PROJECT REPORT

ON

NavIC-Based Automated Optimal Path Selection & Obstacle Avoidance Using Machine Learning

AT

ज्ञानेन प्रकाशते जगत्

INDUS UNIVERSITY

In partial fulfilment of the requirement

for the degree of

Bachelor of Technology

in

Computer Engineering

PREPARED BY

PANDYA DARSHIT AMIT (IU1641050034)

UNDER THE GUIDANCE OF

External Guide	Internal Guide
Er. Anusha Singamaneni Scientist / Engineer – ‘SD’ NRD/NTAG/SSAA Division SAC – ISRO, Ahmedabad	Mr. Hiren Mer Assistant Professor, Dept. of Computer Engineering, Indus University, Ahmedabad

SUBMITTED TO

INSTITUTE OF TECHNOLOGY AND ENGINEERING

INDUS UNIVERSITY CAMPUS, RANCHARDA, VIA-THALTEJ

AHMEDABAD-382115, GUJARAT, INDIA,

WEB: www.indusuni.ac.in

MAY 2020

CANDIDATE'S DECLARACTION

I declare that the final semester report entitled "**NavIC-Based Automated Optimal Path Selection & Obstacle Avoidance Using Machine Learning**" is my own work conducted under the supervision of **Er. Anusha Singamaneni** (External Guide) and **Prof. Hiren Mer** (Internal Guide).

I further declare that to the best of my knowledge, this report for B.Tech final semester does not contain part of the work which has been submitted for the award of B.Tech Degree either in this university or any other university without proper citation.

____

Candidate's Signature

PANDYA DARSHIT AMIT (IU1641050034)

Internal Guide:
Mr. Hiren Mer
Assistant Professor
Department of Computer Engineering
Institute of Technology and Engineering
INDUS UNIVERSITY– Ahmedabad
State: Gujarat

External Guide:
Er. Anusha Singamaneni
Scientist / Engineer – 'SD'
NRD/NTAG/SSAA Division
Space Applications Centre (SAC)
ISRO – Ahmedabad
State: Gujarat

भारत सरकार
अंतरिक्ष विभाग
अंतरिक्ष उपयोग केन्द्र
आंबावाडी विस्तार डाक घर,
अहमदाबाद-380 015. (भारत)
दूरभाष : +91-79-26913050, 26913060
वेबसाईट : www.sac.isro.gov.in/www.sac.gov.in

Government of India
Department of Space
SPACE APPLICATIONS CENTRE
Ambawadi Vistar P.O.
Ahmedabad - 380 015. (INDIA)
Telephone : +91-79-26913050, 26913060
website : www.sac.isro.gov.in/www.sac.gov.in

May 19, 2020

CERTIFICATE OF PRACTICAL TRAINING

Name of Student	:	DARSHIT PANDYA
Name of the Institution	:	INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING, AHMEDABAD
Period of Training	:	07/01/2020 to 08/05/2020
Division where Training was taken	:	NRD/NTAG/SSAA

DETAILS OF TRAINING

"NAVIC-BASED AUTOMATED OPTIMAL PATH SELECTION & OBSTACLE AVOIDANCE USING MACHINE LEARNING"

PROFICIENCY SHOWN BY THE CANDIDATE

Attendance	:	REGULAR
Conduct	:	VERY GOOD
Practical Ability	:	He has excellent practical knowledge and understanding of Machine Learning. He is quick learner. He has good proficiency in python. He is sincere and hardworking

(J Ravisankar)
Head, HRDD

भारतीय अंतरिक्ष अनुसंधान संगठन Indian Space Research Organisation

IU/ITE/CE/2020/IDP-35

INDUS INSTITUTE OF TECHNOLOGY AND ENGINEERING
COMPUTER ENGINEERING
2019 -2020

CERTIFICATE

Date: 24/05/2020

This is to certify that the project work entitled "**NavIC-Based Automated Optimal Path Selection & Obstacle Avoidance Using Machine Learning"** has been carried out by **PANDYA DARSHIT AMIT** under my guidance in partial fulfillment of the degree of Bachelor of Technology in **COMPUTER ENGINEERING (Final Year)** of Indus University, Ahmedabad during the academic year 2019 - 2020.

Mr. Hiren Mer
Assistant Professor,
Department of Computer Engineering,
I.T.E, Indus University
Ahmedabad

Dr. Seema Mahajan
Head of the Department,
Department of Computer Engineering,
I.T.E, Indus University
Ahmedabad

ACKNOWLEDGEMENT

On the successful completion of this work, I am really grateful to all the individuals who have assisted me during this journey and without whom this work would not have been possible.

Firstly, my sincere thanks to Mr. K.S.R. Swamy (Associate Dean, Indus University) for believing in my potential and recommending me for internship to the officials at the Indian Space Research Organization (ISRO). Again, I would not have been at this stage without the kind support and encouragement from Swamy Sir.

Furthermore, I am really grateful to Ms. Anusha Singamaneni (Scientist, ISRO) for assisting and guiding me throughout the completion of the project. My sincere thanks to Ms. Saumi S. (Head, SNTD Department, SAC, ISRO) for providing me with an opportunity to work on such a magnificent project. Also, I am grateful to all the members of the ISRO community who have supported and encouraged me throughout my journey at ISRO.

Next, I would like extend my sincere thanks to Prof. Hiren Mer (Assistant Professor, Department of Computer Engineering, Indus University) for his assistance, guidance and motivation. Also, I am really grateful for the support provided by Dr. Seema Mahajan (Head, Department of Computer Engineering, Indus University) during the project tenure.

Finally, I am grateful to my parents for their constant support and encouragement.

~ Darshit Pandya

TABLE OF CONTENTS

Title **Page No**

ABSTRACT

This report presents a novel approach for autonomous driving within the Indian terrain. Taking into account the advantages of the Navigation with Indian Constellation (NavIC) satellite system for the Indian terrain, this work propounds a unique approach for the functioning of an autonomous Rover prototype within the Indian subcontinent.

The primary focus of this work comprises of developing algorithms for the functioning of an autonomous rover prototype under the presence as well as absence of position coordinates (NavIC data). Accordingly, two novel Path Planning methods are proposed in this report which are focussed on assisting the rover in presence / absence of NavIC-data.

Furthermore, numerous techniques for Obstacle Avoidance have been penned in this report. Considering the complexity of the Indian terrain, numerous techniques dealing with such complex obstacles have been discussed within this report. In essence, this work proposes several obstacle avoidance techniques considering mainly three types of obstacle: Stationary/Moving Obstacles, Potholes and Slopes. Both, obstacle detection and obstacle avoidance techniques, have been proposed and elaborated in this report.

Cost optimization represents the secondary aim of this work. In order to minimize the cost of development as well as the cost of functioning of the rover, minimal amount of sensors have been used. Accordingly, data from only two sensors: Ultrasonic and INS, has been utilized in this work. Moreover, this work comprises of only the software modules and no hardware materials have been utilized. However, only the data from the hardware parts has been utilized. Also, this work is purely based on Machine-Learning technology.

In addition, numerous tests have been carried out for the evaluation of the algorithms proposed in this report. A comprehensive test report has been presented in this report for better understanding of the performance of the proposed algorithms under real-world conditions. Only real-world test cases have been considered and the simulation of the rover under such cases has been presented in this report.

ORGANIZATION PROFILE

Indian Space Research Organization (ISRO)

The Indian Space Research Organisation is the space agency of the Government of India and has its headquarters in the city of Bengaluru. It foundation dates back to August 15, 1969 on the 22^{nd} Independence day of the Republic of India. It was founded under the presence of India's most celebrated scientist Dr. Vikram Sarabhai. The vision of this reputed organization is and has been:

"To harness space technology for national development while pursuing space science research & planetary exploration"

The Indian Space Research Organization is currently among the Top – 5 space research organizations in the world. Major launchings of ISRO includes:

- Chandrayaan – 1 *(October, 2008)*
- Chandrayaan – 2 *(July, 2019)*
- Mangalyaan *(November, 2013)*

Apart from space applications, ISRO is directly involved in numerous social-work activities. Some of such activities include educating school kids about space technologies, conducting various seminars for enlightening the knowledge of space research among the general public and many more. In addition, ISRO provides various internship and apprenticeship opportunities to the undergraduate & graduate students in order to assist them for their final semester thesis.

On the technical end, ISRO holds an outstanding record of more than 5000 major publications in both national as well international journals. Moreover, ISRO is home to more than 10, 000 scientists and researchers, excluding the supporting staff.

As of now, ISRO has become the pride of India. ISRO symbolizes India's talent, hard work and determination on international level. Accordingly ISRO has been named the best government organization in India.

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Abbreviation	Text
GPS	Global Positioning System
NavIC	Navigation with Indian Constellation
IRNSS	Indian Regional Navigation Satellite System
ISRO	Indian Space Research Organization
IDE	Integrated Development Environment
SVM	Support Vector Machine
SVC	Support Vector Classifier
INS	Inertial Navigation System
SMOA	Stationary/Moving Obstacle Avoidance
PDA	Pothole Detection & Avoidance
SAA	Slope Analysis & Avoidance
DBM	Distance-Based Method
GM	Greedy Method
SAS	Side-Angle-Side
SSS	Side-Side-Side

CHAPTER – 1
INTRODUCTION

- PROJECT SUMMARY
- PROJECT PURPOSE
- PROJECT SCOPE
- PROJECT OBJECTIVES
- TECHNOLOGY OVERVIEW
- SYNOPSIS

1.1 PROJECT SUMMARY

This work presents a new approach for reliable autonomous driving, with minimal power consumption, for the Indian terrain. Confined by the limitations of Global Positioning System (GPS) and stimulated by the advantages of Navigation with Indian Constellation (NavIC), this project presents a contemporary approach to autonomous driving. Two main concerns regarding autonomous driving, namely Obstacle Avoidance and Path Planning, are addressed in this work. The primary goal of this work is to fabricate a well-grounded Machine-Learning model for facilitating the autonomous rover to work competently in the absence of position coordinates, basically NavIC coordinates. Also, this system solely relies on ultrasonic & INS sensors only for less power consumption and less computational complexity. Minimal efforts were required for training the rover to work in the presence of NavIC data due to the presence of currently existing techniques. However, the challenging part of making the rover work in the absence of NavIC has been unravelled by the two methods proposed in this work, namely Greedy and Direction-Based.

Considering the complexity of the Indian terrain, the Obstacle Detection & Avoidance domain has been addressed precisely. All sorts of obstacles including, but not limited to, slopes, humps, potholes, moving & stationary objects have been contemplated thoroughly and the rover’s Machine-Learning model is trained accordingly. In addition, diversion decisions have been manipulated according to the real-world conditions. Tremendous amount of data has been collected and utilized to train the model in order for it to work as meticulously as possible. Although bounded by the time constraints, the models have been trained and tested according to the real-world conditions to the best extent possible.

Furthermore, over 1 TB of data has been collected and has been utilized for training the Machine-Learning models. This work would surely form the base for research on autonomous driving over the Indian lands. Adding to it, one more application has been kept in mind during the development phase: Extra-terrestrial Applications. Though this project primarily focusses on the Indian terrain, the proposed Path Planning methods and Obstacle Avoidance techniques have been constructed such that the rover could be used for Planet-Explorations in Space.

Being a valuable research project, this work has been carried on with utmost care and precision. Resultant accuracy of all the Machine-Learning models employed in this work have been refined to the best possible extent considering the time constraints.

1.2 PROJECT PURPOSE

Following the launch of India's own navigation satellite system – NavIC, numerous unique applications of NavIC have emerged, one of which is discussed in this report. After reviewing plenty of research articles on GPS-based autonomous vehicles, there appeared a scope for improvement of the existing techniques with the help of NavIC. Considering the advantage of NavIC's precision for the Indian subcontinent, the project is focussed at improving the performance of an autonomous rover by training the rover to work in presence as well as in absence of NavIC along with reliable obstacle avoidance for the Indian terrain. The project is primarily focussed on optimization of path-planning & obstacle avoidance within the boundaries of Indian land along with minimal power consumption and less complexity. Developing such a robust system would ameliorate the possibilities of implementing autonomous vehicles on the Indian terrain. The project would serve various end-user applications which are listed later on in the report. The project is also aimed to support autonomous rovers used in the extra-terrestrial applications, ex. Vikram Lander (Chandrayaan-II, 2019), to work in absence of orbiter's geographic data.

1.2.1 NavIC - Navigation with Indian Constellation

As a replacement of Global Positioning System (GPS), NavIC is India's first navigation satellite system primarily focussed on providing positioning and timing services for the Indian subcontinent. NavIC is the operational name for the Indian Regional Navigation Satellite System (IRNSS) programme initiated by the Indian Space Research Organization (ISRO) [1]. The NavIC system consists of seven satellites with the first satellite launched in 2013 and the latest one in 2018. The orbital height is 36,000 kilometres. The operational accuracy of NavIC has been proved to be better than the Global Positioning System (GPS) for the Indian subcontinent.

On comparison to GPS, NavIC provides better accuracy and compatibility with respect to the Indian subcontinent. This benefit serves as an opportunity to develop new applications pertaining to the scope of the Indian land. Accordingly, this project aims to utilize this advantage of NavIC over GPS in order to build a reliable autonomous rover for the Indian land. The project aims to train the rover to work effectively even in the absence of NavIC. This feature contributes to the space applications as well. This scope of space applications & explorations strengthens the future scope of the current project and would be a topic of interest for further research.

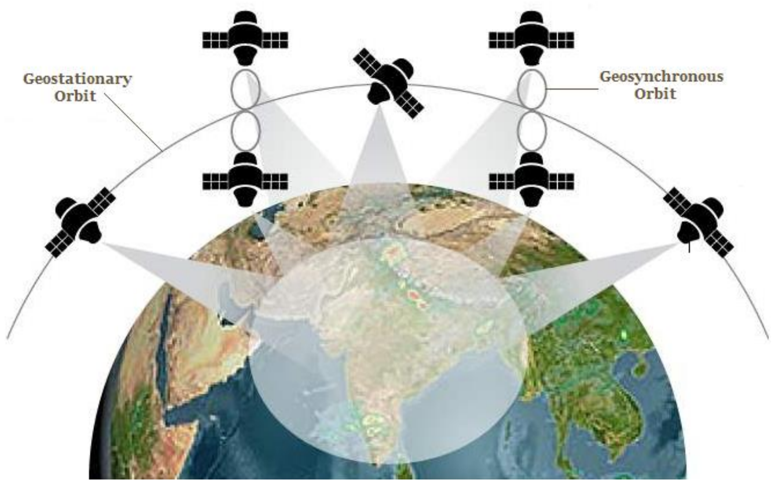

Fig 1.1 NavIC Satellite Formation

1.2.2 Indian Terrain

Terrain is a dominant feature in the autonomous driving domain. In developed countries, the autonomous driving works great due to the surfaced roads as well as abundant markings. India, being a developing country, neither has consistent roads nor sufficient markings. This serves as a major challenge while developing an autonomous system for the Indian terrain. This project work aims to accept this challenge and output a breakthrough performance on the Indian Terrain.

Considering the Indian Terrain, there are basically two dominant categories of uneven land: Potholes and Slopes. Both of these asymmetrical formations serve as a major challenge for the autonomous rover. These challenges are addressed precisely in this work and are prioritized highly for obstacle avoidance.

1.2.3 Motivation from Chandrayaan-II

Chandrayaan-II was India’s second Moon mission. It was launched on 22 July, 2019. This mission was aimed at landing a rover, named Vikram Lander, on the Moon’s South Pole. However, due to loss of connection, the lander couldn’t land safely on the lunar surface and it crashed. However, the orbiter is still orbiting the Moon successfully.

From this incident, it is clearly evident that connectivity can decide the success of a mission. Accordingly, this work focusses on training the rover for working under such connectivity loss. It could also be the case that the rover would have landed successfully but couldn’t move due to an upcoming connectivity loss. In such scenarios, it becomes essential to develop a robust system that can work on its own. Such a system is discussed in this report.

1.3 PROJECT SCOPE

This work involves working with data from satellites and sensors. Accordingly, it was essential to acquire knowledge of each of these satellites and sensors. This generated a major scope for understanding the data generated from such entities and also directed on the usage of such data for the required output. This project, being a product-based system, has numerous target entities. Consequently, it possesses a much broader scope. Considering the end-user applications, the three major target users are listed in Table 1.1.

Table 1.1 List of Target Users

End Users	Role(s)
Logistics Firm	Delivery of small goods
Mining Firm	Search operations
Municipality	Inspection of roads / highways

Considering the research perspective, this work would serve as a base of research for scientists working in the domain of autonomous driving. For example, the Machine Learning models used in this work may be changed as well as the pre-processing of data may be improved. As this work represents a base prototype, it can be used for varied applications by just adding some required sensors or any other entity as and when needed. Furthermore, considering the Space Research domain, the researchers may work on the improvement of the proposed methods as this work has Considered Space Applications too.

1.4 PROJECT OBJECTIVES

Following are the main objectives of this work:

1. To understand the scope of usage of NavIC data in autonomous driving.
2. To develop a Machine-Learning model for Obstacle Avoidance.
3. To develop a Machine-Learning model for Path-Planning.
4. To test the Machine-Learning models against previously planned paths & terrain.
5. To evaluate the results for accuracy and consistency.
6. To refine the results to the best extent.

All of the above mentioned objectives have been accomplished, within this project work, to the best extent possible.

1.5 TECHNOLOGY & LITERATURE OVERVIEW

1.5.1 TECHNOLOGY USED

The core of this project is based on Python & some Machine-Learning Libraries. Table 1.2 lists out the complete bundle of technologies used.

Table 1.2 List of Technologies Used

Category	Technology
Programming Language(s)	Python v3.7
IDE	Spyder v3, Anaconda v3
ML Libraries	Scikit Learn, Pandas, NumPy, MatplotLib
ML Models	Polynomial Regression, Support Vector Machine

1. **Python**

   Python is an interpreted, high-level, general-purpose programming language. It was first released in 1991 and the latest version Python v3 in 2008. It is the most widely used programming language in the domain of Machine-Learning and Data-Analytics. This project is based on Python v3. All of the code written for this work is done in Python v3.

   After analysing various object-oriented programming languages, Python v3 has been found the most suitable programming language for this work in terms of its rich Machine-Learning library rack as well its simplicity and readability.

2. **Integrated Development Environment (IDE)**

   For implementation purpose, Anaconda v3 has been utilized as it is a complete pack consisting of Python compiler, Python IDE (Spyder v3) and various Python compatible Machine-Learning libraries.

3. **ML Libraries**

   This project, being fully based on Machine-Learning, has taken into account various well-known Machine-Learning libraries, such as Scikit Learn, Tensorflow, Pandas, Pickle, NumPy & MatplotLib, which are all compatible with Python v3. Each of these libraries have been specifically designed for Machine-Learning task implementation and all are packed into Anaconda v3.

4. **Machine Learning**

Following the development of Artificial Intelligence, Machine-Learning is currently the most booming field. It is utilized widely as it gives the power of learning to the computing machines. Following this capability, Machine-Learning turns out to be the best choice for this project. One major goal of this project is to train the rover prototype based on numerical data. Accordingly, as Machine-Leaning simplifies the task of complex numerical calculations, it becomes the best choice for this work.

Overall, many libraries are used in this project such as Scikit Learn, Tensorflow, Pandas, Pickle, NumPy & MatplotLib. Though they are all equally important, Scikit Learn is the chief library of them all as all the basic Machine-Learning models are bundled within it. Out of the numerous Machine-Learning models available, based on the type of data, two of them are used in this project work:

- **Polynomial Regression**

  According to the type of data and the desired output, Polynomial Regression has been found to be a best-fit for various parts of this work. It has been majorly used in the Obstacle-Avoidance phase. It basically tries to fit a curve of n^{th}-degree polynomial on the training data fed to it. On the output end, it outputs a numerical value by fitting the given testing parameters into the n^{th}-degree polynomial [2].

- **Support Vector Machine (SVM)**

  On the other end, some tasks in this work require classification of the given data. This classification task is fulfilled by the Support Vector Classifier (SVC) – a part of Support Vector Machine (SVM) [3]. After testing many of the classifiers over the training data, Support Vector Machine (SVM) turned out to be the best classifier based on the type of training data fed to the model.

As mentioned above, all of these Machine-Learning models selected for training the data have been tested and compared with other models as well. After the analysis of this comparison, Polynomial Regression and Support Vector Machine (SVM) turned out to be the best available models for this project work.

1.6 SYNOPSIS

This report provides a detailed explanation of the complete procedure carried out for accomplishing the desired outcomes of the project work, done as a part of final year thesis leading to a Bachelor's degree. The problem statement is thoroughly delineated in Section 1.1 of Chapter-1. In addition, the purpose, motivation and objectives of the work are comprehensively covered under Chapter-1. Also, the collection of technology utilized in this work are recapitulated in Chapter-1. Following this, a complete background survey for this work, including research articles and thesis, are detailed out in Chapter-2. Furthermore, the project plan & schedule is discussed in Chapter-3. A glimpse of the hardware and software requirements are listed out in Chapter-4. Next, Chapter-5 details the core techniques and proposed algorithms of the project. This is a key chapter as it covers a detailed description of the techniques used for Obstacle Avoidance as well as the proposed methods for effective Path Planning. All of the derived results and testing outcomes are briefed in Chapter-6 along with the testing techniques used for it. The significance of this work becomes evident by going through this chapter. Every project work has some limitations and a future scope. Chapter-7 covers such limitations. Finally, the conclusion of this project work is penned out in Chapter-8. This chapter concludes this project work.

CHAPTER – 2
LITERATURE SURVEY

- **LITERATURE REVIEW**

2.1 LITERATURE REVIEW

For the purpose of understanding the background of autonomous driving, a bunch of research articles as well as thesis have been reviewed before progressing with this report. Accordingly, the fundamental concepts to be understood for proceeding with this work are the previously implemented algorithms for autonomous driving. Below are the few literatures considered and thoroughly studied for this project:

- **A Path Planning And Obstacle Avoidance Algorithm For An Autonomous Robotic Vehicle** – *Sharayu Yogesh Ghangrekar* [Thesis]

  This thesis provides a similar perspective to what the current work aims at. A number of Path Planning and Obstacle Avoidance techniques have been discussed in great detail [4]. However, as only known data is used, this work has only been considered as a starting point for the current project work.

- **Autonomous Mobile Robot Navigation Using Machine Learning** – *Xiang S.*

  This research article presents a novel approach to move a robot out of a maze using the readings of ultrasonic sensors [5]. This kind of robot navigation can be termed as intelligent navigation as the robot solves the maze optimally with a very few errors. The current project work has been greatly influenced by the decision-making algorithm of this article for navigation.

- **Implementation of Vehicle Speed Reducing System at Speed Breaks by Detecting Potholes and Humps Using Ultrasonic Sensor** – *V. Kalpana*

  This research article proposes different ways for Obstacle Avoidance, specifically humps and potholes, through the readings of ultrasonic sensors [6]. As the current project aims at working solely on the basis of Ultrasonic & INS sensors, this article has been very informative with respect to the current project.

By going through the above articles, it is evident that most of the focus is on algorithms based on reliable navigation and obstacle avoidance using ultrasonic sensors only. This corresponds with the aim of this project as this work aims to use minimal amount of sensors, only ultrasonic and INS, in order to reduce complexity as well as power consumption. Furthermore, many other research articles have also been reviewed. However, as most of them relied on computer vision and image processing techniques, they did not comply with the aim of this project, i.e. only ultrasonic sensors to be used, and hence are disregarded.

CHAPTER – 3
PROJECT MANAGEMENT

- **PROJECT PLANNING OBJECTIVES**
- **PROJECT SCHEDULING**

3.1 PROJECT PLANNING OBJECTIVES

3.1.1 PROJECT DEVELOPMENT APPROACH

Due to the complex nature of the project, it has been completed in two phases:

⦿ **OBSTACLE AVOIDANCE PHASE** **[PHASE - I]**

In this phase, three different modules are implemented simultaneously:

- Stationary/Moving Obstacle Avoidance (SMOA)
- Pothole Detection & Avoidance (PDA)
- Slope Analysis & Avoidance (SAA)

Fig 3.1, Fig 3.2 and Fig 3.3 illustrates the development process of SMOA, PDA and SAA respectively.

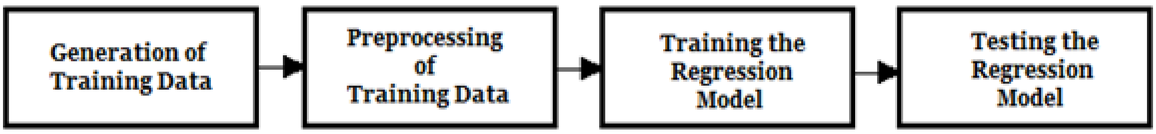

Fig 3.1 Levels of SMOA

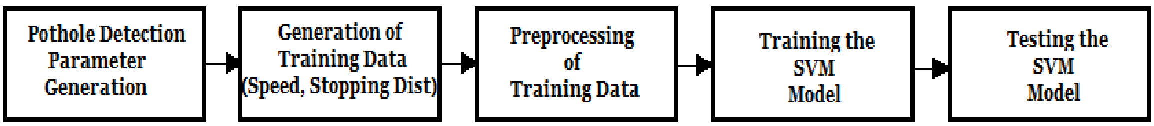

Fig 3.2 Levels of PDA

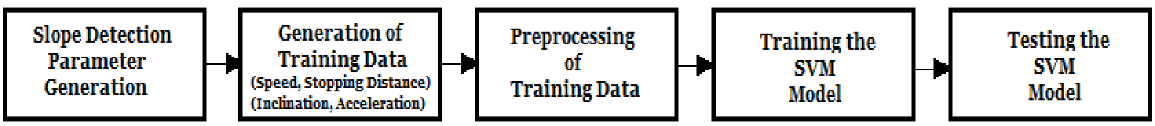

Fig 3.3 Levels of SAA

Basically, some steps of the above processes are common, namely Generation of data, Training and Testing of the Machine-Learning model. Here, maximum time has been consumed by the data generation and training levels. Approximately, around 1 TB data was generated. Here, the training processes for each of these modules have been conducted simultaneously by generating three different processes that run concurrently through Multiprocessing.

⦿ PATH PLANNING [PHASE-II]

In this module, two path planning methods are proposed:

- **Direction-Based Method (DBM)**

  This method is proposed to solve the challenge created by the absence of NavIC data. As this method is proposed without any past reference, it has been a complex and time-consuming task to implement this method. However, an incremental approach has been chosen to improve the algorithm by making necessary changes according to the results obtained in the previous simulation tests. The levels of implementation can be visualized from Fig 3.4.

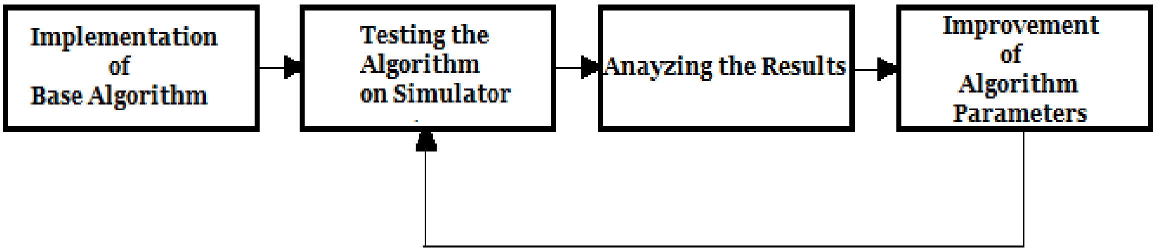

Fig 3.4 Levels of DBM

- **Greedy Method (GM)**

  This method is a new approach towards optimization of paths. A very simple approach is followed in this method. Hence its implementation is not too complex. Fig 3.5 demonstrates the levels of implementation of this method.

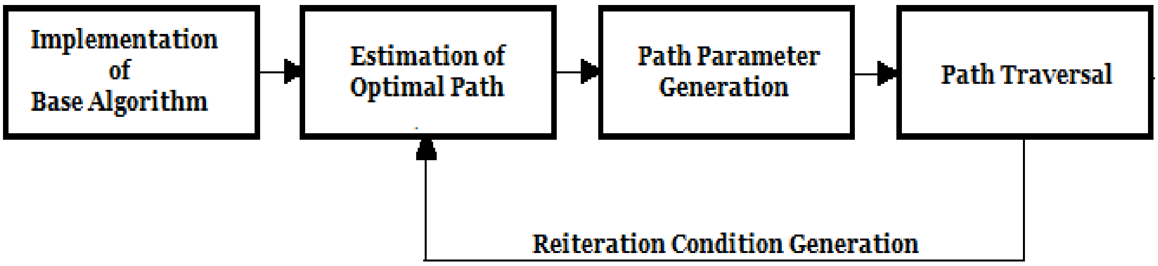

Fig 3.5 Levels of GM

After the successful implementation of the above mentioned algorithms, the final stage of implementation would be the merging of modules of Phase-I & II.

3.2 PROJECT SCHEDULING

3.2.1 TIMELINE CHART

Though the time constraint for this project work had been four months, this project has been completed around three-and-a-half months. Maximum amount of time has been spent on the Data Generation and Model Training sections. Furthermore Module-Testing and Final Simulation Testing also took considerable amount of time.

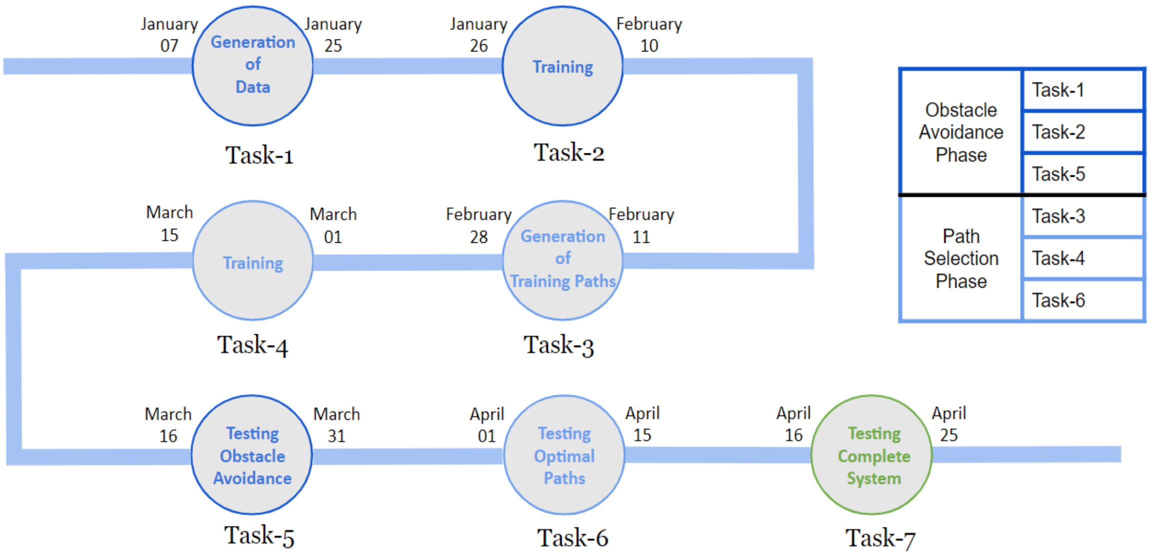

Fig 3.6 Timeline Chart

CHAPTER – 4
SYSTEM REQUIREMENTS

- HARDWARE REQUIREMENTS
- SOFTWARE REQUIREMENTS

4.1 HARDWARE REQUIREMENTS

The complete hardware requirements have been listed out in Table 4.1 below.

Table 4.1 Hardware Requirements

Hardware	Description	
Workstation ***[for Training Models]***	**Processor**	Intel® Xeon®
	Cores	12
	Clock Speed	3.60 GHz
	RAM	64 GB
Sensor(s) ***[for Sensor Readings]***	Ultrasonic Sensor, INS	
Satellite ***[for positioning data]***	NavIC (IRNSS)	

4.2 SOFTWARE REQUIREMENTS

The complete software requirements have been listed out in Table 4.2 below.

Table 4.2 Software Requirements

Category	Software
Programming Language(s)	Python v3.7
Integrated Development Environment	Sypder v3, Anaconda v3
Libraries	Scikit Learn, Pandas, NumPy, MatplotLib
Machine-Learning Models	Polynomial Regression, SVM
Operating System	Windows 10 / Linux

CHAPTER – 5
DETAIL DESCRIPTION

- **PHASE – I : OBSTACLE AVOIDANCE**
- **PHASE -II : PATH PLANNING**
- **INTEGRATION OF PHASE – I & PHASE – II**

5.1 PHASE – I : OBSTACLE AVOIDANCE

As discussed earlier, this project has been implemented in two phases – Obstacle Avoidance and Path Planning. The first phase is detailed in this section. While considering autonomous driving, the first challenge that pops up is obstacle avoidance. This work has addressed this challenge in great depth, following the characteristics of the Indian terrain.

One might think of an obstacle as just a stationary object such as a car. However, such objects forms just a small part of the complete obstacle list. Many other severe obstacles are a part of the Indian terrain, such as potholes, humps and slopes. These obstacles cannot be avoided by using a single common technique. Accordingly, each type of obstacle has been addressed by different avoidance techniques as listed below:

5.1.1 STATIONARY / MOVING OBSTACLE AVOIDANCE (SMOA)

It is evident from the name that stationary obstacles include car, tree or any other stationary object. Moving obstacles might include cars or any other moving object. The technique for avoiding such obstacles is simple. The implementation of this avoidance technique works as follows:

- Firstly, Recording Ultrasonic Sensor reading
- Secondly, Calculating relative speed
- Finally, Diversion prediction using Machine-Learning Model

Here, it is understood that the first step just includes fetching readings from the ultrasonic sensor. The next step consists of calculating relative speed. This is in context of a moving obstacle. The relative speed can be measured with a simple formula:

$$\text{Speed}_{\textit{Relative}} = \text{Speed}_{\text{Rover}} - \text{Speed}_{\text{Obstacle}} \quad (5.1.1)$$

Here, in Equation 5.1.1, if a stationary obstacle is present, then consider $\text{Speed}_{\text{Obstacle}} = 0$. Now, the key functionality of this technique lies in the final step. This stage includes the use of Machine-Learning to make a decision. For the implementation of this step, Training and Testing data should be generated and then some Machine-Learning model has to be deployed against this generated data. It may sound as a complex procedure, however, by attempting it step-by-step greatly simplifies the procedure. Also, diversion decision, i.e. whether to go left or right after avoiding the obstacle, is also included in this final step of the technique. The final step is explained in detail below:

- **Application of Machine-Learning**

  As discussed above, this section describes the complete implementation procedure of Machine-Learning. Here, two procedures are followed:

  - **Avoidance Decision**

    Here, a decision is to be made whether or not to avoid an obstacle based on the distance to the obstacle and the current speed. For this task, a Machine-Learning model has to be trained. Accordingly, data is needed for the implementation of the model. This method includes:

    - **Generation of Training Data**

      This task includes the collection of data. For this project, the data has been generated by simulation of real-world conditions. For Avoidance Decision, two types of data are needed for Training the Machine-Learning model: Speed $_{Rover}$ and Stopping Distance. The Speed $_{Rover}$ has been collected from the INS sensor. Adding to it, the Stopping Distance records are calculated by measuring the distance travelled by the rover after applying brakes at the current speed. Reaction distance is also added to it. This process has been demonstrated in Fig 5.1 below.

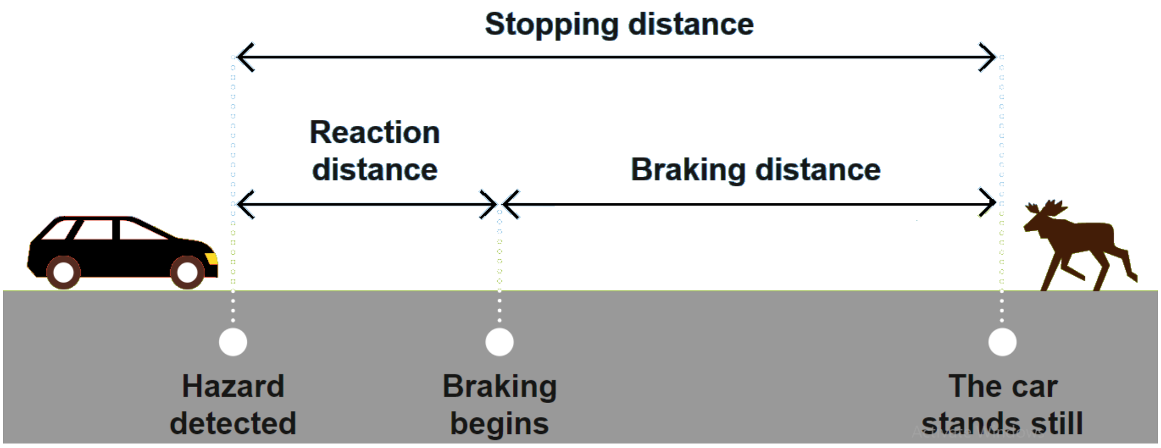

Fig 5.1 Braking Distance Demonstration

As seen from the figure, Stopping Distance is the total of Reaction Distance and Actual Braking Distance. Here, however, the reaction distance is considered as negligible. Following this, around 50,000 records (Speed, Stop. Dist.) have been collected.

- **Training the Machine-Learning Model**

  After the generation of data, the next step is selection of Machine-Learning model. For this work, Polynomial Regression model has been deployed as it has proved to be the most accurate model for the generated data. Next step is Training the Machine-Learning model. This has been accomplished by using the Scikit Learn library. Fig 5.2 visualizes the Training phase of this model.

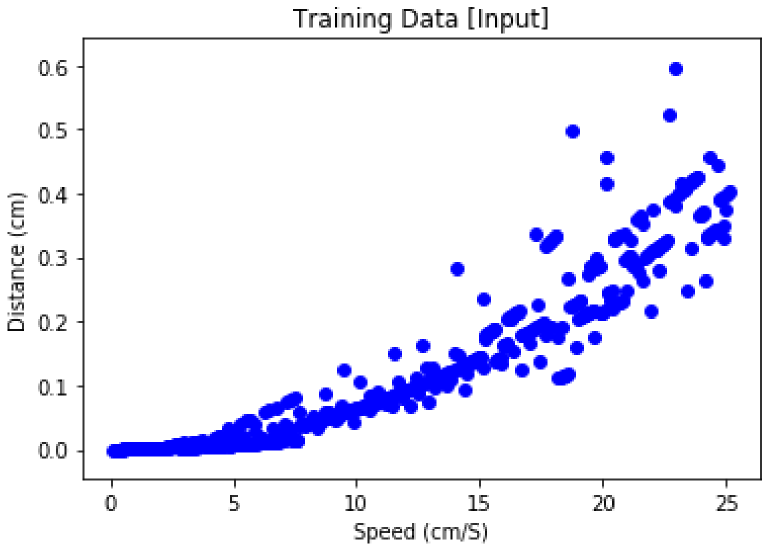

Fig 5.2 Avoidance Decision Training

- **Testing the Machine-Learning Model**

  Testing output, after fitting a polynomial of degree-2, is visualized in Fig 5.3. Accuracy: *99.86%*.

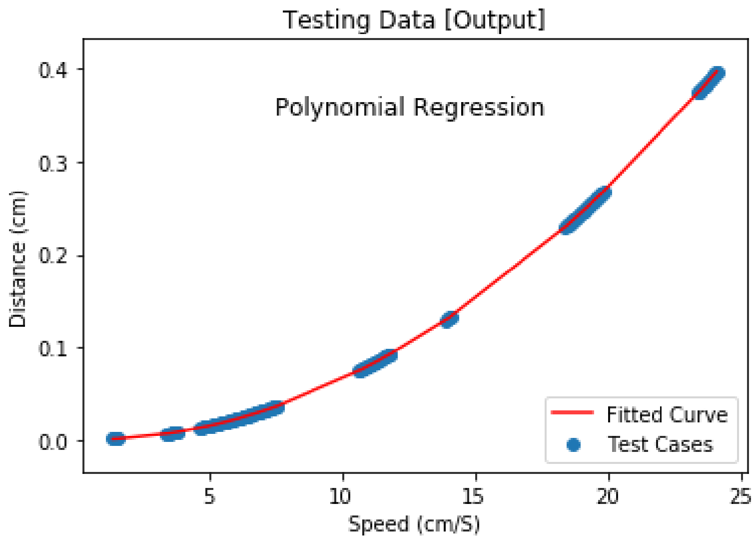

Fig 5.3 Avoidance Decision Testing

- **Core Logic**

  Here, the core logic is to predict the Stopping Distance by giving the Speed $_{Rover}$ as an input to the model. From this distance and the current distance from the Obstacle, available through the ultrasonic sensor, a decision is made regarding whether is it urgent to divert in order to avoid the obstacle or is it safe to move forward.

- **Diversion Decision**

  Here, after confirmation of Avoidance Decision, a decision is to be made on whether to turn Left or Right based on Ultrasonic sensor readings. In other words, if an Avoidance Decision turns out to be a Divert, then the direction of diversion has to be determined. This has been achieved by implementing Machine-Learning algorithm on the data available from the Left and Right Ultrasonic sensors. This has been achieved by following the below mentioned procedure:

  - **Generation of Training Data**

    Data needed for Training the model has been collected by simulation of the rover in real-world conditions. Here, the data from Left and Right Ultrasonic sensors have been taken as input and the direction of diversion has been taken as label.

  - **Training the Machine-Learning Model**

    As this is a classification problem, Support Vector Machine (SVM) has been found to be the best fit model for this type of data. Accordingly, the Machine-Learning model is trained with the data generated in the previous step.

  - **Testing the Machine-Learning Model**

    After testing the trained model, the final output results are available in Fig 5.4. Here, the readings of Left and Right Ultrasonic sensors are provided as input to the trained model and a Diversion Decision is generated as output. Braking option, without any diversion, has also been considered. Accuracy obtained has been around *98.69%*. Total Records: 30,000.

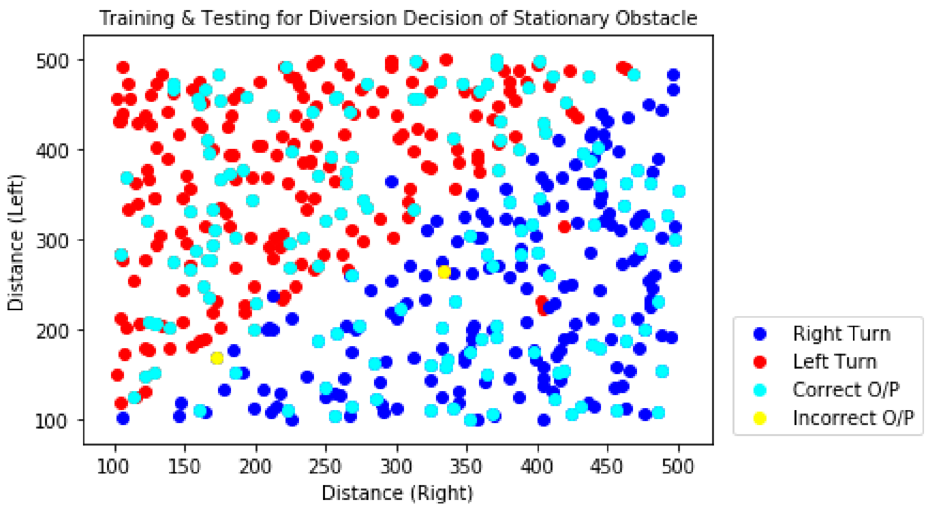

Fig 5.4 Diversion Decision Training & Testing [SMOA]

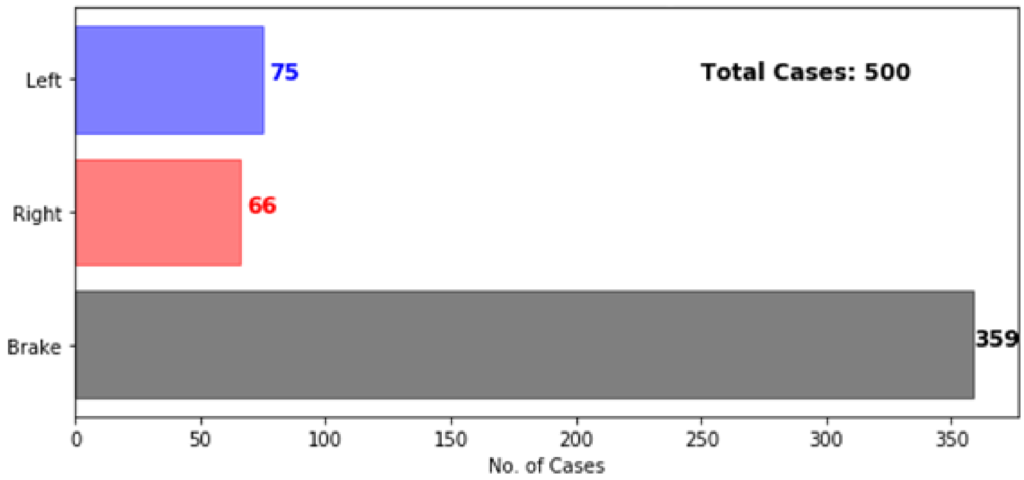

Fig 5.5 Instances for Diversion Decision

The above information provides a detailed insight into the implementation of the first part of PHASE – I (OBSTACLE AVOIDANCE), namely Stationary / Moving Obstacle Avoidance (SMOA). The net accuracy of both the parts, Avoidance Decision and Diversion Decision, has been up-to-the-mark. These models have been refined by pre-processing of the generated data using general noise removal techniques. Consequently, the accuracies obtained are at their best level.

The testing information and plots mentioned above are for the testing purpose of the model only. This does not represent the final testing against a stationary or a moving obstacle. The final Obstacle Avoidance testing for Stationary / Moving Obstacle Avoidance (SMOA) is to be followed in Chapter-6. Next section consists the information on implementation of Pothole Detection and Avoidance (PDA).

5.1.2 POTHOLE DETECTION AND AVOIDANCE (PDA)

One of the most consistent characteristics of Indian Terrain is Pothole. Almost every 15 to 20 meter distance, there is pothole presence on the India terrain. Due to this high frequency of potholes, the possibility of encountering potholes is high. For this possibility, potholes serve as a major challenge in the Obstacle Avoidance Domain. This section describes in detail, the implementation of the techniques used to detect and avoid potholes.

It is evident from the section heading that this challenge consists of two parts:

- Pothole Detection
- Pothole Avoidance

Each of these challenges have been addresses in detail within this section. The key part of this stage is Pothole Detection. This is explained in detail below:

- **Pothole Detection Technique**

  Pothole Detection is necessary considering the Indian terrain. According to a survey, there are more than 15 million potholes throughout the Indian lands. Accordingly, this work has taken the Pothole detection phase with serious consideration.

  This technique involves placing an Ultrasonic sensor on top of the rover. This sensor is placed in addition to all the other Ultrasonic sensors used previously. The positioning of this sensor is demonstrated in Fig 5.6 below.

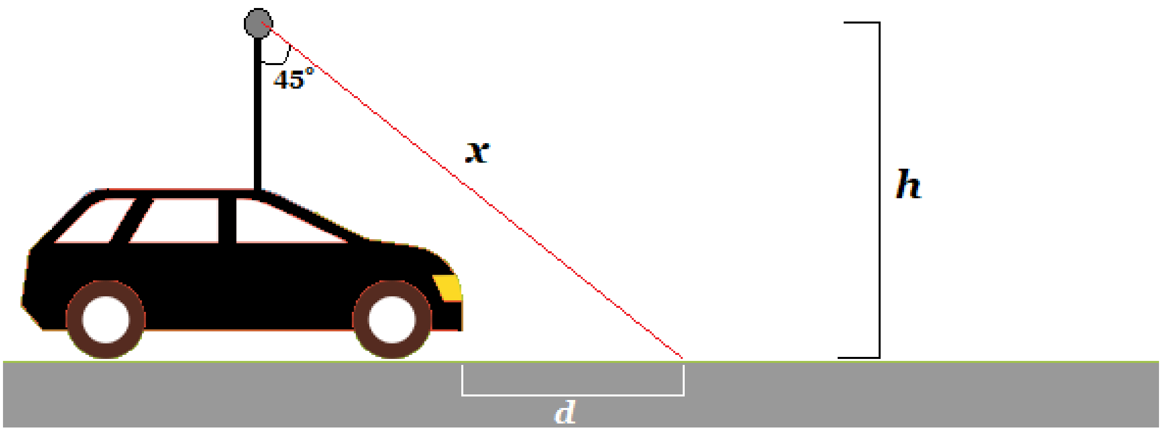

Fig 5.6 Sensor Placement [PDA]

It is evident from Fig 5.6 that as sensor angle (45°) and height (h) are fixed, the value of distance d and the value of reading x are also fixed under any condition.

Now, Fig 5.7 demonstrates the further logic for detecting the presence of a pothole.

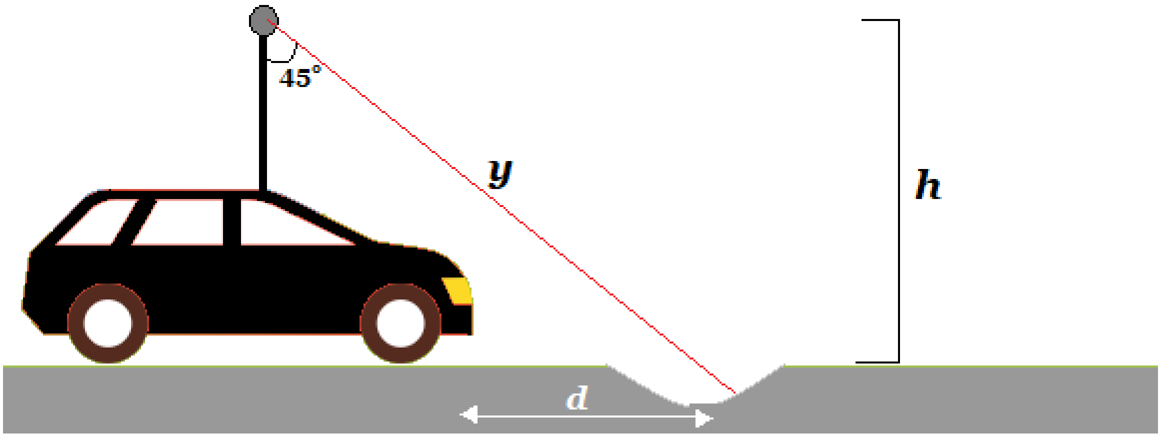

Fig 5.7 Pothole Detection

Now, comparing Fig 5.6 and Fig 5.7, a simple logic can be formed as follows:

```
if y > x:
        Pothole Present
else:
        Pothole Absent
```

As the pothole is detected, next challenge is to determine whether it is safe to cross the pothole or diversion is necessary. For this, Machine-Learning has been utilized. The following process has been followed:

- **Generation of Training Data**

  The data needed for training the Machine-Learning model includes: Speed $_{\text{Rover}}$, Pothole Distance *d* (Fig 5.7), Stopping Distance and the output label (Safe or Unsafe). Such type of data has been collected by simulation of the rover in real-world conditions. Here, the Stopping Distance was predicted using the module illustrated in Section 5.1.1 for Stationary / Moving Obstacle Avoidance (SMOA). Records: 15,000.

- **Training the Machine-Learning Model**

  Coming to the model selection part, this problem depicts a classification requirement. Accordingly, on comparing accuracies of different classification algorithms, SVM turned out to be the best classifier. Consequently, 70% of the generated data has been used for training.

- **Testing the Machine Learning Model**

  On the testing side, 30% of the unutilized generated data has been used. Speed $_{\text{Rover}}$, Pothole Distance *d* (Fig 5.7), Stopping Distance are provided as inputs and Safe/Unsafe label is generated as output. The accuracy turned out to be around 99.96%. Fig 5.8 plots the resultant output.

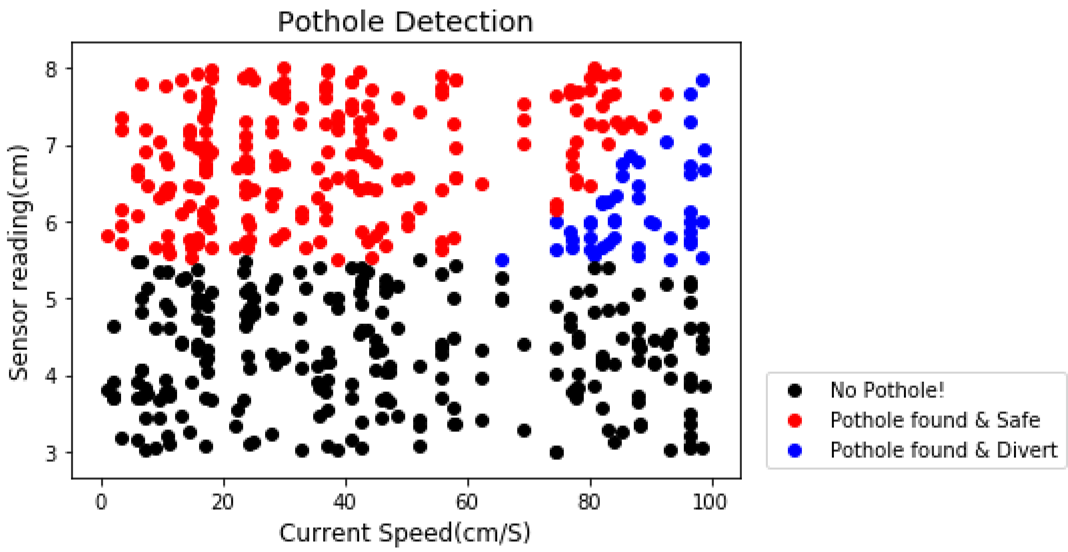

Fig 5.8 Pothole Detection Results

It is evident from the figure that in the areas of low sensor reading and high speeds, the diversion decision is generated, which is fair. Hence, in other words, if at current speed, the stopping distance > pothole distance, diversion occurs.

Another point to be noted from the above plot is, the normal reading for the Top mounted ultrasonic sensor (Fig 5.6) is around 5.5 as for all sensor readings above 5.5 predicts the presence of a pothole.

⊙ **Pothole Avoidance Technique**

Following the decision made in the previous step, Safe or Divert, the next thing to be determined is: in which direction to divert, Left or Right, if the decision in the above step turns out to be a Divert Decision. By taking this decision, potholes can be avoided. Accordingly, for making this decision, Machine-Learning technology has been used. Subsequently, the Machine-Learning model has been trained by sticking to the steps mentioned in the following documentation:

- **Generation of Training Data**

  Data needed for Training the model has been collected by simulation of the rover in real-world conditions. Here, the data from Left and Right Ultrasonic sensors have been taken as input and the direction of diversion has been taken as label.

- **Training the Machine-Learning Model**

  As this is a classification problem, Support Vector Machine (SVM) has been found to be the best fit model for this type of data. Accordingly, the Machine-Learning model is trained with the data generated in the previous step.

- **Testing the Machine-Learning Model**

  After testing the trained model, the final output results are available in Fig 5.9. Here, the readings of Left and Right Ultrasonic sensors are provided as input to the trained model and a Diversion Decision is generated as output. Braking option, without any diversion, has also been considered. Accuracy obtained has been around *98.69%*. Total Records: 30,000.

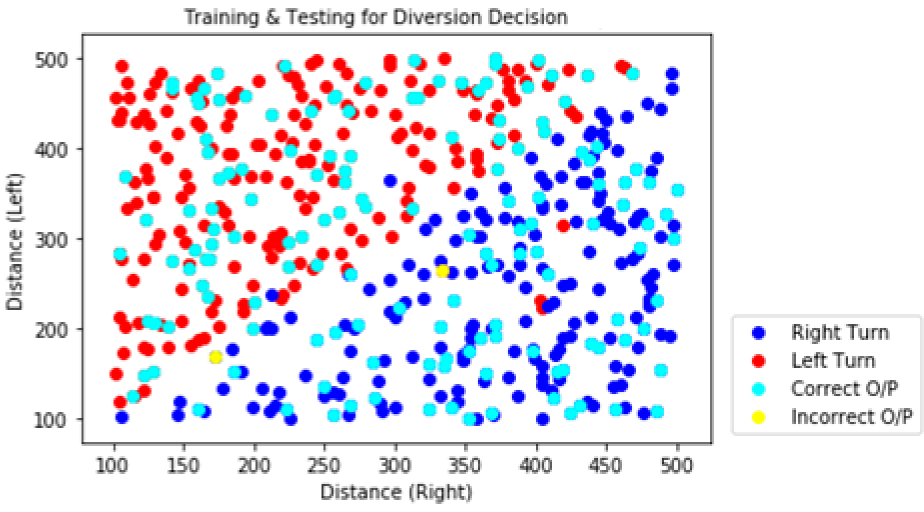

Fig 5.9 Diversion Decision Training & Testing [PDA]

On a concluding note, the methods for Pothole Detection & Pothole Avoidance, illustrated above, have resulted on being a reliable method for detecting potholes and avoiding them as the output accuracy for both the methods are up-to-the-mark. These models have been refined by pre-processing of the generated data using general noise removal techniques.

5.1.3 SLOPE ANALYSIS AND AVOIDANCE (SAA)

Another consistent characteristic of Indian terrain is Slope. In other words, it may be any kind of hump. These humps also pose a great challenge to autonomous driving domain. This module is difficult to implement as well as understand as compared to the pothole detection module as this module includes various complex stages for detection as well as avoidance. However, such complex techniques have been illustrated within this report in the simplest form possible.

Making things simple, this module has been implement in two stages:

- **Slope Analysis & Detection**

  Under this section, as the name suggests, there are two phases namely Analysis and Detection. For easy understanding, the Slope Detection phase is illustrated first.

  - **Slope Detection**

    As the title suggests, this section contains implementation information for the Slope Detection techniques. This technique is quite similar to the Pothole Detection technique discussed in Section 5.1.2. However, there are some minor variations. Here, instead of one ultrasonic sensor, two ultrasonic sensors have been used on the top level. This placement of the sensors have been demonstrated in Fig 5.10 below.

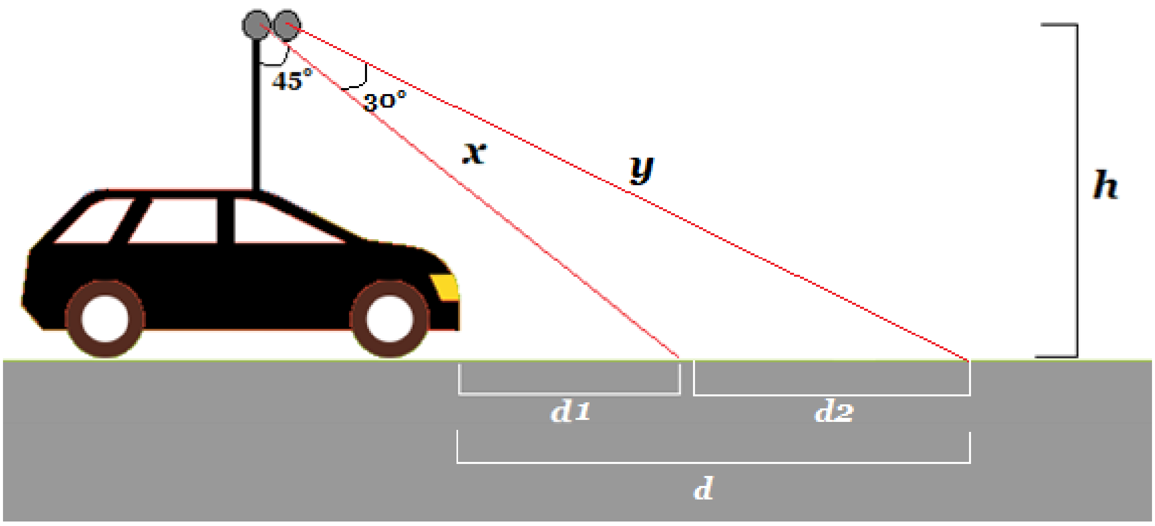

Fig 5.10 Sensor Placement [SAA]

Some of the observations from Fig 5.10 are:

- Angles of sensors are fixed
- x and y are fixed (for plain surface)
- h is fixed
- d = d1 + d2 is fixed

Here, one more point to note is that whenever a slope is present, the reading of y will change first. This is a key observation and serves as a base for proceeding with slope detection. This can be understood by looking at Fig 5.11 below.

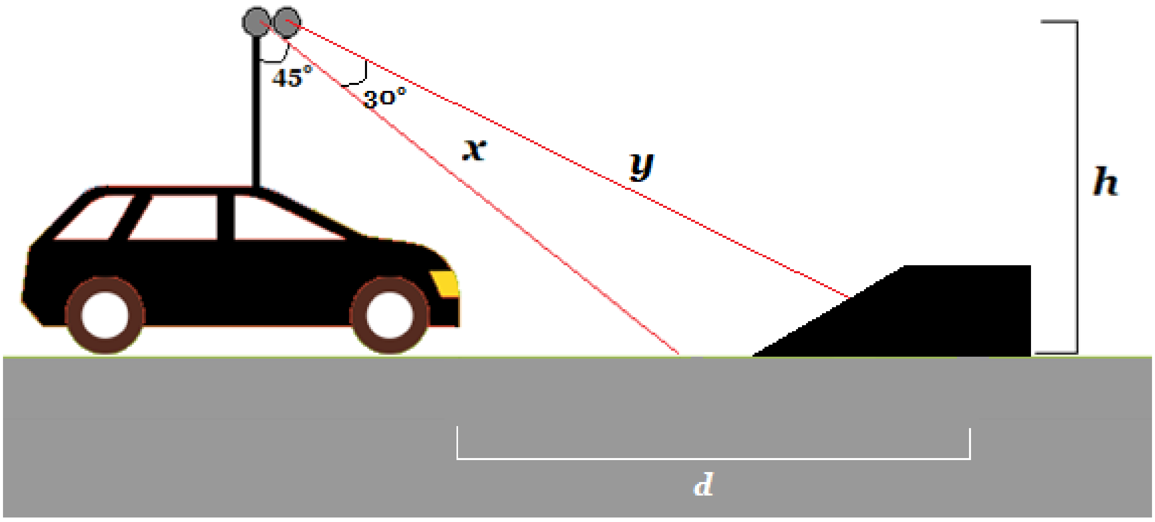

Fig 5.11 Slope Detection

It is clear from the above figure that the reading of y changes first whenever any slope/hump is detected. Now, for the detection of slope, the following logic is implemented:

```
if y < original (y):
    Slope Detected
else:
    Slope Not Detected
```

By using the above logic, the detection of slope has been implemented within this project.

This step concludes the slope detection part. This method has been tested for real-world conditions and has worked competently.

- **Slope Analysis**

  Till now, only the Slope Detection techniques have been discussed. Moving on, the next phase is of Slope Analysis. The need for analysis arises due to the requirement of data for training the Machine-Learning model, which is discussed in the next section. In short, the data that is required for slope avoidance is to be generated by Slope Analysis.

  The main task of this phase is to calculate the inclination of the slope detected in the previous phase. This calculation of inclination is necessary as it is one of the parameters needed to train and test the Machine-Learning model implemented in the Slope Avoidance phase.

  Fig 5.12 assists in understanding the technique for calculation of the inclination of slope.

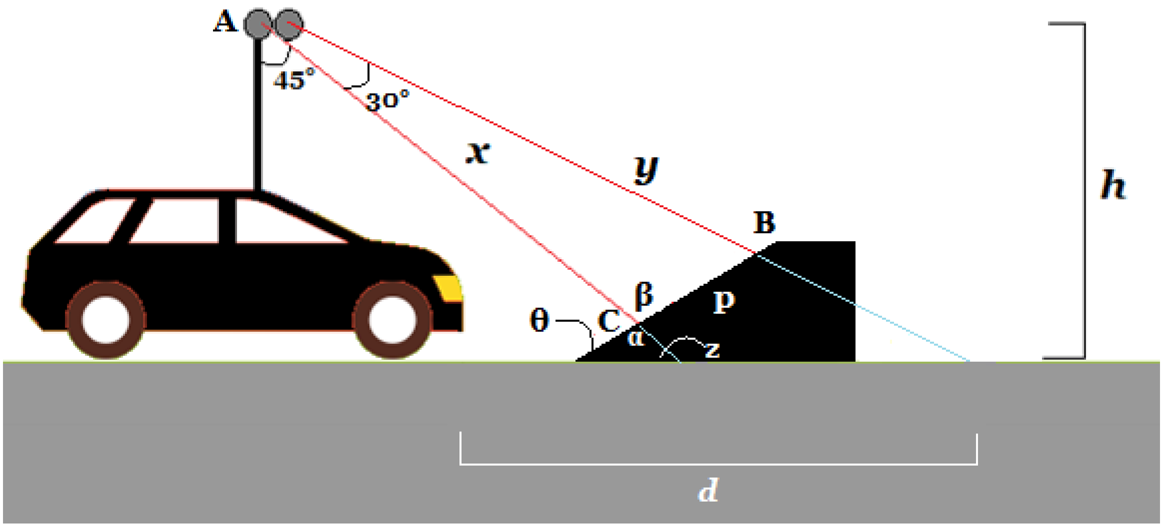

Fig 5.12 Slope Analysis

Considering △ABC, length p can be found using Side-Angle-Side (SAS) theorem. Now, ∠β can be found using Side-Side-Side (SSS). As ∠α and ∠β are vertically opposite angles, ∠α = ∠β. Now, ∠z can be found by applying Pythagoras Theorem on the right angle in Fig 5.10. Now, ∠θ can be found by using the Triangle Law: ∠θ = 180 - ∠α - ∠β. Here, ∠θ is the final inclination angle needed for the training process of the Machine-Learning model used in the Slope Avoidance phase.

This step concludes the Slope Detection and Analysis section.

⦿ **Slope Avoidance**

After Slope Detection & Analysis, the next stage is to avoid slopes, when detected. This task is quite simple as compared to the Slope Analysis part. Some of the Slope Avoidance strategies are similar to the ones discussed in the Pothole Avoidance section. However, there are quite a few changes.

For Slope Avoidance, there are three categories of decisions:

- Safe to climb
- Unsafe to climb & Divert
- Direction of Diversion

To simplify the understanding of this implementation, Climbing Decisions are explained first followed by the Diversion Decision later.

- **Climbing Decision**

  This is an answer to the question: Is it safe to climb the slope? Accordingly, Machine-Learning technology has been utilized to predict the climbing decision. To achieve this, the following steps have been followed:

  - **Generation of Data**

    The data used for this task includes: Speed $_{\text{Rover}}$, Acceleration $_{\text{Rover}}$, Inclination of Slope (determined in Slope Analysis phase) and Climbing Decision for the real-world conditions. Total number of records collected: 35,000.

  - **Training the Machine-Learning Model**

    As this represents a classification problem, Support Vector Machine (SVM) has been selected as the Machine-Learning model and is trained with Speed $_{\text{Rover}}$, Acceleration $_{\text{Rover}}$, Inclination of Slope and Climbing Decision as output label. Refer Fig 5.13 and 5.14 for output plots.

  - **Testing the Machine-Learning Model**

    Input: (Speed $_{\text{Rover}}$, Acceleration $_{\text{Rover}}$, Inclination of Slope) and Output Prediction: (Safe/Unsafe to climb). Accuracy: 99.54%.

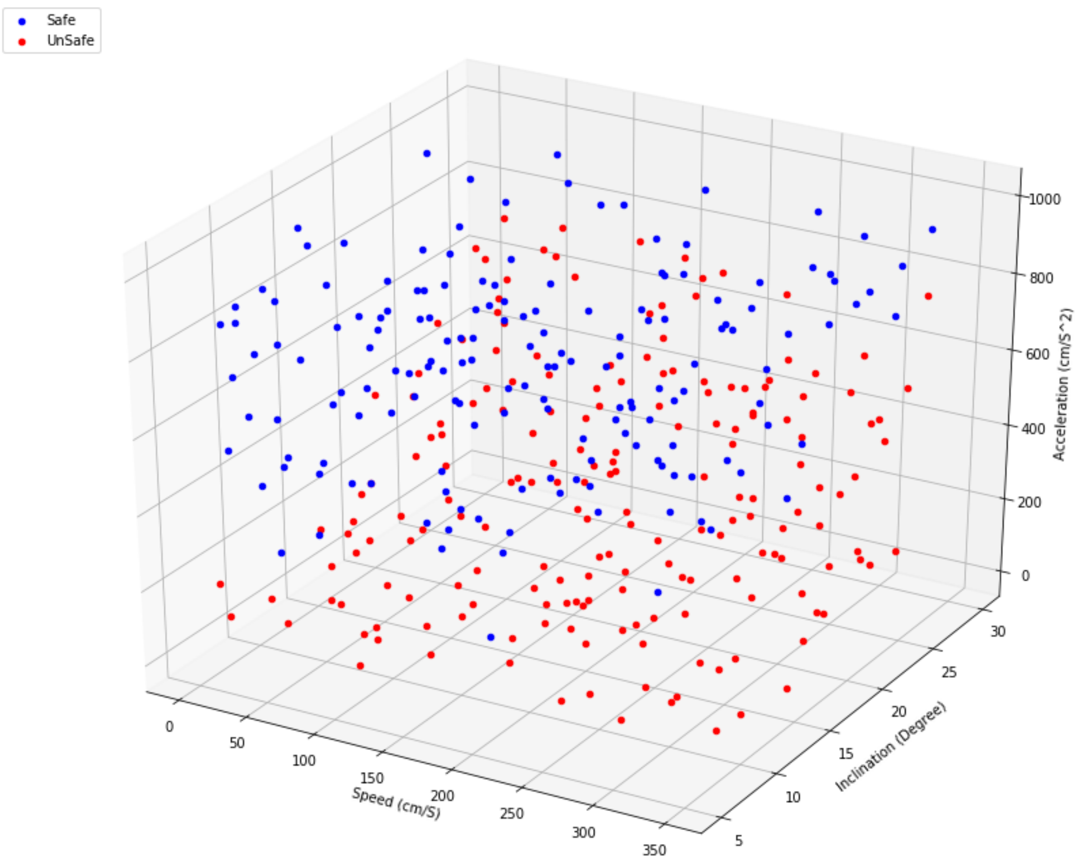

Fig 5.13 Slope Avoidance – I

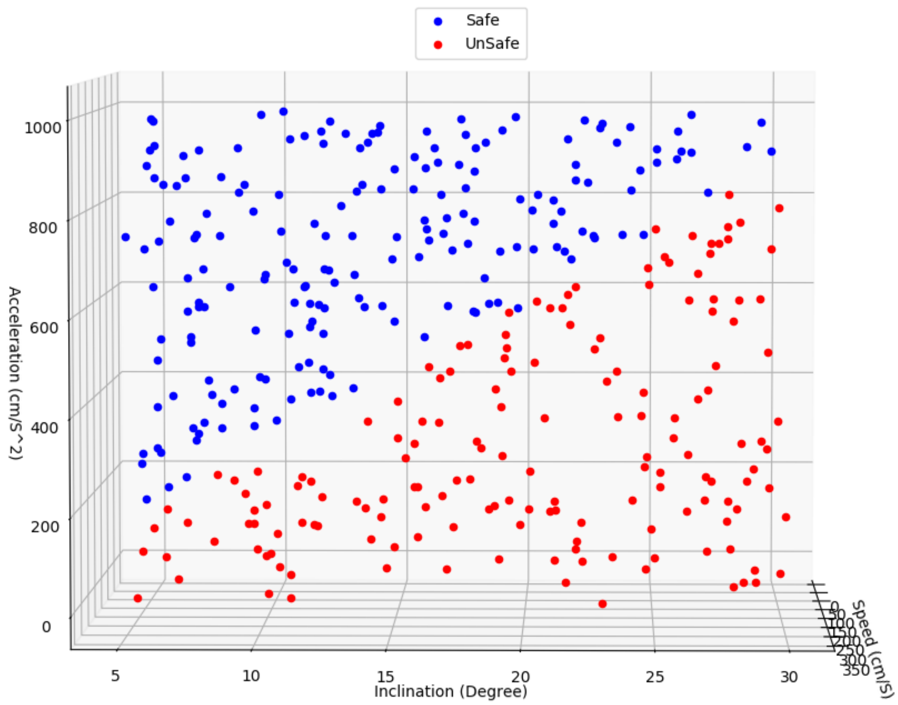

Fig 5.14 Slope Avoidance – II

- **Diversion Decision**

  After getting a Climbing Decision from the above phase, next step is to determine in which direction the rover should divert, if the Climbing Decision turn out to be Unsafe & Divert. This decision is illustrated in this section. Machine-Learning has been utilized to predict the Diversion Decision, i.e. Left / Right direction, based on the readings of the Left and Right Ultrasonic Sensors. Following is the process for the same:

  - **Generation of Training Data**

    Data needed for Training the model has been collected by simulation of the rover in real-world conditions. Here, the data from Left and Right Ultrasonic sensors have been taken as input and the direction of diversion has been taken as label.

  - **Training the Machine-Learning Model**

    As this is a classification problem, Support Vector Machine (SVM) has been found to be the best fit model for this type of data. Accordingly, the Machine-Learning model is trained with the data generated in the previous step.

  - **Testing the Machine-Learning Model**

    After testing the trained model, the final output results are available in Fig 5.15. Accuracy obtained has been around *98.69%*. Total Records: 30,000.

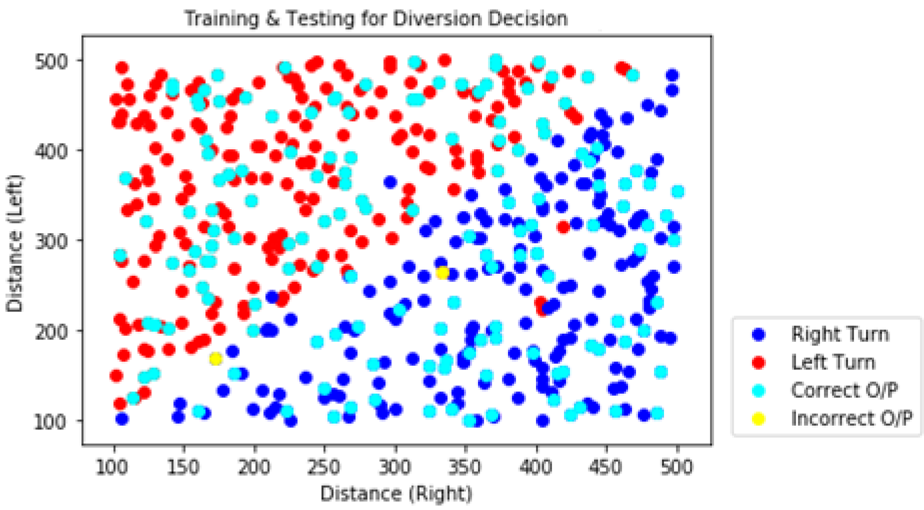

Fig 5.15 Diversion Decision Training & Testing [SAA]

5.2 PHASE – II : PATH PLANNING

This section introduces the second phase of this work: Path Planning. In order to create a reliable autonomous driving module, Obstacle Avoidance alone is not enough. Subsequently, Path Planning is required for assisting the rover to reach its destination. In Path Planning, there are two main tasks:

- Planning
- Optimization

The planning of path can be easily done by choosing any path that leads to the destination. However, optimization task includes optimizing the planned paths in terms of Trip Time and Path Distance. This optimization task is a bit complex to understand as well as implement.

One more concept that has been given significance under this work is: Navigation in absence of position coordinates. This might seem irrelevant but its relevance can be understood by considering Connection-Loss problem. When connection loss occurs, the rover will not be able to continue on its designed path without position coordinates. This scenario does not comply with the primary aim of this project. In addition, considering the space applications, if a rover loses its contact with the orbiter, the space mission would have to be aborted and a huge monetary loss would occur. Such situations cannot be disregarded. Accordingly, this work has considered such scenarios and techniques suitable for avoiding such situations are also proposed.

Two methods are proposed, within this project, for Path Planning:

- Direction-Based Method (DBM)
- Greedy Method (GM)

Here, the Direction-Based Method (DBM) has been developed to work in absence of NavIC data while the Greedy Method has been proposed to work under the presence of NavIC date. Hence, the Direction-Based Method (DBM) may be utilized for extra-terrestrial explorations. Implementations of both of these methods are somewhat difficult to understand, however, each of these methods have been explained in this section in the simplest form possible. Each of the above mentioned methods have been tested under real-world conditions as well as in presence/absence of position coordinates. These methods are explained below with proper implementation details.

5.2.1 DIRECTION-BASED METHOD (DBM)

This is the most reliable method for the working of Rover in the absence of position coordinates (NavIC data). This method overcomes the Position-Loss problem. As the name suggests, this method works on the basis of directions: North, East, West & South. As the directions are needed for the working of this method, a Magnetometer (Compass) is required to be attached to the Rover.

- **Algorithm**
  1. Get Source Direction and Target Direction
  2. Attempt to turn in order to match the Target Direction
  3. If turn is successful, update Source Direction and Target Direction
     Else, continue in current direction
  4. Repeat Step-2 to Step-3 until destination is reached

To get a clear idea of the implementation of this method, the logic is explained below.

- **Core Logic**

if S_{Dir} = = E and T_{Dir} = = S:
    Turn(Right(90°))
    S_{Dir} = S

if S_{Dir} = = E and T_{Dir} = = N:
    Turn(Left(90°))
    S_{Dir} = N

if S_{Dir} = = W and T_{Dir} = = N:
    Turn(Right(90°))
    S_{Dir} = N

if S_{Dir} = = W and T_{Dir} = = S:
    Turn(Left(90°))
    S_{Dir} = S

if S_{Dir} = = N and T_{Dir} = = E:
    Turn(Right(90°))
    S_{Dir} = E

if S_{Dir} = = N and T_{Dir} = = W:
    Turn(Left(90°))
    S_{Dir} = W

if S_{Dir} = = S and T_{Dir} = = W:
    Turn(Right(90°))
    S_{Dir} = W

if S_{Dir} = = S and T_{Dir} = = E:
    Turn(Left (90°))
    S_{Dir} = E

Following the above mentioned logic, the goal of the method has been achieved in this work. Also, this method usually generates an optimal path as it tend to reach the destination via the shortest path, following the path towards the direction of the destination. In addition, this method has been proposed to work under the following conditions:

- Absence of Positioning Data (NavIC data)
- Absence of Terrain Data (Map)

For demonstration purpose, a demo path has been set-up in Fig 5.16 to check the working of the above mentioned method.

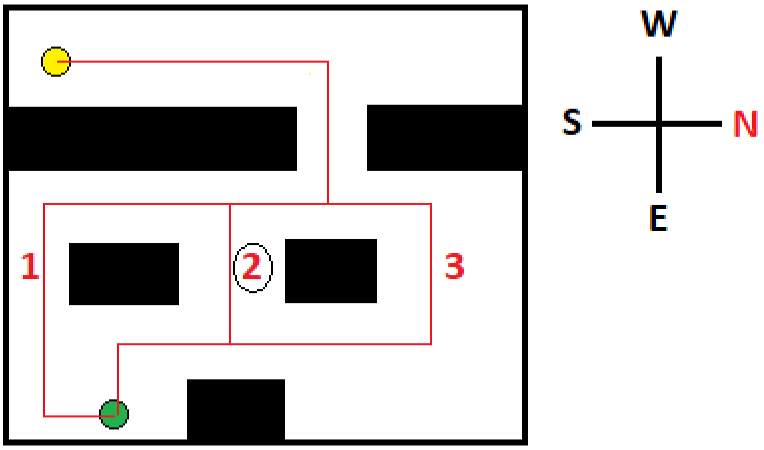

Fig 5.16 Demo Path [DBM]

The simulation output of the rover has been demonstrated in Fig 5.17 below.

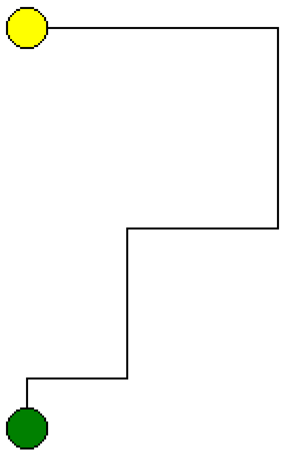

Fig 5.17 Output Path [DBM]

It is evident from the above figure that the Rover followed the most optimal path out of the 3 available paths. *(Note: All the simulation graphics are visualized using Turtle Library).*

- **Limitation**

Although it may seem as the best possible method, this method also possesses some flaws. After testing this method for over 5000 demo paths, one limitation has been determined. This limitation is due to the absence of Terrain data (Maps). This limitation can be visualized by observing Fig 5.18 below.

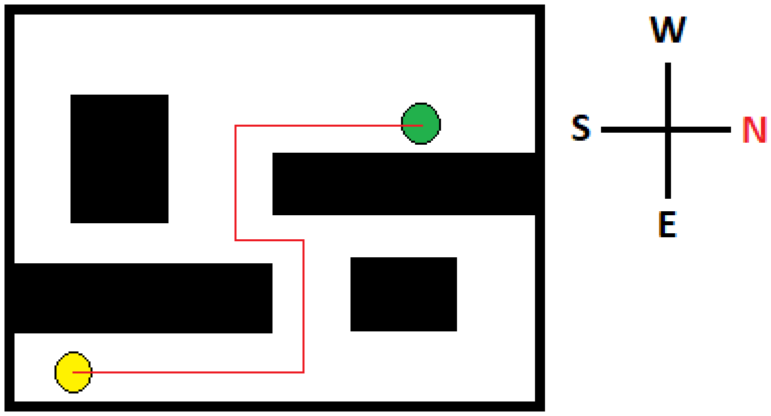

Fig 5.18 Limitation Path [DBM]

By observing the above figure, it is evident that there exists only one optimal path from source to destination. However, the simulation output of the rover has been different. This is demonstrated in Fig 5.19 below.

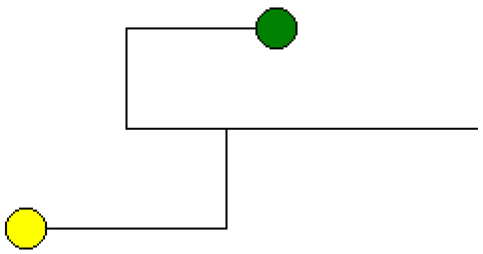

Fig 5.19 Limitation Output [DBM]

It is clearly evident from the above figure that the rover followed a wrong path that led to a dead-end and had to return backwards to reach the destination. This occurs as at the middle point, its Source Direction is W and Target Direction is N. Accordingly, it took a right turn and hence, could not follow the optimal path.

By observing this limitation, the algorithm has been tried to be improved and such changes are incrementally made in the algorithm.

5.2.2 GREEDY METHOD (GM)

The next proposed method of Path Planning is the Greedy Method (GM). This method, as the name suggests, is based on a greedy approach, i.e. it tends to follow a direct straight line path from the Source to the Destination. This method works under the following conditions:

- Presence of Positioning Data (NavIC data)
- Presence/Absence of Terrain Data (Map)

The core logic of this algorithm is stated as follows:

- **Algorithm**
  1. Record the Coordinates (Latitude, Longitude) of Source & Destination
  2. Compute a straight-line path between the Source & Destination
  3. Follow the path
  4. If obstacle is detected, avoid the obstacle (using Obstacle Avoidance techniques) & regenerate the path based on current coordinates
  5. Repeat Step-2 to Step-4 until the Destination is reached

It is clear from the above algorithm that whenever an obstacle is detected, a new straight-line path is generated based on the latest coordinates of the Rover. Hence, the presence of NavIC coordinates is required for the flawless working of this method. The algorithm logic has been implemented using core Python programming. Also, NavIC data has been provided by the Indian Space Research Organization (ISRO).

One sample implementation of this approach is demonstrated in Fig 5.20 below.

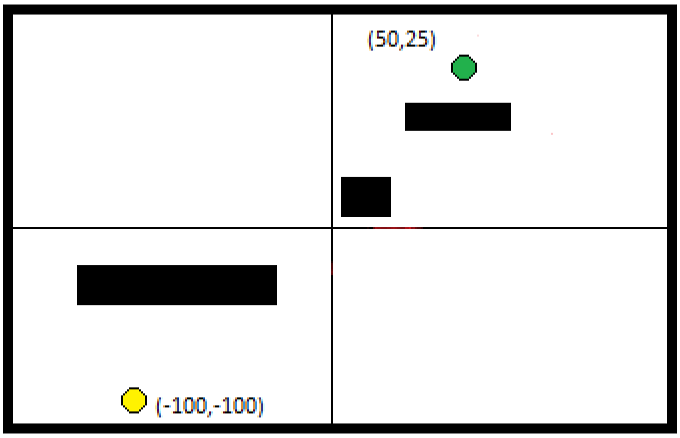

Fig 5.20 Demo Path [GM]

The simulation output is shown in Fig 5.21 below.

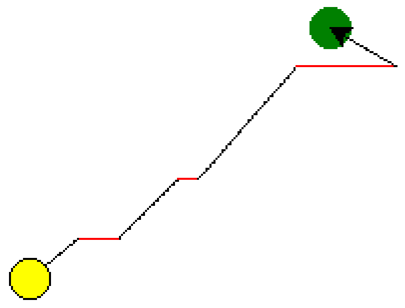

Fig 5.21 Output Path [GM]

It can be clearly seen that the Terrain Data (Map) is not considered. It is evident from the simulation that the Rover tends to move in a straight-line path towards the destination. Also, whenever an obstacle is detected, it avoids the obstacle according to the Obstacle Avoidance techniques discussed in Phase-I of this chapter. After the obstacle is avoided, the Rover generates a new straight-line path to the destination, based on the current NavIC coordinates. The performance of this algorithm under known terrain is discussed in Chapter-6. *(Note: All the simulation graphics are visualized using Turtle Library).*

5.3 INTEGRATION OF PHASE – I & PHASE – II

Following the implementation of Phase – I and Phase – II separately, now the next step is the integration of the two phases. All the modules developed in both the above mentioned phases have been integrated and tested for the combined output performance. The final simulation has been demonstrated in Chapter – 6.

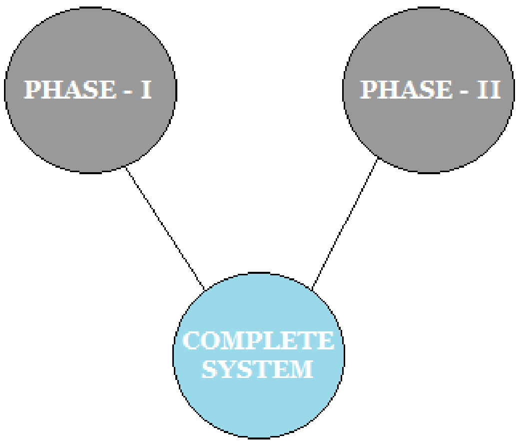

Fig 5.22 Integration of Phase – I & Phase – II

CHAPTER – 6
TESTING & RESULTS

- PHASE – I: OBSTACLE AVOIDANCE TESTING
- PHASE – II: PATH PLANNING TESTING
- COMPREHENSIVE SYSTEM TESTING

6.1 PHASE – I: OBSTACLE AVOIDANCE TESTING

Till now, all the techniques for Obstacle Avoidance have been discussed along with their implementation. Now, this section includes the complete testing procedural details as well as the final experimental results. As discussed earlier, in Chapter – 5, the testing results of all the Machine-Learning models deployed have been set forth along with their implementation description. Accordingly, this section demonstrates only the final system testing results, without repeating the Machine-Learning model accuracies.

For Obstacle Avoidance, three types of modules have been deployed:

- Stationary / Moving Obstacle
- Pothole
- Slope

Now, in order to simplify the testing procedure, one demo path has been constructed consisting of all the three categories of obstacles mentioned above. This demo path is visualized in Fig 6.1 below.

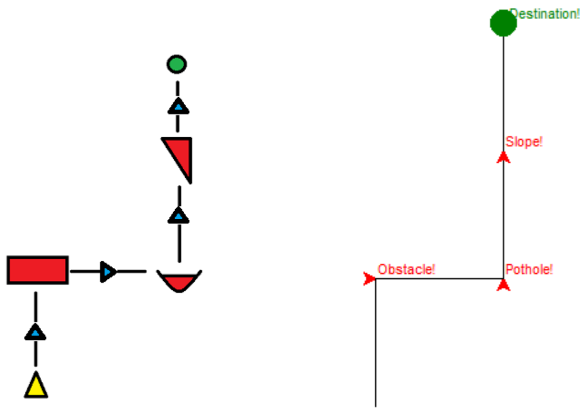

Fig 6.1 Obstacle Avoidance Testing

As seen from Fig 6.1, all the three obstacles have been detected correctly and the Diversion Decisions have been found to be 100% accurate according to the sensor readings provided. Around 500 such paths have been tested. The console output is shown in Fig 6.2.

```
Obstacle Avoided   ---   Right_Turn
Pothole Avoided   ---   Left_Turn
Slope Detected   ---   Climbing

In [4]:
```

Fig 6.2 Console Output [Testing]

6.2 PHASE – II: PATH PLANNING TESTING

Referring to Section 5.2, demo paths with their testing outputs have been presented along with their respective algorithms' description. Considering Section 5.2.1, for Distance-Based Method [DBM], demo path testing for unknown terrain has already been discussed. Considering Section 5.2.2, for Greedy Method [GM], demo path testing for unknown terrain has already been discussed.

This section illustrates the remaining testing procedures: Greedy Method [GM] testing for known terrain (Map). This testing has been conducted by using a known Map of an area. NavIC coordinates have also been utilized for the same. The Map has been visualized in Fig 6.3 below.

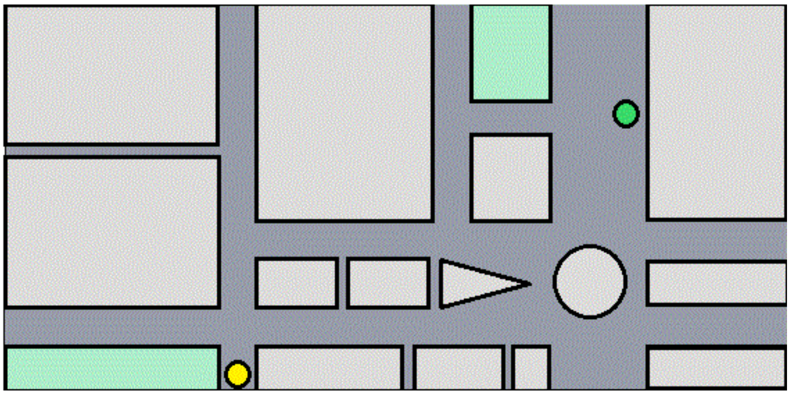

Fig 6.3 Demo Path Testing Map [GM]

Here, the yellow dot represents the starting position while the green dot represents the destination. Now, the simulation output is shown in Fig 6.4 below.

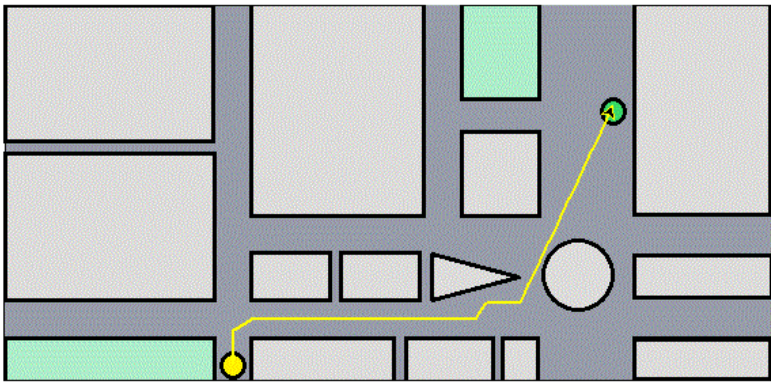

Fig 6.4 Demo Path Testing Output Map [GM]

It is evident that the Greedy Method follows the optimal path and works effectively on a known terrain. *(Note: All the simulation graphics are visualized using Turtle Library).*

6.3 COMPREHENSIVE SYSTEM TESTING

Following the individual module tests of Obstacle Avoidance and Path Planning, this section comprises of the tests performed on the integrated system (Obstacle Avoidance + Path Planning) and their respective results.

For simulation purpose, the designed graphics are demonstrated in Fig 6.5 below. In addition, all the parameter values (Sensor values, Position coordinates) have been collected under real-world conditions. However, for simulation purpose, only the graphical output has been demonstrated. Also, the output performance is shown in Fig 6.6.

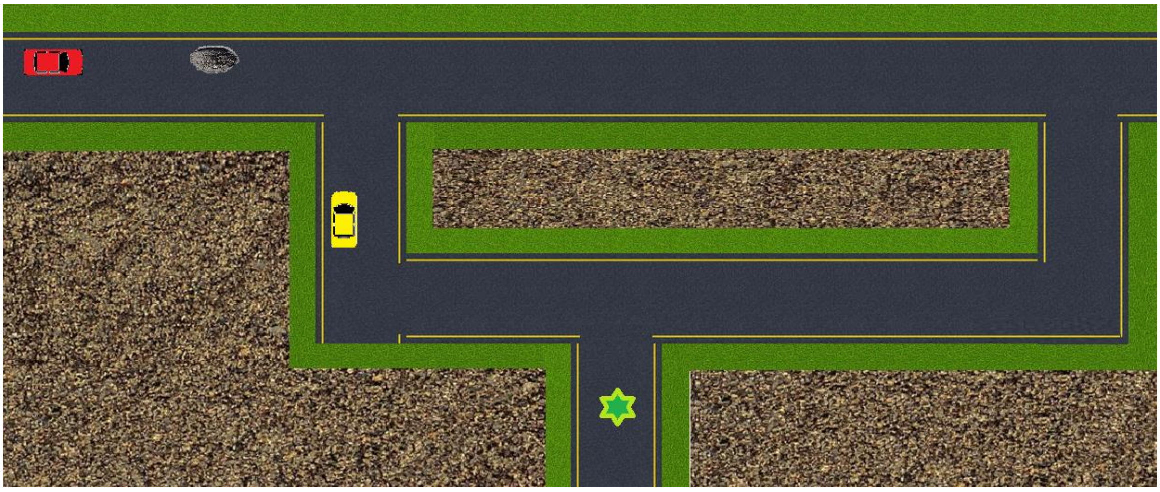

Fig 6.5 Comprehensive Testing Path

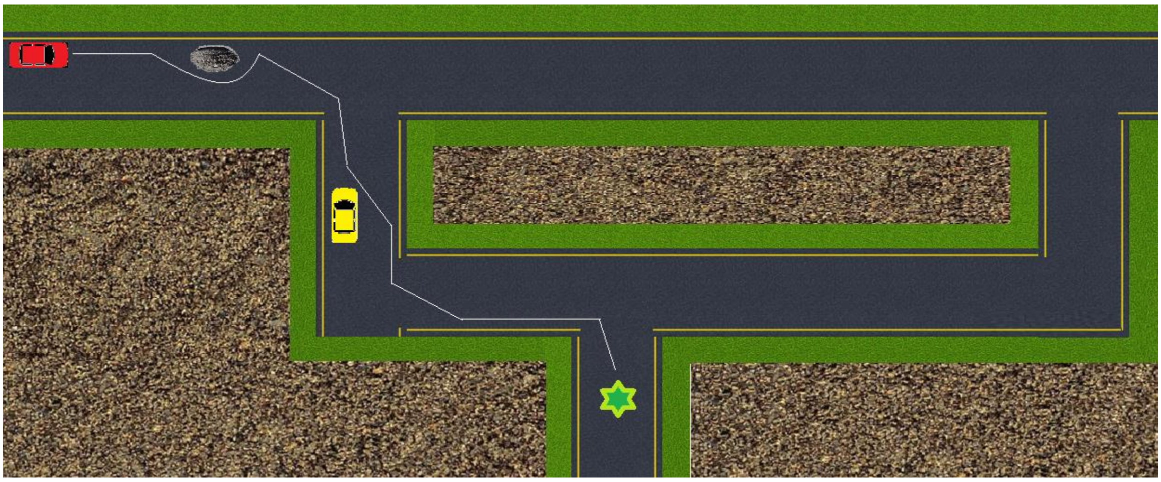

Fig 6.6 Comprehensive Testing Output

Here, Greedy Method (GM) has been deployed. It is evident from the above figures that the path followed by the rover is the most optimal path considering the given scenario. Also, the Obstacle Avoidance strategies have also performed well. Total Test Paths: 5000.

CHAPTER – 7
LIMITATION & FUTURE ENHANCEMENT

- **LIMITATION**
- **FUTURE ENHANCEMENT**

7.1 LIMITATION

It is evident from Chapter – 5 and Chapter – 6, that the methods proposed within this project work have proved to be effective under real-world conditions. However, there exists some shortcomings. These limitations have been discovered after the rigorous testing of the proposed methods, as mentioned in Chapter-6.

Though the proposed methods turned out to be working perfectly well, there still exists some limitations considering the data and time constraints. As this project work has been done as a part of final year thesis leading to a bachelor's degree, it has been completed within a time bound of less than four months.

The following are the limitations of this work:

- **Sensor Accuracy**

  Considering the limited budget and size constraints, the sensors used within this project work have certain limitations. The accuracy of these sensors affects the final output performance of this work. As the sensors used in this project do not have very high precision, the data generated from the sensors are not precisely accurate and hence serve as a major limitation.

- **Machine-learning Model Accuracy**

  One major limitation of using Machine-Learning technology is that it cannot provide 100% output accuracy. Accordingly, as Machine-Learning models have been used multiple times within this project work, the output accuracy obtained is not a perfect 100%.

- **Data Constraints**

  As this project work focusses primarily on software modules, data used in this project becomes a vital component for achieving the desired results. It is evident that this project requires some real-world data for training various Machine-Learning models. Although the data used in this work has been collected under real-world conditions, it is apparent that simulation of the complete real-world environment is not possible. Subsequently, the data used in this project work can be considered reliable, however, it may not completely symbolize the real-world conditions. Accordingly, such data constraints constitute a major limitation of this work.

7.2 FUTURE ENHANCEMENT

After going through the limitations mentioned in Section 7.1, it is evident that there exists a major scope of improvement in various areas. Considering such areas, the scope of future enhancements includes:

- **Inclusion of Computer Vision**

  Following the goal of this project, i.e. less complexity and less power consumption, only Ultrasonic and INS sensors have been utilized. However, in order to deploy the product into the real world, a more reliable system is needed. For achieving this, Computer Vision technology can be deployed in addition to the currently used Machine-Learning technology. This enhancement would serve as a major breakthrough as it would substantially improve Obstacle Detection & Recognition. Also, this would open up paths for numerous end-user applications.

- **Upscaling of the Rover Prototype**

  As the final deployment of the rover prototype is constrained for limited end-user applications, the size constraint also applies to it. However, as new applications are discovered within this domain, the rover can be upscaled to a bigger size in order to satisfy the generated demands.

- **Integration with Google Maps®**

  For using the methods proposed in this work on a map accurately, this work can be integrated with the Google Maps. This would largely improve the Path Planning as the path planning algorithm deployed by Google is one of the best optimal path planning algorithms currently existing in the world. Accordingly, by this integration, accurate positioning as well as optimized path generation would be available. Also, Google Maps are updated frequently. Subsequently, this integration would improve the coverage area for the rover prototype.

- **Enhancement of Data & Algorithms**

  Although the accuracy obtained from the various Machine-Learning models used in this work are up-to-the-mark, there is still a scope of improvement by improving the pre-processing of data. This can be done by using more advanced pre-processing and noise-removal algorithms.

CHAPTER – 8
CONCLUSION

- **CONCLUSION**

8.1 CONCLUSION

A novel approach on autonomous driving within the Indian terrain has been discussed in this report. This project elaborates on the process of developing software for an autonomous rover by utilizing the Navigation with Indian Constellation (NavIC) services as well as reducing the power consumption and simplification of working algorithms. In addition, a unique approach has been presented for the working of rover in absence of positioning data (NavIC data).

Following the aim, two phases have been discussed in this report: Obstacle Avoidance and Path Planning. Three categories of obstacles have been discussed in detail, namely Stationary / Moving Obstacle, Slope and Pothole. Adding to it, techniques for the detection and avoidance of obstacles have been presented within this report.

Furthermore, two novel approaches have been proposed for Path Planning. One of which, namely Direction-Based Method (DBM), has been proposed for assisting the rover to reach its destination in absence of positioning data (NavIC data). In addition, a new optimal path generator method (Greedy) has been proposed which works in the presence of NavIC data.

The test results of all the techniques proposed in this report have been discussed in detail with elaboration of all the test parameters and test cases considered. The accuracy of all the Machine-Learning models used have been in the range of 98% to 99.96%. Adding to it, the proposed Path Planning algorithms have been tested under real-world conditions. Subsequently, both the proposed algorithms have shown remarkable performance. In addition, the comprehensive system test has also generated astounding results.

In conclusion, the proposed system along with the proposed techniques for Obstacle Avoidance and Path Planning have proved to be reliable for functioning over the Indian terrain along with some scope of application in the extra-terrestrial explorations.

REFERENCES

[1] A.S. Ganeshan ; Rathnakara Shuntikatte ; R. Gupta ; A.K. Jain, (2005), Indian Regional Navigation System (IRNSS) concept, *(Published by Research Gate)*.
Retreival No.: 296928143

[2] Richard M. Heiberger ; E. Neuwirth, (2009), Polynomial Regression, *R Through Excel*
DOI: 10.1007/978-1-4419-0052-4 11

[3] Yijun Yang ; Jianping Li ; Yimei Yang, (2015), The Research of the Fast SVM Method, *2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing.*
DOI: 10.1109/ICCWAMTIP.2015.7493959

[4] Sharayu Yogesh Ghangrekar, James M. Conrad, (2009), A Path Planning And Obstacle Avoidance Algorithm For An Autonomous Robotic Vehicle, *2009 IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems*.
DOI: 10.1109/MASCOT.2009.5366693

[5] Xiyang Song ; Huangwei Fang ; Xiong Jiao ; Ying Wang, (2012), Autonomous Mobile Robot Navigation Using Machine Learning, *2012 IEEE 6th International Conference on Information and Automation for Sustainability*.
DOI: 10.1109/ICIAFS.2012.6419894

[6] V. Kalpana, (2019) Implementation of Vehicle Speed Reducing System at Speed Breaks by Detecting Potholes and Humps Using Ultrasonic Sensor, *International Journal of Recent Technology and Engineering (IJRTE).*
Retrieval No.: E11870275S19