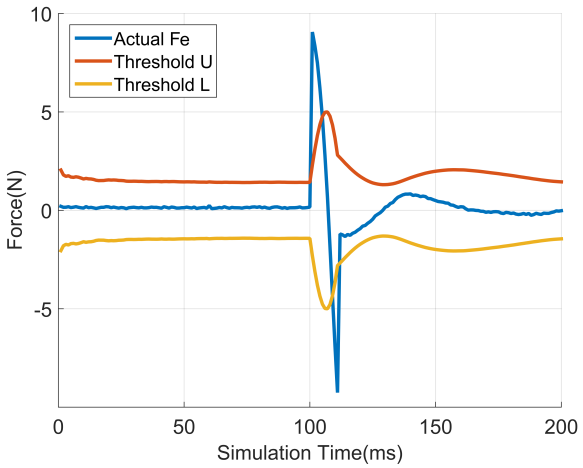
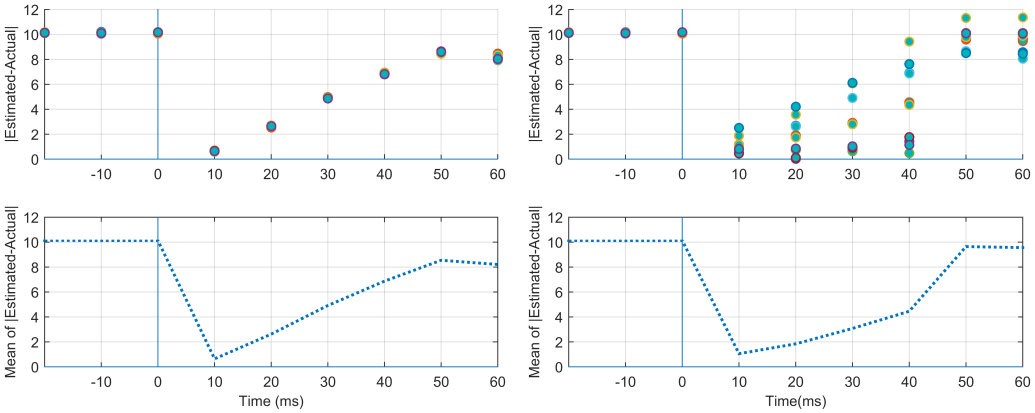


Figure 3.12: Upper(U) and lower(L) threshold vs. estimated impact force

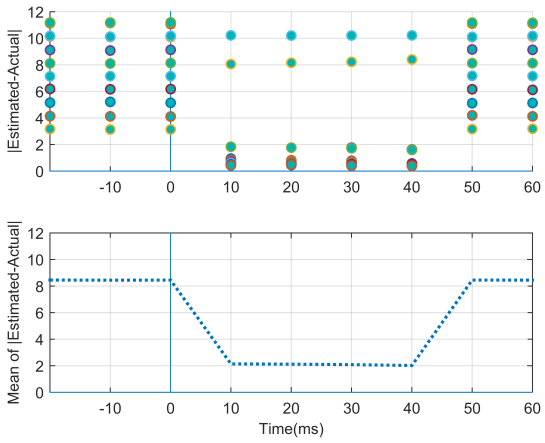
When the estimated impact force exceeds the threshold, it is marked as a potential strike, and the module is alerted.

In order to respond the question “When the estimation is the most accurate?“, the following scenarios are applied to the quadrotor model on Gazebo[20]. Figures 3.13 plot results for each test case.



(a) Case 1, the resultant force of 10N in each axis applied to the same point 11 times

(b) Case 2, the resultant force of 10N in each axis, applied to randomly selected different points 11 times



(c) Case 3, the resultant force of 12N to 4N forces in each axis, applied to randomly selected different points 13 times

Figure 3.13: Absolute value of the estimation error (*estimated-actual*) for impact force and its mean for each test cases

During each test, the IMU data and RPM readings for each rotor data are recorded. These data are then processed, and the impact force is estimated. The accuracy of the estimation is measured as the absolute value of the difference between the applied and estimated impact forces.

Experimental results indicate that the SMAGM module can estimate the strike within

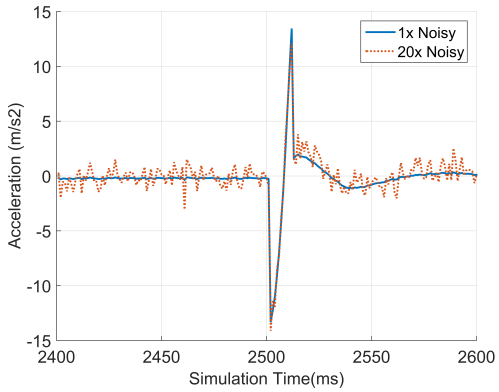
10ms to 20 ms. However, this duration may change in real-life tests, as it would depend upon the impact and collision model of bullet and quadrotor, sensor sensitivities, and inertia of the quadrotor.

3.1.2.7 Analysis of the IMU Noise Level on the Estimation Accuracy

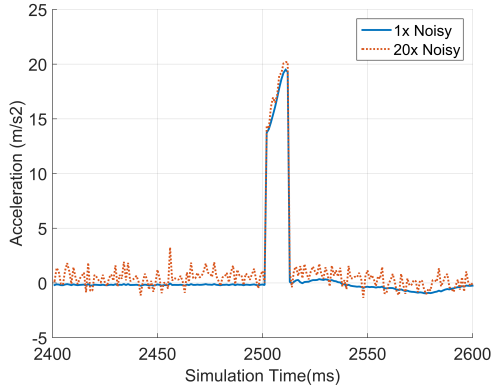
In the Gazebo simulation environment, the effect of the IMU noise level to the accuracy of the estimation is analyzed thanks to RotorS[45] framework due to allowing us to change noise level for both gyroscopes and accelerometers.

In our experiments, a hovering quadrotor is hit by a resultant impact force of 10N from each axis, both RotorS[45] default IMU noise level coefficients and 20 times multiplied IMU noise level as it is shown in Table 3.5. In each test, IMU readings are recorded, and the orientation of the quadrotor is calculated using this data. Finally, both the IMU readings and the orientation estimations are used to estimate the impact force.

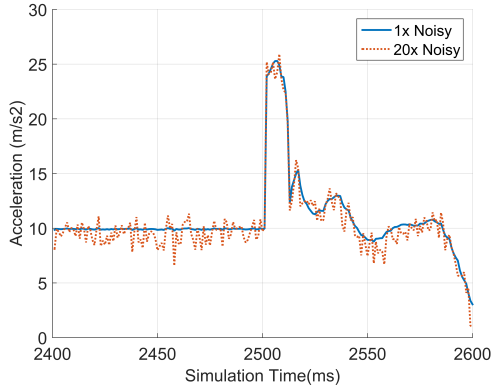
Figures 3.14 plot the acceleration values read from IMU, for each axis.



(a) X Axis



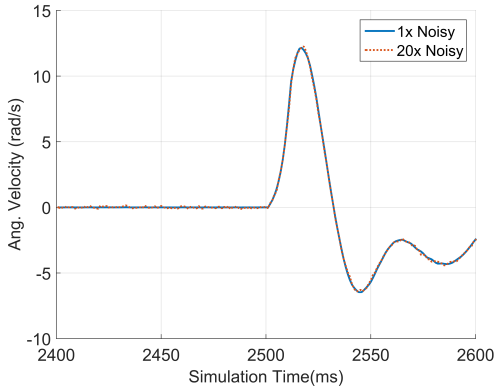
(b) Y Axis



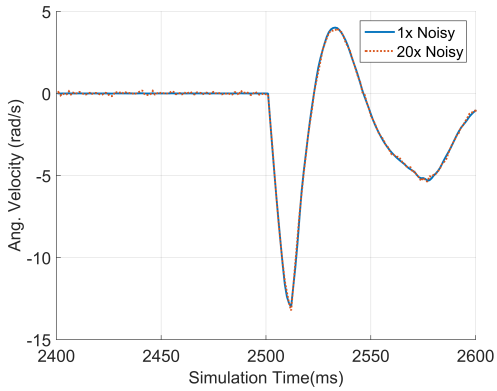
(c) Z Axis

Figure 3.14: Default vs. 20x noisy linear acceleration data during simulation for each axis.

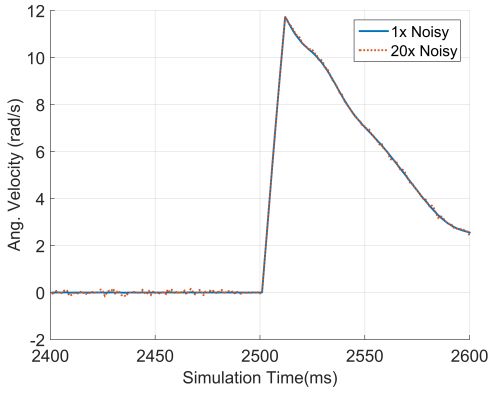
Figures 3.15 plot the angular velocity readings coming from gyroscopes of IMU, for each axis.



(a) X Axis



(b) Y Axis



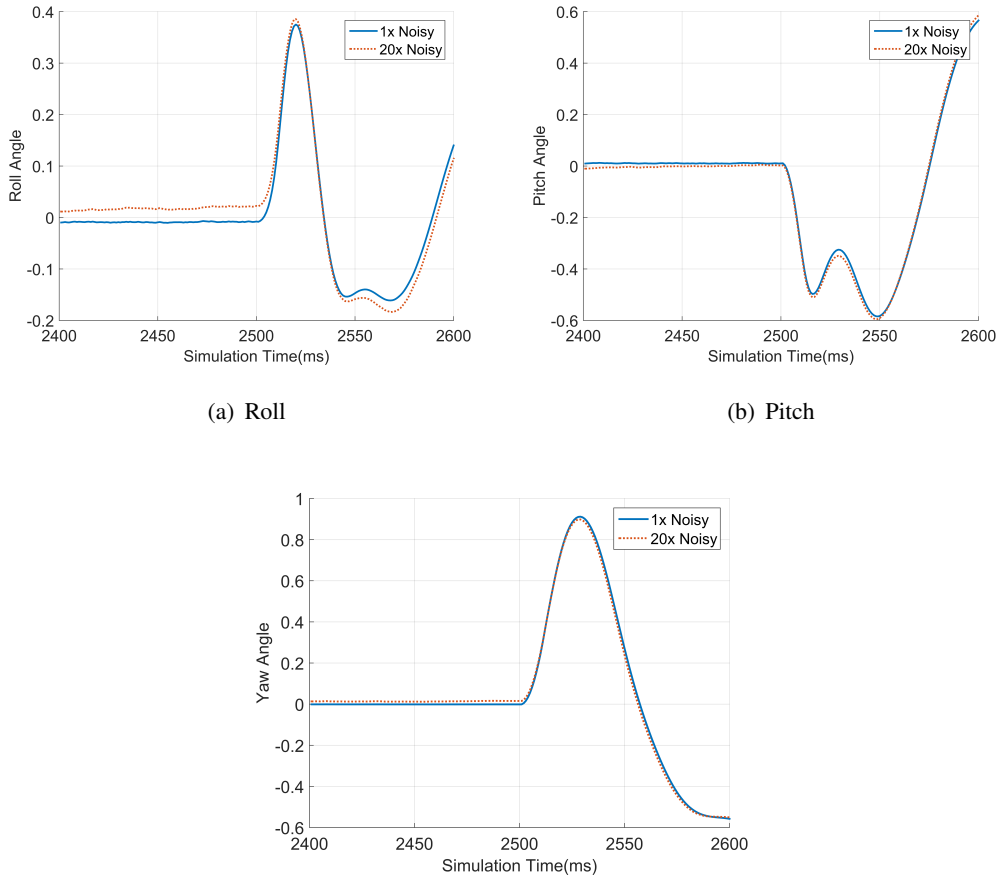
(c) Z Axis

Figure 3.15: Default vs. $20\times$ noisy angular velocity readings for each axis

Table 3.5: RotorS ADIS16448 IMU plugin noise level coefficients

Coefficient	**Default**	**Multiplied**
Gyroscope Noise Level	3,391e-4	6,391e-3
Accelerometer Noise Level	2.0e-3	4.0e-2

Using angular velocity and linear acceleration readings on each axis, the orientation filter estimates the pose of the quadrotor. The effect of IMU noise level over orientation shown in Figure 3.16.



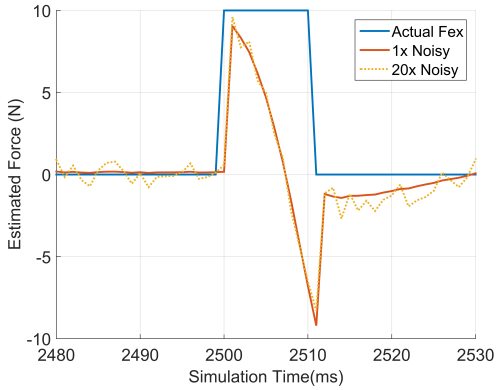
(a) Roll

(b) Pitch

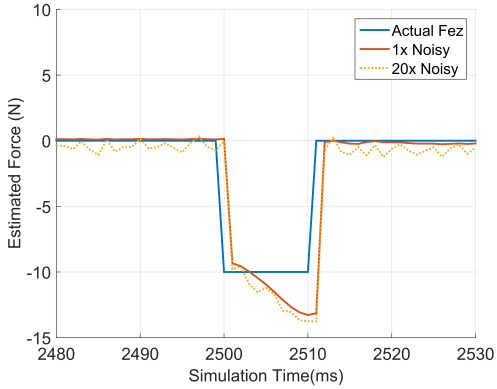
(c) Yaw

Figure 3.16: Euler Angles under noisy data

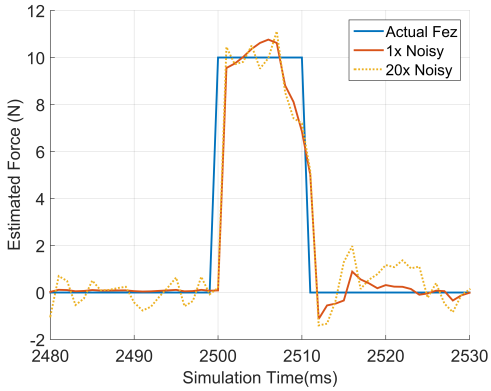
Finally, estimated impact forces for each axis for each noise level are shown in Figures 3.17 and Table 3.6.



(a) X Axis



(b) Y Axis



(c) Z Axis

Figure 3.17: Estimated impact force for each axis

Table 3.6: Estimated impact forces at different noise levels.

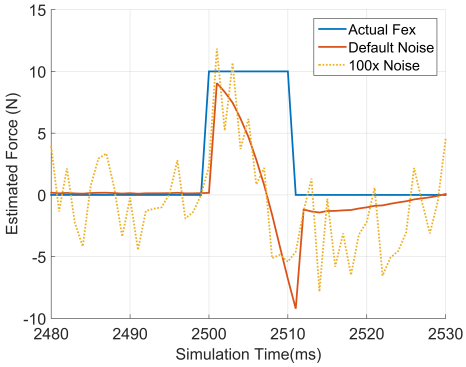
Hit Time	**Default Noise Level**			**20x Noise Level**		
	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	0.16	0.13	0.08	0.58	-0.02	0.20
10	9.05	-9.33	9.55	9.59	-9.78	10.45
20	8.3	-9.56	9.74	7.75	-9.59	9.71
30	7.46	-9.97	10.02	8.10	-10.93	9.81
40	6.17	-10.45	10.36	5.71	-11.53	10.51

Furthermore, using Equation 2.20, location of the sniper is detected and shown in Table 3.7. According to the table, even increased noise level affects the estimated force magnitudes, it does not dramatically affects the location of sniper. The reason for this, even the magnitudes of the estimated impact force is changes, their rate is nearly remains same.

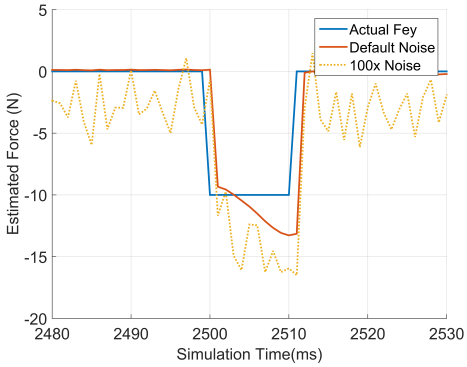
Table 3.7: Estimated sniper location for different noise levels

Sniper Location	**Actual**	**Default Noise**	**20x Noise**
X-Axis	5 m.	4.7 m.	4.58 m.
Y-Axis	5 m.	4.8 m.	4.67 m.

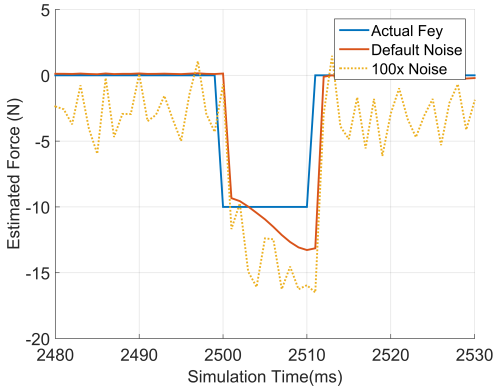
When the IMU noise level is multiplied by 100, estimated impact force is considerably deteriorated as shown in Figure 3.18 and Table 3.8.



(a) Estimated F_{e_x}



(b) Estimated F_{e_y}



(c) Estimated F_{e_z}

Figure 3.18: Estimated impact force for each axis

Table 3.8: Estimated impact forces for each noise level

Hit	**Default Noise Level**			**100x Noise Level**		
Time	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	0.16	0.13	0.08	2.32	-0.67	0.70
10	9.05	-9.33	9.55	11.84	-11.68	14.28
20	8.3	-9.56	9.74	5.20	-9.72	9.66
30	7.46	-9.97	10.02	10.7	-14.92	8.97
40	6.17	-10.45	10.36	3.71	-16.08	12.08

In summary, a reasonable IMU noise level, which is directly related to the specifications of the sensor, slightly affects the estimation accuracy.

3.1.3 Verification of the Proposed Model

Four tests with different scenarios are applied in Gazebo[20] environment using RotorS, *A Modular Gazebo MAV Simulator Framework* [45]. In tests, a model quadrotor AscTec Hummingbird[46] is used as shown in Figure 3.19 and the constants of this model are listed in Table 3.9 where they are the same for each test.

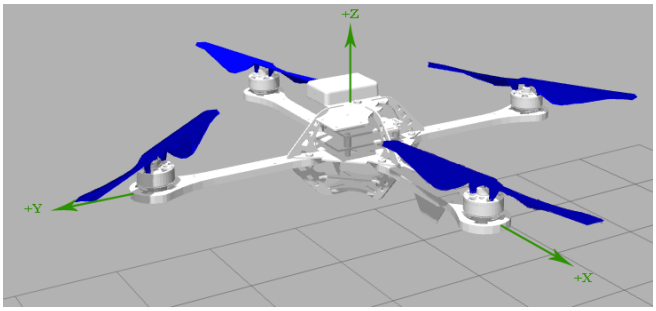
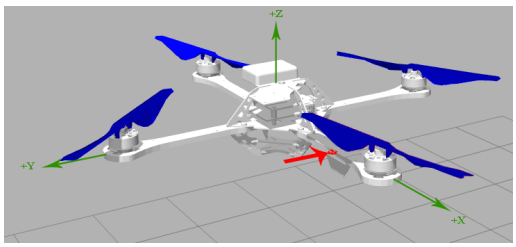


Figure 3.19: Hovering Hummingbird in Gazebo environment

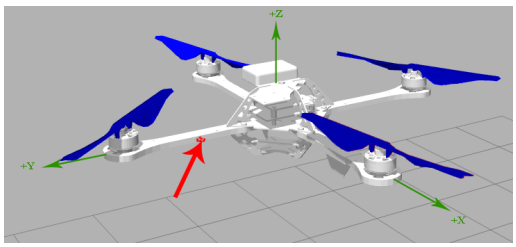
Table 3.9: Simulation model parameters, used in the Gazebo simulation environment

Name	**Value**	**Unit**
m: Mass of Quadrotor	0.68	kg
g: Gravity	9.8	m/s^2
L: Arm Length	0.17	m
I_{xx}, I_{yy}, I_{zz}	0.01	$kg.m^2$

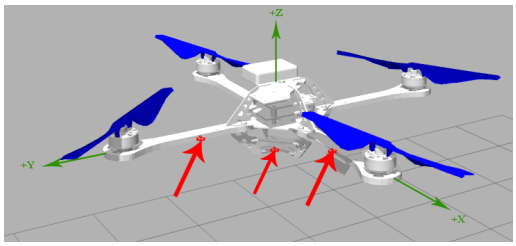
AscTec Hummingbird quadrotor model has an IMU sensor as a plug-in and provides angular and linear acceleration along 6 degrees of freedom (DOF) at 100 *Hz* frequency. The Robot Operating System (ROS)[47] allows these data to be accessed outside the Gazebo. Moreover, rotor speeds are also accessible outside of the Gazebo to evaluate generated force by propellers. Finally, during the simulation, a known impact force(s) can be applied to any point over the flying device at a predefined time of simulation throughout predefined time.



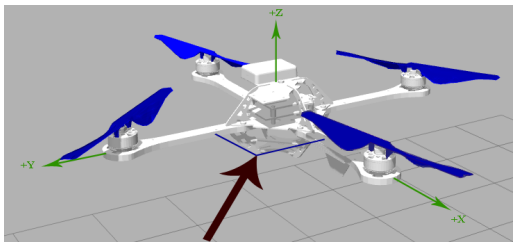
(a) Hit the quadrotor on X-axis on test 1 and 2



(b) Hit the quadrotor on Y-axis on test 3



(c) Hit the quadrotor on multiple points on test 4



(d) Resultant impact of multiple points on test 4

Figure 3.20: Impact force applied points representation for each tests

In this section, the following tests are applied to verify the proposed impact force estimation model.

- Test 1, apply impact force vector throughout 100 ms while quadrotor is hovering, which depicted in Figure 3.20-a.

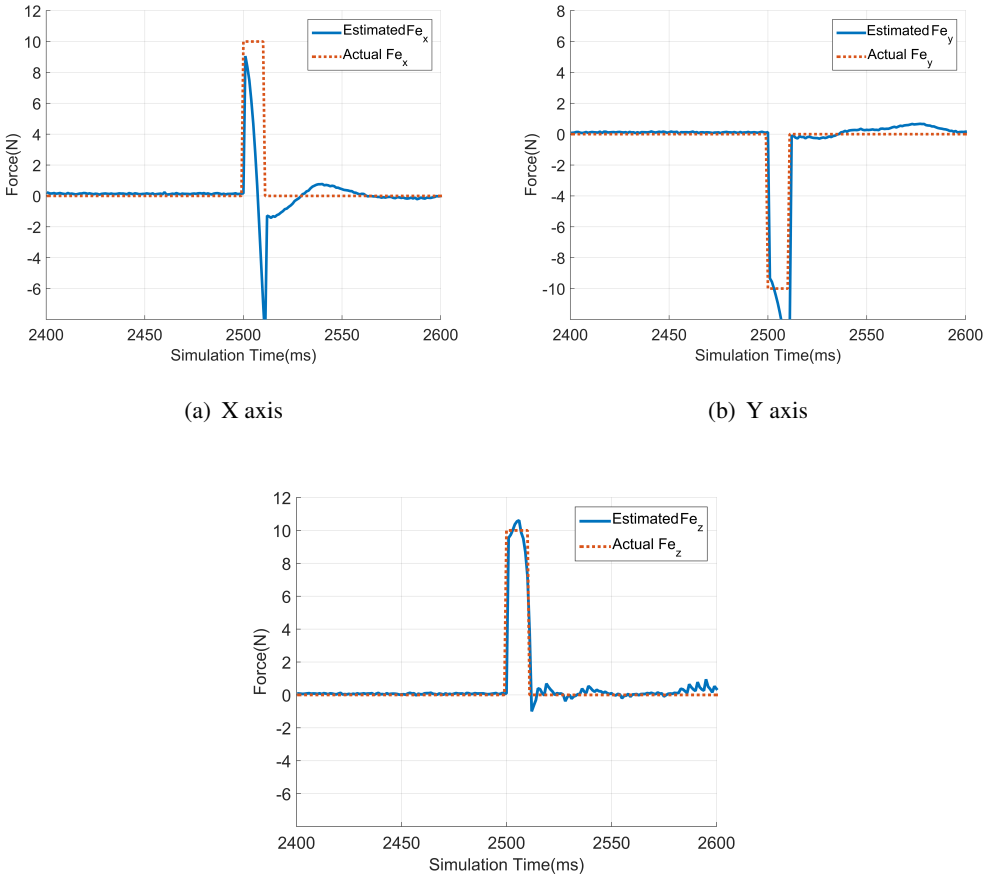
- Test 2, apply impact force vector throughout 100 ms while quadrotor is flying from one point to another, which depicted in Figure 3.20-a.

- Test 3, apply impact force vector throughout 100 ms while quadrotor is hovering, which depicted in Figure 3.20-b.

- Test 4, apply three impact forces upon quadrotor body from different points at the same time throughout 100 ms, which depicted in Figure 3.20-c and Figure 3.20-d.

3.1.3.1 Test 1: Shooting a Hovering Quadrotor on the X-Axis

In order to verify our estimation model, in test 1, we applied a known force vector upon a hovering quadrotor in Gazebo[20] simulation world. Throughout the test, IMU data and Rotor Speeds are recorded outside of the simulator. At 25*th* second of the simulation, an impact force vector F_e which is resultant of 10N in the X-axis, -10N in Y-axis and 10N in Z-axis, is applied throughout 100 *ms* to the point on quadrotor arm (x: 0.145, y: 0.0, z: 0.0) as depicted in Figure 3.20-a.



(a) X axis

(b) Y axis

(c) Z axis

Figure 3.21: Actual vs. estimated impact force F_e for each axis on test 1. Red dotted line represents the actual impact force that we applied and blue line represents the estimation of SMAGM module.

After test, recorded sensor data are applied into SMAGM module in Figure 3.1. Ac-

tual vs. estimated impact forces in each axis are depicted in Figures 3.21 for each axis.

Table 3.10 shows test outcomes for a short time period that starts from hit moment. Actual represents that we applied and estimated represents SMAGM module outcomes.

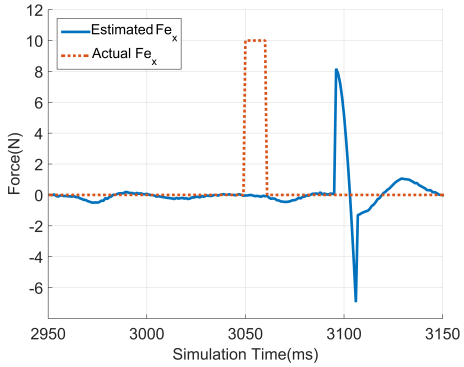
Table 3.10: Actual vs. estimated impact forces on test 1

Hit	**Actual**			**Estimated**		
Time	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	10	-10	10	0.160	0.135	0.08
10	10	-10	10	9.007	-9.344	9.545
20	10	-10	10	8.198	-9.594	9.727
30	10	-10	10	7.177	-10.05	10.03
40	10	-10	10	5.680	-10.58	10.39

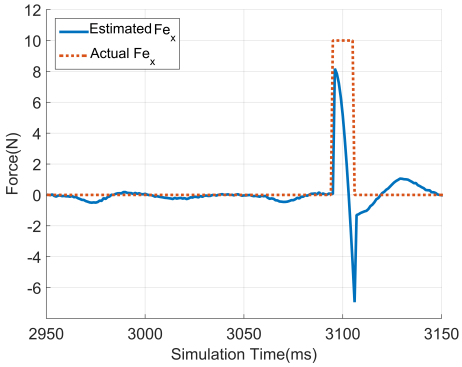
3.1.3.2 Test 2: Shooting a Flying Quadrotor on X-Axis

In order to verify our estimation model that, in test 2, we applied a known force vector upon a quadrotor that flies from one point to another in Gazebo[20] simulation world. Throughout the test, IMU data and Rotor Speeds are recorded outside of the simulator. At 30.05*th* second of the simulation, an impact force vector F_e which is resultant of 10N in the X-axis, -10N in Y-axis and 10N in Z-axis, is applied throughout 100 *ms* to the point on quadrotor arm (*x*: 0.145, *y*: 0.0, *z*: 0.0) as depicted in Figure 3.20-a.

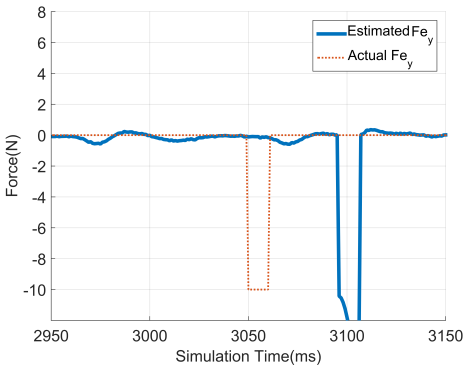
After test 2, recorded sensor data are applied to SMAGM module on 3.1. In graphs in Figures 3.22, actual vs. estimated impact forces in each axis are shown for test 2. Furthermore, in graphs *b* of each figure depicts the *40 ms* shifted estimations.



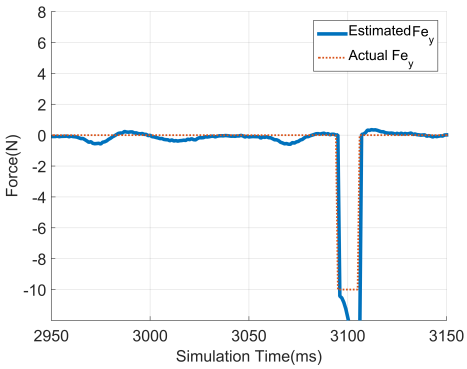
(a) X axis



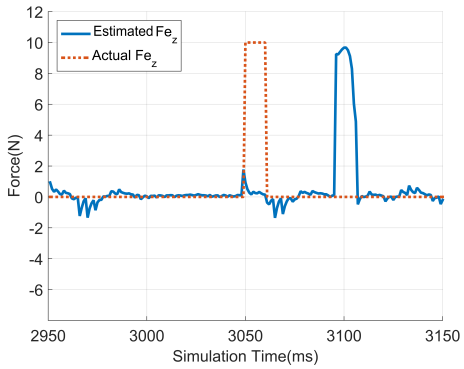
(b) X axis *40 ms* shifted



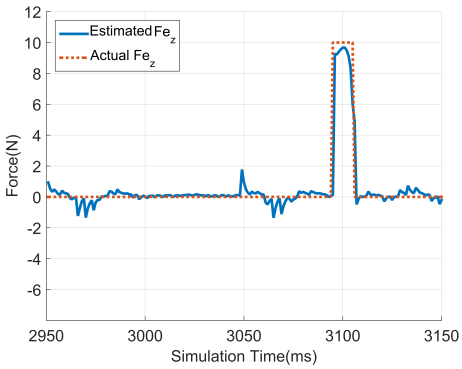
(c) Y axis



(d) Y axis *40 ms* shifted



(e) Z axis



(f) Z axis *40 ms* shifted

Figure 3.22: Actual vs. estimated impact force F_e for each axis on test 2. Red dotted line represents the actual impact force that we applied and blue line represents the estimation of SMAGM module.

According to test 2 results, the impact force F_e is estimated with a *40 ms* delay as it is more clear on Table 3.11. This shifting is done due to the weakness of the simulators IMU plug-in and is expected to be very short in real life. Already, to estimate impact force immediately is not very crucial in our proposed approach. Because our model decides either it is an external impact or normal noise using continuously updating threshold values.

Table 3.11 shows test outcomes for a short time period that starts from hit moment. Actual represents that we applied and estimated represents SMAGM module outcomes.

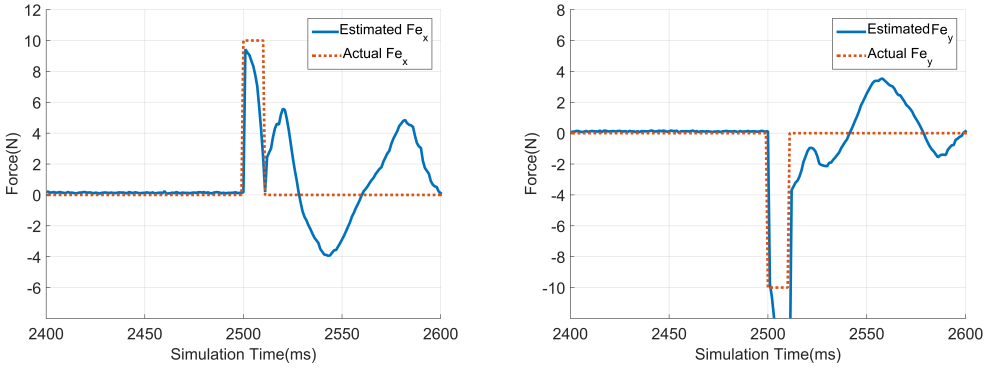
Table 3.11: Actual vs. estimated impact forces on test 2

Hit	**Actual**			**Estimated**		
Time	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	10	-10	10	0.092	0.099	0.16
10	10	-10	10	0.042	0.025	0.09
20	10	-10	10	0.18	0.05	-0.006
30	10	-10	10	0.07	0.02	0.08
40	10	-10	10	0.05	0.018	0.11
50	10	-10	10	8.17	-10.423	9.256
60	10	-10	10	7.80	-10.54	9.2279

3.1.3.3 Test 3: Shooting a Hovering Quadrotor nn Y-Axis

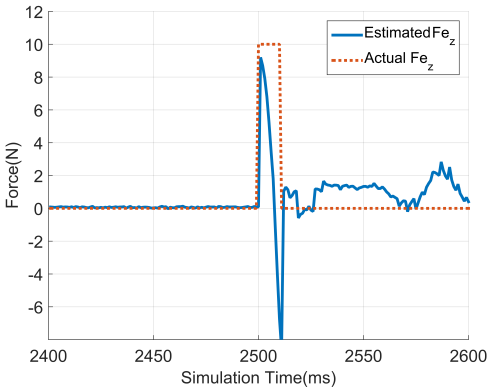
In order to verify our model that estimates the impact force, in test 3; we applied a known force vector upon a hovering quadrotor, same as test 1. Differently, in this test, the impact force applied to another arm of the quadrotor. At 25*th* second of the simulation an impact force F_e with compounds in each axis: 10.0N in X, -10.0N in Y and 10.0N in Z, is applied throughout 100 *ms* to point on quadrotor arm (x: 0.0, y: 0.145, z: 0.0) as depicted in Figure 3.20-b.

After the test, captured sensor data are applied to the SMAGM module in Figure 3.1.Figures 3.23 shows actual vs. estimated impact forces for each axis.



(a) X axis

(b) Y axis



(c) Z axis

Figure 3.23: Actual vs. estimated impact force F_e for each axis on test 3. Red dotted line represents the actual impact force that we applied and blue line represents the estimation of SMAGM module.

Table 3.12 shows test outcomes for a short time period that starts from hit moment. Actual represents that we applied and estimated represents SMAGM module outcomes.

Table 3.12: Actual vs. estimated impact forces on test 3

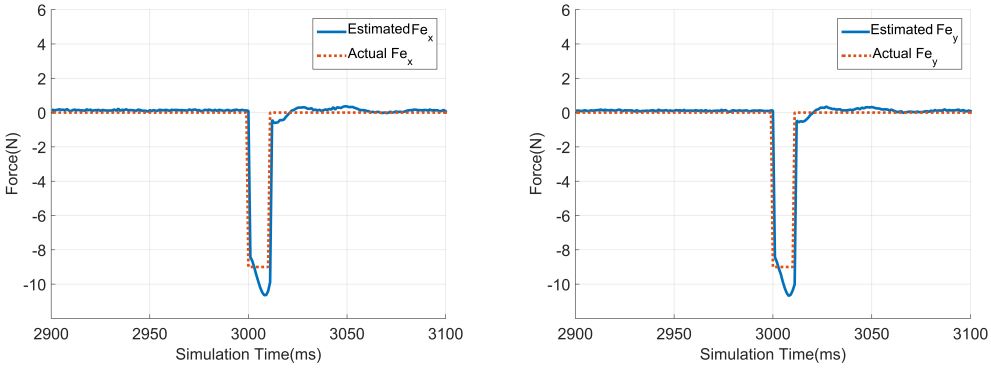
Hit	**Actual**			**Estimated**		
Time	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	10	-10	10	0.166	0.13	0.08
10	10	-10	10	9.429	-9.76	9.20
20	10	-10	10	9.223	-10.60	8.69
30	10	-10	10	9.064	-11.83	7.91
40	10	-10	10	8.702	-13.24	6.79

3.1.3.4 Test 4: Shooting a Hovering on Multiple Points

In order to verify our model that estimates the impact force, in test 4, we applied 3 equivalent force vectors to a hovering quadrotor at 30*th* second of the simulation from different points on quadrotor body. In Figure 3.20-c, each red arrow represents the resultant force of 3N's in each axis. In Figure 3.20-d, resultant force vector of these three equivalent impact force vectors are represented.

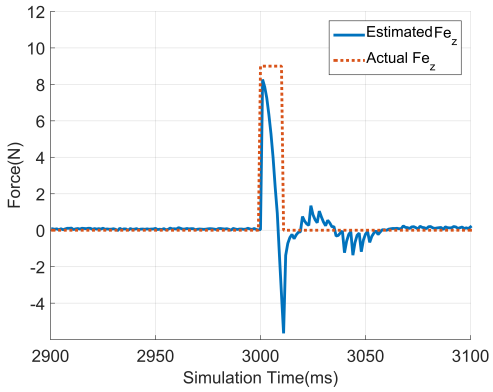
This scenario is similar to the expected real-world scenario where one or more buckshots hit the quadrotor body from different points simultaneously. Moreover, the magnitude of these vectors is nearly equivalent due to their source is the same. Furthermore, their initial velocities are equivalent due to the same reason. Even the resultant force vector of these buckshots is not on the body of the quadrotor, as shown in Figure 3.20-d, our model can estimate it.

After the test, recorded sensor readings are applied to the SMAGM module in Figure 3.1. Even if this resultant force is not on the body of the quadrotor, our proposed model performs well, as shown in Figures 3.24.



(a) X axis

(b) Y axis



(c) Z axis

Figure 3.24: Actual vs. estimated impact force F_e for each axis on test 4. Red dotted line represents the actual impact force that we applied and blue line represents the estimation of SMAGM module.

Table 3.13 shows test outcomes for a short time period that starts from hit moment. Actual represents that we applied and estimated represents SMAGM module outcomes.

Table 3.13: Actual vs. estimated impact forces on test 4

Hit	**Actual**			**Estimated**		
Time	F_{e_x}	F_{e_y}	F_{e_z}	F_{e_x}	F_{e_y}	F_{e_z}
0	-9	-9	9	0.09	0.12	0.03
10	-9	-9	9	-8.42	-8.41	8.27
20	-9	-9	9	-8.65	-8.70	7.86
30	-9	-9	9	-9.03	-9.09	7.24
40	-9	-9	9	-9.46	-9.41	6.33

3.2 SNIPER LOCALIZATION

Sniper localization process takes the estimated impact force vector from the impact force estimation process. It estimates the altitude from air pressure data of the barometer sensor. Then, it estimates the location of the sniper with respect to the *Vehicle Body Fixed* frame. Using magnetometer data, the location of the sniper is rotated to *Earth Inertial* frame. Finally, using GPS sensor data of quadrotor at the hit time, the estimated coordinate of the sniper is detected. The overall process of sniper localization is shown in Figure 3.25.

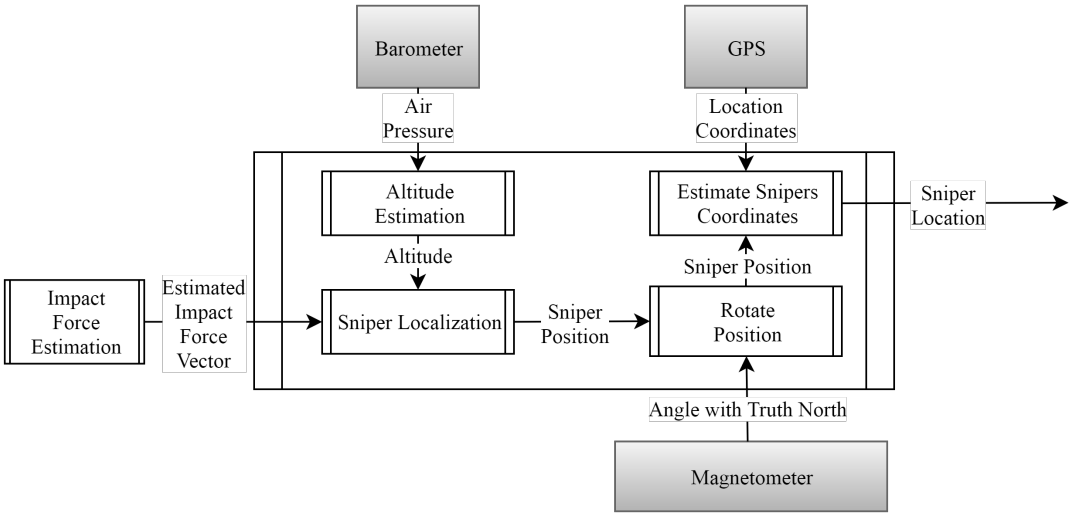


Figure 3.25: Sniper localization process overview

As depicted as 2D in Figure 3.26, the initial velocity of the bullet is V_i. At the time of the impact t_h, the velocity of the bullet is represented by V_h. V_q represents the velocity of the quadrotor at impact. After the collision, the velocity of the collided bodies is represented by V_c. The mass of the bullet is represented by m_b, and the mass of the quadrotor is represented by m_q.

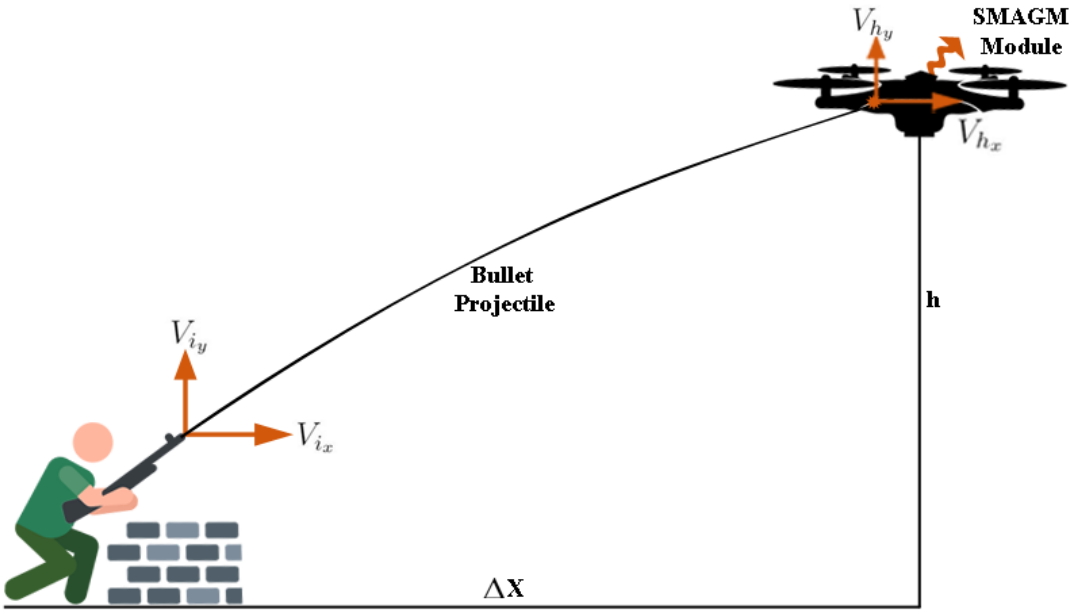


Figure 3.26: Sniper hits the quadrotor representation

The relationship between estimated impact forces and distance from the source of threat to quadrotor needs to be derived. It is assumed that bullet makes an inelastic collision with the quadrotor and after collision of the quadrotor and the bullet move together preservation of momentum. The following equations are generated with respect to the preservation of momentum.

$$m_b \cdot V_h + m_q \cdot V_q = (m_b + m_q) \cdot V_c$$
$$V_h \cdot m_b = m_q \cdot (V_c - V_q)$$
$$V_h \cdot m_b = m_q \cdot \Delta V$$

Assuming that the change in the velocity ΔV, is generated by a constant impact force, F_e, generated by the collision for a duration Δt.

$$m_q \cdot \Delta V = m_q \cdot a \cdot \Delta t = F_e \cdot \Delta t$$

The equation above represents the relationship between the velocity and the moment of impact. The bullet has initial velocity and hitting time velocity for each axis. Their hitting time velocities are linearly related to the estimated impact force vector, as shown above. The velocity vector may tail off its magnitude, but not changes its direction. Therefore, in order to estimate the threat location or distance from the quadrotor, the final velocity vector is used to estimate taken distance because travel time is equal for each axis. Hence, the following relationships are used:

$$\begin{bmatrix} V_{h_x} \\ V_{h_y} \\ V_{h_z} \end{bmatrix} \cdot m_b = \begin{bmatrix} F_{e_x} \\ F_{e_y} \\ F_{e_z} \end{bmatrix} \cdot \Delta t \tag{2.12}$$

where m_b and Δt are the same for each axis. We can then write:

$$F_e = \frac{m_b}{\Delta t} \cdot V_h \tag{2.13}$$

The equations above represent the linear relationship between impact velocity and the estimated impact force. It is expected that the initial velocity of the bullet, V_i is greater than impact velocity V_h where air resistance exist. The distance from the quadrotor to the sniper is:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} V_{h_x} \\ V_{h_y} \\ V_{h_z} \end{bmatrix} \cdot \Delta t + \frac{1}{2} \cdot \left(\begin{bmatrix} V_{h_x} \\ V_{h_y} \\ V_{h_z} \end{bmatrix} - \begin{bmatrix} V_{i_x} \\ V_{i_y} \\ V_{i_z} \end{bmatrix} \right) \cdot \Delta t \tag{2.14}$$

Applied forces during the flight to the rigid body of bullet reduces the velocity, as:

$$\begin{bmatrix} \Delta V_x \\ \Delta V_y \\ \Delta V_z \end{bmatrix} = \begin{bmatrix} F_{t_x} \\ F_{t_y} \\ F_{t_z} \end{bmatrix} \cdot \frac{\Delta t}{m_b} \tag{2.15}$$

where F_t refers to the total force acting during the flight on a specific axis.

Substituting Equation 2.15 into Equation 2.14 yields:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} V_{h_x} \\ V_{h_y} \\ V_{h_z} \end{bmatrix} \cdot \Delta t + \frac{1}{2} \cdot \begin{bmatrix} F_{t_x} \\ F_{t_y} \\ F_{t_z} \end{bmatrix} \cdot \frac{\Delta t^2}{m_b} \tag{2.16}$$

The relation between impact velocity and the estimated impact force in the Equation 2.12 substitute to the previous equation yields:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} F_{e_x} \\ F_{e_y} \\ F_{e_z} \end{bmatrix} \cdot \frac{\Delta t}{m_b} + \frac{1}{2} \cdot \begin{bmatrix} F_{t_x} \\ F_{t_y} \\ F_{t_z} \end{bmatrix} \cdot \frac{\Delta t^2}{m_b} \tag{2.17}$$

Simplification of Equation 2.17, for each axis, through a simplification of notation using variables yields:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} F_{e_x} \\ F_{e_y} \\ F_{e_z} \end{bmatrix} \cdot C + \begin{bmatrix} F_{t_x} \\ F_{t_y} \\ F_{t_z} \end{bmatrix} \cdot T \tag{2.18}$$

The effect of slowing down forces ignored on Equations 2.18 due to it is assumed that slowing down forces F_{t_x}, F_{t_y} and F_{t_x} are same for each axis and very short flight time of the bullet. Therefore, Equations 2.18 simplified as:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} F_{e_x} \\ F_{e_y} \\ F_{e_z} \end{bmatrix} \cdot C \tag{2.19}$$

The only measurable variable on Equations 2.19 is ΔZ which is altitude. Therefore, Therefore, the constant C is calculated with respect to the altitude, which is represented by h in Figure 3.26. When we write the equation set on 2.19 according to the altitude, the following final equations are yielding:

$$\begin{aligned} \Delta Z &= h \\ \Delta X &= h \cdot \frac{F_{e_x}}{F_{e_z}} \\ \Delta Y &= h \cdot \frac{F_{e_y}}{F_{e_z}} \end{aligned} \tag{2.20}$$

In order to verify the proposed approach, 10 different tests are applied on Gazebo[20] to hit a hovering quadrotor. The test scenario is to throw a sphere with 0.1 kg mass where the gravity is 9.8 m/s^2 and quadrotors' location is accepted as the origin. These

test cases are listed in the Table 3.14 where ΔX and ΔY refer to the initial ground location of the thrown sphere, h refers to the altitude of the quadrotor ,and F_x, F_y, and F_z refers to the applied forces over the sphere.

Table 3.14: Throwing a sphere from the ground test results

Case#	ΔX	ΔY	h	F_x	F_y	F_z	**Hits?**
1	1m.	2m.	5m.	-1N	-2N	5.98N	OK
2	1m.	20m.	5m.	-1N	-20N	5.98N	OK
3	4m.	3m.	5m.	-4N	-3N	5.98N	OK
4	10m.	10m.	5m.	-10N	-10N	5.98N	OK
5	20m.	20m.	5m.	-20N	-20N	5.98N	OK
6	20m.	15m.	5m.	-20N	-15N	5.98N	OK
7	15m.	30m.	5m.	-15N	-30N	5.98N	OK
8	25m.	25m.	5m.	-25N	-25N	5.98N	OK
9	20m.	30m.	10m.	-20N	-30N	10.98N	OK
10	20m.	30m.	20m.	-20N	-30N	20.98N	OK

The last column of the Table 3.14 shows the success of the sphere on hitting the quadrotor. According to Table 3.14, our proposed approach is valid.

In real-world, a single bullet or multiple buckshots will hit the quadrotor. The deviation of a standard bullet is provided by bullet producers to inform its users. For a normal bullet that does not include guidance, such as a 308 Winchester bullet, Table 3.15 shows the deviation from different ranges [48]. Quadrotors generally fly below 100 mt. altitude, hence deviation is negligible.

Table 3.15: 308 Winchester bullet deviation

Distance	**Bullet Deviation**
91 mt.	0 cm
137 mt.	3.12 cm
182 mt.	10.05 cm
228 mt.	20.98 cm
274 mt.	36.19 cm

Finally, in order to find the location of the sniper, we need to rotate the estimated point based on IMU axes. GPS gives the location information according to the Earth Inertial Frame, as detailed in subsection 2.4.1. The magnetometer sensor data is used to align the orientation of the quadrotor with the orientation of the world, to find the exact location of the sniper.

$$x' = x \cdot cos\theta - y \cdot sin\theta$$

$$y' = y \cdot cos\theta + x \cdot sin\theta$$

where θ is an angle to be rotated. This angle is obtained from the magnetometer, which gives the angle between IMU's X and the true North. After x' and y' are calculated, the exact location of the threat is automatically calculated by adding these values to the GPS outputs for the quadrotor.

As a summary, using the relationship between the estimated impact force for each axis and the velocity vector, the location of the sniper can be estimated, under the following assumptions:

- Flight duration of the bullet is fairly short. This is a realistic assumption since quadrotors mostly fly under 100mt. altitude and bullet can hit a quadrotor under less than 1 seconds.
- During the flight, for each axis, air drag is approximately the same. Although it might differ in the real environment, in short flight duration, its effect might be trivial.

- The only impact force effects on the body of the bullet is the gravitational force.
- The only measurable distance between quadrotor and ground located sniper is the altitude by barometer sensor. The altitude of the gun and the height of the sniper are neglected.

Another specific issue for this computation was the impact model of the quadrotor. In this thesis, it is assumed that incoming effects such as bullet directly hit to the body of a quadrotor, and we can measure its location before effect change its direction or acceleration after hitting. It is apparent that this model is open to extend and to be precise by applying real-time tests in the real world.

3.3 BROADCASTING ESTIMATED SNIPER LOCATION

SMAGM module should broadcast an estimated sniper location. After the quadrotor hit by the sniper, the quadrotor possibly damaged and falls. Our module can start to broadcast an estimated location thanks to its design as an independent add-on unit from the quadrotors. We propose that it is enough to broadcast threat location data to inform the rest of the swarm and other friendly forces.

CHAPTER 4

CONCLUSION

This thesis proposed a conceptual and theoretical proof-of-concept of SMAGM module, which is designed as a stand-alone module that can be affixed on quadrotor, and can act as a deterrent to attacks coming from the adversaries. Specifically, the module is designed to estimate the position of the sniper using readings from its on-board sensors to compute the instantaneous force vector at the time of the impact and uses this information to localize the position of the sniper. A method is proposed to estimate the force vector generated by the impact of the bullet using disturbance estimation methods. Then this vector and the altitude of the UAV is used to localize the position of the sniper. The methods are verified and systematically evaluated on a quadrotor model simulated in the Gazebo simulator [20] under four different test cases. The estimation rate of the simulated SMAGM module was limited at 100 Hz due to the limitations of the simulated IMU model.

Problems that were not addressed within this thesis and left for future work are:

- The cases where the propellers take a hit,
- The case where the quadrotor body losing its structural rigidity before making estimation,
- Testing changing under air drag,
- Variations in the time latency of power transfer from body to motors due to battery of quadrotor,
- The quadrotor geometrical center is different from the center of mass (CoM) nor the center of gravity (CoG) of the platform.

REFERENCES

[1] R. E. Weibel and R.J. Hansman, "Safety Considerations for Operation of Different Classes of UAVs in the NAS," *The 4th Aviation Technology, Integration and Operations Forum, AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*, 2004.

[2] A. Chapman, "Types of Drones: Multi-Rotor vs Fixed-Wing vs Single Rotor vs Hybrid VTOL," https://www.auav.com.au/articles/drone-types/ [Online; accessed January 02, 2020].

[3] "Matlab." https://ch.mathworks.com/products/matlab.html [Online; accessed June 26, 2019].

[4] "The FLIR Black Hornet PRS." https://www.flir.com/support/products/black-hornet-prs [Online; accessed January 15, 2019].

[5] "Bayraktar Mini İHA - BAYKAR." https://www.baykarsavunma.com/iha-16.html [Online; accessed January 10, 2020].

[6] "Bayraktar TB2 - BAYKAR." https://www.baykarsavunma.com/iha-15.html [Online; accessed January 10, 2020].

[7] "Türk Havacılık ve Uzay Sanayii A.Ş, AKSUNGUR - Yüksek Faydalı Yük Kapasiteli İHA." https://www.tusas.com.tr/urun/aksungur [Online; accessed September 01, 2019].

[8] "The Top Professional Drones for Serious Commercial UAV Pilots." https://tr.gearbest.com/rc-quadcopters/pp009677692189.html [Online; accessed September 01, 2019].

[9] "Bayraktar DİHA - BAYKAR." https://www.baykarsavunma.com/iha-17.html [Online; accessed January 18, 2020].

[10] U. A. U. C. of Excellence, "US Army Unmanned Aircraft Systems RoadMap 2010-2035," 2010.

[11] ESA, "IMU, Multi Wafer Hybrid Integration: Rover IMU 1." https://robotik.dfki-bremen.de/en/research/projects/esa-imu.html [Online; accessed September 11, 2019].

[12] Syed Ali RazaWail Gueaieb, "Intelligent Flight Control of an Autonomous Quadrotor." https://www.intechopen.com/books/motion-control/intelligent-flight-control-of-an-autonomous-quadrotor [Online; accessed April 30, 2020].

[13] S. O. Madgwick, "An Efficient Orientation Filter for Inertial and Inertial/Magnetic Sensor Arrays," 2010.

[14] D. Salunkhe, S. Sharma, S. Topno, C. Darapaneni, A. Kankane, and S. Shah, "Design, Trajectory Generation and Control of Quadrotor Research Platform," 2016.

[15] B. R. Wager and M. D. Breed, "Does Honey Bee Sting Alarm Pheromone Give Orientation Information to Defensive Bees?," *Annals of the Entomological Society of America*, 2000.

[16] M. D. Breed, E. Guzman-Novoa, and Greg J. Hunt, "Defensive Behavior of Honey Bees: Organization, Genetics, and Comparisons with Other Bees," *Annual review of entomology*, 2004.

[17] A. Turgut, H. Çelikkanat, F. Gökçe, and E. Sahin, "Self-organized flocking with a mobile robot swarm," vol. 1, pp. 39–46, 01 2008.

[18] H. Çelikkanat and E. Sahin, "Steering self-organized robot flocks through externally guided individuals," *Neural Computing and Applications*, vol. 19, pp. 849–865, 09 2010.

[19] E. Sahin, S. Girgin, and E. Ugur, "Area measurement of large closed regions with a mobile robot," *Autonomous Robots*, vol. 21, pp. 255–266, 11 2006.

[20] Gazebo Simulation Software. https://www.gazebosim.org [Online; accessed January 11, 2019].

[21] O. Shea, "A Short History of Unmanned Aerial Vehicles," 2017. https://www.consortiq.com/media-centre/blog/short-history-unmanned-aerial-vehicles-uavs [Online; accessed September 02, 2019].

[22] S. Minaeian, J. Liu, and Y.-J. Son, "Vision-Based Target Detection and Localization via a Team of Cooperative UAV and UGVs," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 2016.

[23] M. Campion, P. Ranganathan, and S. Faruque, "A Review and Future Directions of UAV Swarm Communication Architectures," 2018.

[24] Y. Tan and Z. yang Zheng, "Research advance in swarm robotics," *Defence Technology*, vol. 9, no. 1, pp. 18 – 39, 2013.

[25] C. Karaağaç, "İHA Sistemleri Yol Haritası Geleceğin Hava Kuvvetleri 2016-2050," 2016.

[26] S. Kurak and M. Hodzic, "Control and Estimation of a Quadcopter Dynamical Model," 2018.

[27] S. Widnall, "Vectors, Matrices and Coordinate Transformations." https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-07-dynamics-fall-2009/lecture-notes/MIT16_07F09_Lec03.pdf [Online; accessed September 11, 2019].

[28] B. Erginer, "Quadrotor VTOL Aracın Modellenmesi ve Kontrolü," 2007.

[29] B. Siciliano and O. Khatib, "Springer Handbook of Robotics Springer Berlin Heidelberg," 2008.

[30] U. Kayasal, "Modeling and simulation of a navigation system with an IMU and a Magnetometer," 2007.

[31] M. Kumon, I. Mizumoto, Z. Iwai, and M. Nagata, "Wind estimation by unmanned air vehicle with delta wing," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 1896–1901, 2005.

[32] G. Jiang and R. Voyles, "A nonparallel hexrotor uav with faster response to disturbances for precision position keeping," in *2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)*, pp. 1–5, 2014.

[33] S. Eppinger and W. Seering, "On dynamic models of robot force control," in *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 29–34, 1986.

[34] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. 2005.

[35] J. Hu and R. Xiong, "Contact Force Estimation for Robot Manipulator Using Semiparametric Model and Disturbance Kalman Filter," *IEEE Transactions on Industrial Electronics*, 2018.

[36] F. Ruggiero, J. Cacace, H. Sadeghian, and V. Lippiello, "Passivity-based control of VToL UAVs with a momentum-based estimator of external wrench and unmodeled dynamics," *Robotics and Autonomous Systems*, 2015.

[37] J. Cayero, B. Morcego, and J. Cugeró, "Estimating External Disturbances in UAVs Using Linear and Spin Momenta," 2018.

[38] T. Tomić, C. Ott, and S. Haddadin, "External Wrench Estimation, Collision Detection, and Reflex Reaction for Flying Robots," *IEEE Transactions on Robotics*, 2017.

[39] H. S. Fabio Ruggiero, Jonathan Cacace and V. Lippiello, "Impedance control of VToL UAVs with a momentum-based external generalized forces estimator," 2014.

[40] P. J. Venhovens and K. Naab, "Vehicle Dynamics Estimation Using Kalman Filters," *Vehicle System Dynamics*, 1999.

[41] "Interactive Physics, Physics Simulation Software for the Classroom." http://www.design-simulation.com/ip [Online; accessed June 26, 2019].

[42] P. Salmony, "Attitude Estimation," http://philsal.co.uk/projects/imu-attitude-estimation [Online; accessed June 11, 2019].

[43] T. Zhang, Y. Kang, M. Achtelik, K. Kühnlenz, and M. Buss, "Autonomous Hovering of a Vision/IMU Guided Quadrotor," 2009.

[44] T. Ligard, "Methods to Measure Speed on DC Motors: RPM Measurement." https://www.brighthubengineering.com/consumer-appliances-electronics/64590-rotational-speed-characteristics-of-dc-motors [Online; accessed June 11, 2019].

[45] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*. 2016.

[46] "The AscTec Hummingbird." http://www.asctec.de/en/uav-uas-drones-rpas-roav/asctec-hummingbird [Online; accessed February 06, 2019].

[47] The Robot Operating System https://www.ros.org [Online; accessed September 11, 2019].

[48] T. Coach, "Basic Trajectories and Ballistics." http://marksmanshipresources.blogspot.com [Online; accessed October 30, 2019].