T.R.
VAN YUZUNCU YIL UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES
MECHANICAL ENGINEERING DEPARTMENT

THE DESIGN OF AN UNMANNED GROUND VEHICLE AND ITS TRAJECTORY-TRACKING CONTROL BY USING GPS

M. Sc. THESIS

PREPARED BY: Firas Muhammad Saib M.M. AL-NAQSHBANDI
SUPERVISOR: Asst. Prof. Dr. Atilla BAYRAM

VAN-2018

T.R.
VAN YUZUNCU YIL UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES
MECHANICAL ENGINEERING DEPARTMENT

THE DESIGN OF AN UNMANNED GROUND VEHICLE AND ITS TRAJECTORY-TRACKING CONTROL BY USING GPS

M.Sc.THESIS

PREPARED BY: Firas Muhammad Saib M.M. AL-NAQSHBANDI

This project was supported by Scientific Research Projects Coordination Unit of Van Yuzuncu Yil University with project no: **FBA-2016-5062**.

VAN-2018

ACCEPTANCE and APPROVAL PAGE

This thesis entitled "**The Design of an Unmanned Ground Vehicle and Its Trajectory-Tracking Control by Using GPS**" presented by Firas Muhammad Saib M.M. AL-NAQSHBANDI under supervision of Assist. Prof. Dr. Atilla BAYRAM in the department of Mechanical Engineering has been accepted as a M. Sc. thesis according to Legislations of Graduate Higher Education on 18/07/2018 with unanimity of votes members of jury.

Chair: Assoc. Prof. Dr. Mehmet Nuri ALMALI Signature:

Member: Assist. Prof. Dr. Ergin KILIÇ Signature:

Member: Assist. Prof. Dr. Atilla BAYRAM Signature:

This thesis has been approved by the committee of The Institute of Natural and Applied Science on ..20../..07../.2018. with decision number .2018/.33-.I.

Signature
Prof. Dr. Suat ŞENSOY
Director of Institute

THESIS STATEMENT

All information presented in the thesis obtained in the frame of ethical behavior and academic rules. In addition all kinds of information that does not belong to me have been cited appropriately in the thesis prepared by the thesis writing rules.

Signature
Firas Muhammad Saib M.M. AL-NAQSHBANDI

ABSTRACT

THE DESIGN OF AN UNMANNED GROUND VEHICLE AND ITS TRAJECTORY-TRACKING CONTROL BY USING GPS

AL-NAQSHBANDI, Firas Muhammad Saib M.M.
M.Sc. Thesis, Mechanical Engineering Department
Supervisor: Asst. Prof. Dr. Atilla BAYRAM
July 2018, 120 pages

The autonomous mobile robot (AMR) is an automated machinery which is able to self-motion and has the ability to navigate in their intended environment without the need for special guidance hardware.

The aim of this study is at first to design a prototype of an unmanned ground vehicle (UGV) that can move fully automatically. This vehicle consists of three main parts: mechanical parts (the DC-Motors, wheels have been purchased and the chassis, junction pieces, covers of the vehicle etc.), Electronic parts (DC-Motor driver Card, Microcontroller, Battery), and Sensors (Global Positioning System (GPS), Inertial Measurement Unit (IMU), Encoder, and Magnetometer). These sensors are used to know the precise real position and orientation of the unmanned ground vehicle (UGV) synchronously.

The second part of the thesis consists of the modeling and trajectory/path tracking control of the unmanned ground vehicle (UGV). The control algorithm based on the kinematic model was proposed to control the position, speed, and orientation of the unmanned ground vehicle (UGV) to follow the desired reference routes. The proposed control system using a modified PD (Proportional-Derivative) controller has proved highly effective in the process of controlling the vehicle. In order to prove the efficiency of the control system, the results obtained in simulations were compared with the sliding mode control (SMC) method adopted from the literature. The simulation results showed the success of the unmanned ground vehicle (UGV) in designing and control.

Keywords: Global Positioning System (GPS), Mobile robot, Path-Following, Trajectory-Tracking, Unmanned ground vehicle.

ÖZET

İNSANSIZ BİR YER ARACININ TASARIMI VE GPS İLE YÖRÜNGE KONTROLÜ

AL-NAQSHBANDI, Firas Muhammad Saib M.M.
Yüksek Lisans Tezi, Makine Mühendisliği Bölümü
Tez Danışmanı: Dr. Öğr. Üyesi Atilla BAYRAM
Temmuz 2018, 120 sayfa

Otonom mobil robotlar (AMR), kendi kendine hareket edebilen ve özel yönlendirme donanımı gerektirmeksizin gezinme kabiliyetine sahip olan otomatik makinelerdir.

Bu çalışmanın amacı ilk olarak tam otomatik olarak hareket edebilen insansız bir kara aracının (UGV) prototipini tasarlamaktır. Araç üç ana bölümden oluşmaktadır: mekanik parçalar (DC-Motorlar, jantlar gibi bazı parçalar satın alınmış olup şasiler, bağlantı parçaları, araç kapakları vb. diğer parçalar doğrudan ihtiyaçlara göre üretilmiştir), elektronik parçalar (DC-Motor sürücü kartı, Mikrokontrolcü, Batarya) ve sensörler (Küresel konumlama sistemi (GPS), Ataletsel ölçüm birimi IMU, Enkoder, ve Magnetometer). Bu sensörler senkronize UGV’in kesin gerçek konumunu belirlemesi ve yönünün tayininde kullanılmıştır.

İkinci kısım, UGV'nin tasarım ve yörünge/yol izleme kontrolünü içermektedir. İstenen referans rotalarının takip edilmesi için, UGV'nin pozisyonunun, hızının ve yönünün belirlenmesini sağlayan kinematik tabanlı bir kontrol algoritması önerilmiştir. Algoritmada modifiye edilmiş bir PD kontrolör kullanılmış ve bu kontrolcü ileoldukça etkili sonuçlar elde edilmiştir. Kontrol sisteminin başarısı, bilgisayar benzetimleri yardımıyla elde edilen sonuçların literatürde bulunan SMC kontrol yöntemi sonuçları ile karşılaştırılması yardımıyla görülmüştür. Sonuç olarak simülasyon sonuçları UGV’nin başarılı bir şekilde tasarlandığını ve kontrol edildiğini göstermiştir.

Anahtar kelimeler: Küresel konumlama sistemi (GPS), Mobil robot, Yol takibi, Yörünge izleme, İnsansız yer aracı.

ACKNOWLEDGEMENT

Above all, I want to thank ALLAH for his blessings and virtues for his guidance and rewards toward me. Then I want to offer my deep thanks and gratitude to my supervisor Asst. Prof. Dr. Atilla BAYRAM for all advice and guidance during all stages of this research. As well as I want to thank my family, especially my mother and father for their support and encouragement to continue my studies. Thank you all.

2018

Firas Muhammad Saib M.M. AL-NAQSHBANDI

LIST OF CONTENTS

Page

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

Figure **Page**

SYMBOLS AND ABBREVIATIONS

Some symbols and abbreviations used in this study are presented below, along with descriptions.

Symbols	**Description**
a	Distance from CG to rear wheel (axle).
b	Distance from CG to front wheel (axle).
L	Distance from rear wheel to front wheel (axle).
v_u	Linear velocity along the u - axis.
$\dot{W}$	Wheel velocity on the given co-ordinate system.
θ	The angle of the rear wheel with respect to x-axis.
δ	Steering angle between the front wheels and axis of the body.
X_c, y_c	The position of the control point.
ω	Angular velocity.
T_d	Driving torque.
R_w	Radius of the tires (wheels).
u_1	Input voltage of the rear DC-Motors.
u_2	Input voltage of the front DC-Motors.

Abbreviations	**Explanation**
AC	Alternative Current.
AMR	Autonomous Mobile Robot.
AUV	Autonomous Underwater Vehicle.
CG	Center of gravity (center of mass).
CP	Control point.
CW	Clock wise.
DC	Direct Current.
DGPS	Differential Global Positioning System.
DR	Dead Reckoning.
ECEF	Earth-centered, Earth-fixed
FLC	Fuzzy Logic Control.

FTDI	Future Technology Devices International.
GIS	Geographic Information System.
GNSS	Global Navigation Satellite System.
GPRS	General Packet Radio Service.
GPS	Global Positioning System.
KF	Kalman Filter.
EKF	Extended Kalman Filter
PC	Personal Computer.
IDE	Integrated Development Environment - or Arduino Software
IMU	Inertial Measurement Unit.
INS	Inertial Navigation System.
LED	Light-Emitting Diode.
NASA	National Aeronautics and Space Administration
NMEA	National Marine Electronics Association.
P	Proportion Controller.
PD	Proportion Derivative Controller.
PI	Proportional Integral Controller.
PID	Proportional Integral Derivative Controller.
PWM	Pulse Width Modulation.
RTK	Real Time Kinematic.
SMC	Sliding Mode Control.
UAV	Unmanned Arial Vehicle.
UGV	Unmanned Ground Vehicle.
USB	Universal Serial Bus.
US	United State.
UTC	Coordinated Universal Time.
WGS84	World Geodetic System.

LIST OF APPEDIX

Pages

1. INTRODUCTION

We use clever tools in every part of our lives and we will not pass a lot of time to see that the electronics will do the greatness of daily tasks. One of the most complex tasks that being carried out every day by the human more than once in a day is driving a car. To facilitate vehicle driving and make it safer, many studies and attempts have been made to make vehicles driving self-propelled on the roads. These studies began many years ago from the 1950s, while the first truly self-driving cars appeared in the 1980s. Carnegie Mellon University in 1984 carried out many experiments in this field also in 1987, Mercedes-Benz and the University of Bundeswehr in Munich, Germany, introduced a project to improve the self-driving cars (Goge, 1995). Since then, many major companies and research organizations such as General Motors, Toyota, Apple, Google, Intel, Audi, BMW, and many other companies have focused on the developing of self-driving vehicles. The interest on this subject has rapidly grown with contributions of these companies. In 2009, Google began manufacturing and developing self-driving technology to build a self-propelled car, Google tested its own self-driving technology with Toyota cars on highways in California in the same year Google announced that its cars completed half million kilometers of self-driving on highways without accidents.

The autonomous Mobile robots (AMR) are automated machinery which is able to self-motion and they have the ability to navigate in their environment. The mobile robots are not fixed to a specified environment. It is possible that these mobile robots with self-control are able to navigate in their intended environment without the need for special guidance hardware or electromechanical devices. Mobile robots can be based on routers that allow them to move on the pre-specified trajectories in the controlled space. They are not like industrial robots that are usually fixed and consisting of pivotal arms and grippers which linked to a fixed surface. The recent mobile robots are becoming more common and have widely used in all areas of life (commercial, industrial, military, etc.). Therefore, this topic has become a focus for the major research centers and universities to study, develop, and use the mobile robots in various fields of our life

(Moubarak and Ben-Tzvi, 2011). There are many areas which mobile robots can be used on:

- Services: the Mobile robots can be used in postal service centers to classify, transport and collect postal parcels, and it is also possible to use them that supplied with arms in hospitals to bring medicines and medical equipment. (For example, the Postal Service of Australia is trialing a new parcel delivery service in the Brisbane suburb of New Farm using an autonomous robot Figure 1.1)

Figure 1.1. Australia Post's Delivery Robot.

- Search and Rescue: A mobile robot can be utilized in rescue and search service, in the event of disasters and building collapses, where it can move between scattered debris to find victims and survivors. Figure 1.2 shows the bomb disposal robot utilized in the search and rescue operations of the West Virginia Mine in 2006.

Figure 1.2.V2 the bomb disposal robot.

- In Space: A mobile robot can be used in space for the purpose of exploration and recognition of outer space. Figure 1.3 shows a mobile robot from NASA's Mars Science Laboratory, for studying Mars's ability to sustain microbial life in the past or present.

Figure 1.3. The space mobile robot of NASA's Mars Science Laboratory.

- Underwater: An autonomous mobile robot can be used to explore undiscovered oceans, to monitor marine life, or to clear marine mines. Robosea Company manufactured an underwater drone propelled by a swishing tail mobile robot named BIKI with the aim of capturing photos of marine life and provides some very interesting exploration possibilities for consumer products Figure 1.4.

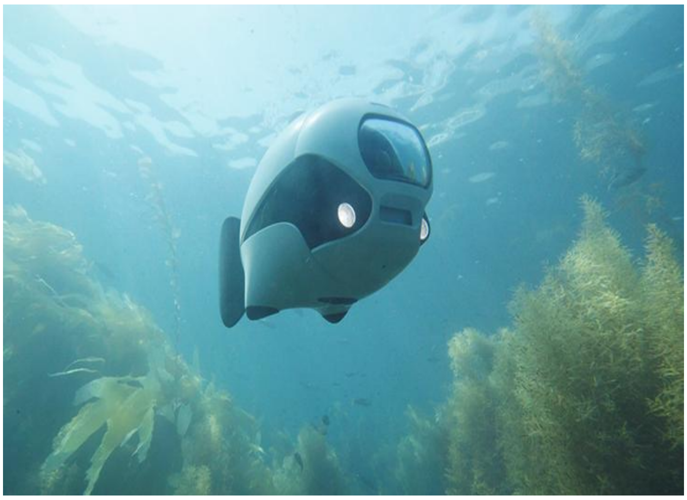


Figure 1.4. BIKI underwater drone mobile robot.

- Nurse: A mobile robot can work as a nurse; certainly work as a nurse is a tired and careful work, especially when it comes to attention to patients, and provision of certain types of medicine. For example, a new robot was made in the form of a bed for patients can do most of these tasks. The special name of these mobile robots is EPush bed. They can do many tasks such as moving from room to room and many other things. Figure 1.3 shows RIBA robot nurse that can raise the patients and transfer patients from one suite to another for surgeries or tests.

Figure 1.5 RIBA robot nurse.

The mobile robots are used for different types or works as mentioned above. They contain different components depending on the field and aim. The most important ones of the mobile robots are given as follows:

1. Control units (They are a processors, microcontrollers or personal computer PC).
2. Control software (It is programs or algorithms that can be written by different computer high-level languages (C, C++, FORTRAN, Pascal, etc.) to control the mobile robots).
3. Sensors and actuators [Actuators give the mobile robots a driving force, and the sensors are a set of different devices used for different tasks dependent upon the requirements of the mobile robot (touch and sense the environment, avoiding obstacles, etc.) (Gopalakrishnan et al., 2004)].

In general, mobile robots can be classified according to the environment in which they move and work through it, or to the appliances and style used by them in the process of moving as shown in Table 1.1.

Table 1.1. Classification of mobile robots

Classifications	Types
Due to the environment in which mobile robots moves and works through it.	1. Ground (land) robots: This type of mobile robot moves on the ground [Unmanned ground vehicle (UGV)], they often use wheels or chains or legged to move. 2. Industrial mobile robot: This type of mobile robots works in closed environments and it is linked to the fixed surface. It is used to transport materials and other works in their own environment. 3. Aerial mobile robots: This type of mobile robot moves in the space [Unmanned Aerial Vehicles (UAV)]. 4. Underwater mobile robots: This type of mobile robot moves under the water [Autonomous underwater vehicles (AUV)]. 5. Polar mobile robots: This type of mobile robot moves in ice environments.
Due to the appliances and the style used by the mobile robot in the process of moving.	1. Wheeled mobile robots: This type of robot uses wheels in the process of moving. 2. Legged mobile robots: This type of robot uses the devices like human legged in the process of moving. 3. Tracks mobile robots: This type of robot utilizes a continuous band of treads which is driven by two or more wheels in the process of moving.

1.1. Unmanned Ground Vehicle (UGV)

Unmanned Ground Vehicle (UGV) is any mechanical device moving on the surface of the earth in order to transport materials, peoples, or used for any other purposes. The motion is fulfilled autonomously without any driver so that it can deal with the cluttered environment. Generally, there are two types of unmanned ground vehicle (UGV) (Nguyen-Huu and Phuoc-Nguyen, 2009):

- Remote Operated UGV: This type is of UGV controlled remotely by a human (operator).
- Autonomous: This type works without the need for human intervention to control its motion.

The driverless ground vehicle is considered self-movement robots that need to be studied for increasing the effectiveness and efficiency. The driving car considered as one of the most complex tasks that being carried out every day by the human and we note that a large number of accidents resulting from road transport that needs quick and effective solutions. These accidents come from different reasons, some of them related to the quality of vehicles and their severity, some related to the streets quality and rudeness, and some come from the mistakes of humans those driving vehicles. Fully automatic driving leads to minimize human error and thereby reduce accidents significantly. Since ancient times, the Human is trying to do all his works by his hand, such as manufacturing, spinning, cultivating, harvesting, building, reconstructing, trading, eating, and drinking. Manual work has become difficult in some cases because of the difficulty of the work or the high accuracy that are not available in some cases in manual work, so he invented machines and tools that help him to do the works with high accuracy and speed, with all of this, remains some works that is not able to do it with his hands or using the machines that invented by him, so he resorted to manufacturing intelligent machines that can remotely control or programmed to carry out precise and sensitive tasks automatically. The unmanned ground vehicle (UGV), the Mobile robot is the machine that can use it to do some delicate and sensitive jobs that human cannot do it. There are a lot of technical, agricultural, industrial and some works that need to continuously monitor can be carried out by a mobile robot. For example, in medical fields, it can be used in micro-medical operations and also in the transfer process of medical appliances. In agriculture, there are specialized mobile robots for different tasks such as picking fruits and vegetables, fertilization, cultivation of plants, clean up the forest, extinguish fires, cutting down trees. One of the important applications of the Mobile Robot is used in hazardous environments and areas of disasters as well as in demining and inspects and cleans the pipes and tanks of oil. It also has applications in the military sphere, where a self-propelled vehicle is manufactured to use in surveillance and monitoring, as well as, border control, and mine clearance. A mobile robot can also be used to clean windows for high-rise buildings and hard-to-reach places. There are many other applications in which the mobile robot can be used in it.

This study consists of two parts. The first part is to design a prototype of an Unmanned Ground Vehicle (UGV) that can move fully automatically (Autonomous). The vehicle was designed with four wheels used in the motion. The front wheels are used to define the direction (Steering angle) of the vehicle by the DC-Motor installed in the middle of the front axle. That is this vehicle is a front wheel steered mobile robot. The rear wheels are fixed and used to drive the car by two DC-Motors that installed on each rear tire. They are driven separately to provide that the right and left wheels turn at different speeds during rotation because the mobile robot does not have differential gear. The DC-Motors used in this study work with DC voltage in the range of 0-24 volts. Any change in the value and sign of the control voltages leads to a change in the speed and the direction of motion of the DC-Motors. For this aim, the input voltage adjustment should be done for the convenient driving by using lower DC voltages. Three DC-Motor drivers are installed to the mobile robot prototype to control rotational and longitudinal motion. The control inputs to the drivers are generated by Arduino microcontroller card connected to each DC-Motor. Arduino can supply a control voltage in the range of 0-5 volt via PWM output pins. All communications between the computer and the vehicle prototype are implemented by Arduino. The other part of this study includes the kinematic modeling and control of the vehicle for the Trajectory-Tracking and Path-Following, to design a control algorithm that makes the vehicle able to move on particular trajectories (or paths) and control the position and speed of the vehicle with the acceptable positional errors. The position and the direction of the mobile robot designed are controlled by a feedback control law which is a PD control with some modification. The control system of this robot involves a GPS (Global Positioning System), an IMU (Inertial Measurement Unit), Encoder and Magnetometer as sensors to find the current position and direction of the vehicle. This system (GPS and IMU) can work in different environments whether it is closed or open environments. In this research, the work area is an open-air environment. The system with GPS and IMU can also work in urban areas, valleys, tunnels, or among others. To get good results, you must use GPS with higher resolution to locate the vehicle accurately and compare it with the predefined trajectories. The measured data from the integration with GPS/IMU gives the real position and orientation of the car and this pose is compared with the predefined trajectory (or path) to find the errors (distance and direction errors) to produce the corresponding output signals. In the meanwhile, an encoder is used to measure the steering angle indirectly and a magnetometer gives the orientation of the vehicle with respect to the Earth frame. These signals define the required speed to the vehicle and the steering angle. Compensating the errors is carried out on a PC by using a user defined program.

The ground vehicle industry, in general, is moving toward self-propelled smart vehicles in whole or in part. The role of these vehicles is specified to provide an assist to the driver. To achieve an independent navigation, the vehicle must have:

- Sensors and special device (units) to process the data and find or locate the vehicle position.
- The trajectory followed by the vehicle.
- Sensors to avoid obstacles.

In this thesis, we will deal with first and second one of these tasks, especially how to receive and analyze the data and compare it with the pre-defined trajectory to determine the location and direction of the vehicle. Here, the success in achieving the tasks depends on the vehicle's ability to determine its current location and next direction with high accuracy. The sensors used to process the data and find or locate the vehicle position are combination between (GPS and IMU). The predefined trajectories or (path) can be drawn from custom software such as Google Earth, GIS, etc. Google Earth is a program that uses for maps, geography, and information. It can be used to draw triple-dimensional map of the land through the installation of images obtained from satellites, aerial photography and Geographic Information Systems (GIS). The pre-defined trajectories or paths are splines or known geometrical curves created with specified parameters. These curves are designed via the related control points which are the longitudinal and lateral coordinates selected on Google Earth. The control of the vehicle need to know the: location, speed, direction, and orientation. These measurements are obtained from the GPS/INS installed on the vehicle (Bradford et al., 1996). There are many types of a control method which can be used to control the mobile robot systems such as PID, Fuzzy Logic (FLC), Slide mode control (SMC), etc. In the thesis, we proposed a *Proportional–Derivative (PD) and* Slide Mode Control (SMC) as a controller. Trajectory and path are different each other. The trajectory depends on time while the path is independent from time. This study investigates both the Trajectory-Tracking and Path-Following control for an Unmanned Ground Vehicle by using control devices and methods.

In the studies, the experimental and theoretical results show the success of the prototype of the mobile robot designed and the control methods proposed. To demonstrate the effectiveness of the proposed controller, a sliding mode control (SMC) from the literature (Solea and Nunes, 2007) was adapted to the proposed vehicle. The outcomes proved the proposed control achieved better performance compared to the SMC with respect to positional error and especially steering angle behavior. Also, the results show that the mobile robot (AUGV) with the control method can successfully track the pre-defined trajectories/paths.

1.2. Motivation of the Studies

As previously mentioned that the topics of smart vehicles have considerable interest. They can be used in many areas of civil, industrial, commercial, military, and also to help man in the driving process. So there are many motives that make us try to study in this area:

1. Control systems in the stability of passenger transport vehicles will depend on (GPS).
2. Need to know more information about the ground vehicles and its motion.
3. Need to improve the quality and accuracy of the information that will help in the development of unmanned ground vehicle (UGV).
4. Unmanned ground vehicle need accurate and powerful navigational information.

5. Mobile robots could be used in agricultural, industry, military areas, and others.

1.3. Aim of the Study

The aims of the study can be summarized in the following points:

1. The aim of this research is to design an unmanned ground vehicle to be driven on a particular trajectory or path with the least percentage of errors.
2. Using the different type of modern control system to control the motion of the vehicle with the specified trajectory (or path).
3. Exam which kind of controller is convenient for our systems.
4. Benefit from the new controller devices.
5. Find and draw a trajectory that can be drawn digitally and access to digital maps on the computer using one of the specialized programs for this purpose such as Google Earth, GIS, etc.
6. In the control operation, process the current position data of the vehicle obtained by GPS/IMU devices installed on the vehicle.
7. Write a computer program to model the vehicle, process the positional data and follow the trajectory/path.

1.4. Organization of the Thesis

1. Chapter one: contains a general introduction to the mobile robots and methods of control of self-propelled vehicles, the motivation for research, aims of the search and organization of the research.
2. Chapter two: introduces the previous similar related works that relate to the ways of control, mobile robots as well as the devices used in research and the types of the control methods.
3. Chapter three: presents all materials used in this research and the vehicle designed in this study, also drives kinematic and dynamic equations, control type and speed control.
4. Chapter four: includes the results that have been obtained through the study.
5. Chapter five: gives a summary of the results obtained, discusses and conclusions of the results, and offers a suggestion for future works.

2. LITERATURE REVIEW

Many authors have recently focused on the mobile robots. These studies can be investigated in different categories such as mobile robot types, design and control of mobile robots, vehicle modeling, global positioning system (GPS) in control, digital Earth maps, and hardware of mobile systems.

2.1. Mobile Robots

Mobile Robot is an automatic machine that has the ability to self-motion in a specified environment and have applications in multiple areas of life such as the fields of military and espionage, to monitor the border of the country, the unmanned aircraft, agricultural fields in harvest vehicle, vehicles in the civilian fields, boats used for transporting goods and passengers etc. Especially mobile robots with wheels are considered the most common and widely used in industrial robots and service robots. Mainly when self-movement is required above the ground and smooth surfaces, due to its ability to rapidly maneuver, his controllers are simple, and it has conservation of energy feature. So there are a lot of researches and studies in this area. Here, we'll present a few of these researches. For example, (Di Paola, 2010) studied the independent mobile robot in a closed environment. They suggested a monitoring system and the system of public functions in a closed environment. They found through this study the proposed monitor system handles a large number of problems related to the environment draw maps, resettlement, independent navigation, as well as monitoring tasks. In life, this type of proposed system can be using to monitor several areas such as museums, airports, public buildings and others. (Xianchao Long, 2013) provide an active way to study tactile sensor based mobile robot navigation. A MATLAB simulator which can simulate the properties of the tactile sensors, the environment and the motion of the robot, is studied. The simulator uses an abstraction model of a compliant tactile sensor to represent an array of sensors covering the robot. The tactile sensor can find out shear and normal forces. The simulator has been used by the set of human subjects to drive the robot in an indoor environment to catch data.

André Guilherme Nogueira Coelho dos Santos (2008), studied the viability of Smartphone embedded system to work as control unit navigation of mobile robots. This study has used a mobile robot named (Lego's NXT Mindstorms) [Lego's NXT Mind storms: is a collection of programmable robots, issued by the LEGO Group in 2006] and a Smartphone. The Smartphone used as a control unit by using a Bluetooth technology Figure 2.1. The mobile robot tasted experimentally. The experiment consist three main tasks: moving of mobile from the start point to the end point of pre-defined path using the Path-Following method and avoiding all obstacles in the environment, mapping an area of the environment, and localization the mobile robot within the environment. The practical experiences and techniques used in this research showed that the system was a success in his three plans.

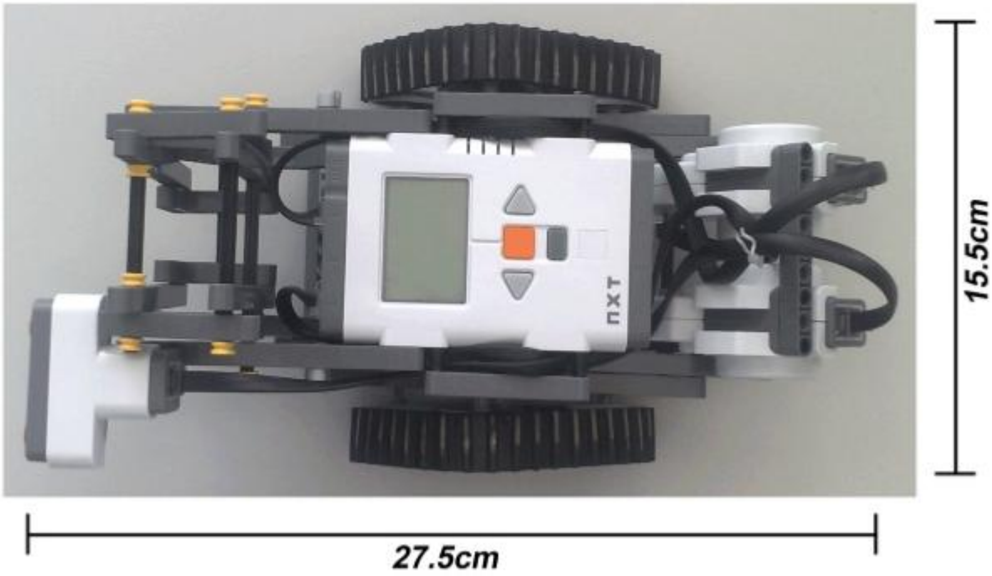


Figure 2.1. Mobile Robot prototype (André Guilherme, 2008).

Bayar et al. (2009), designed a configurable mobile robot. Because there are many motives, make researchers to study this specialty, and increased demand for the mobile robot, this field has become in constant evolution, especially these robots become used in many areas. In this research, in this the researchers designed and manufactured the configurable mobile robot for all terrain applications. This mobile robot can be driven by wheels or tracks, or both (Figure 2.2). This mobile robot can walk on various terrains with high efficiency. The body of the mobile robot has been designed in a way that you can add any equipment without the need for complex manufacturing processes. This research presents the design of the mobile robot (Configurable Mobile Robot for All Terrain Applications) and the details of his industry where they use a simple model of platform as well as it displays the tests performed on it.

Figure 2.2. Manufactured reconfigurable mobile platform (Bayaret al., 2009).

The vehicle that is capable of navigating and sensing its environment without the driver it's called autonomous vehicle (self-driving car, driverless car, and robotic car). There are various systems that work in combination with each other to control a driverless car such as Radar sensors, Lidar sensors, cameras, powerful mapping technologies, and others. The example of driverless car mobile robots is Google's driverless car. The researcher (Guizzo, 2011) explained that how Google’s self-driving car is work. The car is installed with a group of sensors and software's that reveal and predict the behaviors of road users from a long distance (Figure 2.3.).

Figure 2.3. Google's self-driving car.

Loughnane (2001), designed a mobile robot that moves on the wheels and works in the indoor environment as a unit of a security and monitoring. This mobile robot motion is a self-movement without any external interference, and the system consists of sensors mounted on the mobile robot. These sensors detect obstacles and determine the distance that the mobile robot traveled and the mobile robot have the ability to maneuver within a distance of less than one meter.

Elkaim and Connors (2008), designed the Overbot (mobile robot) shown in Figure 2.4. The Overbot is a self-propelled vehicle with four tires moving on the surface of the earth (Autonomous Unmanned Ground Vehicle UGV).The vehicle designed for off-road environments and has an ability to climb over large obstacles. The top speed of the vehicle reaches to 40 mph. The vehicle is actuated by a combination of controllers and devices (Hardware). For example, it uses PD controller on straight roads. Visioning devices and the devices that find the position of the vehicle are equipped on the vehicle (mobile robot) for navigation and sensing. It also uses GPS to find its position. Different control methods are examined on the vehicle on the different types of road or tracks.

Figure 2.4. The Overbot autonomous UGV.

Another type of mobile robot that moves in the air called unmanned aerial vehicle (UAV). The unmanned aerial vehicle is an aircraft without a pilot on aboard, which flight either dependent on pre-defined program behaviors or dependent on remotely piloted by a ground station. There are many researchers studying this type of mobile robot such as (Suiçmez, 2014). This study proposed the Trajectory-Tracking of a Quadrotor Unmanned Aerial Vehicle (UAV) by controlling UAV position and attitude simultaneously. The Quadrotor is a small rotary-wing UAV which has the ability to land

and take off vertically. The quadrotor used in this study called (AscTech Hummingbird) showed in Figure 2.5. AscTech Hummingbird has four propellers operated by four brushless motors, these motors work separately thus providing the necessary force to take off separately and the vehicle also contains electronic units to control the speed of the vehicle. Two control methods are used to track the required trajectory precisely. The first method is a nonlinear control method called (back stepping control) while the second control method is Linear Quadratic Tracking.

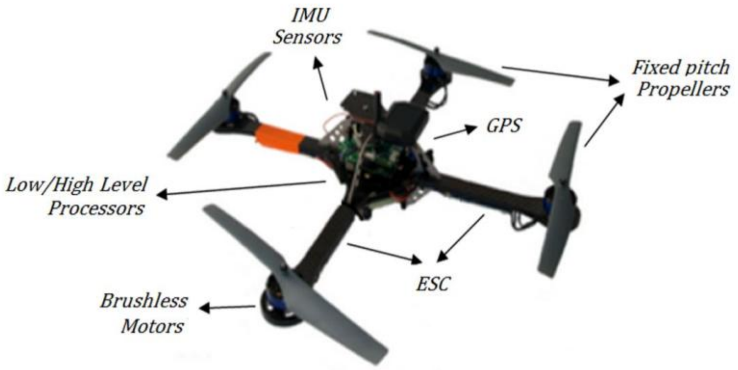


Figure 2.5."AscTech Hummigbird" quadrotor used in this thesis (Suiçmez, 2014).

2.2. Control of Mobile Robots

The control of any vehicle (mobile robot) needs to know the following vehicle parameters:

A. Position.
B. Velocity.
C. The Direction of travel.
D. Orientation.

The previous parameters can be measured by using global position system (GPS), or any other type of sensors. To control the system in this research, there are two main parameters that must be controlled simultaneously [d(error) and ϴ(error)] as shown in Figure 2.6. There are many methods to control the system such as PID, FLC, SMC and etc. We will display some of the previous studies in this area.

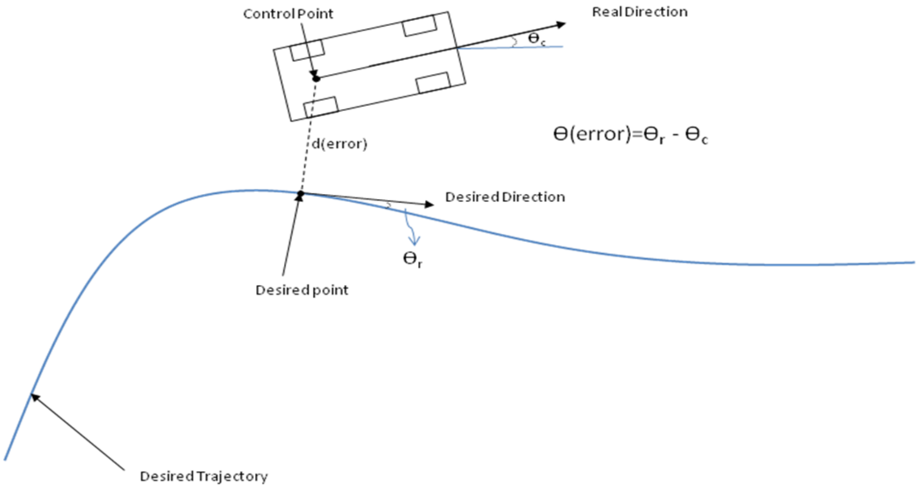


Figure 2.6. Vehicle's real position and the trajectory.

Ouadah et al. (2008), designed a car-like robot called (Robucar) for the purpose of transporting passengers. Fuzzy logic controllers (FLC) are utilized to perform an accurate and efficient positioning of an AMR with correct orientation. To achieve this aim two FLC controllers have been improved:

A. Robot positioning controller (RPC).

B. Robot following controller (RFC).

To exam and show the activity of the suggested technique, both theoretical (computer simulation), and the real-time test was done on the different trajectories. Comparing theoretical and practical results shows the effectiveness of the suggested system. The results showed a similarity in velocity profile in both simulation and experimental testing, and also the change in the steering angle in both practical and theoretical tests was very close to each other.

The Trajectory-Tracking is one of the important things in autonomous self-movement vehicle. So the researchers (Zakaria et al., 2012) proposed a control type for the Trajectory-Tracking of a self-propelled vehicle. This control type includes the control on the direction of the vehicle and the lateral motion of the vehicle as shown in Figure 2.7. The prediction of the future point of the vehicle is the basis for calculating the lateral error. The control system is depending on the steering angle and yaw rate. The system that control the vehicle is based on the relationship between the lateral error of future point, the linear speed of the vehicle, the error in the direction of the vehicle and reference yaw rate.

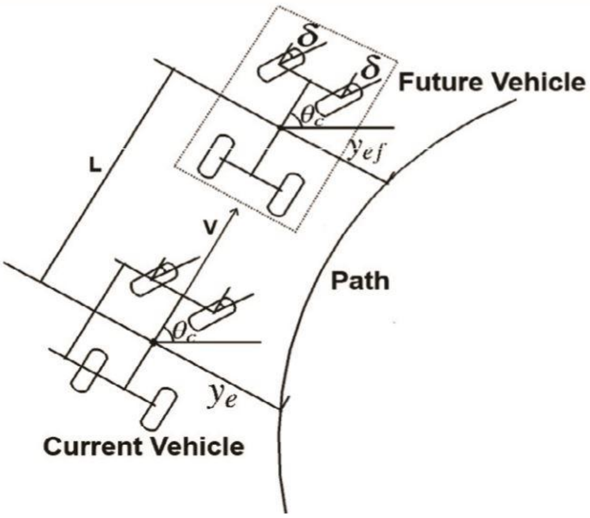


Figure 2.7. Current position and future position of the vehicle (Zakaria et al., 2012).

At the same subject, (Solea et al., 2009) studied the control of the vehicle (mobile robot) moving by the wheels. This work used a slide mode control (SMC) to resolve the problem of vehicle traffic on dedicated track. The most important advantages for the SMC are a rapid response, durability with foreign turbulence, and good transitory. In the different research, the authors (Dagci et al., 2003) treated the problem of following the track by the mobile robot. They designed the vehicle to follow the path and implement it with the SMC. Initially, the original signal aggregation algorithm by using position information is used, after that, two SMC's are implemented, one for the dynamic model and the other for the kinematic model of the vehicle. The proposed algorithms were tested through experiments, as well as through simulations. Where the results show that the vehicle can tracks the path smoothly, velocity profile and steering angle behaviors are in both simulations and experiments are very similar.

The technology of artificial intelligence is used widely in the process of system control. The control methods related to artificial intelligent is trying to imitate human movements and behavior. Especially, the normal control systems (classic control systems) are complex, and their designs need a lot of time and effort. The researchers (Naranjo et al., 2004) used another type of control system which examines controlling the speed and the direction of vehicles with self-navigating vehicles (smart vehicles). This work studied the development of steering control depending on a fuzzy logic controller. In this work, the control system has been installed and tested on real vehicles. The steering system has become a self-navigating with the ability to control it by computer. Through experiments, it was demonstrated that with the use of high-precision Global Positioning System (GPS) device (to determine the location and maps accurately) the car could stay on the right track efficiently.

The fuzzy logic control system is another type of control system which has a widely uses in the control applications. Fuzzy logic is an approach to computing depend on the grade of truth instead of the normal logic that the modern computer is based on (true 1 or false 0). (Lin, 2012) used the fuzzy logic control method in the process of tracing a continuous path. They display a robot simulation through the application (Software) called middleware (middleware: is software that works as a link between applications and database or operating system). With some fuzzy logic reasoning rules, satisfactory results obtained for the motion of the robot on the specified path. The study also provided a way to generate a continuous path from the group of points depends on the cubic B-splines. This path has the ability to avoid stationary obstacles.

Naranjo et al. (2005), provided two-layer control structure to automatically move the steering wheel of a real vehicle (mass-produced vehicles: Vehicles produced in large identical quantities are and available to all people). The first layer is to estimate the target position of the steering wheel at any time with depending on FLC controller. The other layer is a classical control based on the actuation of the steering bar with a drive element that achieves the desired position by the first layer. The RTK-DGPS (real-time differential universal positioning system) is utilized as the main sensor for the aim of locating the position.

Open-loop and closed-loop control system is two type of control system. In the open-loop control system does not depend on the output, it just deals with the input but in the closed-loop control system, it considers the current output as the base to generate new input. (Schworer, 2005) has examined the dynamics of the car differential drive that moves on two wheels Figure 2.8. Because of the flexibility of this model, the different control methods can be applied to control the motion. This thesis suggested both open and closed loop control schemes. Control of an autonomous lawn mower is simulated and evaluated by using optimal control for Path-Following and Trajectory-Tracking.

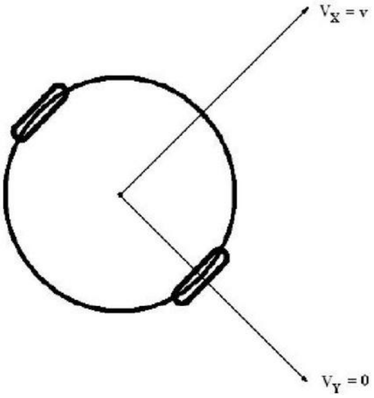


Figure 2.8. Schematic of two-whelled vehicle (Schworer, 2005).

In literature, there are many different methods used also in classical control systems. One of them is sliding mode control. (Solea and Nunes, 2007) targeted to design an autonomous self-propelled vehicle using the slide mode control to control the vehicle on the specified path. They proposed a new design of sliding surface which suggests that lateral and angular errors are internally coupled with each other in a one sliding surface, and this new design leaded to the convergence of both lateral and angular errors.

Proportional integrated derivative (PID) control system is a common control loop feedback mechanism (close-loop control system) used in deferent applications. (Zhao et al., 2012) studied an independent self-propelled vehicle manufactured to follow a specified path in unknown environments. To control the vehicle on the path without errors, an adaptive PID controller has been used (In Adaptive PID the control parameters (kp, kd, ki) does not remain constant, it will automatically be changed when faced with changing environments). By using this method, the vehicle control process could be more effective and get a good flexibility, stability, and adaptability.

Kada and Ghazzawi (2011), described the installation and configuration of the PID control system for complex systems. The unmanned aerial (UAV) vehicle was controlled by this type of controller (PID controller). Through the simulation by the computer appeared that there are improvements in the performance of the vehicle after using PID control system.

Demir et al. (2016), studied the Trajectory-Tracking of an unmanned aerial vehicle (UAV) on the specified trajectory using a self-tuning fuzzy PID controller. Initially, a classic PID controller is utilized for the autonomous tracking of the trajectory by this aerial vehicle. At the first, the control parameter values of PID controller are defined by Ziegler–Nichols method. Then a fuzzy controller is utilized to regulate the control parameters of PID controller. With these control system, the behaviors of the aerial vehicle to follow the specified trajectory with least uncertainty are recognized in the real-time application.

2.3. Vehicle Modeling

The mobile robots try to move on the pre-defined routes. On the planned road, the motion of the vehicle is a superposition of three motions:

1. Longitudinal direction (forward direction) (u).
2. Lateral direction (v).
3. Rotation about the center of gravity (r).

There are several models such as four wheel vehicle model, bicycle model, etc., to calculate the forces affecting to the vehicle and to calculate the speed, direction, and steering angle of the vehicle thus to find the vehicle location at the moment of [(X(t)), (Y(t)), (Ɵ(t))]. (Anderson and Bevly, 2004) explained how to calculate the state of the main vehicles and biases sensor, using (GPS) (Global Positioning System). The Estimation of yaw rate, sideslip, and heading is done by a model based on KF with GPS velocity measurements (Kalman filter: is an algorithm that uses a series of measurements that monitored over a period of time.). The Bicycle model was used to calculate the forces affects on the vehicle. The lateral dynamics of the bicycle model can be explained by taking a sum of the moments and forces about the vehicle's center of gravity. The second estimator utilized in this study is a model-based estimator. The state space of model-based estimator can be obtained from linearizing the bicycle model dynamics equations. The parameters of the main vehicle such as cornering stiffness are determined and used to correct the estimated data from the theoretical model. In this study, the vehicle compares the data measured practically with the calculated values of the theoretical model continually, thus giving the controller a correct model.

Bascetta et al. (2016), presents a method for the Trajectory-Tracking control of an Ackermann steering vehicle and they use kinematic bicycle model to designing the Trajectory-Tracking controller for the Ackermann steering vehicle. In the experiment, they use deferent trajectories such as eight-shape trajectory, and rectangular-shaped trajectory.

Kouros and Petrou (2017), offers a control system for an autonomous four-wheel-drive (4WD) four-wheel-steering (4WS) vehicle. The vehicle's modeling depends on kinematic bicycle model but with demonstrating the slipping of the wheels. In order to draw the environment, localization, planning, and implementation of motions, the vehicle uses several motors, sensors, and algorithms.

The dynamic models of the vehicle are another type of vehicle modeling. (Pepy et al., 2006) discusses a control system for a mobile robot tracking predefined path. In the study, they use a five degree of freedom dynamic model to describe the forces act on the vehicle. The state of the car is defined by four variables(X and Y coordinate in the CG, θ is the vehicle direction, and r is the yaw rate). The bicycle model was used to simplify the model by combining the four wheels of the vehicle on two wheels located at the center of the vehicle Figure 2.9.

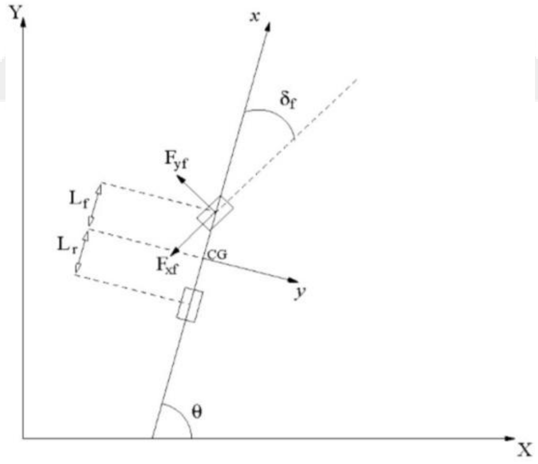


Figure 2.9. Simplified dynamic model (Pepy et al., 2006).

Chen and Jiao (2011), suggested a control system depend on the non-linear dynamic model. This dynamic model combines between the constraints of kinematic and vehicle dynamics. the side slipe friction forces of the wheel are taken into account which exists in the high speeds vehicles or in the massive vehicle.

Path-Following is a modeling control system that drives an mobile robot to reach and follows the given path. (Kavya and Arun, 2016) presented a Path-Following control method to control a vehicle movement on the pre-defined path. The vehicle is moving with two DC-Motors which controlled by a PID control methods. The path specified and followed by the vehicle in this study is a circular path.

2.4. Global Positioning System (GPS)

GPS is a navigation system based on the space satellite consisting of a network of 24 satellites in the space. The GPS provides important information (location, time) that can be used in industrial, civil and other fields. This system is used in this research as a sensor mounted on the unmanned ground vehicle to find the current vehicle location. There are many previous researches used the GPS for different of purposes. For example, the authors (Daily and Bevly, 2004) provide a way to use GPS velocity measurement to get a better lateral stability control system. Without knowing the vehicle model, the sideslip angle can be measured for the vehicle. This measurement can be collected with conventional measurements of the vehicle to control the lateral movement of the car. This work presented a way to control a vehicle using GPS, based on measurements of slip angle. To get a high update for the data used in the process control of the vehicle, the GPS measurements are combined with internal sensors measurements of the vehicle. At present, the use of GPS has expanded, and now a lot of passenger vehicles, as well as agricultural vehicles and industrial machinery, are relied on a GPS device.

Hernandez and Kuo (2003), tested the steering control of the passenger vehicle in the automated highway. It is performed several simulations for the possibility of automatic routing, depending on the information obtained from the GPS which relates to the coordinates of the current location (position) of the vehicle and magnetic sign directing system. This process takes place through the feedback of information and signals coming from the GPS device and the magnetic sign system and compares them in order to correct the mistakes that means correcting the direction of the vehicle. The current vehicle location (position) can be identified by monitoring the edges of the vehicle with respect to the magnetic signs. (Abbott and Powell, 1999) studied, the global positing system GPS and its potential to use in many applications for the motion and transport of vehicles. This paper discussed the applications of GPS's use in navigation systems, as well as shows the development of the navigation systems. Also, this paper presents a quantitative test of the effectiveness that individual navigation sensors have on the performance of a ground vehicle navigation systems. A range of navigation sensor performance levels and their effect on the vehicle positioning accuracy are tested.

Since it is most self-navigation and finding the vehicle position based on the GPS.

Georgy (2010), studied, the methods to improve the performance of the vehicle to find a position of the vehicle in the different environments for a long time. To get accurate navigation process, the vehicle should be able to determine the location and the direction precisely and clearly, however, it is noted that the GPS device may not work or work with less accurate in some indoor environments such as in tunnels. So, this research aims to improve the process of finding a vehicle position with high accuracy in different environments and for a long distance.

Kehl et al. (2005), shows a method of the vehicle which follows the predefined path with the assistance of the GPS device. A feedback and feedforward control planner for this purpose has been introduced. The basic aim of this research was to reduce errors and increase the accuracy of the GPS device.

2.5. Digital Earth Mapping Systems

Google Earth is a cartographic and geographic computer program called originally Earth Viewer 3D, created by Keyhole, a company acquired by Google in 2004. The program draws a map of the Earth through the composition of images obtained from satellite imagery, atmospheric photography and geographic information systems of the earth.

We will offer some of the studies conducted in this area. For example, the researchers (Lin et al., 2005) presented, a solution for real-time remote control for vehicles that using mobile communications. In this study, the GPS, Geographical Information Systems (GIS) and GPRS are combined as a mechanism and used as a remote control observation and control system. GPRS mobile communication technology has been adapted as a bridge to transmit control commands and satellite observations. The GPRS can control a vehicle at a greater range within the range of mobile communications coverage.

Henry (2009), examined the possibility of using Google Earth in the geographic information system (GIS) applications. This project was specifically to test the use of vectors and placement of data and the ability to process and display the data in new methods using Google Earth Platform. In this project, they designed a methodological structure to reach its purpose that consists of three basic parts: a database tier, an application tier, and a client tier. Also, (Levie, 2008) presented two methods for visualizing large geographic datasets in Google Earth. The problem is presented in the context of a typical geographic web application. Web applications are generally subject to constraints not found in other client-server system architectures. The nature of the World Wide Web is such that careful consideration must be made in order to simplify rich applications such as Google Earth. The other method that was presented can be used for any geographic dataset. A first method is a simple approach which makes no allowance for network conditions or topology.

2.6. Hardware of Mobile Robots

One of the important parts of the mobile robots is DC-Motor. DC-Motor is an electric motor that converts electrical energy into kinetic energy and works only on DC (Direct Current) power systems. The most popular types rely on the forces produced by the magnetic fields. We are reviewing some of the research conducted on the DC-Motor and how to control the direction and the speed of the DC-Motor using new control devices used in this area, such as Arduino and motor driver card. Also we will review some studies about other hardware devices used in this research such as IMU and encoders.

Găşpăresc (2016), targeted to control automated vehicle speed by controlling the DC-Motor speed utilizing PID controller with reference to obstacles that will be confronted by the vehicle. The researcher displays a technique for programmed tuning of PID controller to control the speed of DC-Motor utilizing Arduino (Uno) microcontroller. The hardware that utilized in the system is shown in Figure 2.10 consists of five elements, which are:

1- Microcontroller (Arduino Uno board).

2- DC-Motor driver card.

3- Optical encoder.
4- DC-Motor.
5- PC.

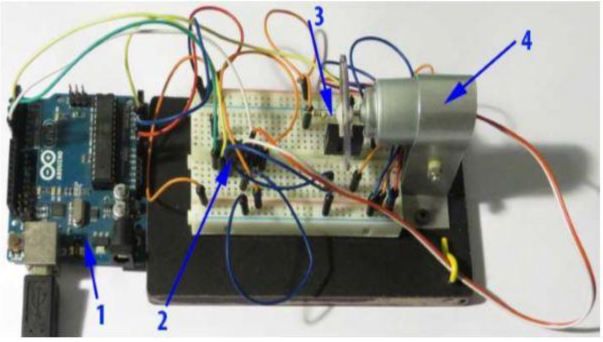


Figure 2.10. System hardware components (Găşpăresc, 2016).

DC-Motor will be interfaced with Simulink using an Arduino Uno board. Microcontroller Arduino Uno board functions the role of low-cost information acquisition board. Utilizing object distance information calculated by sensor PID controller will control the speed of the DC-Motor inside the set point limits.

PID controllers are most popular and more types of controllers used in Industry. The publicity of the PID controllers is due to their wide domain of operating situations and functional simplicity. Different types of tuning principles have proposed which can fine-tune the system to get desired response. So the researchers (Allam and Matla Raju, 2016) present the way to control robotic vehicle speed by controlling DC-Motor speed using PID controller with reference to obstacles that will be faced by vehicles.

Gavran et al. (2017), shows a technical system for DC-Motor speed control. The speed of DC-Motor is controlled by utilizing Arduino (microcontroller) programming platform and MATLAB’s Simulink. This study contains an introduction to using Arduino board and Simulink PI controller in the closed-loop system. As well as the process of programming Arduino with the Simulink is described.

Jayetileke et al. (2014), survey the evolution of a real-time FLC controller to control the speed of a DC-Motor using a microcontroller board. The proposed FLC depends on Mamdani approach and has been examined by high-performance microcontroller board (Arduino Due 32-bit) and utilizing MATLAB program. Through the real-time process, the DC-Motor conduct and the fuzzy logic controller’s reaction were plotted, and the information was put away in MATLAB without interfering with the fuzzy logic controller. The examination was not constrained just for a PC simulation but also for obtaining the system reactions through an experimental setup. Achievement of the microcontroller Arduino Due board has been examined using five various kinds of reference trajectories.

Vikhe et al. (2014), studied a method to control the speed of DC-Motors. The main aim of this research is control the speed of the DC-Motor using a control unit (proportional integral derivative (PID). The process to control the speed of DC-Motor is an important process because any change in the speed of DC-Motor may affect the stability of the system. In the control process, they have used an open source computer hardware and software (Arduino), to facilitate the process of the control system.

Rambabu (2007), studied the speed control of brushless DC-Motor (Brushless DC-Motors are a type of synchronous motor, where magnetic fields generated by the stator and rotor rotate at the same frequency, and it is available in single-phase, 2-phase, and 3-phase configurations), by using two methods of control systems, namely (Fuzzy logic control) and (Proportional Integral Derivative control). So as to obtain the best results in the process control, they used MATLAB software to simulate the system. Other researcher (Awang, 2010) studied a method to control the speed of DC-Motor. This study presents a method used as a controller to control electric current of the DC-Motor named Programmable Interface Controller (PIC) microcontroller.

Another hardware part which is used in our vehicle (mobile robot) is IMU. An Inertial Measurement Unit (IMU) is an electronic device fixed on the vehicle to calculate the orientation, position, velocity, and acceleration of a vehicle.

Radhamani and Thomas (2016), offers a process to combine a low-cost GPS with IMU sensors using KF in order to increase the precision of navigation information. Experimental test for the combining of GPS and IMU was done through KF estimation technique. The results showed that the combination of the data coming from the GPS and Data that comes from the IMU leads to an improvement in the accuracy of the data (orientation, position, velocity, and acceleration of a vehicle in motion), thus improving navigation process.

Brunner et al. (2017), suggest an approach that calculates the relative motion by dint of linear accelerometers. This technique, known as Gyroscope-Free (GF), enables the relative motion to be calculated utilizing only data from various accelerometers installed and spread all around the object. An IMU (Inertial Measurement Unit) which is a relative motion estimator utilizing a KF/EKF without any system model is introduced and performed on real-world test benches. The results display the success of this device in the convenience as a suggested estimator. The objective of this study was to develop multiple IMU-based observers which are independent of any system model. This lets accuracy improvement in navigation algorithms with low-cost sensors by comparing signals from the various IMUs.

In a study about IMU/GPS integration, the researchers (Kim and Lee, 2016) suggested an algorithm for the aim of:

- Evaluating the vehicle's position as a distribution form.
- Controlling the system.
- Measuring the noise covariance in order to compensate for this main disadvantage.

The method that proposed to control noise covariance is processed separately (independently), using sensor error and fading factor While considering the driving situation Figure 2.11.

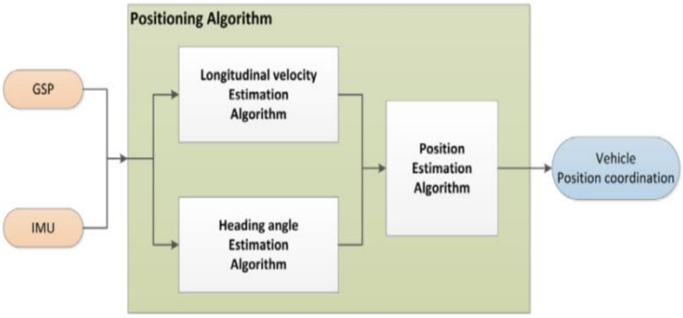


Figure 2.11. Block diagram of the proposed algorithm for vehicular positioning (Kim and Lee, 2016).

Encoder is a hardware used in this research to calculate the value of steering angle. (Ye et al., 2016) presented a navigation system using a GPS-based implemented the navigation to appreciate the auto-steering performance with both the active front and rear steering and Ackermann steering modes. The vehicle was designed as shown in Figure 2.12 and used in the research to investigate the influencing factors for various auto-steering strategies. The steering angles of the vehicle were calculated utilizing four absolute encoders. The maximum steering angle is equal to 90° for all steering wheels which makes it possible for the four wheels independent steering (4WIS) system to place its instantaneous center of rotation (ICR) at any location. This gives the vehicle more flexibility to move at different angles. The optical incremental encoder was utilized to calculate the rotational speed of each wheel.

Figure 2.12. A Four wheel independent steering (4WIS) robotic (Ye et al., 2016).

Toledo et al. (2018), presented an intelligent odometry system for an autonomous vehicle (Figure 2.13). A mathematical model was derived to calculate the vehicle orientation and position. The odometry encoders are used as inputs of the system and the model utilize the distance between wheels and wheels diameter as parameters. A least square minimization is made in order to earn the nominal best parameters. This model will be updated in the real-time to improve the precision of the results. Neural Network is used in experiments and the results are compared to the mathematical model, showing that the neural network based optimization can be a highly robust system to get better precision of the odometric system.

Figure 2.13. Shows the link between the odomatric sensor and the encoder (Toledo et al., 2018).

3. MATERIALS and METHOD

In this chapter, we present the designing, modeling, and control of the Unmanned Ground Vehicle (UGV) as a whole system. The UGV is an autonomous mechanical machine which can move on the rough terrain for in various fields of industry, agriculture, military, and especially in the works that cannot be done by human or difficult for the human. In this chapter, we give first the designing of the UGV by a CAD program and the hardware of the mobile robot with their special features and technologies. Then, the kinematic model of this vehicle is presented by using a simplified model. After that, a control method proposed for the Trajectory-Tracking and Path-Following with the given pre-defined references is studied and compared by another control method from the literature in terms of its performance.

This chapter is divided into two parts, the first part is the materials involving the design and equipments used in the unmanned ground vehicle (UGV) and the second part consists of the methods with the modeling and control procedure for the Trajectory-Tracking and Path-Following control of the vehicle.

3.1. Materials

In this research, we used different mechanical and electronic devices that are required to equip the Unmanned Ground Vehicle. These devices are to ensure the position control of the vehicle motion on the given trajectory or path and also to regulate the speed of the vehicle. For example, to locate the vehicle's position and orientation accurately, we need to install some sensors on this mobile robot to find the current position and direction of the vehicle. For this purpose, a high-precision Global Positional System (GPS) device, an accurate Inertial Measurement Unit (IMU), magnetometer, and an encoder were used to measure the current vehicle's location (position and orientation). Also, these sensors can define the steering and actuation system requirements. In the connecting the devices with each other and programming process, we used a microcontroller of Arduino, which is considered as a miniature computer that can interact with the instruments and tools around it. The following

Table3.1 can clarify all the sensors, actuators and other appliances needed in the prototype design of the mobile robot.

Table 3.1. The devices and tools with their properties

Device or material	Properties
GPS	X360T Multipurpose Portable GNSS
Arduino microcontroller	Arduino Mega 2560
DC-Motors with reduction	300 Watt, 1200 rpm, 24Volt, 13A Rate:1:20
DC-Motor driver card	IBT-2 H-Bridge with Arduino.
Computer	Laptop with the Intel (R) core (TM) i5-4300 CPU @1.90 GHZ 2.5 GHZ and 8.00GB RAM
Battery	Two rechargeable 12V, 120Ah
Encoder	Incremental Rotary Encoder
IMU	MTi-10
Magnetometer	LSM303DLHC
Vehicle chassis	

The following diagram defining the whole system used in this thesis as shown in Figure 3.1.

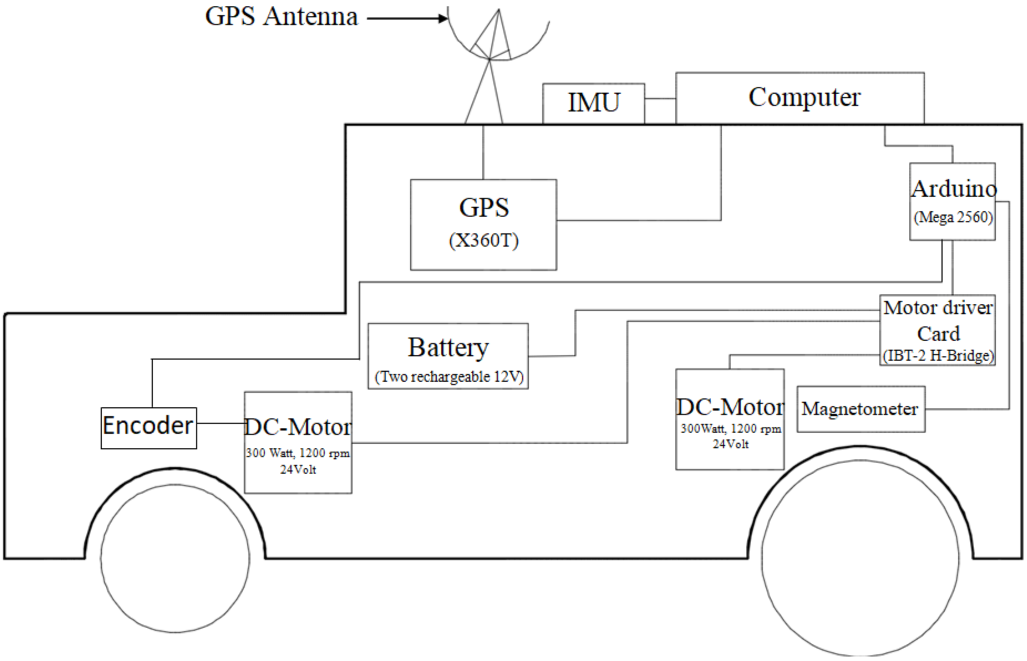


Figure 3.1. Schematic explanation of the basic equipment of the unmanned ground vehicle.

3.1.1. Mechanical parts

In this study, a self-propelled ground vehicle (Autonomous Unmanned Ground Vehicle) with the ability to track a pre-defined trajectory or path was designed first by the SOLIDWORKS CAD program as shown in Figure 3.2. The 3D designs of all mechanical parts of the vehicle were separately and in detail drawn in their real dimensions and mounted for the final shape. In this design procedure, some parts such as the DC-Motors, wheels have been purchased and the others such as the chassis, junction pieces, covers of the vehicle etc. have been manufactured directly according to the needs. According to these CAD drawings, each mechanical and also electrics parts were installed to completed the unmanned ground vehicle. The entire chassis with the welded connections is composed mainly of various pipe profiles, the majority of which are 40 mm in diameter. This mobile robot consists of four wheels. The rear and the front tires have the same diameter of 0.48 m, the distance between the tires on the same axle is equal to 0.93 m, and the distance between rear wheels axle and front wheels axle is 1.10 m.

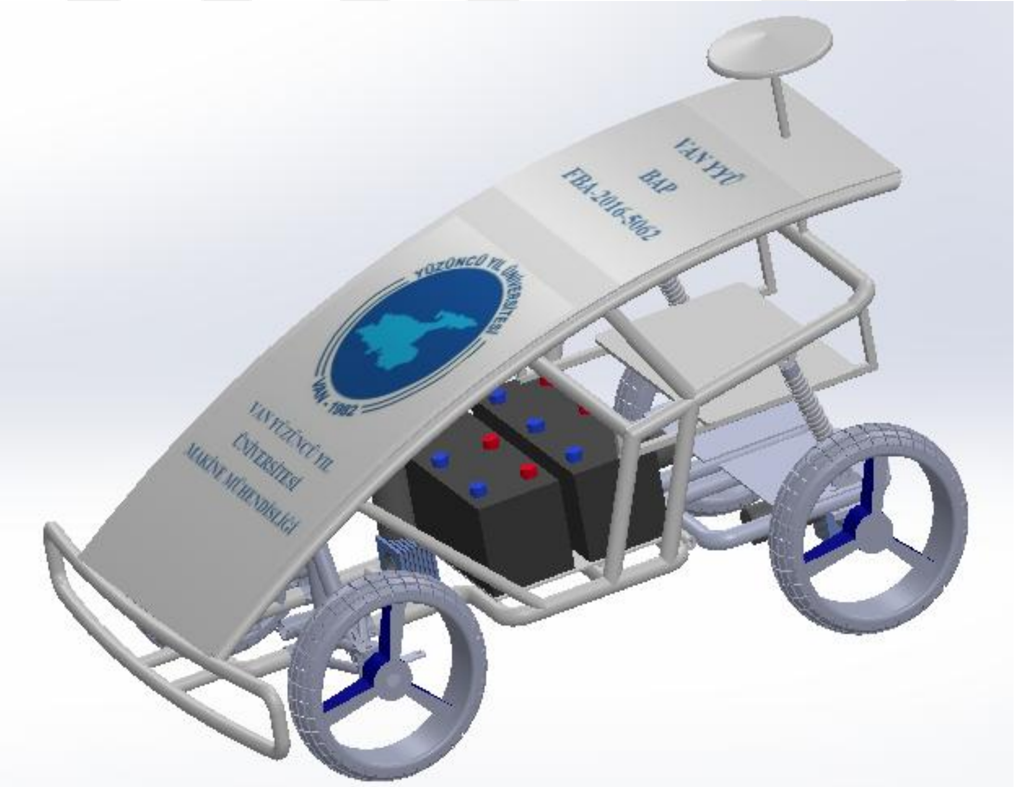


Figure 3.2. Vehicle Model (Car Prototype).

The wheels are attached to the body of the vehicle with the dashpots which involves helical spring and shock absorber to restore the vibration, improve handling during motion, absorb the relief of road defects from bumps and drilling, or even sharp turns. Figure 3.3 shows the suspension system for both the front and rear wheels. The role of the suspension system is not only to adjust the vertical motion and the pace of the motion, but also to adjust the orientation of the wheels in the straight line as in the turns, or when reducing the speed and starting to move.

Figure 3.3. Suspension system.

The rear wheels are connected to the vehicle body separately (not rotating together) and rotate to the clockwise and anticlockwise to supply both forward and backward directions. For this aim, the required torqueses to the wheels are generated via the DC-Motors that attached to the each of the rear wheels, so that the torque is evenly distributed on the rear wheels. In addition, the actuated system of the robot does not have a differential gear. Therefore, to ensure the left and right wheels on the rear to rotate without lateral slippage in case of turning, two DC-Motors driven by different control voltages with respect to the kinematic calculations are used for the different tire speeds. In the meanwhile, the front wheels are connected in a way that they can rotate to the left and right sides with a specified steering angle to give the vehicle the ability to determine the direction of motion. The steering system consists of a DC-Motor that produce a rotational motion converted to the linear motion by using a mechanism called rack-and-pinion (uses a fixed jagged or toothed bar or rail engaging with a smaller gear). The pinion gear is attached to the steering shaft. When the DC-Motor rotates the gear spins, this leads to move the rack. The tie rod at each end of the rack connects to the steering arm on the spindle thus leading to the motion of tires to the sides and giving the required angle. The longitudinal actuation and steering systems are pictured in Figure 3.4 with the picture of the real system.

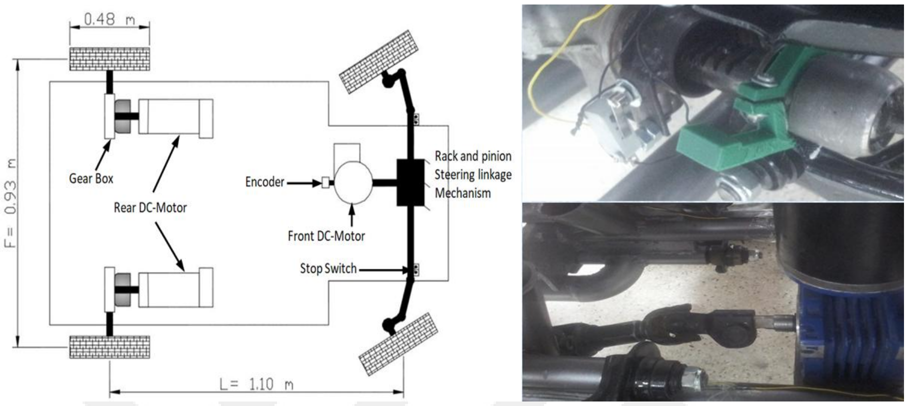


Figure 3.4. Mechanical Actuation and Steering system.

An encoder was connected to the output shaft of the front DC-Motor to measure the angular speed of the tires as well as to know the value of steering angle. The largest value of the steering angle of the vehicle is restricted to $\pm 31^{o}$ to preserve the motor drivers. For this aim, two stop switches were installed on the steering system to cut the power that comes to the DC-Motor when steering angle reach its specified maximum value.

A magnetometer was connected directly to the Arduino for the aim of measuring the real orientation of the vehicle with respect to the Earth coordinate.

3.1.2. Electronic parts

In this section, we will explain the electronic devices used in the study The electronic setup of the whole system is shown in Figure 3.5. This part consists of DC-Motors, DC-Motor drivers, Arduino microcontroller, battery and electric connectors.

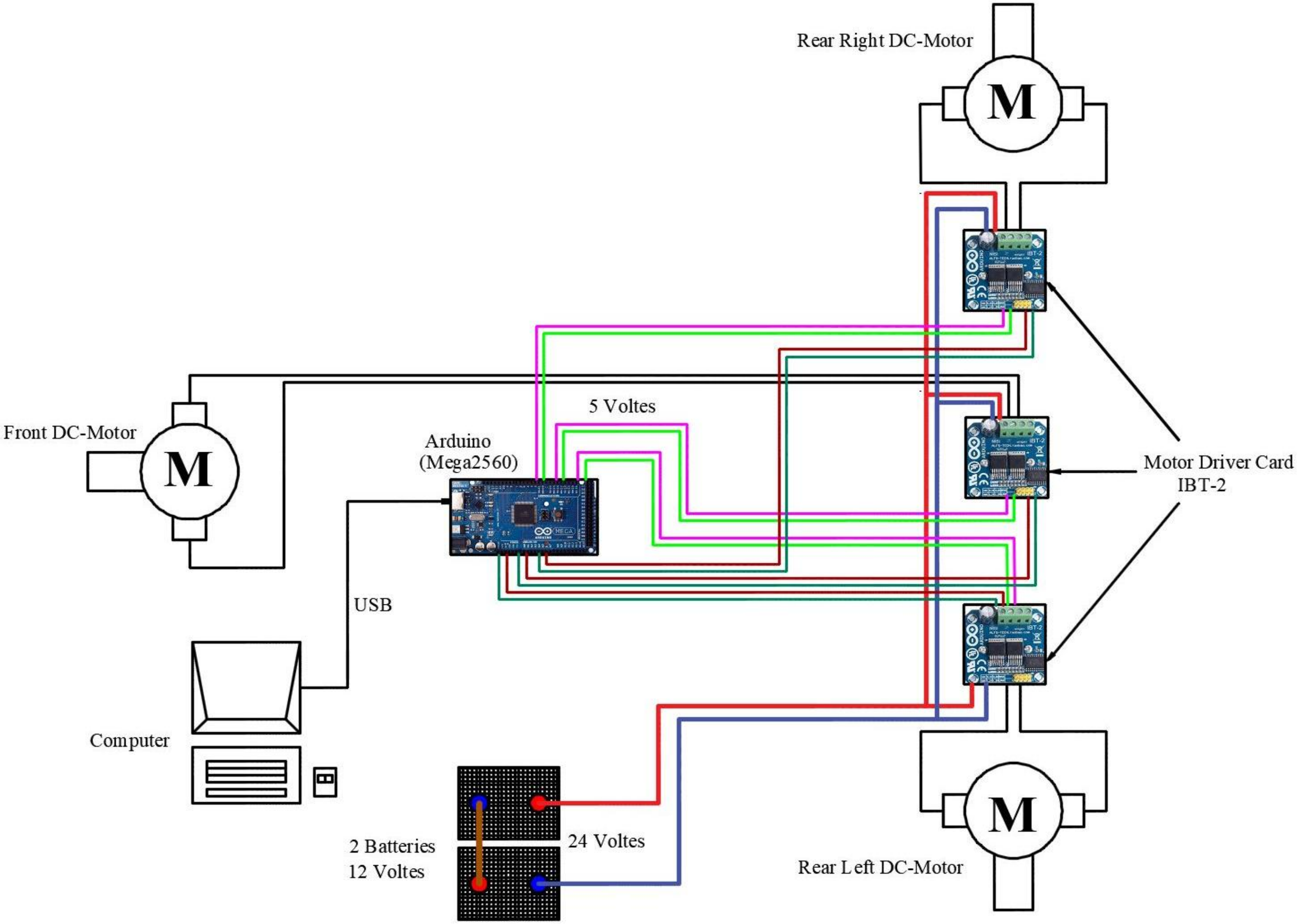


Figure 3.5. The electronic setup of the whole system.

3.1.2.1. DC-Motors

DC-Motor is a machine which converts electrical energy into mechanical kinetic energy and works only on DC power systems. The most popular types rely on the forces produced by magnetic fields. Most types of DC-Motors provide rotary motion. In DC-Motor, you can control the speed smoothly and in most cases can reverse the direction of rotation.

The model designed in this research used three DC-Motors with the reduction with the rate of 1:20. These motors are the Company of KORMAS with the features of 24 DC volts, 300 Watt, 1200 RPM and the maximum current of 13A as shown in Figure 3.6. Two DC-Motors were installed in the back of the vehicle to give the necessary thrust for the translation motion. The torque is disturbed evenly on its rear wheels. The microcontroller (Arduino) with the maximum output voltage of 5V cannot provide required DC volts up to 24 volts so we need to use a motor driver card to provide the required voltages and the directed signals of CW and CCW. The third DC-Motor with a DC-Motor driver was installed to the front wheels to generate the steering angles of the vehicle.

Figure 3.6. DC-Motor used in the Prototype of the Unmanned Ground Vehicle.

3.1.2.2. DC-Motor driver card

DC-Motor driver card is necessary to operate DC-Motor and facilitate the connection between DC-Motor and microcontroller. Using this device you can define the speed as well as the direction easily. Since the microcontroller or other processors cannot provide the voltage and the current that require operating electrically heavy loads devices such as the motors. A DC-Motor driving card was used for providing the voltages and the current which are necessary to run the front and rear DC-Motors.

The microcontroller cannot provide more than 5 volts. However, we need the range of the voltages from 0 to 24 volts to operate the DC-Motors. For this aim, a DC-Motor driver should be used to ensure these required voltages. In the prototype, three DC-Motor drivers were used for each DC-Motor. These motor drivers are the IBT-2 H-bridge which is high power motor driver card based on two BTS7960 chips shown in Figure 3.7. Also, this figure shows the connection a DC-Motor with Arduino. It is Arduino compatible and very suitable to drive the 0-24 DC-volts via Arduino with 0-5 DC volts which are the output of Pulse Width Modulation (PWM). The IBT-2 board has got four power inputs and 8 output-input pins for the connection and operation with Arduino. Some technical features of the IBT-2 are given in the Table 3.2.

Table 3.2. The technical features of IBT-2 H-bridge board

DC-Motor Driver Card	IBT-2 H-bridge board
Input voltage	6V-27V
Input level	3.3V-5V
Maximum Current	43A
Control mode	PWM or level
Duty cycle	0 to 100%
Current conditioning output	yes
Size	50mmx40mm/1.97"X1.58"(inch) (approx)
Weight	66 g

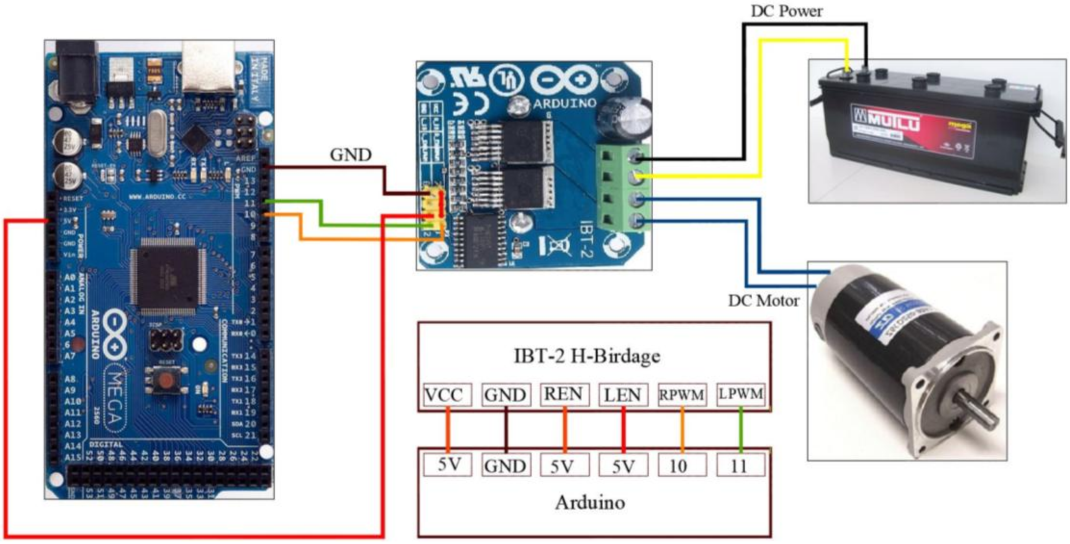


Figure 3.7. The IBT-2 H-bridge motor driver control card and its connections with Arduino.

3.1.2.3. Microcontroller

Arduino is a microcontroller device that can work facilely and powerful. Arduino is an open-source device meaning that it has the possibility of obtaining their software freely and you can change the program codes by yourself. The primary goal of Arduino is to build a reactive digital device that can sense the physical world and control it. The programming language of Arduino is a simplified version of the C or C++ program. Arduino hardware and software were designed for the inventor, designers, hobbyists, hackers, and anyone interested in design interactive objects or environments. Arduino can interact with the surrounding environment. Arduino has a small microcontroller board with a USB plug to connect to the computer and a number of connection sockets that can be wired up to external electronic devices or sensors, buttons, LEDs, speakers, motors, GPS units, electronic control cards, cameras, internet, and even your TV or smart-phone. This flexibility combined with the reality that the Arduino software is free, the hardware boards are pretty cheap, and both of the hardware and software are easy to learn. The most important part of the Arduino is a microcontroller that contains the processor (such as computers) and memory, and some input/output pins that can be controlled, which often called GPIO (General Purpose Input Output) (Smith, 2011). Arduino is a programmable circuit board often referred to it as a microcontroller and also a part of the software, or IDE that runs on computers.

IDE is used to write and upload the computer code to the physical board where they can run these programs even after the separation with the computer connection.

What you need to run Arduino:

- Arduino device board.
- Programming cable type (A to B) that can connect to USB.
- External power source if you need it to work alone (9V battery).
- Special board for external circuits, and wire to connections.
- Computer with suitable operating program (Windows, Mac, etc.) to run the Arduino. (Durfee, 2011).

Arduino can be connected to the computer with a USB cable, micro USB, or FTDI breakout and communicates using standard serial protocol, also can run in standalone mode. Arduino has too many advantages such as:

- Ready to Use: Arduino can be used directly with any manipulation. The package comes in a complete set which includes a microcontroller, 5V regulator, a burner, an oscillator, serial communication interface, LED and headers for the connections.
- Examples of codes: One of the big advantages of the Arduino is its library of examples existing inside the software of Arduino which you just have to click on it and they will serve your objective.
- Effortless functions: There are many functions present in the software of Arduino which makes coding so easy and fast that is not possible with other simple microcontrollers.
- Large community: There are many people (such as engineers, hobbyists, and professionals) are talking and displaying on the internet the projects that they do it on the Arduino. You can simply find assist in everything. Moreover, the Arduino website describes all details about all functions of the Arduino.
- Can be run independently of a computer.
- Can communicate with many programming languages.

There are many kinds of Arduino which differ in their characteristics. The popular differences between the modules are in the number of outputs and inputs, capacity and the speed of the memory, the type of microcontroller processor.

In this research, the most important part is Arduino as a controller. We selected Arduino Mega 2560 shown in Figure 3.8. **Arduino Mega 2560** is a microcontroller board based on the ATmega 2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (Universal Asynchronous Receiver-Transmitter hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP (in-circuit serial programming) header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started (Anonymous, 2016). It is recommended for more spaces such as control, 3D projects, robotic applications and many electronic hobbies. Some technical features of this board are given in the Table 3.3.

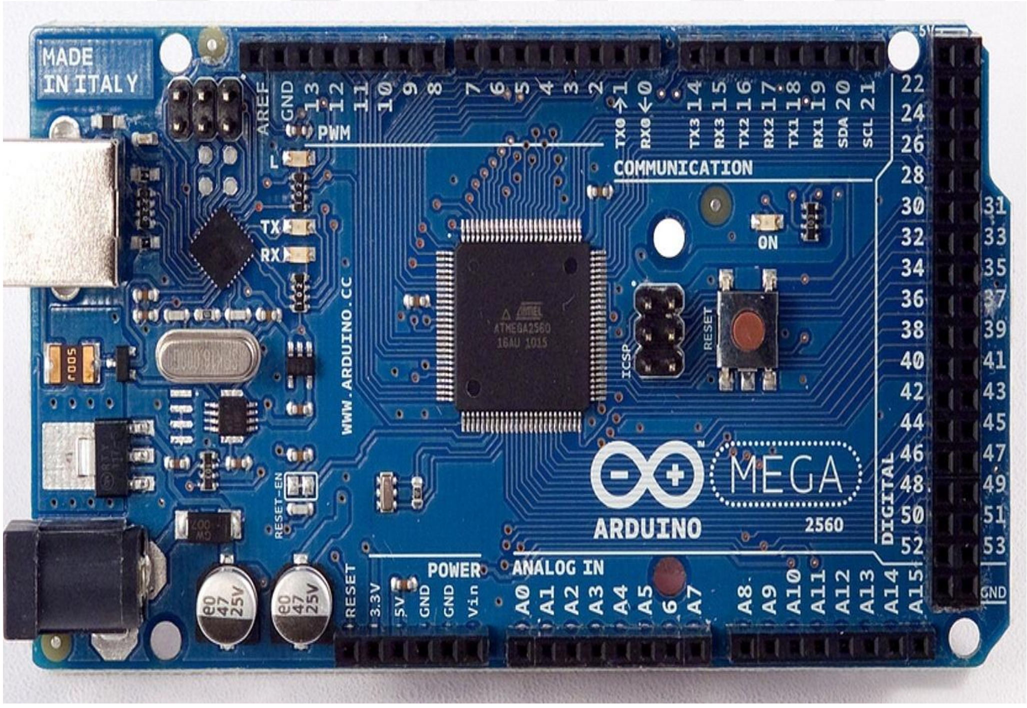


Figure 3.8. Arduino Mega 2560.

Table 3.3. The technical features of Arduino Mega 2560

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by boot loader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16Hz

3.1.2.4. Battery

The battery is the device that produces electricity through a chemical reaction. The battery consists of one or more units called electric cells. Each cell contains all chemical materials and components that enable them to generate electric current. The battery consists of three main parts: the cathode which is the positive terminal, the anode which is the negative terminal and some of the electrolyte. In our work, we used two rechargeable batteries with a mass of 35.3 kg; each provides 12 volts and 120 ampere-hours. They are wired each other in series to give 24 volts (Figure 3.9).

Figure 3.9. Battery.

3.1.2.5. Electric-Electronic connectors

An electrical connector is an electro-mechanical tool used to connect electrical terminations and build an electrical circuit. Electrical connectors consist of plugs and jacks. The connection can be temporary, as for portable devices, require a tool for assembly and plucking out, or work as a permanent electrical link between two wires or appliances. In this study, the male and female jumpers and breadboard were used to connect the drivers, encoder and switches to Arduino. The connections of the devices such as Arduino, GPS, and etc. to the computer were supplied with USB cables. In addition, the electric supply from battery to the motors and drivers via the cooper wires.

3.1.3. Sensors

This study involves controlling the position and orientation of the unmanned ground vehicle for the application of the Trajectory-Tracking. For this aim, we must know the precise real position and orientation of the mobile synchronously. Some sensors such as GPS, IMU, Magnetometer and encoder were used to measure these data.

3.1.3.1. Global positioning system (GPS)

The Global Positioning System (GPS) is a satellite navigation system that gives us information on the time, date and place (provides a three dimensional position in absolute coordinate) on the earth or near it for any ground vehicle, aircraft, and boat in all weathers conditions (Siciliano and Khatib, 2008).

The first uses of this system are in the US military; however, since about 1995, the system has been used in commercial and civil applications, in addition to their military uses. Now, the GPS system currently has 31 active satellites in orbits revolves in different paths around the ground, at the rate of two cycles per a day. The orbits are designed so that there are always 6 of satellites in the view in the most places on the earth. Every time there is a need to connect with four satellites at least to give the information about the vehicle's location. When the number of satellites increases the accuracy of the results that we get are lead to increases, the largest number of satellites that can be seen at once time is 13 satellites (Figure 3.10).

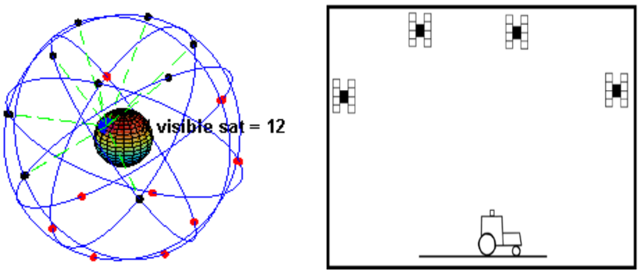
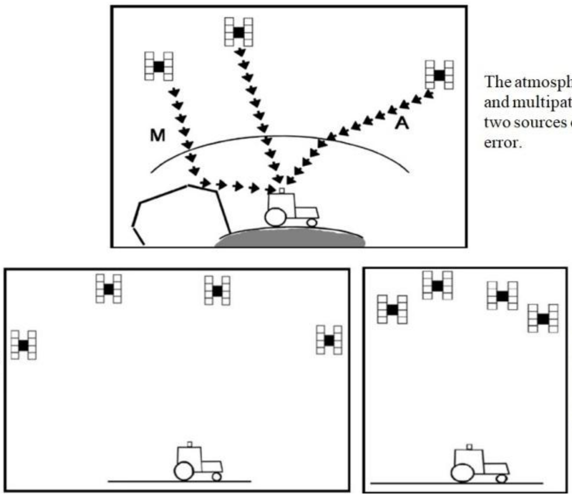


Figure 3.10. Four satellite signals are enough to determine the location on the ground.

The term "accuracy of a GPS" Means the ability of the device to determine the coordinates of a particular location on the surface of the earth with high accuracy. If the coordinates that captured by a GPS receiver were compared to true/known coordinates of any point, there would probably occur errors (or discrepancy) between the captured and known knowledge. The most common factors affecting the GPS accuracy is (Perry and Rains, 2009):

- Clock errors.
- Ephemeris data.
- Satellite geometry (Figure 3.11).
- Multipath errors (Figure 3.11).
- Atmospheric effects (Figure 3.11).



The atmosphere (A) and multipath (M) are two sources of GPS error.

Good satellite constellation. Satellites are spaced far apart for better position calculation.

Poor satellite constellation. Satellites are spaced closely together for a poor position calculation.

Figure 3.11. Some types of possible errors (Perry and Rains, 2009).

Generally, GPS consist of three parts:

- Satellites.

- Ground control stations.
- Receivers (GPS).

The receivers on the vehicle supply to capture the coordinates of the vehicle at the time. The device can be programmed to take the coordinates for each period of time or can be programmed to take the coordinate every certain of driving distance. These points are the real location of the vehicle at the moment and when comparing these points with the actual (reference) points of the trajectory shows us the errors. This operation helps us in the process of the control of the vehicle to follow the trajectory. Also these operations can be used to control process of the speed and acceleration of the vehicle.

In this research, we use the CHC X360T Multipurpose Portable GNSS shown in the Figure 3.12. CHC X360T is a portable, rugged, versatile GNSS receiver which provides high accuracy positioning data to any mobile device through cable or wireless communication.

Accuracy and reliability specifications of the CHC-X360T device that was used in this research are affected by multipath, satellite geometry and atmospheric conditions. This GPS device needs to follow up by five satellites for obtaining satisfactory results. Some features of this GPS device are given in Table 3.4.

Figure 3.12. CHC-X360T Multipurpose Portable GNSS.

Table 3.4. The features of the CHC X360T Multipurpose Portable GNSS

GNSS Characteristics	Type:X360T
Channel	372
Satellite Signals	GPS: L1; GLONASS: L1; BeiDou: B
Update Rate	1 Hz
Protocols	RTCM2.x/RTCM3.x, CMR/CMR+, NMEA 018
Cold Start	< 60 s
Hot Start	< 10 s
Stand alone GPS	1.2 m
Real-time Correction	Horizontal: 2cm
With SBAS	Horizontal: 50cm
Operating system	Linux
Flash memory	16 GB
Processor	454 MHz
Size & Wight	110×81×52mm & 490g (With battery)
Operating temperature	-20 °C to +60 °

In this experimental study, TUSAGA-Aktif / CORS-TR which is constantly observing GPS stations network used to improve the accuracy of the GPS data. This system is a Turkish Governmental service provided by *Tapu ve Kadastro* Genel Müdürlüğü. This system works on under-meter and can give results in sensitivities with the position error up to 2cm.

3.1.3.2. Inertial measurement unit (IMU)

IMU is an electronic device that utilizes measurement systems such as gyroscopes and accelerometers to estimate and reports the orientation, position, velocity, and acceleration of a vehicle in motion (Siciliano and Khatib, 2008). One of the main components of inertial navigation systems used in unmanned aerial vehicles, unmanned ground vehicles, and other unmanned systems is IMU. The computer processes the data compiled from the IMU to specify current position based on velocity and time through DR (DR, dead reckoning: is the process of calculating current position of the vehicle by using a previously specified position, or fixed position, and advancing the current position based upon calculated velocities over prior times and paths or previously known velocity). The GPS system is sometimes unable to provide reliable and continuous positioning due to its inherent reliance on external electromagnetic signals that are affected by external factors such as weather conditions or others. Generally, there are errors produced by the GPS device, whether those related to weather conditions or the multipath errors because of its proximity to the high buildings and other errors mentioned earlier.

GPS and IMU sensors separately display large errors. Therefore, it is recommended to combine the GPS system with the IMU system for the purpose of increasing accuracy because they complement each other by maximizing the features of both two systems. Common applications for IMU include navigation and correction, control and stabilization, unmanned systems control, measurement and testing, and mobile mapping (Radhamani and Thomas, 2016). In this study, MTi-1S-DEV IMU from XSENS Company is used as shown in Figure 3.13. The table 3.5 shows some features of this device.

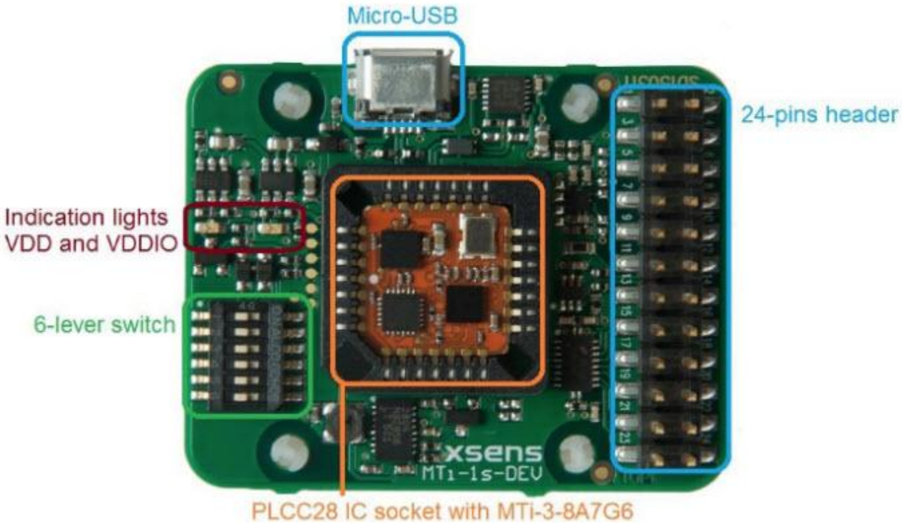


Figure 3.13. MTi-1S-DEV IMU IMU.

Table 3.5. System specifications of MTi-1

The main feature of the MTi-1 IMU			
Size	Width/Length	12.0 mm	PLCC-28 compatible
	Height	2.45 mm	
Wight		0.66 gram	
Temperature		-40 to 85 ºC	Ambient temperature, non-condensing
Power consumption		44 MW	VDD 3.0V; VDDIO 1.8
Timing accuracy		10 ppm	

3.1.3.3. Encoder

An encoder is a type of sensor that detects the mechanical motion and converts it to digital signals according to this motion. Encoder is also an electro-mechanical device to able to generate motion control unit for position, velocity and direction in terms of electrical voltages. There are two various kinds of encoders: linear and rotary.

Incremental Encoder can provide high resolution outputs at a reasonable price. A pulse is generated on encoder in each step of rotation. For instance, a tachometer as an encoder with a single code track converts the speed changes into pulse signals. However, the output of the single-channel encoder does not give the value of direction. To measure the direction of the rotation, a two-channel, or quadrature, encoder utilizes two detectors and two code tracks. The most popular type of incremental encoder utilizes two output channels A and B to sense position. Using two code tracks with sectors positioned 90° out of phase; the two output channels of the quadrature encoder indicate both position and direction of the rotation. If channel A leads channel B, the disk is rotating in a CW direction. If channel B leads channel A, then the disk is rotating in a CCW direction. Therefore, by monitoring both the number of pulses and the relative phase of signals A and B, you can track both the position and direction of rotation (Anonymous, 2008). In this research Incremental Rotary Encoder (E38S6-2-N-24) was connected to the front DC-Motor for the aim to measure the angular velocity and calculate the steering angle of the front wheels. In the meanwhile, incremental encoders don't give an absolute position.

Figure 3.14 Incremental Rotary Encoder (E38S6-2-N-24).

3.1.3.4. Magnetometer

The digital compass that's usually based on a sensor called the magnetometer. A magnetometer is a sensor tool that calculates magnetism or the magnetization of a magnetic material such as a ferromagnetic, direction, strength, and relative variation of a magnetic domain at a particular location.

A magnetic field is a vector quantity with both magnitude and direction. For the aim of calculating the direction more than one axis sensor should be used because the scalar sensor calculates the field magnitude only. The Magnetometer's accuracy readings are influenced by several factors: temperature effects, nearby iron materials, earth field variation, magnetic sensor errors.

The magnetic sensor is classified into three types due to its field sensing range:

- Low field-sensing.
- Medium field-sensing.
- High field-sensing.

In the real-time application of this research, the magnetometer is used with the encoder and IMU and GPS to calculate the real position of the mobile robot accurately. The magnetometer is utilized to define the orientation of the vehicle with respect to the Earth coordinate system especially at the beginning of the navigation process when there is no information from GPS and IMU (Uğur Kayasal, 2007).

In this research, (LSM303DLHC) magnetometer with three digital axis accelerometers and three magnetometer axis were mounted on the center of the rotation of the vehicle and connected to Arduino for the aim to measure the orientation of the vehicle.

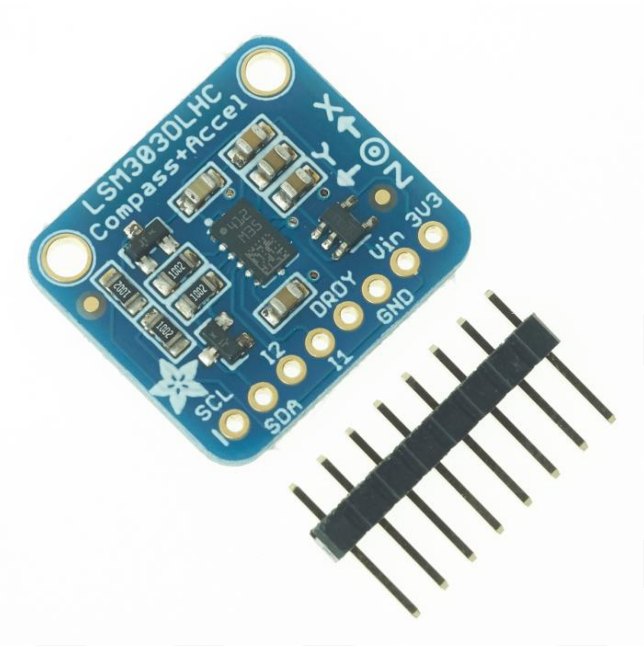


Figure 3.15. Magnetometer Sensor (LSM303DLHC).

Table 3.6. The main futures of (LSM303DLHC) magnetometer

The main features of LSM303DLHC	
Type	GY-511
Size	20.5mmx14.5/0.80"X0.57"
Pin Pich	2.54 mm
Power Supply	3-5V
Magnetic field range	± 1.3 / ± 1.9 / ± 2.5 / ± 4.0 / ± 4.7 / ± 5.6 / ± 8.1 gauss
Acceleration range	± 2 g / ± 4 g / ± 8 g
Communication	100-400Hz
Using the chip	LSM303DLHC
I2C serial bus interface supports	

3.1.4. Computer

The linkage between the computer and the other devices like the microcontroller Arduino and GPS is made by USB. All the process to control the system is executed by a program written with MATLAB software. This program includes many special MATLAB commands related to Arduino to recognize the driver, GPS, IMU, encoder, magnetometer and microcontroller. The controller is run in the computer software and also the pre-defined trajectory or path loaded to the computer through this software. The data taken from the GPS can be recorded, manipulated, and displayed by the computer in the MATLAB environment. In the experiment, a notebook with the Intel (R) core (TM) i5-4300 CPU @1.90 GHZ 2.5 GHZ and 8.00GB RAM was used.

3.2. Method

In this chapter, we will talk about the ways and the methods that used to study the kinematic model and control the vehicle's motion on the pre-designed trajectory or path.

The methods include the mathematical model (kinematic model), control algorithm, signal process, parameter calculation, Google earth mapping system, and others. We will clarify and explain each part separately in this section.

3.2.1. Kinematic model

The goal of this part is to derive kinematic model by placing the nonholonomic state into consideration. These restrictions of the nonholonomic system are used under the premise that there is no sliding at the wheels.

The kinematic model studies the mathematics of motion without considering the forces and moments that affect on the motion. In this study, only the kinematic modeling has been studied to control the unmanned ground vehicle for the Trajectory-Tracking or Path-Following because the vehicle mass is not large and most especially the inertial effects are not important due to the low speed of the robot. The kinematic model of the mobile robot describes the motion with respect to the given reference Cartesian frame. The kinematic model deals with the geometrical relationships that control the system, and the relationship between the control parameters and the behavior of the system in the state space.

There are many different methods to describe the model in this research. The kinematic Bicycle model is considered. The Bicycle model simplifies the four-wheel vehicle by combining the rear wheels together and the front wheels together to take shape of the two-wheeled model, such as the bicycle as shown in the Figure 3.16.

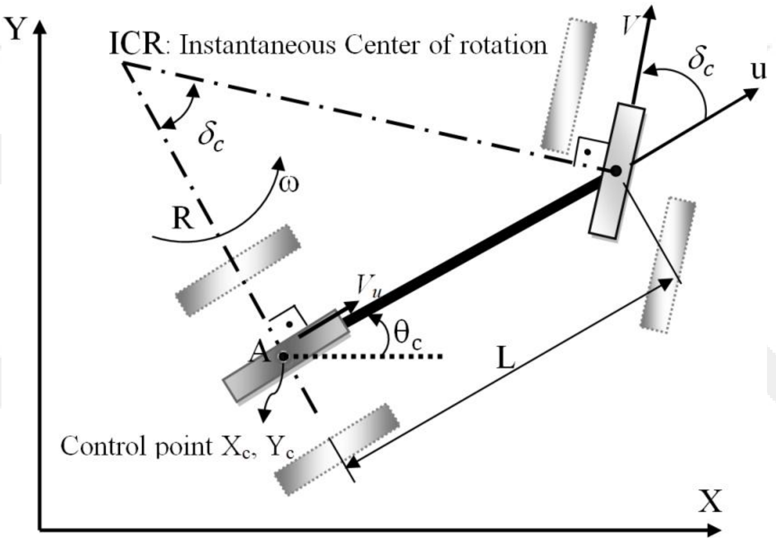


Figure 3.16. Overall coordinates of the vehicle.

The inputs for the Bicycle Model are the speed of the vehicle along the u-axis (V_u), and the steering angle of the vehicle (δ_c). The outputs are the (X_c, Y_c) coordinate of the control point and the direction of the vehicle (θ_c). The vehicle velocities in terms of the components in the X and Y direction are given as follows.

$$\dot{X}_c = V_u \cos\theta_c \tag{3.1}$$

$$\dot{Y}_c = V_u \sin\theta_c \tag{3.2}$$

From the Figure 3.16 above, the steering angle can be calculated according to the instantaneous center of rotation.

$$\tan\delta_c = \frac{L}{R}$$

$$R = \frac{L}{\tan\delta_c}$$

$$\Delta\theta_c = \frac{\Delta d}{R}$$

Dividing both side of the above equation by Δt and take the limit

$$\dot{\theta}_c = \lim_{\Delta t\to 0}\frac{\Delta\theta_c}{\Delta t} = \lim_{\Delta t\to 0}\frac{\Delta d}{\Delta t} * \frac{1}{R}$$

Where, $\lim\limits_{\Delta t\to 0}\frac{\Delta d}{\Delta t} = V_u$

$$\dot{\theta}_c = \frac{V_u}{R}$$

$$\dot{\theta}_c = \frac{V_u}{L}\tan\delta_c \qquad (3.3)$$

Finally, the set of the full state equations is given as follows.

$$\dot{X}_c = V_u\cos\theta_c$$

$$\dot{Y}_c = V_u\sin\theta_c$$

$$\dot{\theta}_c = \frac{V_u}{L}\tan\delta_c$$

In practice, since we don't have a differential gear to give a suitable motion for the tiers during the rotation, each wheel should be actuated separately. This problem was solved by a simple kinematic model to give the different speed values with respect to the steering angle. That is, the problem solution was achieved by a software program written on MATLAB. Figure 3.17 is used to model the differential driving.

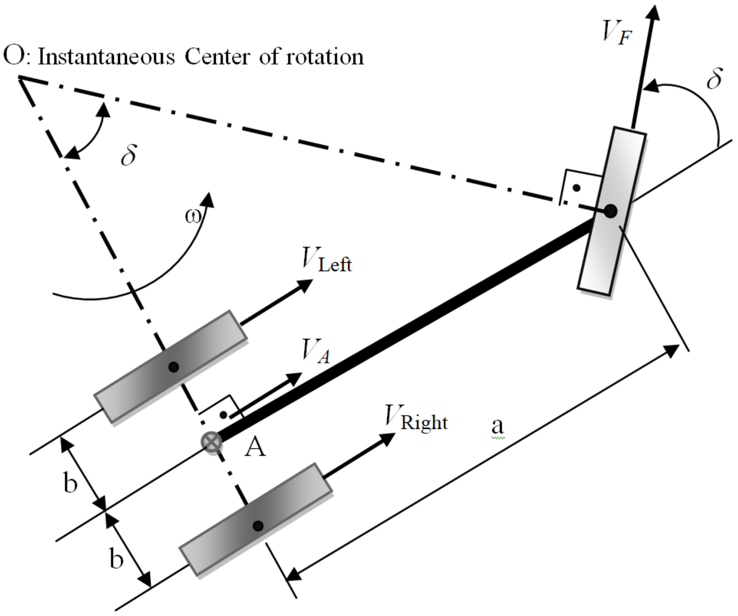


Figure 3.17. Overall tire velocities.

Where:

$|\mathrm{OA}| = \rho$: The radius of curvature for the instantaneous longitudinal velocity of the rear axle.

V_{Left} : The speed of the left wheel

V_{Right} : The speed of the right wheel

V_A : The velocity of the rear axle at the mid point

V_F : The velocity of the front wheel

Here,

$$V_A = \frac{V_{\mathrm{Left}} + V_{\mathrm{Right}}}{2}$$

For the same ω angular velocity for instantaneous speed of the car,

$$V_A = \omega \rho \quad \Rightarrow \omega = \frac{V_A}{\rho}$$

$$V_{\mathrm{Right}} = \omega(\rho + \mathrm{b}) \quad \Rightarrow V_{\mathrm{Right}} = \frac{V_A}{\rho}(\rho + \mathrm{b})$$

$$V_{\mathrm{Left}} = \omega(\rho - \mathrm{b}) \quad \Rightarrow V_{\mathrm{Left}} = \frac{V_A}{\rho}(\rho - \mathrm{b})$$

At the same time,

$$V_{\mathrm{Right}} = \omega_{\mathrm{Right}} \mathrm{r_W} \quad \Rightarrow \omega_{\mathrm{Right}} = \frac{V_A}{\mathrm{r_W}} \frac{(\rho + \mathrm{b})}{\rho}$$

$$V_{\mathrm{Left}} = \omega_{\mathrm{Left}} \mathrm{r_W} \quad \Rightarrow \omega_{\mathrm{Left}} = \frac{V_A}{\mathrm{r_W}} \frac{(\rho - \mathrm{b})}{\rho}$$

Where ω_{Right} and ω_{Left} are the angular velocity of the right and left wheel, $\mathrm{r_W}$ is the radius of the wheels.

From the above figure;

$$\rho = \frac{\mathrm{a}}{\tan \delta}$$

Where δ is steering angle a is the distance between the front and rear axles. Consequently, each angular velocity for the left and right wheels is given as follows.

$$\omega_{\mathrm{Right}} = \frac{V_A}{\mathrm{r_W}} \frac{\left(\mathrm{a} + \sigma_{\mathrm{s}} \mathrm{b} \tan\delta\right)}{\mathrm{a}} \tag{3.4}$$

$$\omega_{\mathrm{Left}} = \frac{V_A}{\mathrm{r_W}} \frac{\left(\mathrm{a} - \sigma_{\mathrm{s}} \mathrm{b} \tan\delta\right)}{\mathrm{a}} \tag{3.5}$$

σ_{s} is changed by the direction of the rotation.

$$\sigma_{\mathrm{s}} = \begin{cases} +1 & \mathrm{CCW} \\ -1 & \mathrm{CW} \end{cases}$$

The angular velocity of the left and right wheels depends completely on the steering angle regardless of slipping and skidding. In turning of the mobile robot, the actuated voltages applied onto each DC-Motor will be adjusted by the control voltages from the microcontroller according to "Eq 3.4" and "Eq 3.5". A relation between the applied voltage and speed of the car was found by the experiment, i.e., the longitudinal speeds were defined in terms of the applied voltage from the microcontroller into DC-Motor drivers. After finding the nominal speed of the vehicle by assuming no slippage, then, the speed of the left and right wheel can be adjusted by the nominal speed and steering angle. So, "Eq's 3.1-3.3" define the kinematic equations used to control the unmanned ground vehicle for the Trajectory-Tracking or Path-Following. As previously stated, we don't need the dynamic equation of motion since the inertial effects can neglected due to the lower speed of the mobile robot.

3.2.2. Control algorithm

In order to achieve the independent navigation process, the vehicle should contain sensors to determine a location of vehicle, identification of a specific trajectory for the vehicle, and to avoid the obstacles (Georgy, 2010) as previously mentioned in the introduction. In this research, we will be interested with the first two points only, in particular, the process of controlling of the vehicle motion on the given trajectory. We designed the control of the position for the autonomous unmanned ground vehicles in the applications of the Trajectory-Tracking and Path-Following by using a classical PD controller with some modifications. In the meanwhile, for the performance evaluation of the controller designed, the results obtained from the proposed control algorithm have been compared with another control method. Sliding mode controller used successfully in literature for Trajectory-Tracking of mobile robots was selected for the comparison.

3.2.2.1. Trajectory-Tracking control

The trajectory is the route that a moving object traces it as a function of time. Trajectory-Tracking involves time as a constraint. This means that you have to be at a certain point in a certain time. So the vehicle might return back in its try to be at a given reference point in the specific time. A vehicle needs smooth generated trajectories with lower associated acceleration and no sudden changes. Lateral and longitudinal acceleration depends on the linear speed. So, the trajectory planning must implement both of the curve planning and the profile of the speed (Solea and Nunes, 2007).

After the mathematical model of the vehicle has been defined, we can now clarify the process and the methods to control the vehicle and try to reduce the positional errors in the tracking applications. Generally, we have three kinds of errors (X_e, Y_e, θ_e), these errors must be controlled simultaneously Figure 3.18.

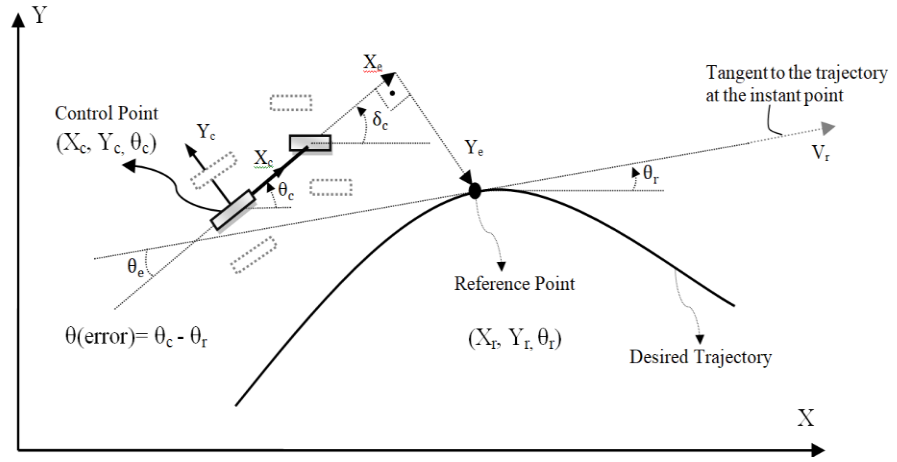


Figure 3.18. The positional features of a vehicle in the Trajectory-Tracking.

The mobile robot in the world frame has three degrees of freedom (X_c, Y_c, θ_c). For this aim, the three errors (X_e, Y_e, θ_e) can be defined as the difference between the real vehicle position and the reference point on the desired trajectory in the world frame. The comparison process to find the errors (X_e, Y_e, θ_e) is repeated with all of the trajectory points.

The errors can be written in the matrix form which is transformed into the coordinate system attached on the reference curve as follows.

$$\begin{bmatrix} X_e \\ Y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos\theta_r & \sin\theta_r & 0 \\ -\sin\theta_r & \cos\theta_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c - X_r \\ Y_c - Y_r \\ \theta_c - \theta_r \end{bmatrix} \tag{3.6}$$

In this equation, the only unknown term is the orientation of the velocity of the reference curve, θ_r as calculated as follows.

$$\theta_r = \operatorname{atan2}\left(V_{rx}, V_{ry}\right) \tag{3.7}$$

Where,

$$V_{rx} = \frac{dX_r}{dt} = \dot{X}_r \text{ and } V_{ry} = \frac{dY_r}{dt} = \dot{Y}_r \tag{3.8}$$

The steering angle in many researches is calculated by (Moret, 2003):

$\dot{\delta}_c = \frac{1}{\tau_s} \cdot \delta_c + c_s \cdot U_2$ (This equation is generally used in fast moving robot but in our study we will use our steering angle equation and we will explain it in parameter estimation section)

In the beginning, you must select the appropriate type of controller that must be simple and easy to implement in the control process and easily adapted to the system by regulating its parameters. In this research, we choose to use two types of controller and using it in the system separately and then comparing the results obtained from each type together. The first controller that used in this research is PD (Proportional-Derivative) controller.

PD (Proportional-Derivative) control compares the real output with the desired output and regulates it to meet the desired output. The aim of using the PD controller is to supply the stability of the system by improving the control since it has a capability to predict the future errors of the system response. In order to avoid the effects of the sudden change in the value of the errors signal, PD controller is often used in control of moving objects (such as flying and underwater vehicles, ships, rockets etc.) (Temel et al., 2013). The general equation for PD controller is:

$$u(t) = k_p\, e(t) + k_d \frac{de(t)}{dt}$$

The inputs of the Bicycle model are the speed of the vehicle (V_u) and the steering angle (δ_c). To control the motion of the vehicle on the pre-defined trajectory correctly with a smaller positional error we need to regulate the inputs. The speed of the vehicle depends on the real location of the vehicle with respect to the corresponding point on the trajectory and also the speed of the vehicle relies on the amount of the longitudinal error (X_e). If the amount of X_e is positive that means the vehicle passes the comparison point, in this case, we need to decrease the speed of the vehicle by decreasing the input voltage represented u_1. Either if the amount of X_e is negative that means the vehicle is behind the comparison point, in this case, we need to increase the speed of the vehicle by increasing the input voltages (u_1). The input voltage ranging from 0 to 5 volts where 5 volt means the maximum speed of the vehicle, which is equal to 1.31 m/s. As for to compensate both the lateral error (Y_e) and the orientation error (θ_e), the steering angle should be adjusted by its amount and direction. If the amount of (Y_e) error is positive, this means the vehicle is on the positive side of the lateral axis with regard to the coordinate frame attached to the reference curve, in this case, the steering angle must be negative (CW). Either if Y_e error is negative that means the vehicle is on the negative side of the lateral axis, the steering angle must be positive (CCW). The value of the steering angle depends on the input voltage (u_2) which is ranging between -5 to 5 volts. The comparison process are repeated with all given trajectory points to find errors (X_e, Y_e, θ_e) and thus to determine the modified inputs. The modified input can be adjusted by changing the coefficients of PD controller so that keep the vehicle on the correct track. In this research, the coefficient of PD is calculated by trial and error methods. Some modification has been made to the classic PD controller for the aim of: (1- The controller firstly tries to push the car into the specific border around the predefined path. 2- Inside the border, the controller supply the car to stay inside the border and follow the reference smoothly (smooth behavior in steering angle). 3- So the steering angle behaves convenient motion in real-time applications. (That means no big oscillation because it’s a mechanical device and most behaves smoothly).

The other type of control system used in this study is SMC. In the definition of any practical control issue, there will usually be an error between the real plant and its mathematical model utilized for its control system design. These errors emerge from unknown outer troubles, plant parameters, and others. Designing of the control system that gives the desired performance output to the closed-loop system in the presence of these outer troubles is a very difficult task. This has led to being a great interest in the field of designing high-precision control systems to control the outer troubles and to get good output results in the approach to its basic design. SMC technique is considered as one of a particular approach to robust controller design.

Sliding mode control is a nonlinear control technique that modifies the dynamics of a nonlinear system by implementation of a discontinuous control signal that makes the system to travel along sliding surface. The state feedback control rule is not a continuous function of the time and it can change the states from one continuous structure to another one based on the current position in the related space. Therefore, sliding mode control is a variable structure control method.

For designing a robust controller for the vehicle that tracks the desired trajectory correctly, the desired trajectory for the vehicle is must be pre-defined by a trajectory designer.

Previously the mathematical model of the vehicle and the errors vector for Trajectory-Tracking was derived by multiplying the matrices and taking the time derivative of "Eq 3.6":

The corresponding error derivatives are

$$\begin{bmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} -V_r + V_u \cos\theta_e + Y_e \dfrac{V_r}{L} \tan\delta_r \\ V_u \sin\theta_e - X_e \dfrac{V_r}{L} \tan\delta_r \\ \dfrac{V_u}{L} \tan\delta_c - \dfrac{V_r}{L} \cdot \tan\delta_r \end{bmatrix} \tag{3.9}$$

Where V_r is the desired reference speed, δ_r is the desired steering angle and L is the inter axle distance.

$$V_r = \sqrt{\dot{X}_r^{\,2} + \dot{Y}_r^{\,2}} \tag{3.10}$$

The angular velocity of the reference is

$$W_r = \dot{\theta}_r = V_r \frac{\left|\dot{X}_r \ddot{Y}_r - \ddot{X}_r \dot{Y}_r\right|}{\sqrt{\left(\dot{X}_r^{\,2} + \dot{Y}_r^{\,2}\right)^{3/2}}} = \frac{V_r}{L} \tan\delta_r \tag{3.11}$$

The typical sliding surface (s) (Slotine and Li, 1991) is give as follows.

$$s = \dot{\tilde{x}} + k\,\tilde{x} \tag{3.12}$$

Where

$$\tilde{x} = x - x_{desired}$$

In this study, the sliding mode control to control this mobile robot has been adapted from two studies (Solea and Nunes, 2007) and (Solea et. al., 2009). The authors designed two sliding surfaces: the first one for X_e (longitudinal error) and the other for the combination of Y_e (lateral error) and θ_e (angular error).

$$s_1 = \dot{X}_e + k_1 X_e$$

$$s_2 = \dot{Y}_e + k_2 Y_e + k_0 \, \mathrm{sat}\left(Y_e\right) \theta_e$$

Where: k_0, k_1, and k_2 are constants.

A practical general form of the reaching law is:

$$\dot{s} = -Q \; g(s) \; - P \; \mathrm{sgn}(s)$$

Where:

$$Q = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix}, \quad P = \begin{bmatrix} p_1 & 0 \\ 0 & p_2 \end{bmatrix},$$

g(s)=s And sgn(s) represents the signum function. However, instead of the signum function, a saturation function has been used to eliminate the chattering effects.

* Different selection of P and Q assign different rates for s and yield various structures in the reaching law.

Finally, the two control input (V_u, δ_c) was calculated by the authors (Solea and Nunes, 2007) and (Solea et. al., 2009) as follows.

$$\dot{V}_u = \frac{1}{\cos\theta e} \cdot \left(-q_1 \cdot s_1 - p_1 \cdot \mathrm{sat}(s_1) - k_1 \cdot \dot{X}e - \dot{W}_r \cdot Ye - W_r \cdot \dot{Y}e + V_u \cdot \dot{\theta}e \cdot \sin\theta e + \dot{V}_r\right) \tag{3.13}$$

$$\delta_c = \arctan\,\Big(\frac{L}{V_u} \cdot W_r + \frac{L}{V_u(V_u \cdot \cos\theta e + k_0 \cdot \mathrm{sat}(Ye)} \cdot \left(-q_2 \cdot s_2 - p_2 \cdot \mathrm{sat}(s_2) - k_2 \cdot \dot{Y}e - \dot{V}_u \cdot \sin\theta e + \dot{W}_r \cdot Xe + W_r \cdot \dot{X}e\right)\Big) \tag{3.14}$$

In this study, the saturation function used in the SMC is proposed with ± 0.5 thresholds.

3.2.2.2. Path-Following control

Path-Following is dealing with the design of a control system that drives an object (robot arm, mobile robot, ship, aircraft, etc.) to reach and follow the given path. In the Path-Following, the mobile robot's convergence to the path is smoother than in

Trajectory-Tracking as shown in Figure 3.19. The path is given in a time-free parameterization and pre-defined by the trajectory planner whereas trajectory is time dependent, i.e., in Trajectory-Tracking, you have to be in the certain location at a specified time. In the Path-Following process, the mobile robot moves with either a constant speed or variable speed depending on the planner. Path-Following is widely used in applications in which temporal errors are less sensitive than spatial errors (position error is more important than time reaching).

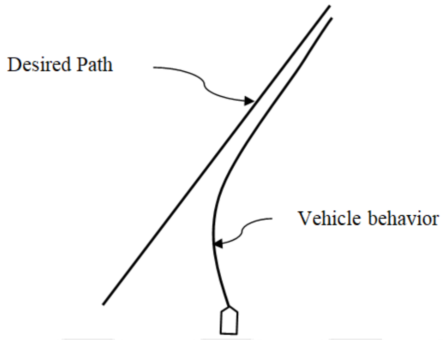


Figure 3.19. Vehicle behaviors in the Path-Following.

In Path-Following, to make the vehicle follow on the pre-specified path correctly, both the θ_e (orientation error) and Y_e (lateral error) must be controlled simultaneously. These errors can be expressed as:

$$\begin{bmatrix} Y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} -\sin\theta_r & \cos\theta_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c - X_r \\ Y_c - Y_r \\ \theta_c - \theta_r \end{bmatrix} \tag{3.15}$$

Where Y_e is the distance between the real position of the vehicle and the closest point to the vehicle on the desired path.

Two types of control systems were used to control the motion of the vehicle on the specified path:

1- PD (proportional-Derivative) control: The control algorithm here depends on the comparison between the real mobile robot position and the position of the closest reference point on the desired path. This comparison process continues until the vehicle reaches the goal. Controlling the value and direction of the steering angle depends on the quantity of both Y_e and θ_e. The procedure for the Path-Following of the unmanned ground vehicle is the same as the Trajectory-Tracking control in the previous section. The only difference is that the Path-Following is free to define the longitudinal speed of the vehicle. However, in the applications, to obtain a smoother motion and the vehicle to complete the path as soon as possible, the speed of the vehicle is also allowed to be adjusted. "Eq's 3.6-3.15" has been used in the Path-Following. In addition, finding the closest point to the vehicle on the pre-defined path was added to the control procedure. The procedure (MATLAB Function) that used to finding the nearest point on the path to vehicle is given in the appendix 1.

2- Sliding Mode Control: The method has again been adopted from the paper in related literature by (Solea and Nunes, 2007). For the errors (Y_e, θ_e), "Eq's 3.6-3.15" is also valid in this method. The only differences are the equations below.
The corresponding error derivatives are:

$$\dot{Y}_e = V_u \sin\theta_e$$

$$\dot{\theta}_e = \dot{\theta}_r = \frac{V_u}{L} \tan\delta_c$$

Sliding surface internally coupled the Y_e (lateral error), and θ_e (orientation error) with each other, which leads to the convergence of both variables. For this aim the following sliding surface is suggested:

$$s = \dot{Y}_e + k\,Y_e + k_0 \operatorname{sat}\left(Y_e\right)\theta_e$$

A practical general form of the reaching law for the SMC is:

$$\dot{s} = -Q\,.\,s - P.\ \operatorname{sat}\left(s\right)$$

In Path-Following, there are only two variables (Y_e, θ_e) and just one control input, which means that we have only one sliding surface (Y_e and θ_e are coupled in one sliding surface). The steering control law is obtained as follows.

$$\delta_c = \arctan\left(\frac{L}{V_u}.\frac{-Q\ s - P\ \operatorname{sat}\left(s\right) - kV_u \cdot \sin\theta_e}{V_u\cos\theta_e + k_0 \operatorname{sat}\left(Ye\right)}\right) \tag{3.16}$$

3.2.3. Parameter estimation

In this study, the vehicle model involves many parameters for the control of the Trajectory-Tracking or Path-Following. As expressed previously, the inertial effects have been neglected due to the lower speed of the mobile robot. The only remaining parameters are the kinematic parameters of the vehicle to be defined. These parameters are the longitudinal speed versus the related control voltage and the angular velocity of the front wheel for the calculation of the steering angle. It would be more realistic to specify these parameters experimentally since they depend on some coefficients and gains of the motors and gear systems and also to define some features of the vehicle especially the frictional effects is very difficult. The definition procedure is given as follows.

1- Applied control voltage definition for longitudinal speed of the mobile robot:

In order to calculate the speed constant of driver motor, some experiments were done on the mobile robot. The experiment results are given for the applied control voltage versus the measured distance as follows. The experiment was performed in a straight line and flat way. The control voltage is the voltage applied on the PWM pins of Arduino. The average speed is also calculated for the given specific time. Each test was performed at the specific time of 20 seconds and the results shown in Table 3.7.

Table 3.7. Experimental result of average speed

Experiment No	Applied Control Voltage (VOLT)	Measure Distance (METER)	Average Speed (m/s)
1	0.5	0	0
2	0.75	1.95	0.0975
3	1.00	3.46	0.173
4	1.25	4.75	0.2375
5	1.50	6.20	0.31
6	1.75	7.66	0.383
7	2.00	9.00	0.45
8	2.25	10.45	0.5225
9	2.50	12.00	0.60
10	2.75	13.40	0.67
11	3.00	15.10	0.755
12	3.25	16.50	0.825
13	3.50	17.84	0.892
14	3.75	19.40	0.97
15	4.00	21.20	1.06
16	4.25	22.30	1.115
17	4.50	23.70	1.185
18	4.75	24.75	1.2375
19	5.00	26.20	1.31

The driver speed of the vehicle was created from Table 3.7 with respect to the control voltage in the range of 0-5V shown as in Figure 3.20. However, the mobile robot cannot move at the voltages below 0.5V. The relation between the linear speed and the control voltage is specified for this speed change as follows.

$$\mathrm{V_u}\left(u_1\right)=\begin{cases}0.2908u_1-0.1265 & \text{if} \quad u_1>0.5[\text{volt}]\\ 0 & \text{if} \quad u_1\leq 0.5[\text{volt}]\end{cases} \tag{3.17}$$

$\mathrm{V_u}\left(u_1\right)[\text{m/s}]$: Longitudinal speed of the mobile robot

u_1 $[\text{m/s}]$: Applied control voltage from Arduino.

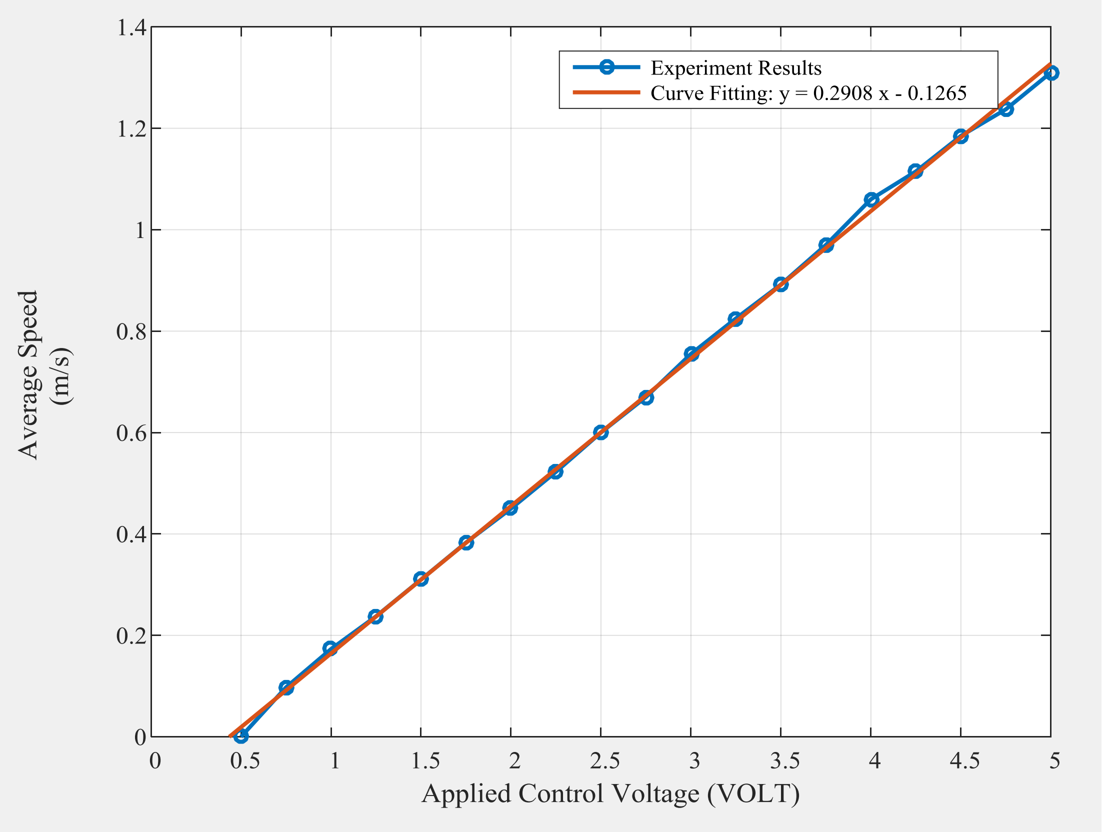


Figure 3.20. Applied control voltage and average speed of the vehicle.

2- Applied control voltage definition for steering angle of Mobile Robot.

To carry out the steering angle depending on the control voltage, the same procedure like the definition of the longitudinal speed was applied to the front wheel steering. In this case, the time is measured for the different control voltages from Arduino between the limited steering angles which are in the range of $+31^0$ and -31^0.

The restricted angles are supplied with two stop switches. The results are given in the Table 3.8 and the relation is obtained from Figure 3.21. Again, because of the friction, the front wheels cannot be steered in all voltages of 0-5V. Therefore, a relation is also assigned for this control action.

Table 3.8. Experimental result of average angular speed of the steering

Experiment No	Applied Control Voltage (VOLT)	Angular Speed (rad/s)
1	1.5	0.599
2	2	0.1119
3	2.5	0.1536
4	3	0.2020
5	3.5	0.2661
6	4	0.3030
7	5	0.3967

The angular speed of the vehicle steering angle was created from Table 3.7 with respect to the control voltage in the range of 1.5-5V shown as in Figure 3.21. However, the mobile robot steering system cannot move at the voltages below 1.5V. The relation between the angular speed and the control voltage is specified for this speed change as follows.

$$\dot{\delta}_c(u_2) = \begin{cases} 0.0968u_2 - 0.0842 & \text{if } u_2 > 1.5\ [Volt] \\ 0 & \text{if } u_2 < 1.5\ [Volt] \end{cases} \tag{3.18}$$

$\dot{\delta}_c(u_2)$ [rad/s]: Angular speed steering angle of mobile robot.

(u_2) [rad/s]: Applied control voltage from Arduino.

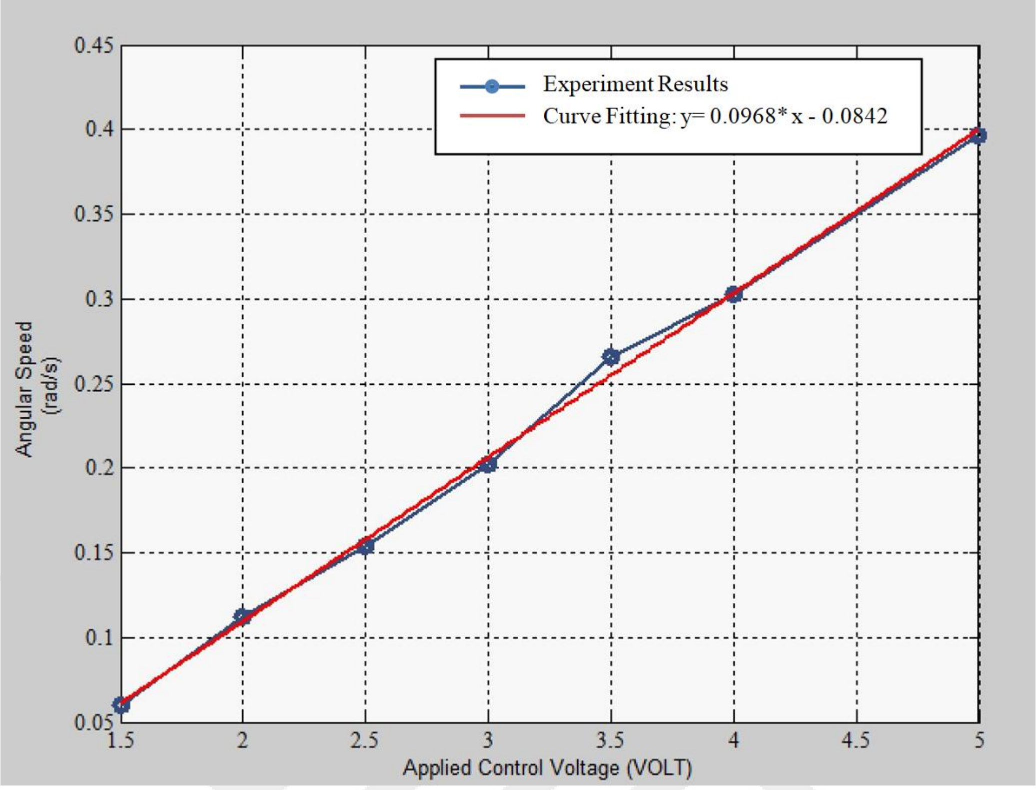


Figure 3.21. Applied control voltage and angular speed steering angle of the vehicle.

3-Encoder data

To measure the steering angle of the vehicle, an encoder was installed on the output shaft of the DC-Motor. It is necessary to obtain the conversion coefficient between the steering angle and the pulse number of the encoder since the steering system consists of some reduction and conversion elements. In the tests, from -31^0 to 31^0, a total of 62^0 rotations occur at 3060 pulses of the encoder. The encoder used in the setup is a quadrature incremental encoder and has 400 pulses in each full rotation. The coefficient for the conversion can be calculated by the test results.

$$\theta_{Steer} = K\, N_E \tag{3.19}$$

Where

θ_{Steer} : Steering angle (degree)

N_E : Pulse number from the encoder

$K = 0.020261 \left[\frac{\text{degree}}{\text{pulse}}\right]$: The conversion coefficient

3.2.4. Google Earth mapping system and coordinate conversion

Google Earth is a computer program that gives the user the maps of the Earth by composing the GIS data, aerial photography, and satellite images onto a 3D globe. The Google Earth program let the users search and sees on any point (cities or landscapes) on the surface of the Earth in diverse angle by writing the name or coordinate of the place by the keyboard or searching by the mouse directly. The Coordinates can be captured on the surface of Earth from the Google Earth program by:

- Zoom into the area you want coordinates for.
- Click add place mark tool.
- Move thumbtack (click left mouse button down and drag) to place you want coordinates for.
- Copy latitude/longitude (lat/long) values from new place mark window.

The location of the mobile robot can be defined by using a GPS receiver. However, a GPS gives us only the latitude and longitude angles with the height and receiver clock. Latitude and longitude are angles to identify uniquely the position of a geographic place on the Earth. Latitude is an angle defined with respect to the equatorial circle and can be in the range of +90 degrees (or 90 degrees north) and -90 degrees (or 90 degrees south). Longitude is an angle varying from the prime meridian (0^0) at Greenwich to 1800 East and West. The position of GPS receiver should be given with a coordinate system rotated with the Earth. These coordinates are not Cartesian and a geographical coordinates system. For this aim, one of the more convenient coordinate systems is Earth-centered Earth-fixed (ECEF). World Geodetic System (WGS84) is used by The Global Positioning System as a reference coordinate system. Since geographic coordinates do not specify a plane, projections must be applied so that they can be plotted as a map on a two-dimensional surface. Therefore, the position of the GPS defining the unmanned ground vehicle on an X-Y World plane can be calculated by converting the WGS84 coordinates to the ECEF coordinate system. The formulations for this conversion have been adapted from (Kaplan and Hegarty, 2005).

$$\begin{bmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \end{bmatrix} = \begin{bmatrix} \left(\frac{a}{R}+h\right)\cos\phi\cos\lambda \\ \left(\frac{a}{R}+h\right)\cos\phi\sin\lambda \\ \left(\frac{a\left(1-e^{2}\right)}{R}+h\right)\sin\phi \end{bmatrix} \tag{3.20}$$

Where

$R=\sqrt{1-e^{2}\sin^{2}\phi}$

a, h and e^2 are the semi-major axis of the satellite orbit, the ellipsoid height of the ground and the numerical eccentricity of the Earth respectively. a and e^2 are constant while h depends on the location measured.

$a = 6378137.0\ \mathrm{m}$

$e^2 = 0.00669438000426082 7$

ϕ : The latitude angle

λ : The longitude angle

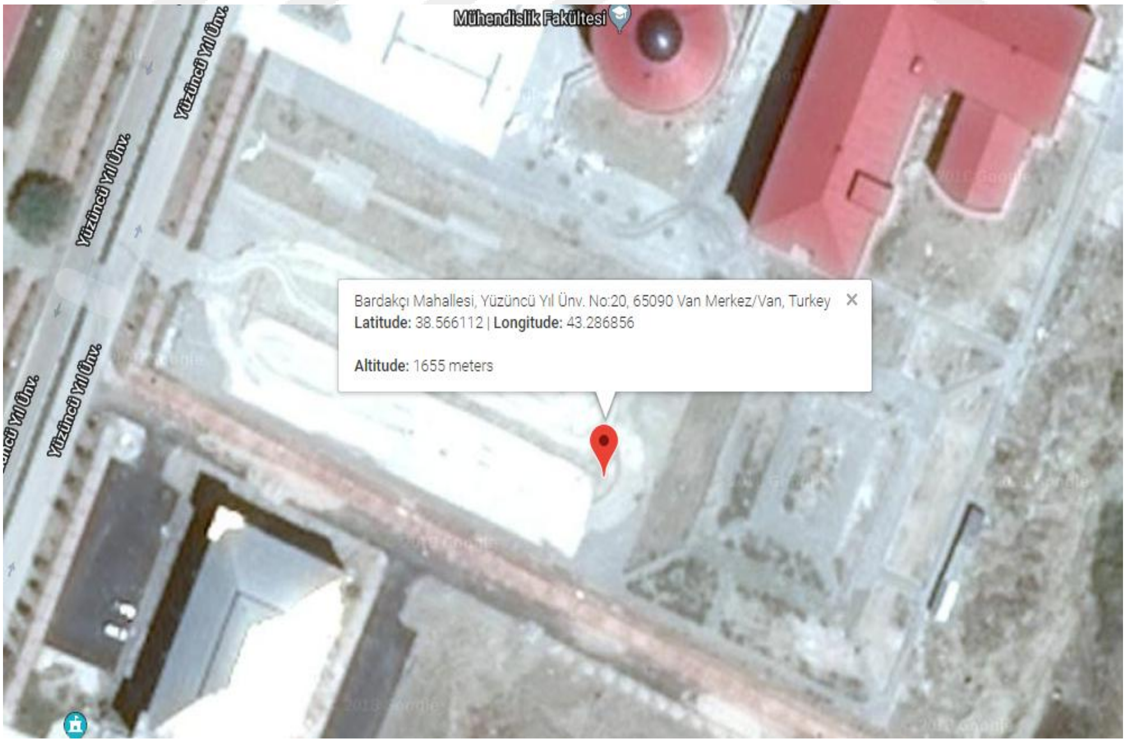
For the trajectory or path planning, the control points for the geometric or spline curves were selected by clicking on the related points on the Google Earth map. These give us the latitude and longitude of these points. These angles are the format of [dd.ddddd dd.ddddd] or [degree minute second = dd mm ss.sss]. For example, the coordinates of lat/long =[38.566112, 43.286856] or [38°33'58.003N, 43°17'12.681E] with the elevation of 1655 m are seen in Figure 3.22. According to this coordinate, X and Y coordinates with respect to the ECEF coordinate system can be found as a numerical data.

$\mathrm{X} = 3635858.93[\mathrm{m}]$ And $\mathrm{X} = 3424685.68\ [\mathrm{m}]$

The GPS connected with the computer directly through a cable, and the data was directly taken to the MATLAB programming in the form of NMEA file. NMEA file consists of many different massages that give us the data that we need in different types of message. To understand the NMEA message structure, in the below we will examine the common $GPGGA message. This private message was output from a GPS receiver (Betke, 2001):-

$GPGGA,235317.000,4003.9039,N,10512.5793,W,1,08,1.6,1577.9,M,-20.7,M,00,,*5F

- All NMEA messages start with the $ character, and each data field is separated by a comma.
- GP represent that it is a GPS position (GL would denote GLONASS).
- Time: 235317.000 is 23:53 and 17.000 seconds (UTC time).
- Latitude: 4003.9039, N is latitude in the (DDMM.MMMMM) format, north.
- Longitude: 10512.5793, W is longitude in the (DDDMM.MMMMM) format, west.
- Data is from a GPS fix: 1 (Fix Quality: 0 = Invalid, 1 = GPS fix , 2 = DGPS fix).
- Number of satellites seen: 08
- 1.6: denotes the HDOP (horizontal dilution of precision).
- Altitude: 1577.9 meters (meters above mean sea level).
- Height of geoids above WGS84 ellipsoid: -20.7Meter.
- Time since last DGPS update: 0 blank (No last update).
- DGPS reference station id: 0 blank (No station id).
- Checksum: *5F (Used by program to check for transmission errors).



Bardakçı Mahallesi, Yüzüncü Yıl Ünv. No:20, 65090 Van Merkez/Van, Turkey

Latitude: 38.566112 | **Longitude:** 43.286856

Altitude: 1655 meters

Figure 3.22. An application of GPS coordinates.

3.2.5. IMU/GPS integration

This section is related to the application of the unmanned vehicle in real time. So it's considered as outside the scope of this research. This research involves only a theoretical study based on the real equations obtained from the actual experiments on the vehicle. However, in this section, we will give some explanations and basic rules about the GPS/IMU integration studied later in the real time application of the mobile robot as a future work.

Navigation and control of the vehicle is an important subject and widely utilized in civil and military applications. The GPS is considered as one of the popular navigation systems which provide accurate information in ideal environments about the speed, position, and time. GPS accuracy may reduce in urban environments because of high buildings or in forests due to tree leaves or because of weather conditions, where GPS signals can be partially or completely interrupted. IMU is a sensor calculates angular rates and specific force with respect to an inertial frame to provide position, velocity, and attitude. IMU is also a stand-alone system; it is an independent system and operates continuously in any environments conditions. The precision of an IMU is reduced across the time by the accumulation of system errors. Therefore, the integration of the GPS system with the IMU improves the quality and safety of each navigational system as they complement each other. The navigation system takes the data from IMU during the information interruption on the GPS and GPS will calibrate the error of the IMU when the signal returns and receive accurate data. In another meaning, both disadvantages and advantages of every individual system are analyzed.

The Kalman filter has an important role in the integrations of GPS/IMU because it works as a recursive estimator. Kalman filter can work and manipulate the linear model with the lowest variance, and depends on the current information value and its previous value in the past of the linear model. Kalman filter integrates information (navigation data) coming from both the IMU and GPS systems, calculates the value of the errors, compensates them and returns it to the original value. However, the problem is that the navigation system is a nonlinear system and Kalman filter works with linear systems only. To solve this problem, you must use the Extended Kalman filter that works with non-linear systems (EKF: Converts nonlinear systems to linear systems using Taylor Series) (Pham et al., 2015). There are three types of GPS / IMU integration methods:

- Loosely coupled integration.
- Tightly coupled integration.
- Ultra tight coupled integration.

In real time application, we will use the loosely-coupled integration which integrates the velocity and position data coming from both GPS and IMU devices since this research uses a high-accurate GPS. The other advantage of the loosely-coupled integration is that it has redundancy behavior since both of GPS and IMU can provide their navigation data independently. Figure 3.23 shows the block diagram of INS/GPS loosely coupled integration.

The state system of the Kalman filter can be represented by the following equation (Lauszus, 2012):

$X_k = F.X_{k-1} + B.u_k + w_k$

Where:

X_k= the state system at k time (Required values)
X_{k-1}: the previous state system at k-1 time (Required values)
F= state transition model
B= control input model
w_k= process noise vector
$E(w_k . {w_k}^T) = Q_k$
Q_k= process noise covariance matrix
$z_k = H_{X_k} + v_k$
$z_{k=}$ observation or measurement process.
H=observation model (used to map the true state space into the observed space)
v_k= The noise of the measurement
$R = E.(v_k . {v_k}^T) = var(v_k)$
$\hat{\chi}_{k|k-1} = F.\hat{\chi}_{k|k-1} + B.\dot{\theta}_k$ (Prediction equation)
$P_{k|k-1} = F.P_{k|k-1}.F^T + Q_k$ (Priori error covariance prediction)
$\hat{Y}_k = z_k - H.\hat{\chi}_{k|k-1}$ (Innovation Matrix: Update equation)
$K_k = P_{k|k-1}.H^T.{S_k}^{-1}$ (Kalman gain)

$S_k = H \, . \, P_{k|k-1} \, . H^T + R$ (Innovation covariance)

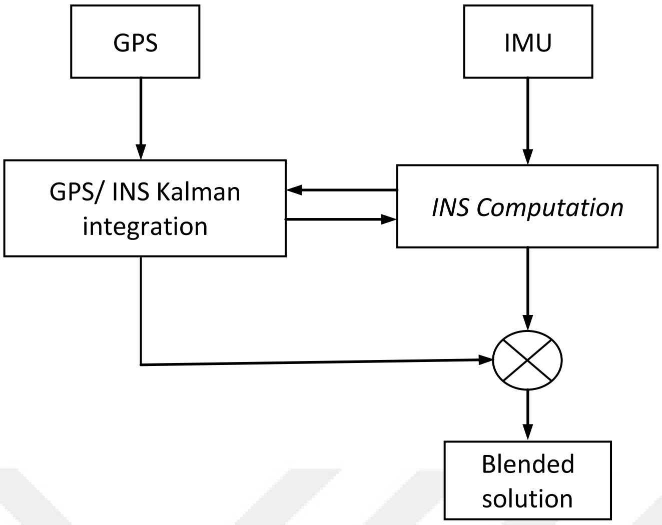


Figure 3.23 Block diagram of GPS/INS integration (Loosely coupled).

3.2.6. Computer program

The computer is connected to the microcontroller (Arduino) as well as equipped with GPS by a USB which is a serial communication interface. The computer has an important role in the operations and can be summarized by the following points:

- All communications among the sensors, actuators and control software are executed on the computer.
- The computer is used to calculate all kinematic equations and the values of controller coefficients, by using MATLAB programs.
- Draw and define the predefined trajectories using one of the specialized programs in this field (Google Earth).
- Calculate the actual speed of the vehicle through the information taken from the GPS/IMU system about the current position of the vehicle and comparing it with the corresponding point on the trajectory. The speed of the vehicle depends on the input volts (u_1) that provided by the microcontroller (Arduino). The value of the volts provided by Arduino is between 0 volts (no movement) and 5 volts (maximum speed). The amount of the volts depends on the current condition of the vehicle and the rate of errors.

- The computer is used in the control process by the control algorithm written by MATLAB program. All manipulations to control the system are implemented via a program written with MATLAB software. The control process depends on comparing between the points taken from the GPS with the pre-defined trajectory or path for the purpose of finding the errors. Then, this determines the control input of the vehicle.
- All results obtained, such as the steering angle, the speed of the vehicle, vehicle's positions, errors, and others are recorded by the computer and they are displayed in multiple sketches by MATLAB program.

In the experiment A notebook with the Intel (R) core (TM) i5-4300 CPU @1.90 GHZ 2.5 GHZ and 8.00GB RAM is used.

In this study, due to the requirement of the control process of the vehicle to track the pre-defined trajectory or path, the flow chart of control algorithm is depicted in Figure 3.24. The steps of the system's work can be summarized as follows:

- Firstly, for the aim, defining the control point of the vehicle on the Google Earth map
- Defining the trajectory or path from the control part by using Spline or any geometric formulation.
- Defining the real position of the vehicle by reading the data from the sensors GPS, IMU, Magnetometer and Encoder (Encoder reads the steering angle value which rate between -31^o to 31^o). Then integrate the GPS/IMU data and with encoder data, thus defining the position and the steering angle of the vehicle.
- Measured data from the sensors (GPS, IMU, Magnetometer and Encoder) will be compared with the reference data for the purpose of finding errors (errors in position and direction).
- The errors will be compensated with a new control system method (PD controller with some modification).
- The results of PD controller are named modified input (control input) which load in MATLAB program.
- The modified input (control input) will be sent to the microcontroller Arduino by USB cable.

- The microcontroller Arduino sent the modified input to the motor driver card (Since the microcontroller Arduino cannot provide the voltage and the current that require to operating the DC-Motors. A DC-Motor driving card was used for providing the voltages and the current which are necessary to run the front and rear DC-Motors).
- DC-Motor driving card sending the required voltage and current to run the DC-Motors to determine the linear speed and steering angle of the vehicle.

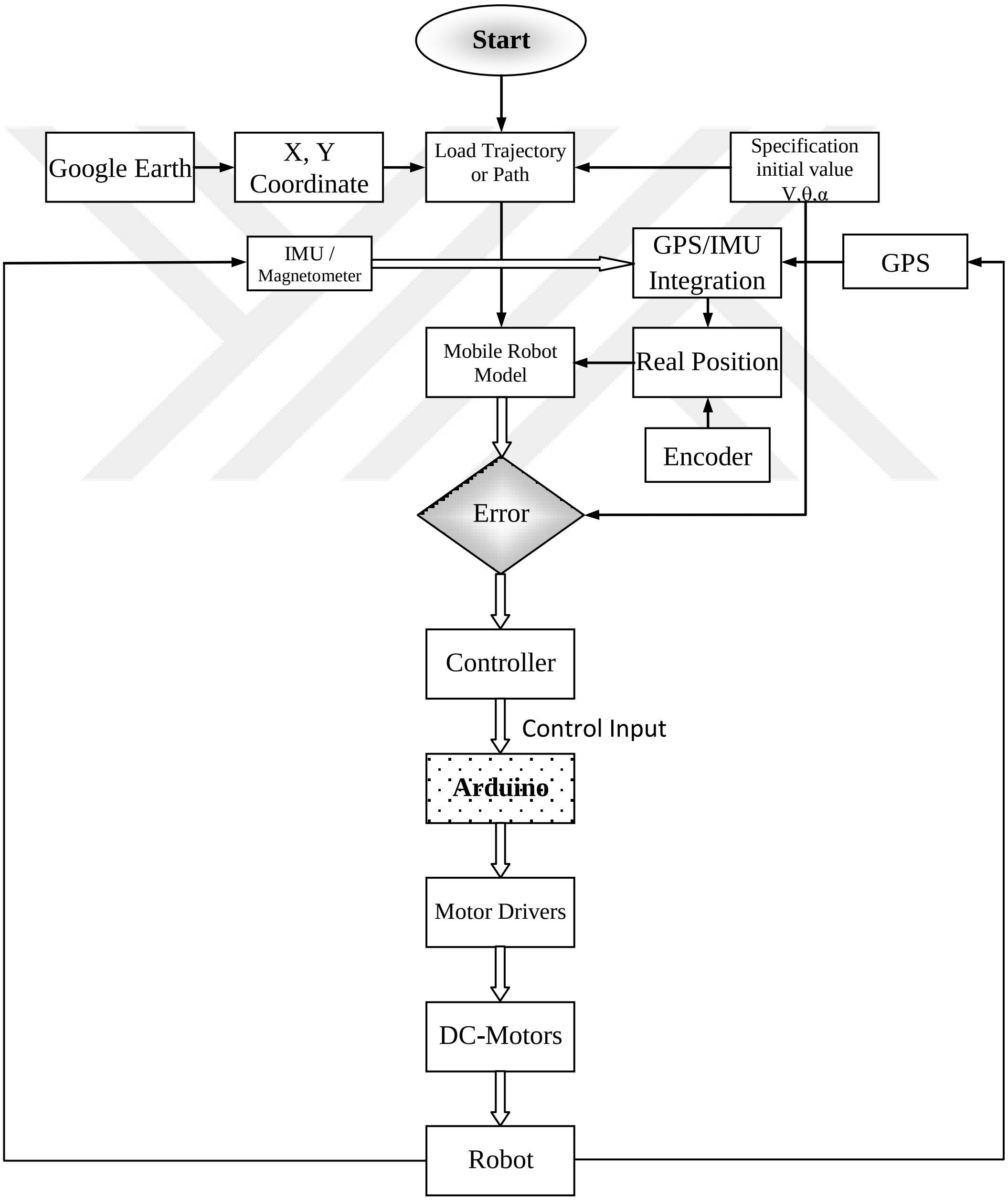


Figure 3.24.Experimental flow chart of the setup.

4. RESULTS

This thesis study involves two parts. The first part is a prototype design of an unmanned ground vehicle while the second one involves the kinematic modeling and control of the vehicle for the Trajectory-Tracking and Path-Following. In this study, the control of the vehicle was designed theoretically with both the classical PD and SMC. The real-time control of this mobile robot has been left to another future work.

In the prototype design, the mechanical, electronic devices and sensors were mounted successfully on the vehicle. Also, the communication among these devices through the computer has been accomplished synchronously. The required parameters to drive and control the mobile robot could be obtained by the real time tests.

The control study depending on the kinematic model of the unmanned ground vehicle was conducted under two titles for the purpose of displaying the behavior of the control system used in detail, which includes the Trajectory-Tracking and Path-Following given in the subsections.

4.1. The Results of the Trajectory-Tracking

In this section, the proposed control system is simulated on three different trajectories for the performance evaluation. In order to properly control tracking the trajectories by the autonomous vehicle (UGV), a control system including a PD controller has been proposed to reduce the rate of errors (distance and direction errors) and oscillations in steering angle or the vehicle velocity. To verify the efficiency of the proposed system, the results obtained from the simulations are compared with another control system (SMC). In Trajectory-Tracking, there are three types of errors (X_e, Y_e, θ_e) with their rates that must be controlled simultaneously, the Inputs for the system are the linear speed (v_u) and the steering angle of the vehicle (δ_c).

In the simulation procedure, at first, the reference trajectory is loaded; here, it is considered as an eight shape trajectory. This eight shape trajectory has been defined by the following equations:

$$R_8 = \begin{bmatrix} X_t \\ Y_t \end{bmatrix} = \begin{bmatrix} 30 * \sin(0.025 * t) \\ 30 * \sin(0.025 * t) * \cos(0.025 * t) \end{bmatrix}$$

The eight shape trajectory started from X = 0 and Y = 0. Then, all parameters related to the vehicle prototype and controllers are specified. The control parameters in the simulation for both PD and SMC controllers which were calculated by trial and error are shown in Table 4.1. After that, in the loop, the simulations have been performed by adjusting the control parameters for both methods. The results of the simulation are plotted in Figure's 4.1-4.4.

Table 4.1. Control parameters for eight-shape

PD control constant		SMC control constant	
Linear speed compensate control parameters		Sliding surface control parameters	
Linear error proportional gain K_{xp}	0.5	k_0	0.02
Linear error derivative gain K_{xd}	0.05	k_1	1.75
Input voltage (u2) compensate control parameters		k_2	1.75
Orientation error proportional gain K_{tp}	95	Constant rate reaching control parameters	
Orientation error derivative gain K_{td}	35	p_1	0.5
Lateral error proportional gain K_{yp}	95	p_2	0.5
Lateral error derivative gain K_{yd}	25	q_1	0.5
		q_2	0.5

The Figure 4.1 shows that the unmanned ground vehicle can track the desired trajectory correctly in both of the PD control system proposed in this research and in the SMC system.

The eight shape trajectory is obtained with non-constant reference velocity. The reference velocity is ranging between 0.4961-1.0607 m/s. Figure 4.2 shows that the linear velocity follows the reference velocity with the desired trajectory, in both PD and SMC controller.

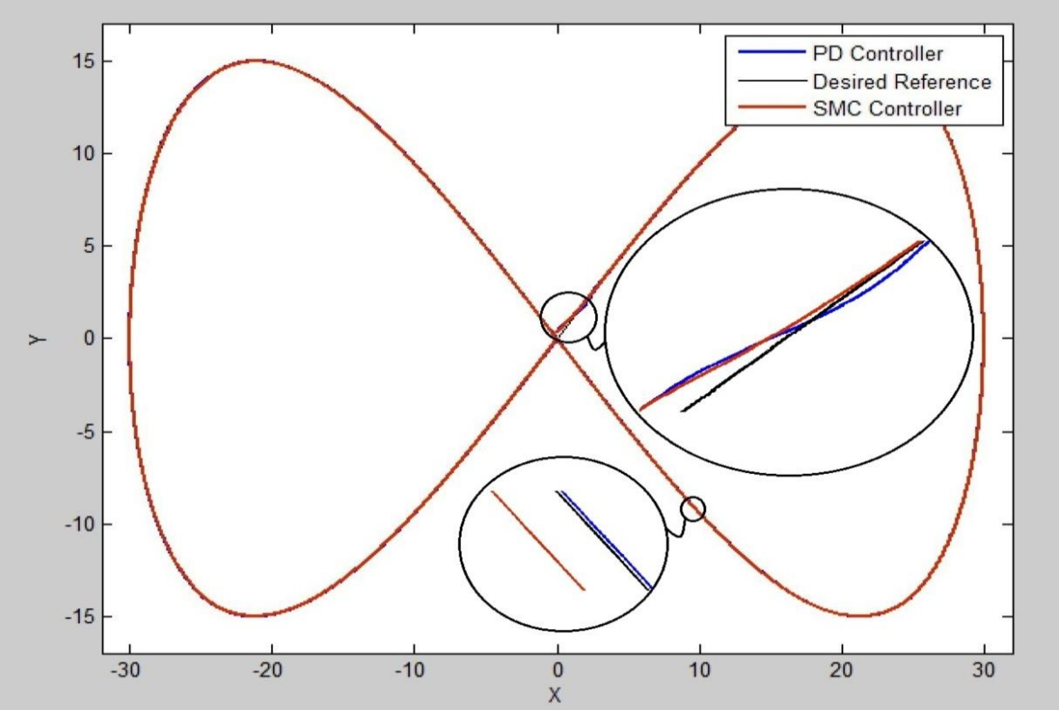


Figure 4.1. The Trajectory-Tracking for the eight-shape trajectory in X-Y coordinates (ECEF coordinate system).

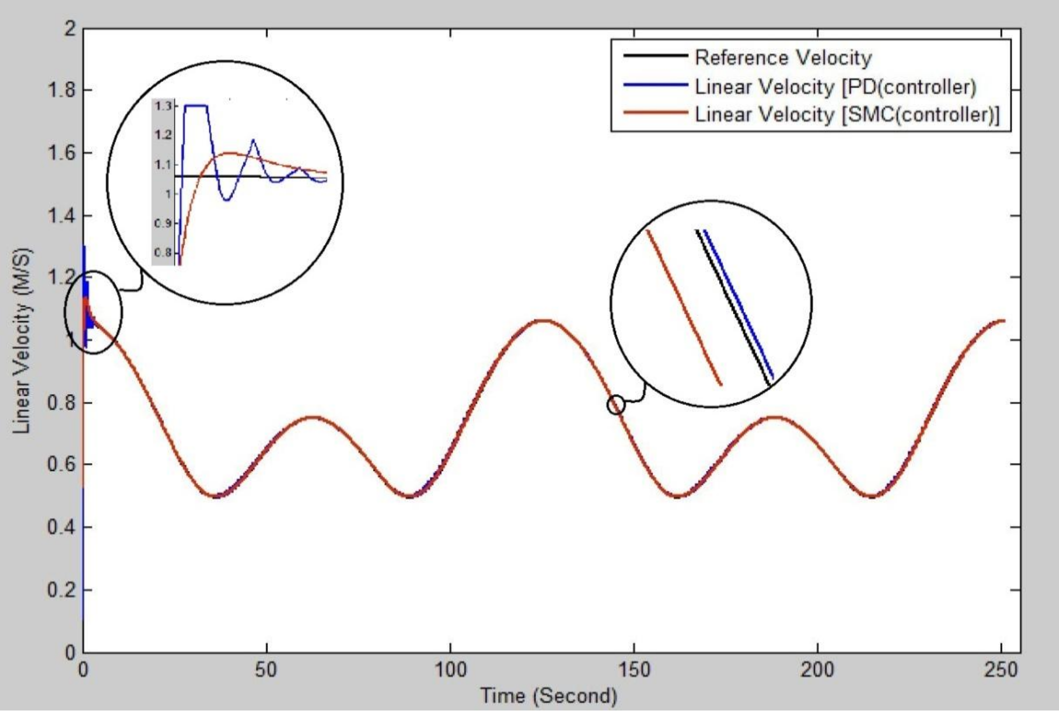


Figure 4.2. The linear velocity behaviors for the Trajectory-Tracking on the eight-shape with PD&SMC controllers.

From the figures, we note that firstly there is some oscillation in the linear velocity of the vehicle until they reach the designed velocity of the trajectory. This stems from the different initial linear velocities. Then, they can follow the designed speed correctly with very small oscillations, which the system cannot sense it. Notwithstanding, the results indicated with the real and desired speeds are highly compatible with each other in both control systems.

The position errors for both types of the controller can converge to near-zero as shown in Figure 4.3. (The position error (tracking error) it means the distance between the control point on the vehicle and the comparison point on the reference trajectory (Figure 2.6). From the curve we note that after the arrival of the vehicle to the defined trajectory the position error is very small where in some areas the position error of the SMC controller larger than the PD controller and in some areas position error of PD controller larger than the SMC controller and both of the result are considered within the acceptable result, where the biggest position error of PD controller is equaled to 0.03 m and for the SMC controller is equal to 0.02 m.

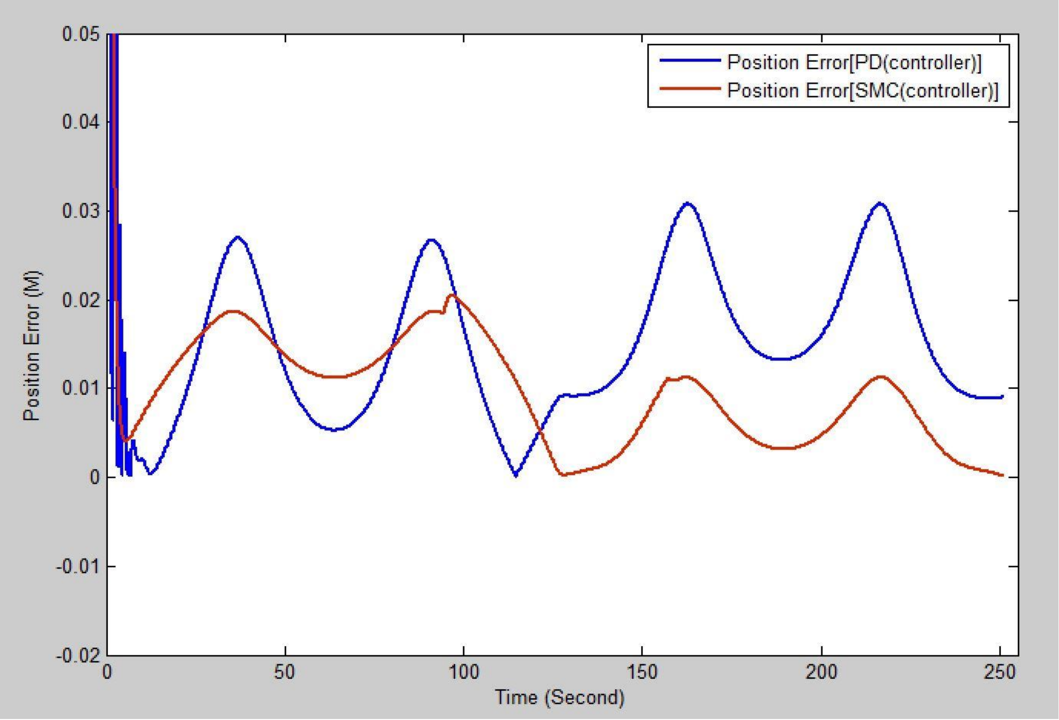


Figure 4.3. The absolute position errors for the Trajectory-Tracking with eight-shape for PD&SMC.

In control operation on the vehicle, the most important variable to be controlled is the steering angle of the robot. The control depends on a great extent on this control variable. Figure 4.4 shows the behavior of the steering angle (δ_c) for both controller types. The vehicle is designed to have the highest value of the steering angle equaling to ±31^{o}degrees. This was taken into consideration when designing the control systems. We note from the figure that the maximum value of the steering angle (δ_c) of the vehicle for both types of the controller that make the vehicle trace the eight shape trajectory correctly is equal to approximately ±10.5^o degrees.

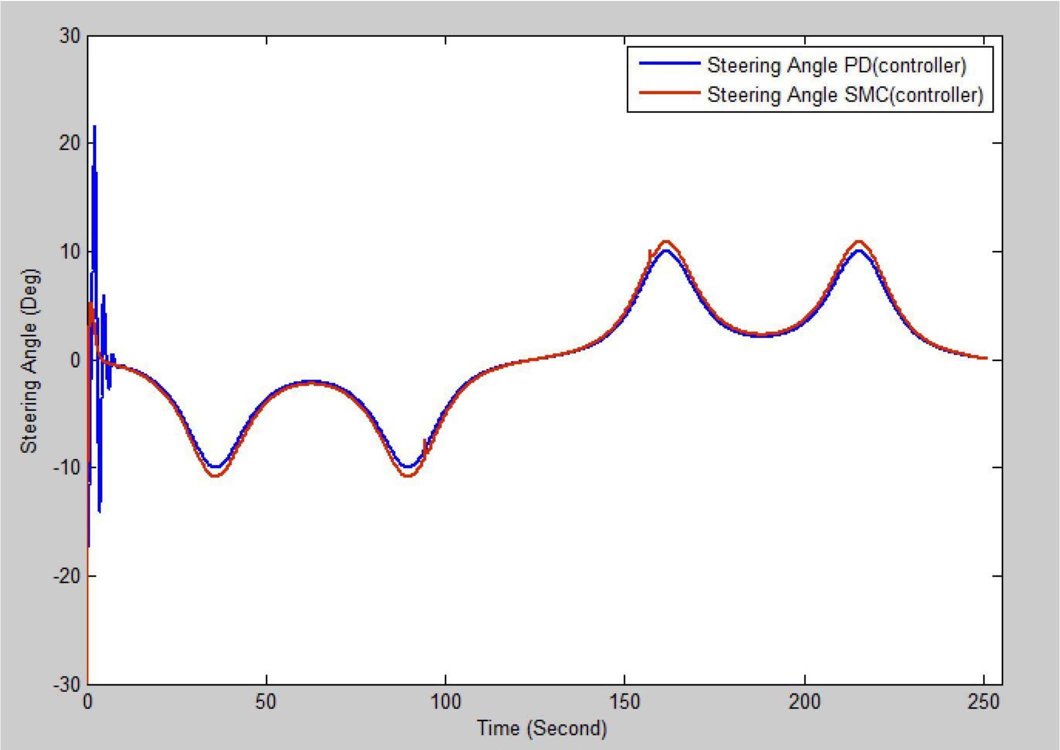


Figure 4.4. Steering angle (δ_c) for Trajectory-Tracking (8-Shape) (PD&SMC Controller).

The second trajectory was considered as straight lines with some sharp turnings. This trajectory has been defined by the following equations:

$$R_{<} = \begin{bmatrix} X_t \\ Y_t \end{bmatrix} = \begin{bmatrix} a_x * t + b_x \\ a_y * t + b_y \end{bmatrix}$$

Where:

$$\mathrm{V_{Ref}} = \frac{X}{t} \quad \Longrightarrow \quad t = \frac{X}{\mathrm{V_{Ref}}\, t}$$

$\mathrm{V_{Ref}} = 1$ and 0.8 m/s.

X= length of the trajectory =1000 m.

Sharp turnings have angles assigned more than 90^o degrees to evaluate the performance of the controllers in unexpected cases. The control parameters used in PD control system and SMC control system are shown in Table 4.2.

Table 4.2. Control parameters for the second trajectory

PD control constant		SMC control constant	
Linear speed compensate control parameters		Sliding Surface control parameters	
Linear error proportional gain K_{xp}	1	k_0	0. 55
Linear error derivative gain K_{xd}	1	k_1	0.55
Input voltage (u2) compensate control parameters		k_2	0.55
Orientation error proportional gain K_{tp}	95	Constant rate reaching control parameters	
Orientation error derivative gain K_{td}	35	p_1	0.001
Lateral error proportional gain K_{yp}	95	p_2	0.001
Lateral error derivative gain K_{yd}	25	q_1	0.003
		q_2	0.003

Figure 4.5 shows that the vehicle can track the desired second trajectory accurately with PD control system proposed in this study.

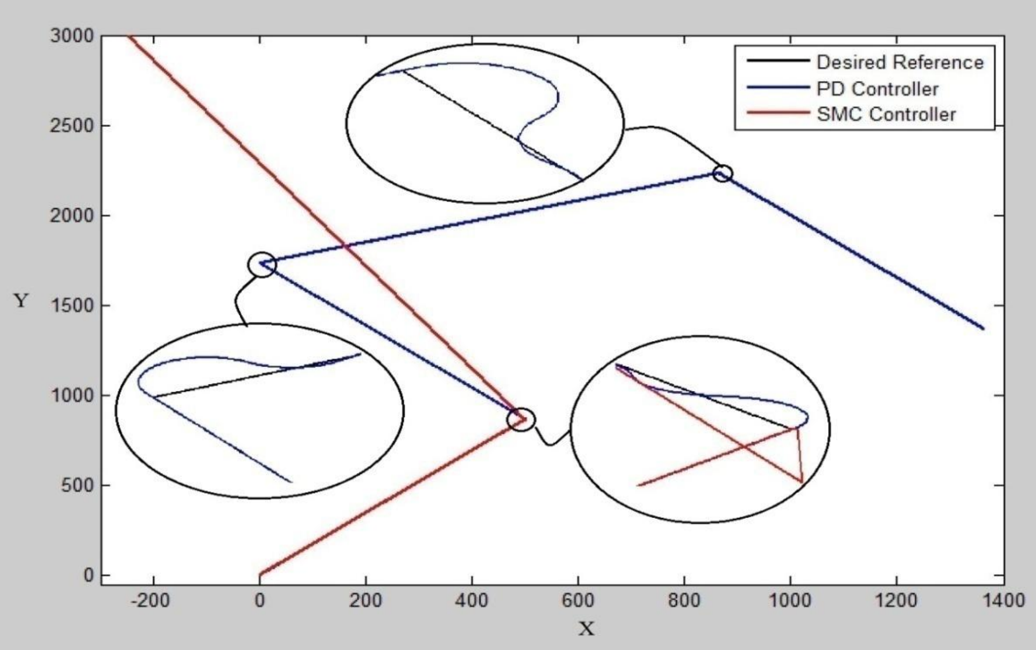


Figure 4.5. The Trajectory-Tracking for the straight lines with sharp turnings in X-Y coordinates (ECEF coordinate system).

The straight-line trajectory is obtained with non-constant reference speed, where initially the reference speed is equal to 1 m/s and after a while, the reference speed is decreased to 0.8 m/s. Figure 4.6 shows that the linear velocity of the reference and desired one corresponds to each other for the PD controller. As it has been mentioned earlier, the SMC method suggested in this study from the literature does not work on this type of trajectories. This method was originally designed for that the orientation error (θ_e) must be less than 90^o degrees. In the mean while, the absolute positional error can be seen in Figure 4.7. This error converges to exactly zero except for the corner points it reach to approximately 2.3 m. However, after each corner points the control method can supply the vehicle to converge to the desired road.

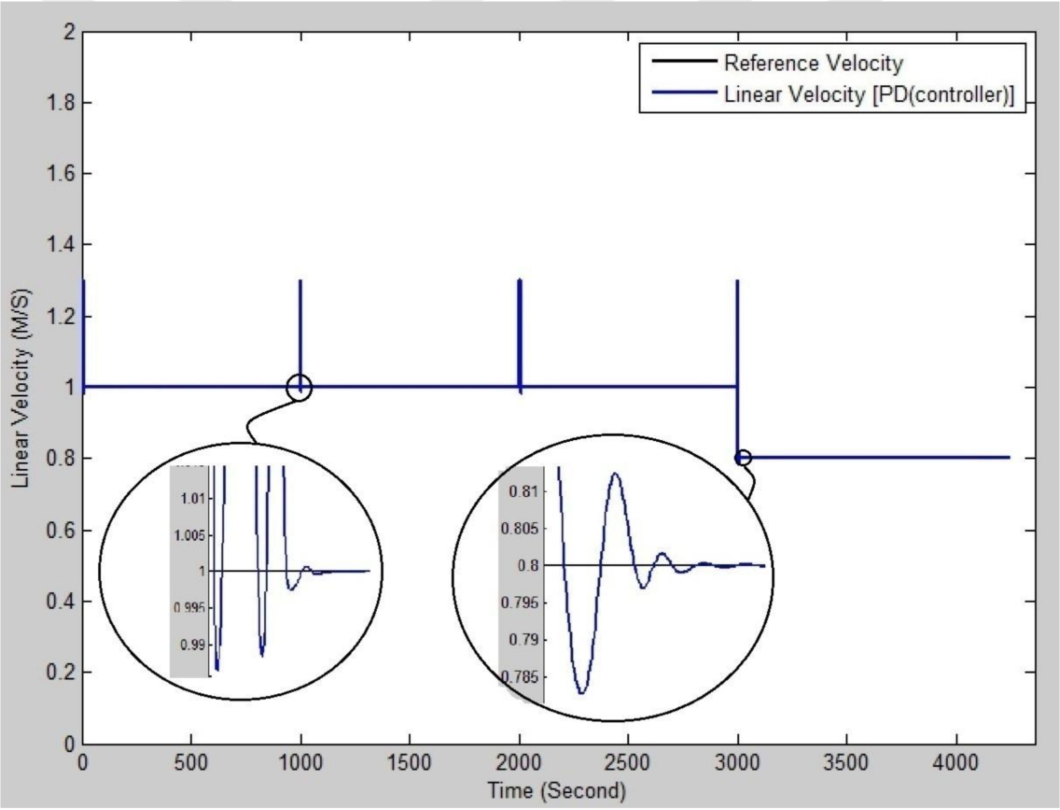


Figure 4.6. The Linear velocity behaviors for Trajectory-Tracking with the straight lines.

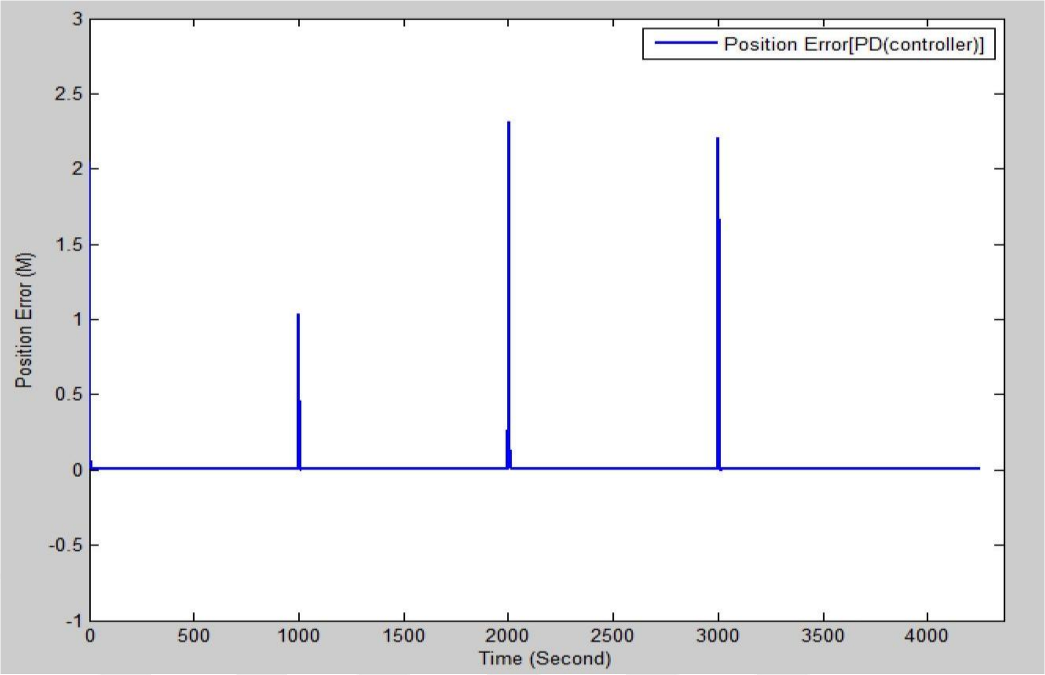


Figure 4.7. The absolute positional error for the Trajectory-Tracking with the straight-lines.

Figure 4.8 shows the behavior of the steering angle (δ_c) of the PD controller. It can be observed from the figure that the vehicle was able to track correctly the road within the specified limits of the steering angle (δ_c) of approximately 31^o. The SMC method cannot succeed in bringing the vehicle back to the reference road.

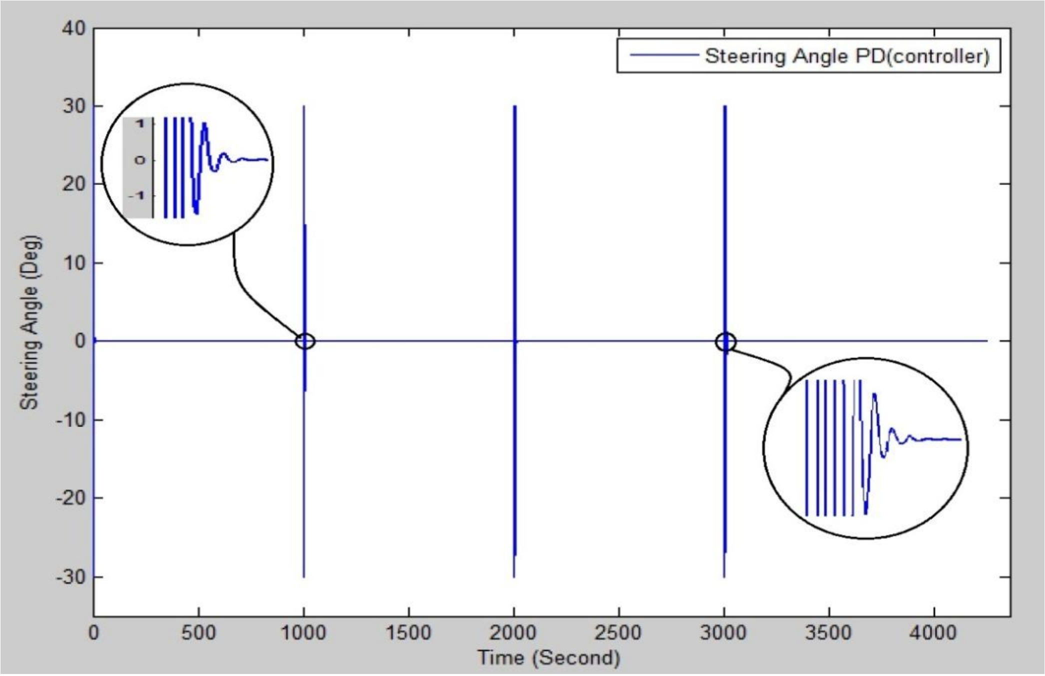


Figure 4.8. The Steering angles (δ_c) for the straight lines.

The final trajectory considered for testing the control system proposed is an irregular road shown in Figure 4.9. The control parameters used in both PD and SMC are given in Table 4.3. After defining the control points, the trajectory can be created by using Spline command in AutoCAD program. Spline draws a smooth curve called non-uniform rational B-splines that pass through or close to a group of fit points or that is defined by the vertices in a control frame then the trajectory is sent from the AutoCAD program to the MATLAB program. This gives us an irregular trajectory with X and Y coordinates. An irregular trajectory means that it does not comply with any mathematical equation or a known geometric curve. Because of this, in irregular trajectory, the rates of the reference can be calculated only by numerically. Then, the first and second rates, the orientation and radius of curvature related to this reference curve are obtained by using the "Eq's. 3.10-3.11".

Table 4.3. Control parameters for irregular road

PD control constant		SMC control constant	
Linear speed compensate control parameters		Sliding Surface control parameters	
Linear error proportional gain K_{xp}	1	k_0	0. 75
Linear error derivative gain K_{xd}	1	k_1	0.75
Input voltage (u2) compensate control parameters		k_2	0.75
Orientation error proportional gain K_{tp}	185	Constant rate reaching control parameters	
Orientation error derivative gain K_{td}	150	p_1	0.15
Lateral error proportional gain K_{yp}	95	p_2	0.15
Lateral error derivative gain K_{yd}	25	q_1	1
		q_2	1

Figure 4.9 show that the vehicle can correctly track the trajectory in both types of control systems (PD & SMC controller).

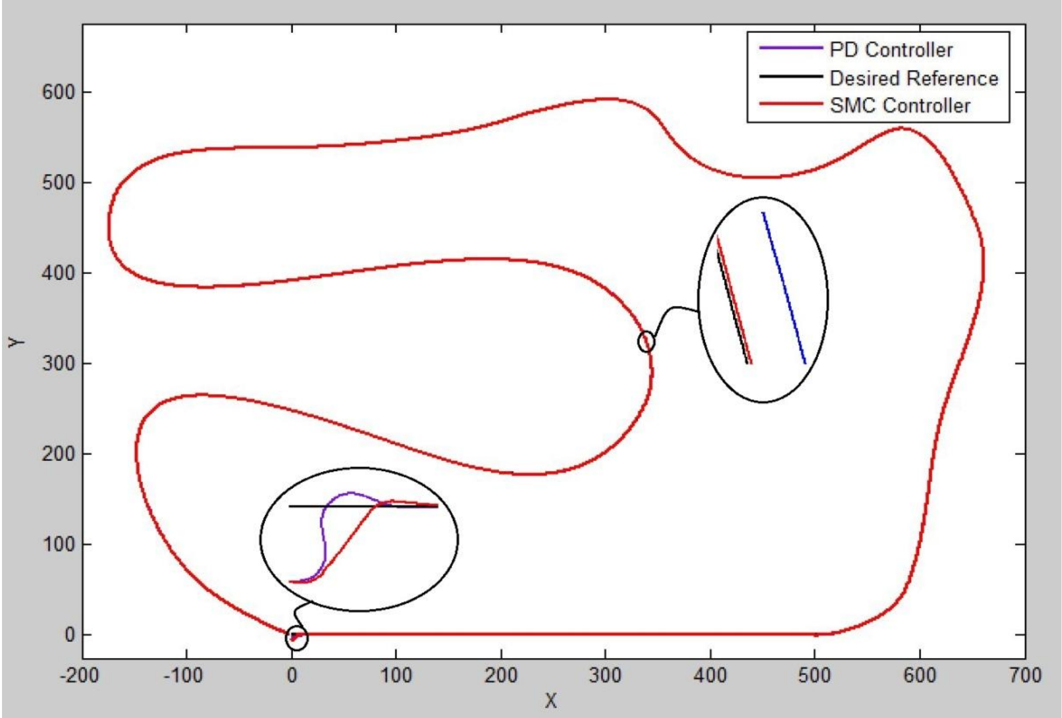


Figure 4.9. The Trajectory-Tracking for the irregular road in X-Y coordinates (ECEF coordinate system).

Figure 4.10 shows that the linear speed (v_u) of the vehicle can trace the reference speed of the desired trajectory. There are some oscillations in the linear speed of the vehicle until it reaches the desired velocity. This stems from the error between the reference speed and the linear velocity of the vehicle because of the initial linear velocity of the vehicle is different from the desired velocity. Nonetheless, both methods continue to track the designed speed correctly with a small oscillation, which the system cannot sense it.

The position error for both types of the controller is almost converging to zero as shown in Figure 4.11. In Figure 4.11, it is noted the PD controller has given a good results with compare to the SMC control system but both results are considered acceptable. Sometimes, the errors increase at some points because the vehicle start to change its direction or the radius of curvature of the reference curve becomes smaller.

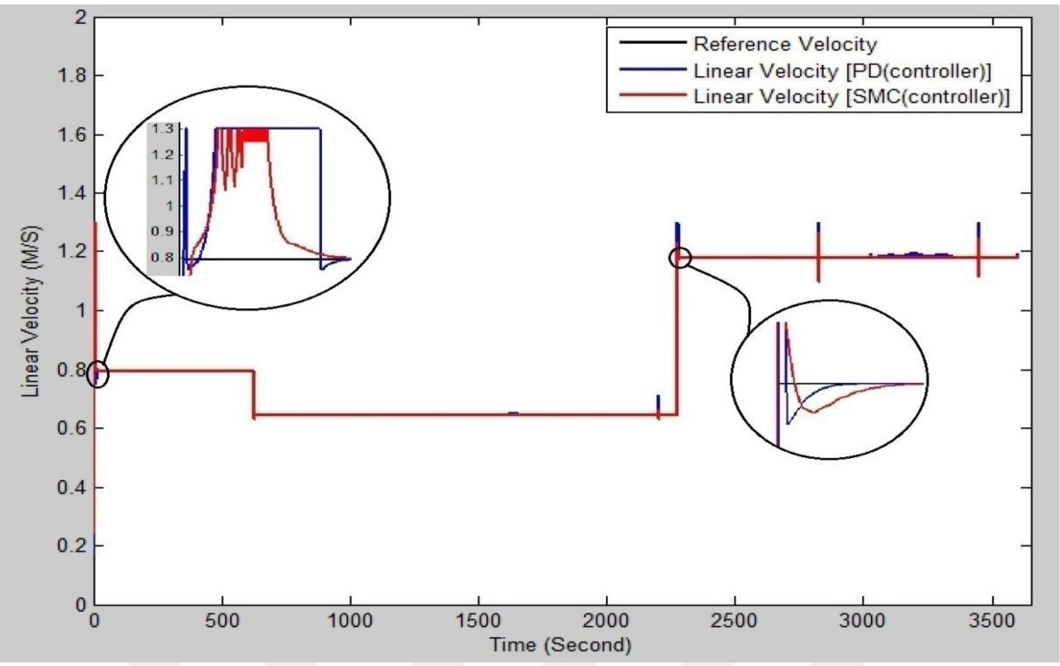


Figure 4.10 The linear velocity behaviors for the tracking of irregular road with PD&SMC controller.

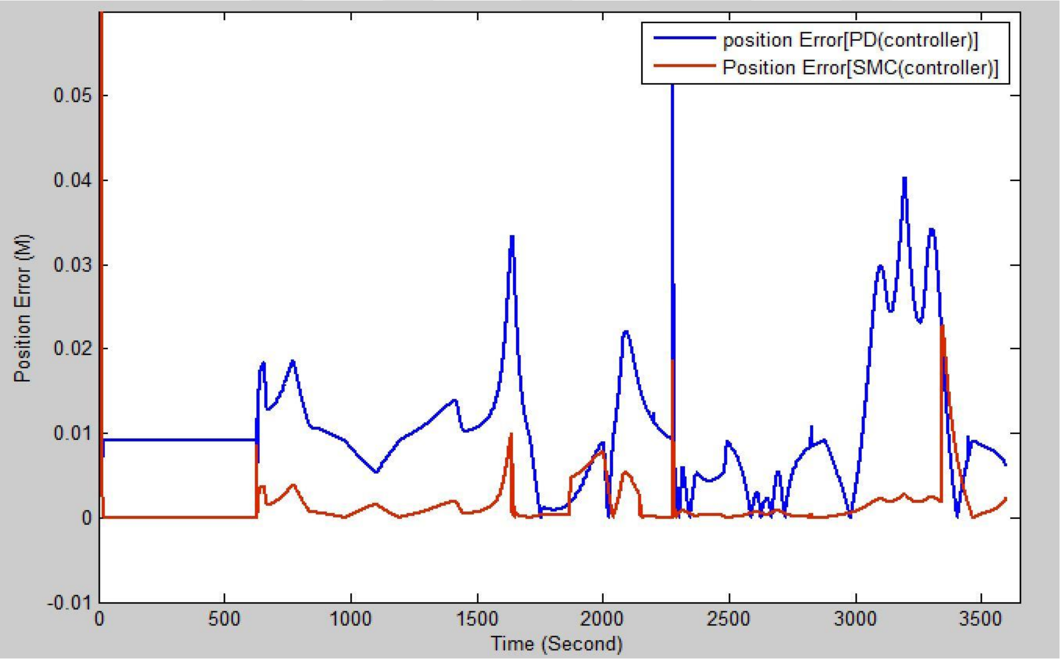


Figure 4.11. The absolute position errors for the tracking of irregular road with PD & SMC controller.

Figure 4.12 indicates the behavior of the steering angle (δ_c) for both controllers. The vehicle must be designed to have the highest value of the steering angle equal to 31^o. This was also taken into consideration when designing the control systems. We note from Figure 4.12 that the proposed system has given a better result compared to the SMC method. The proposed system has fewer oscillations while the steering angle with the SMC reveals swinging behavior. The steering angle is a mechanical feature, not a voltage or current. Therefore, this is not an acceptable situation and the vehicle cannot accomplish like a steering profile. However, the steering angle from the PD controller proposed in this study gives very smooth motion to the vehicle and also the angles remains in the limits of ±31^{o}.

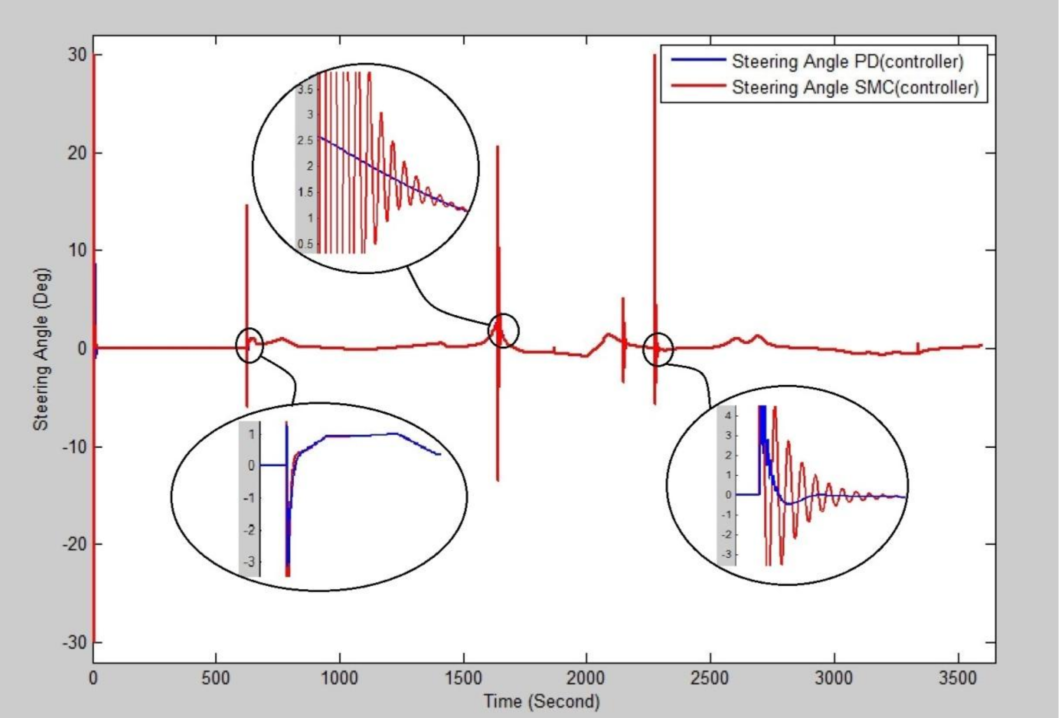


Figure 4.12. The steering angle (δ_c) for the tracking of the irregular road with PD&SMC controller.

4.2. The Results of the Path-Following

In this section, the proposed control system is simulated on one path for the performance evaluation. In order to properly control tracking the path by the autonomous vehicle (UGV). The path is given in a time-free parameterization and pre-defined by the trajectory planner whereas trajectory is time-dependent, i.e., in Trajectory-Tracking, you have to be in the certain location at a specified time. In the Path-Following process, the mobile robot moves with either a constant speed or variable speed depending on the planner. The control systems PD controller has been proposed to reduce rate of errors (distance and direction errors) and oscillations in steering angle or the vehicle velocity. In Path-Following, there are two types of errors (Y_e, θ_e) that must be controlled simultaneously. The control input for the system is the steering angle of the vehicle (δ_c). In Path-Following, the linear speed of the vehicle is not taken into consideration to compensate the errors. The vehicle can travel at constant speed. However, to complete the path as soon as possible, this speed can be increased or decreased with respect to the radius of curvature which the vehicle tries to follow. In the control algorithm for the Path-Following, the comparison process between the current vehicle locations and all points of the path do not depend on a condition, but always the comparing process is done with the closest point on the path to the vehicle.

In the simulation procedure for the Path-Following control, at first, the reference path is loaded; here, it is considered as an irregular road. The path can be created by using Spline command in AutoCAD program. The path is sent from the AutoCAD program to the MATLAB program. The control parameters used in PD control system are shown in Table 4.4.

Table 4.4. Control parameters for Path-Following control

PD control constant	
Input voltage (u2) compensate control parameters	
Orientation error proportional gain K_{tp}	1.5
Orientation error derivative gain K_{td}	0.001
Lateral error proportional gain K_{yp}	0.01
Lateral error derivative gain K_{yd}	0.25

Figure 4.13 shows that the vehicle can track the desired path accurately with PD control system proposed in this study.

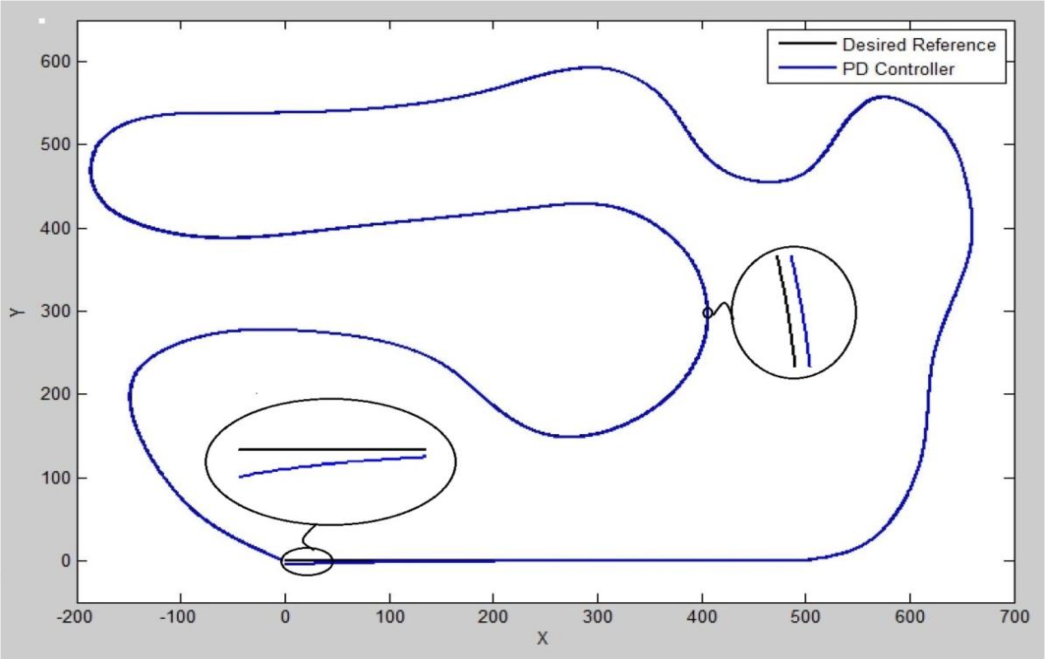


Figure 4.13. The Path-Following for the irregular road in X-Y coordinates (ECEF coordinate system).

In the Path-Following the vehicle uses a constant speed, or the speed can be regulated with respect to the radius of curvature of the path. In the simulation and with no any regulation to the speed of the vehicle the velocity was equal to 1.037 m/s and for the regulated velocity test the velocity is of the vehicle was equal to 1.31 m/s and due to the small value of radius of curvature the value of velocity is decreased to approximately 0.5 m/s and then increases to its desired value when the value of radius of curvature is increased. The reference velocity is equal to (1.202 m/s). Figure 4.14. Shows that in both states, the vehicle has the same behavior for tracking the path correctly. The deferment will be at the time of traveling. When using the constant linear speed the vehicle needs a longer time (3636 seconds) to complete the path compared to the traveling time with variable speed (2886 seconds).

The position error for the Path-Following control is almost converging to zeroin the straight lines as shown in Figure 4.15. In Figure 4.15, it is noted that the PD controller has given a good results. In the first, there is a high value of position errors because of the initial state of the vehicle after that the position error is decreased. There are also some errors in sharp turns up to approximately 1 m. But then we notice that the vehicle after a while corrects its course successfully and this leads to reducing in the position error.

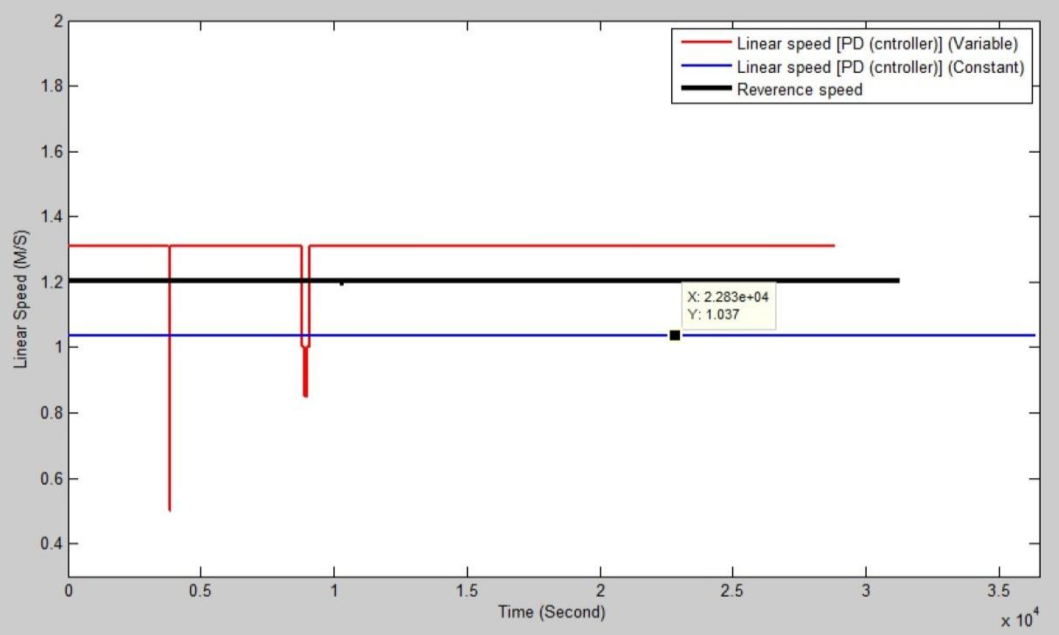


Figure 4.14. The linear velocity behaviors for the tracking of irregular road path with PD controller.

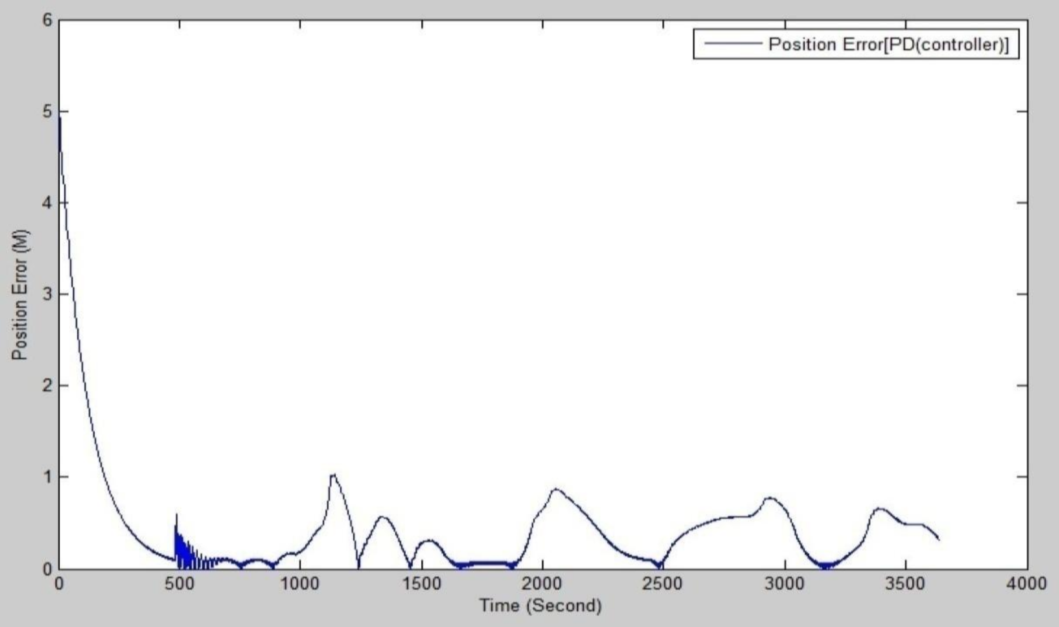


Figure 4.15. The absolute position errors for the tracking of irregular road path with PD controller.

The most important variable to be controlled is the steering angle of the robot. The control depends in great extent on this control variable. Figure 4.16 shows the behavior of the steering angle (δ_c) for in the Path-Following with PD controller. The vehicle is designed to have the highest value of the steering angle equaling to $\pm 31^o$

degrees. This was taken into consideration when designing the control systems. We note from the figure that the maximum value of the steering angle (δ_c) of the vehicle that make the vehicle trace irregular road path with PD controller correctly is equal to approximately $\pm 11.8^o$ degrees. We note that there are some oscillations in some areas. This is due to the nature of the irregular road path. These oscillations are acceptable because the vehicle can be carried out it due to the time, since the time between the one behavior and others in the oscillations area are equal approximately 8 seconds for 6 degrees of steering angle and this is an acceptable time for our vehicle steering angle system, and the vehicle can do it and follow the path correctly.

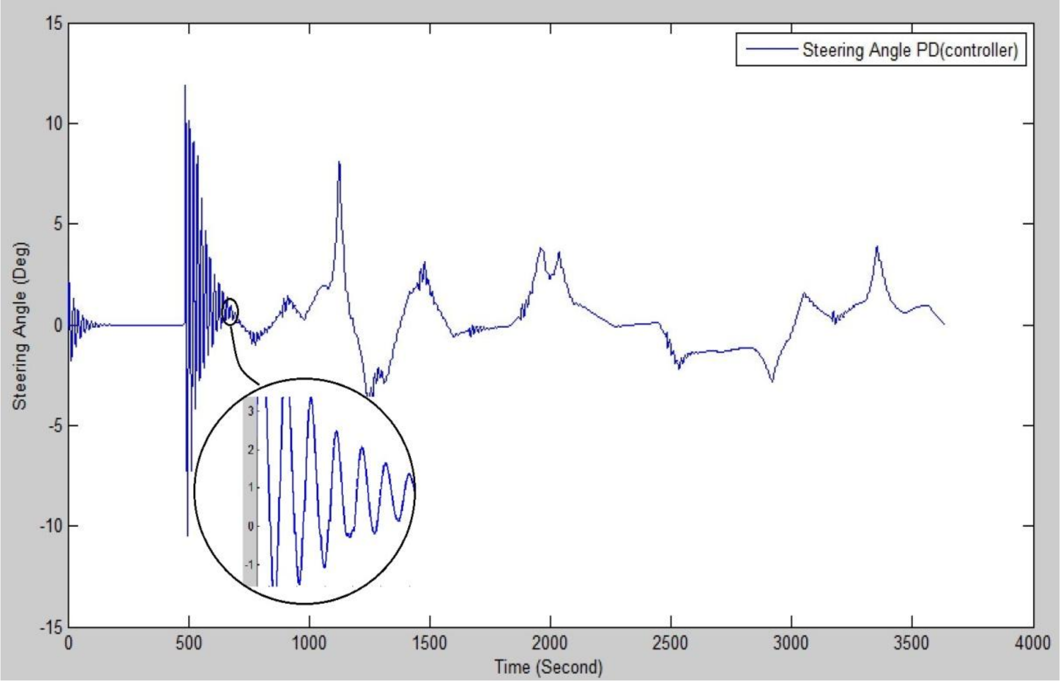


Figure 4.16. The steering angle (δ_c) for the tracking of the irregular road path with PD controller.

5. DISCUSSION and CONCLUSION

This study consists of two parts; the first one is a prototype design of an unmanned ground vehicle and the second is the control of this vehicle for the Trajectory-Tracking and Path-Following.

The mobile robot was at first designed with a CAD program and manufactured according to this designing. The vehicle involves a GPS (Global Positioning System), an IMU (Inertial Measurement Unit), and an Encoder, and magnetometer as sensors to find the current position and direction of the vehicle. In addition, the prototype consists of mechanical and electronic parts such as DC-Motors, DC-Motor driver's card, microcontroller and laptop (PC). In the prototype design, all parts worked in compatible each other and especially Arduino as a microcontroller could successfully fulfill its task which is supplying the communication between DC-Motor drivers and the computer, reading the data of the sensors and driving the DC-Motors.

In the second part, the unmanned ground vehicle was designed to be able to move on a particular trajectories (or paths) by controlling the position and speed of the vehicle with the acceptable positional errors. For this aim, two feedback controls method were used on the vehicle. The first one proposed by the author in this study is a classical PD control with some modification and the other for comparison with the proposed method is the SMC (sliding mode control) adapted from a study form the literature (Solea and Nunes 2007; Solea et al., 2009).

The main aims of this work can be summarized as follows:

- Prototype design: A four-wheeled vehicle was designed to work autonomously. Two wheels were mounted on the rear side of the vehicle and two in the front side. The rear wheels are fixed and rotate only for forward and backward by two DC-Motors installed on each of them. The DC-Motors need 0-24 volts to work. These voltages come from the DC-Motor drivers which amplify the voltages adjusted by Arduino in the range of 0-5V. The front wheels are used to determine the direction (Steering angle) of the vehicle by the DC-Motor with a rack-and-pinion gear and the maximum steering angle can be obtained is $\pm 31^{o}$ this was taken into account when designing the control algorithm, and we notice that the vehicle has the highest steering angle ($\pm 31^{o}$) in the turning of the straight lines with some sharp turnings trajectory in the proposed control algorithm PD controller Figure 4.8. While the maximum linear velocity of the vehicle is 1.31 m/s this also was taken into account in the proposed control algorithm. Usually the velocity of the vehicle is approximately equal to the speed of the reference (The linear velocity of the vehicle following the reference velocity) in Trajectory-Tracking but in Path-Following the speed of the vehicle was constant and equal to 1.0367 m/s (or regulated and equal due to radius of the curvature) , unless if there are high position errors that require an increases or decreases in the speed of the vehicle to reach the right trajectory point (comparison point) and this situation is common at the start of the movement where there is a large position error between the location of the vehicle (control point) and the starting point of the trajectory (or in the turns sometimes the error amount in the position is somewhat large and near 2 m), so the vehicle increases the speed until they reach the speed limit 1.31 m/s in an attempt to reach the correct trajectory, or in sometimes they need to decrease the speed if the vehicle precedes the comparison point Figures 4.10, 4.2. The vehicle that was designed, shown high efficiency and flexibility during practical experiments and all mechanical part and electronic parts was worked synchronously.

- To define the real current position of the vehicle in real time. Four sensors were used which are GPS, IMU, Encoder and Magnetometer. The data which come from the GPS and IMU is integrated with each other and with encoder and Magnetometer data, lead to defining the real current position and the steering angle of the vehicle. This system has shown high efficiency to locate the current real position.
- Modeling an unmanned ground vehicle (UGV). The equations that describe the vehicle's behaviors are presented by a kinematic model. Since the linear dependability of the straight stationary moving of the vehicle will be examined the supposed kinematic bicycle model can be utilized to describe the motion with respect to the given reference Cartesian frame. The dynamics effects were neglected and the kinematic of the vehicle is only used because the speed of the system is quite lower. The results showed that the vehicle can be controlled using the kinematic model without taking advantage of the complexity of the dynamic model, especially that the vehicle is moving at low speeds.

- Trajectory-Tracking and Path-Following control: Two control systems based on linear techniques were designed to control the movement of an unmanned ground vehicle (UGV). This vehicle involves basically Arduino card as a microcontroller, GPS, IMU, encoder, and magnetometer as sensors and a computer. The control programs written on MATLAB which enables to keep the vehicle to follow the desired reference road created by Google Earth Map. A control action is performed to compensate the positional and orientation errors between the real pose of the vehicle from the sensors and the pose of the desired reference curve. The PD controller proposed in this study and the SMC adapted from the literature (Solea and Nunes 2007; Solea et al., 2009) are used to control the vehicle for the Trajectory-Tracking and Path-Following. The results of this research showed the high efficiency of the proposed control system to control the movement of the vehicle on the pre-defined trajectory or path with acceptable position errors and also the proposed control system gave a great smoothness in the steering angle of the vehicle.
- The results obtained from the proposed system are compared with another control system that uses the SMC as a controller. The SMC controller is based on a sliding surface that internally coupled angular errors and lateral errors with each other, that leading to the convergence of both angular errors and lateral errors. From the comparison, we note that both control methods have an acceptable performance in control of the vehicle for the Trajectory-Tracking and Path-Following with respect to the positional error and the change of the steering angle.

The proposed control system has shown a high efficiency in controlling the movement of the vehicle on the specified trajectory or path. Sometime the position errors of the proposed control system are higher than SMC control system but it is acceptable due to the fact that in the proposed control system we use the real equations obtained from the practical tests. These equations correspond to the design and actual performance of the vehicle. In the SMC Control system, they use general theoretical equations that cannot be applied in practice to the model used in this research. But on the other hand, these real equations gave the vehicle a greater smoothness in the behavior of the steering angle. We observe that the behavior of the steering angle of the proposed system is almost free of oscillations, especially in the turns where there are some oscillations in the steering angle for the SMC control system and this can be seen from Figure 4.12, which shows the steering angle behavior for the tracking of the irregular road.

Based on the above aims some remarks are made on this work:

1. The vehicle can often be controlled using the kinematic model without utilizing the complication of the dynamic model since the vehicle moves at low speeds.
2. It concludes that, the vehicles which operate with high velocities need to cover the slip angle in the mechanical model. Thus, we need to increase the complexity of the system and use a more complex mechanical system, i.e., a dynamic model.
3. It can be concluded from the results of the experiment that the desired trajectories or paths with constant speed (which falls within vehicle's speed limits) gives better result from that obtained from trajectories with variable and oscillating velocities.
4. The proposed control system works with small and acceptable errors and it has a preference in roads with sharp turns, where the error of orientation angle exceeds 90^{o}. However, The SMC could not successful in these types of roads. In addition to this, the steering angle behavior with the SMC shows very oscillatory swings since it inherently has a chattering effect.

In the future work related to this study, the following suggestions are recommended to improve the system:

1. It would be exciting to test the unmanned ground vehicle in the real time experiments with the proposed controller.
2. In addition to the kinematic model, the dynamic model will be considered in theoretical and experimental studies for higher speeds.
3. Adding new sensors to the vehicle prototype and developing new control system, it is considered that the vehicle will be able to avoid obstacles in any road.
4. PD coefficients that used in the experiments were chosen by trial and error. These coefficients can be defined by any other methods used in the literature such as neural network, Ziegler-Nichols.
5. Instead of PD controller, other controller types such as the fuzzy-logic, neural-network can be used to improve the system behavior and provide more accurate results. Also it would be exciting to study an artificial intelligent technique in free motion of mobile robot with a complex environment.

REFERENCES

Abbott, E., Powell, D., 2009. Land-vehicle navigation using GPS. ***Proceedings of the IEEE***, **87**(1): 145-162.

Allam, T., Matla Raju, S., 2016. Design of PID controller for DC Motor Speed Control Using Arduino Microcontroller. ***International Research Journal of Engineering and Technology (IRJET)***. p-ISSN: 2395-0072.

Anderson, R., Bevly, D. M., 2004, June. Estimation of slip angles using a model based estimator and GPS. ***InAmerican Control Conference***, 2004. Proceedings of the 2004 (Vol. 3, pp. 2122-2127). IEEE.

Andr´e Guilherme Nogueira Coelho dos Santos, 2008. ***Autonomous Mobile Robot Navigation using Smartphones.*** Master thesis (not published). Technical University of Lisbon,Information Systems and Computer Engineering, Lisbon, Portugal.

Anonymous, 2008. Pc Automation and robotics for everyone, ***https://www.pc-control.co.uk/incremental_encoders.htm***. Date accessed: 2008.

Anonymous, 2018. The Arduino Official Website, ***https://store.arduino.cc/usa/arduino-mega-2560-rev3***. Date accessed: 21.02.2018.

Awang, M. A. F., 2010. ***DC Motor Speed Controller*** (Doctoral dissertation, UMP). Faculty of Electrical and Electronics Engineering. Universiti Malaysia Pahang.

Bascetta, L., Cucci, D. A., Matteucci, M. 2016. Kinematic trajectory tracking controller for an all-terrain Ackermann steering vehicle**.** ***IFAC-PapersOnLine***, **49**(15): 13-18.

Bayar, G., Koku, A. B., ilhan Konukseven, E., 2009. Design of a configurable all terrain mobile robot platform. ***International Journal of Mathematical Models and Methods in Applied sciences.*** Volume 3, Issue 4: 366-373.

Betke, K., 2001. ***The NMEA 0183 Protocol***. Standard for Interfacing Marine Electronics Devices, National Marine Electronics Association, Severna Park, Maryland, USA.

Bradford, P. W., Spilker, J., Enge, P., 1996. ***Global Positioning System: Theory and Applications***. AIAA Washington DC, 109.

Brunner, T., Ledy, J., Lauffenburger, J. P., Manzano, M. J. R. S. Q., Changey, S., Lambert, J., Basset, M., 2017. Road vehicle relative motion estimation using gyro-free IMUs and system independent observers. ***IFAC-PapersOnLine***, **50**(1): 14824-14829.

Chen, B., Jiao, Z. 2011, June. Adaptive path following control of car-like mobile robot using dynamic model. In ***Industrial Electronics and Applications (ICIEA)***, 2011 6th IEEE Conference on (pp. 239-244). IEEE.

Dagci, O. H., Ogras, U. Y., Ozguner, U., 2003, June. Path following controller design using sliding mode control theory. In ***American Control Conference, 2003. Proceedings of the 2003*** (Vol. 1, pp. 903-908). IEEE.

Daily, R., Bevly, D. M., 2004. The use of GPS for vehicle stability control systems. ***IEEE Transactions on Industrial Electronics,*** **51**(2): 270-277.

Demir, B. E., Bayir, R., Duran, F., 2016. Real-time trajectory tracking of an unmanned aerial vehicle using a self-tuning fuzzy proportional integral derivative controller. ***International Journal of Micro Air Vehicles***, **8**(4): 252-268.

Di Paola, D., Milella, A., Cicirelli, G., Distante, A., 2010. An autonomous mobile robotic system for surveillance of indoor environments**. *International Journal of Advanced Robotic Systems***, **7**(1): 8.

Durfee, W., 2011. ***Arduino Microcontroller Guide***. University of Minnesota.

Elkaim, G., Connors, J., 2008. Trajectory Generation and Control Methodology for an Autonomous Ground Vehicle. In ***AIAA Guidance, Navigation and Control Conference and Exhibit***(p. 7261).

Găşpăresc, G., 2016, October. PID control of a DC motor using Labview Interface for Embedded Platforms. In ***Electronics and Telecommunications (ISETC), 2016 12th IEEE International Symposium on*** (pp. 145-148). IEEE.

Gavran, M., Fruk, M., Vujisić, G., 2017, May. PI controller for DC motor speed realized with Arduino and Simulink. In *Information and Communication Technology,* ***Electronics and Microelectronics (MIPRO), 2017 40th International Convention on*** (pp. 1557-1561). IEEE.

Georgy, J., 2010. ***Advanced nonlinear techniques for low cost land vehicle navigation*** (Doctoral dissertation). Queen's University, Kingston, Ontario, Canada.

Goge, D. W., 1995. A Brief History of Unmanned Ground Vehicle (UGV) Development Efforts. ***Unmanned System Magazine, United States of America.*** **13** (3): 1-9.

Gopalakrishnan, B., Tirunellayi, S., Todkar, R., 2004. Design and development of an autonomous mobile smart vehicle: ***a mechatronics application. Mechatronics***, **14**(5): 491-514.

Guizzo, E., 2011. How google's self-driving car works. ***IEEE Spectrum Online***, October, 18.

Henry, A., 2009. ***Using Google Earth for Internet GIS.*** Master thesis (not published). Institute of Geography School of GeoSciences Edinburgh, University of Edinburgh. MSc in Geographical Information Science, 24.

Hernandez, J. I., Kuo, C. Y., 2003. Steering control of automated vehicles using absolute positioning GPS and magnetic markers. ***IEEE Transactions on Vehicular Technology,*** **52**(1): 150-161.

Jayetileke, H. R., De Mei, W. R., Ratnayake, H. U. W., 2014, December. Real-time fuzzy logic speed tracking controller for a DC motor using Arduino Due. ***InInformation and Automation for Sustainability (ICIAfS), 2014 7th International Conference on*** (pp. 1-6). IEEE.

Kada, B., Ghazzawi, Y., 2011, October. Robust PID controller design for an UAV flight control system. In ***Proceedings of the World Congress on Engineering and Computer Science*** (Vol. 2, No. 1-6).

Kaplan, E., Hegarty, C. 2005. ***Understanding GPS: principles and applications***. Artech house. Boston, London

Kavya, S., C., Arun, S., M., 2016. Kinematic Car Model Driven by Actuators. ***International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering***. **5**(7): 2320-3765.

Kehl, S., Pölsler, W. D., Zeitz, M., 2005. Vehicle path-following with a GPS-aided inertial navigation system. In ***Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems*** (pp. 285-300). Springer, Berlin, Heidelberg.

Kim, J., Lee, S., 2016. A vehicular positioning with GPS/IMU using adaptive control of filter noise covariance. ***ICT Express***, **2**(1): 41-46.

Kouros, G., Petrou, L. 2017, June. PANDORA monstertruck: A 4WS4WD car-like robot for autonomous exploration in unknown environments. In ***Industrial Electronics and Applications (ICIEA), 2017 12th IEEE Conferenceon*** (pp. 974-979). IEEE.

Lauszus, K., 2012. A practical approach to Kalman filter and how to implement it. ***Guides, TKJ Electronics, September***.

Levie, E., 2008. ***VISUALISING LARGE WEB APPLICATION DATASETS IN GOOGLE EARTH*** (Doctoral dissertation), Rensselaer Polytechnic Institute, Troy, New York.

Li, Y., He, L., Yang, L. (2013, July). Path-following control for multi-axle car-like wheeled mobile robot with nonholonomic constraint. In ***Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*** (pp. 268-273). IEEE.

Lin, C. E., Hsu, C. W., Li, Y. S., 2005, February. Verification of remote vehicle control using mobile communication. In ***Sensors for Industry Conference****, 2005* (pp. 111-117). IEEE.

Lin, W., 2012. ***Path Following for a Car-like Mobile Robot based on Fuzzy-logic***. Master thesis (not published). Örebro University,Department of Technology, Stockholm, Sweden.

Londhe, P. S., Santhakumar, M., Patre, B. M., Waghmare, L. M., 2017. Task space control of an autonomous underwater vehicle manipulator system by robust single-input fuzzy logic control scheme. ***IEEE Journal of Oceanic Engineering***, **42**(1): 13-28.

Long, X., 2013. ***Tactile-Based Mobile Robot Navigation*** (Doctoral dissertation, Worcester Polytechnic Institute).

Loughnane, D. J., 2001. ***Design and Construction of an Autonomous Mobile Security Device*** (Doctoral dissertation, University of Waikato).

Moret, E. N., 2003. ***Dynamic modeling and control of a car-like robot*** (Doctoral dissertation, Virginia Tech). Electrical Engineering Department, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Moubarak, P. M., Ben-Tzvi, P., 2011, September. Adaptive manipulation of a hybrid mechanism mobile robot. In ***Robotic and Sensors Environments (ROSE), 2011 IEEE International Symposium on*** (pp. 113-118). IEEE.

Naranjo, J. E., González, C., García, R., De Pedro, T., Haber, R. E., 2005. Power-steering control architecture for automatic driving. ***IEEE Transactions on Intelligent Transportation Systems***, **6**(4): 406-415.

Naranjo, J. E., González, C., García, R., de Pedro, T., Revuelto, J., Reviejo, J., 2004, June. Fuzzy logic based lateral control for GPS map tracking. ***In Intelligent Vehicles Symposium***, 2004 IEEE (pp. 397-400). IEEE.

Nguyen-Huu, P. N., Titus, J., Tilbury, D., Ulsoy, G., 2009. Reliability and failure in unmanned ground vehicle (UGV). ***GRRC Technical Report***. UNIVERSITY OF MICHIGAN.

Ouadah, N., Ourak, L., Boudjema, F., 2008. Car-like mobile robot oriented positioning by fuzzy controllers. ***International Journal of Advanced Robotic Systems***, **5**(3): 25.

Pepy, R., Lambert, A., Mounier, H. 2006, April. Path planning using a dynamic vehicle model. In ***Information and Communication Technologies****, 2006. ICTTA'06. 2nd* (Vol. 1, pp. 781-786). IEEE

Perry, C. D., Rains, G. C., 2009. Explaining GPS: the Global Positioning System. The University of Georgia, College of Agriculture and Environmental Sciences and Family Consumer Sciences.

Pham, V. T., Nguyen, V. T., Chu, D. T., Tran, D. T., 2015. 15-State extended Kalman filter design for INS/GPS navigation system. ***Journal of Automation and Control Engineering***, **3**(2).

Radhamani, N. P., Thomas, B. A., 2016. Combined Information Processing of GPS and IMU Sensor using Kalman Filtering. ***International Journal of Innovative Science, Engineering & Technology*** , Vol. 3 Issue 5. (ISSN2348-7968).

Rambabu, S., 2007. ***Modeling and control of a brushless DC motor***. Master of Thesis In Power Control and Drives Technology, National Institute of Technology Rourkela.

Schworer, I. J., 2005. ***Navigation and Control of an Autonomous Vehicle*** (Doctoral dissertation, Virginia Tech).

Siciliano, B., Khatib, O. (Eds.)., 2008. ***Springer Handbook of Robotics***. Springer Science & Business Media, Dudek, Gregory Jenkin, Michael 477-490.

Slotine, J. J. E., Li, W., 1991. ***Applied Nonlinear Control*** (Vol. 199, No. 1). Englewood Cliffs, NJ: Prentice hall.

Smith, A. G., 2011. ***Introduction to Arduino a Piece of Cake***. *Book, Published, ISBN, 1463698348.*

Solea, R., Nunes, U., 2007. Trajectory planning and sliding-mode control based trajectory-tracking for cybercars. ***Integrated Computer-Aided Engineering***, **14**(1): 33-47.

Solea, R., Filipescu, A., Nunes, U., 2009, August. Sliding-mode control for trajectory-tracking of a wheeled mobile robot in presence of uncertainties. In ***Asian Control Conference, 2009. ASCC 2009. 7th*** (pp. 1701-1706). IEEE.

Suiçmez, E. C., 2014. ***Trajectory Tracking of a Quadrotor Unmanned Aerial Vehicle (uav) via Attitude and Position Control*** (Doctoral dissertation), Middle East Technical University.

Temel, S., Yağli, S., Gören, S., 2013. P, PD, PI, PID CONTROLLERS. ***Ee402-Discrete Time Control Systems Recitation 4 Report*** Middle East Technical University, Electrical and Electronics Engineering Department.

Toledo, J., Piñeiro, J. D., Arnay, R., Acosta, D., Acosta, L., 2018. ***Improving Odometric Accuracy for an Autonomous Electric Cart***. (Master Thises) Computer Science and System Department, Universidad de La Laguna, 38200 Santa Cruz de Tenerife, Spain. *Sensors*, **18**(1):200.

Uğur Kayasal, 2007. ***Modeling And Simulation Of A Navigation System With An Imu And A Magnetometer*** (Master thesise), Mechanical Engineering, The Graduate School Of Natural And Applied Sciences, Middle East Technical University, Ankara Turkey.

Vikhe, P., Punjabi, N., Kadu, C., 2014. Real Time DC Motor Speed Control using PID Controller in LabVIEW. ***International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering***, **3**(9): 12162-12167.

Ye, Y., He, L., Zhang, Q., 2016. Steering control strategies for a four-wheel-independent-steering bin managing robot. Center for Precision & Automated Agricultural Systems, Washington State University, Prosser, WA 99350, USA.***IFAC-PapersOnLine***, **49**(16): 39-44.

Zakaria, M. A., Zamzuri, H., Mazlan, S. A., Zainal, S. M. H. F., 2012. Vehicle path tracking using future prediction steering control. ***Procedia Engineering***, **41**, 473-479.

Zhao, P., Chen, J., Song, Y., Tao, X., Xu, T., Mei, T., 2012. Design of a control system for an autonomous vehicle based on adaptive-pid. ***International Journal of Advanced Robotic Systems***, **9**(2): 44.

APPENDIX INDEX

Appendix 1: MATLAB software codes for Trajectory-Tracking and Path-Following control.

% MATLAB codes for Trajectory-Tracking PD controller.

```
L =1.1;% inter axle distance
XX=30*sin((1/40).*t);                                    % X coordinate of the reference
YY=30*sin((1/40).*t).*cos((1/40).*t);                    % Y coordinate of the reference
deltaT=0.01;
Xd=diff(XX)/deltaT;
Yd=diff(YY)/deltaT;
Xdd=diff(Xd)/deltaT;
Ydd=diff(Yd)/deltaT;
Xd=Xd(1:25131);
Yd=Yd(1:25131);
cs=abs(((Xd.*Ydd)-(Xdd.*Yd))./(sqrt(Xd.^2+Yd.^2)));
VRef=sqrt(Xd.^2+Yd.^2);                                   % Reference Velocity
VRefd=diff(VRef)/deltaT;
wr=VRef.*cs;
u1=0.5;                                                   % Input voltage of front DC-Motor
u2=0;                                                     % Input voltage of front DC-Motor
vu=0.2908*u1-0.1265;                                      % Linear Velocity of the vehicle
th_car=45*pi/180;                                  % Vehicle's Orientation
alf=0*pi/180                                       % Vehicle's steering angle
Xc=-10;                                             % X- coordinate of the control point of the Vehicle's
Yc=10;                                             % Y- coordinate of the control point of the Vehicle's
d0=0;
therini= atan2(Yd,Xd);
ther=hizsmooth(therini);
sayac=25130;
T=t(1:25130);
for i= 1:sayac
a1=(XX(i)-Xc)^2;
b1=(YY(i)-Yc)^2;
d= sqrt (a1+b1);
D(i)=d;                                             % Position error
d0=d;
XS=XX(i);
YS=YY(i);
THR=ther(i);
Xref=XX(i);
Yref=YY(i);
VR=VRef(i);
Wr=wr(i);
xd= cos(th_car)* vu;
yd= sin(th_car)* vu;
thed= ((tan(alf))/L)*vu;
xe=cos(THR)*(Xc-Xref)+ sin(THR)*(Yc-Yref);          % Lateral Error
ye=-sin(THR)*(Xc-Xref)+ cos(THR)*(Yc-Yref);         %Longitudinal Error
Th_err=th_car-THR;                                  % Orientation Error
Xed=-VR+(vu*cos(Th_err))+(ye*Wr);
Yed=vu*sin(Th_err)-(xe*Wr);
 Th_errd=(vu/L)*tan(alf)-Wr;
  if Th_err>pi
```

```
    Th_err=Th_err-2*pi;
  end
  if Th_err<-pi
    Th_err=Th_err+2*pi;
  end
  if Th_err<0.05;
  Th_err=Th_err;
  end
ANGLEHATA(i)=Th_err;
YE(i)=ye;
XE(i)=xe;
Kxp=0.5;                                  % control parameter
Kxd=0.05;                                 % control parameter
if xe>0
     vu=vu-(Kxp*xe+Kxd*Xed);
elseif xe<0
     vu=vu+(Kxp*abs(xe)+Kxd*abs(Xed));
else
     vu=vu;
end
if vu>1.3
     vu=1.3;
end
if vu<0.1
     vu=0.1;
end
VU(i)=vu;
Vx(i)=xd;
Vy(i)=yd;
x=Xc+xd*deltaT;
y=Yc+yd*deltaT;
th_car=th_car+thed*deltaT;
  if th_car>2*pi
      th_car=th_car-2*pi;
  end

  if th_car<-2*pi
      th_car=th_car+2*pi;
  end
 TH_CAR(i)=th_car;
alfd=(0.0968*u2)-0.0842;
alf=alf+alfd*deltaT;                     % steering angle
  if abs(alf)>pi/6
   alf=(pi/6)*sign(alf);
  end
ALF(i)=alf;
    ktp=95;                               % control parameter
    ktd=35;                               % control parameter
    kyp=95;                               % control parameter
    kyd=25;                               % control parameter
  if ye>0
    u2=-(ktp*Th_err+ktd*Th_errd+(kyp*ye+kyd*Yed));
  elseif ye<0

   u2=-(ktp*Th_err+ktd*Th_errd)+(kyp*abs(ye)+kyd*abs(Yed));
  else
     u2=0;
  end
```

```
  if abs(u2)>5
   u2=5*sign(u2);
  end
Xc=x;
Yc=y;
U2(i)=u2;
Xp(i)=x;
Yp(i)=y;
end
```

% MATLAB codes for Trajectory-Tracking SMC controller.

```
L = 1.2;
k0=0.02;                          % control parameter
k1=1.75;                          % control parameter
k2=1.75;                          % control parameter
p1=0.5;                           % control parameter
p2=0.5;                           % control parameter
q1=0.5;                           % control parameter
q2=0.5;                           % control parameter
u1=0.5;
u2=0;
Voc=vu=0.2908*u1-0.1265;          % Linear speed
theoc=45*pi/180;                  % Orientation of the vehicle
d0=0;
alfoc=0;                          % Steering Angle of the vehicle
deltaT=0.05;
Xoc=-0.2;                           % X coordinate of the control point
Yoc=0.2;                            % Y coordinate of the control point
t=0:0.01:80*pi;
Xr=30*sin((1/40).*t);                         % X coordinate of the reference
Yr=30*sin((1/40).*t).*cos((1/40).*t);         % Y coordinate of the reference
deltaT=0.01;
Xrd=diff(Xr)/deltaT;
Yrd=diff(Yr)/deltaT;
Xrdd=diff(Xrd)/deltaT;
Yrdd=diff(Yrd)/deltaT;
Xrd=Xrd(1:25131);
Yrd=Yrd(1:25131);
ther= atan2 (Yrd,Xrd);
cs=abs(((Xrd.*Yrdd)-(Xrdd.*Yrd))./(sqrt(Xrd.^2+Yrd.^2)));
vr=sqrt(Xrd.^2+Yrd.^2);
vrd=diff(vr)/deltaT;
wr=vr.*cs;                                      % angular velocity

for i= 1:25130;
THER=ther(i);
Cs=cs(i);
Vr=vr(i);
Wr=wr(i);
XR=Xr(i);
YR=Yr(i);
VRD=vrd(i);
a1=(XR-Xoc)^2;
b1=(YR-Yoc)^2;
d1= sqrt (a1+b1);
D1(i)=d1;
```

```
dd=(d1-d0)/deltaT;
dint=d1+(d1-d0)*deltaT;
d0=d1;
Xcd= cos(theoc)* Voc;
Ycd= sin(theoc)* Voc;
thecd= (Voc/L)*(tan(alfoc));
Xe=cos(THER)*(Xoc-XR)+ sin(THER)*(Yoc-YR);
Ye=-sin(THER)*(Xoc-XR)+ cos(THER)*(Yoc-YR);
Thee=theoc-THER;
Xed=-Vr+(Voc*cos(Thee))+(Ye*Wr);
Yed=Voc*sin(Thee)-(Xe*Wr);
Theed=(Voc/L)*tan(alfoc)-Wr;
s1=Xed+k1*Xe;
s2=Yed+k2*Ye+k0*SAT1(Ye)*Thee;
Wrd=(wr(i+1)-wr(i))/deltaT;
Vcd=(/cos(Thee))*(-q1*s1-p1*SAT1(s1)-k1*Xed-Wrd*Ye-Wr*Yed+Voc*Theed*sin(Thee)+VRD);
Alfc=atan((L/Voc)*Wr+(L/(Voc*(Voc*cos(Thee)+k0*SAT1(Ye)))*(-q2*s2-p2*SAT1(s2)-k2*Yed-Vcd*sin(Thee)+Wrd*Xe+Wr*Xed)));
Vc=Voc+Vcd*deltaT;
 if abs(Alfc)>pi/6
  Alfc=(pi/6)*sign(Alfc);
 end
 if Vc>1.30
   Vc=1.30;
 end
Xc=Xoc+Xcd*deltaT;
Yc=Yoc+Ycd*deltaT;
Thec=theoc+thecd*deltaT;
Xoc=Xc;
Yoc=Yc;
theoc=Thec;
alfoc=Alfc;
Voc=Vc;
VC(i)=Vc;
U2(i)=u2;
Xpp(i)=Xc;
Ypp(i)=Yc;
Alfc1(i)=Alfc;
Thec(i)=Thec;
end
```

% MATLAB codes Path-Following PD controller.

```
L = 1.1;
alfc=0;
u1=4;
u2=0;
vu=0.2908*u1-0.1265;
th_car=0*pi/180;
alf=0;
Xc=0;
Yc=-5;
deltaT=0.1;
XRef=AutoT42(1,:);
YRef=AutoT42(2,:);
XRefd=diff(XRef)/deltaT;
YRefd=diff(YRef)/deltaT;
```

```
XRefdd=diff(XRefd)/deltaT;
YRefdd=diff(YRefd)/deltaT;
XRefd=XRefd(1:31297);
YRefd=YRefd(1:31297);
theRef= atan2(YRefd,XRefd);
CS=abs(((XRefd.*YRefdd-XRefdd.*YRefd)))./(sqrt(XRefd.^2+YRefd.^2).^(3/2));
VRef=sqrt(XRefd.^2+YRefd.^2);
WRef=VRef.*CS;
therini= atan2(YRefd,XRefd);
ther=hizsmooth(therini);
sayac=31297;
Nitr=1;
RC=1./CS;                                      % radius of Curvature
ksay=1;
tt=1:0.1:50000;
T=tt(1:36368);
for i= 1:50000
[Nr,Hata_i]=HataPath(XRef,YRef,Xc,Yc,Nitr);    % Function that find the nearest point
HATA(i)=Hata_i;                                % position error
NSAY(i)=Nr;
THR=ther(Nr);
Xref=XRef(Nr);
Yref=YRef(Nr);
VR=VRef(Nr);
wr=WRef(Nr);
Ro=RC(Nr);
xd= cos(th_car)* vu;
yd= sin(th_car)* vu;
thed= ((tan(alf))/L)*vu;
xe=cos(THR)*(Xc-Xref)+ sin(THR)*(Yc-Yref);
ye=-sin(THR)*(Xc-Xref)+ cos(THR)*(Yc-Yref);
Th_err=th_car-THR;
  if Th_err>pi
    Th_err=Th_err-2*pi;
  end
  if Th_err<-pi
    Th_err=Th_err+2*pi;
  end
ANGLEHATA(i)=Th_err;
xed=-VR+(vu*cos(Th_err))+(ye*wr);
yed=vu*sin(Th_err)-(xe*wr);
Th_errd=(vu/L)*tan(alf)-wr;
YE(i)=ye;
  if vu>1.31
     vu=1.31;
end
if vu<0.1
     vu=0.1;
end
VU(i)=vu;
Vx(i)=xd;
Vy(i)=yd;
x=Xc+xd*deltaT;
y=Yc+yd*deltaT;
th_car=th_car+thed*deltaT;
  if th_car>2*pi
     th_car=th_car-2*pi;
```

```
  end
  if th_car<-2*pi
      th_car=th_car+2*pi;
  end
TH_CAR(i)=th_car;
alfd=(0.0968*u2)-0.0842*alf;
alf=alf+alfd*deltaT;
if abs(alf)>pi/6
     alf=(pi/6)*sign(alf);
end
ALF(i)=alf;
    ktd=0.001;
    ktp=1.5;
    kyp=.01;
    kyd=0.25;

if ye>0
   u2=-(ktp*Th_err+ktd*Th_errd+(kyp*ye+kyd*yed));

  elseif ye<0
   u2=-(ktp*Th_err+ktd*Th_errd)+(kyp*abs(ye)+kyd*abs(yed));
end
if  ye>-0.05 & ye <0.05
        if ye>0
        u2=-(ktp*(0.01/(2+(ye^2)))*Th_err+(ktd*(0.01/(2+(ye^2)))*Th_errd))
    elseif ye<0
         u2=(ktp*(0.01/(2+(ye^2)))*abs(Th_err)+(ktd*(0.01/(2+(ye^2)))*abs(Th_errd)));
    end

if abs(u2)>5
    u2=5*sign(u2);
end
DFin=sqrt((XRef(31298)-x)^2+(YRef(31298)-y)^2);

Xc=x;
Yc=y;
Nitr=Nr;
U2(i)=u2;
Xp(i)=x;
Yp(i)=y;
if DFin<0.46
break;
end
ksay=ksay+1;
end
```

% Finding the nearest point of the vehicle (Path-Following).

```
function [N,HATA]=HataPath(XX,YY,Xc,Yc,Nitr)


nb=Nitr;
np=500;
ENBy=1;
hataKucuk=10000;
for i=nb:nb+np;
```

```
if i>31297
        i=31297;
end
xr=XX(i);
yr=YY(i);
    hata=sqrt((xr-Xc)^2+(yr-Yc)^2);
  if hata<hataKucuk
    hataKucuk=hata;
    ENBy=i;
   end
end
N=ENBy;
HATA=hataKucuk;
hataKucuk=10000;
```

Appendix 2. EXTENDED TURKISH SUMMARY (GENİŞLETİLMİŞ TÜRKÇE ÖZET)

İNSANSIZ BİR YER ARACININ TASARIMI VE GPS İLE YÖRÜNGE KONTROLÜ

Otonom mobil robotlar, kendiliğinden hareket edebilen ve özel kılavuz donanım veya elektromekanik cihazlara ihtiyaç duymadan amaçlanan ortamlarda gezinme kabiliyetine sahip olan otomatik makinelerdir.Bu mobil robotlar giderek yaygınlaşmakta ve hemen hemen hayatın her alanında (ticari, endüstriyel, askeri, vb.) yaygın olarak kullanılmaktadırlar. Bu çalışma insansız yer aracı olan bir otonom mobil robotun tasarım ve yörünge/yol takip kontrolünü içermektedir.Bu robot kendi kendine hareket etme yeteneğine sahip olup önceden tanımlanmış bir yörünge veya yol en küçük konum hatasına göre takip edebilmektedir. Önceden tanımlı bir yörünge veya yolu takip edecek bu aracın hareketi konum ve hız seviyesinde bir geri beslemeli sistem ile kontrol edilecektir. Bu geri beslemeli kontrol sistemi aracın geçek konumu ve referans yörünge veya yol arasındaki konum hatasını telafi edecektir. Bu amaç için özerinde bazı düzenlemeler yapılmış klasik PD kontrol metodu mobil robotun kontrolünde kullanılacak olup bu tip manipülatörler için tasarlanmış literatürdeki diğer bir kontrol metodu ile karşılaştırılacaktır.

Önce insansız yer aracı tüm parça ve sistemleri ile detaylı olarak bir bilgisayar tasarım programı tasarlanmıştır. Daha sonra bu tasarıma göre şase üretimi yapılıp gerekli tüm donanım ve cihazların montajı gerçekleştirilmiştir. Bu çalışmada tasarlanan kendinden tahrikli insansız yer aracı, iki arka tekerleği ayrı ayrı iki DC motorla tahrik edilen dört tekerden oluşmaktadır.Bu mobil robotun yönlendirilmesi ön taraftaki tekerlerden yapılmaktadır. Tasarımda ikisi boylamsal hareket için biri de aracın yönünü ayarlamak için toplam üç adet DC motor kullanılmaktadır. Bu DC motorlar redüktörlü olup 0-24 volt aralığında çalışmaktadırlar. Bu motorlar gerekli voltajları mikro kontrolcüden gelen kontrol voltajları ile ayarlanan sürücü kartları üzerinden almaktadırlar.Bu sistemde mikro kontrolcü olarak basit, ucuz ve programlaması kolay olan Arduino kartı kullanılmıştır. Bilgisayar ve mobil robot üzerindeki tüm cihazlarla olan haberleşmeler bir USB bağlantısına sahip Arduino kartı ile gerçekleştirilmiştir. Arduino 0 ile 5 volt arasında kontrol voltajı üreterek DC motor sürücüleri üzerinden volt gerilimini her bir DC motor için kontrol edebilmektedir. Aracın direksiyon sistemi bir tür kremayer dişli sistemi olup bir DC motor ile tahrik edilmektedir. Ön tekerleklerin dikey eksen etrafındaki açısal hızlarını ölçmek için yani dolaylı olarak dümenleme açısının değerini ölçmek için bir enkoder cihazı dümenleme DC motorunun çıkışına yerleştirilmiştir. Bu ölçme sistemi aracın gerçek konumunu ve yönünü doğru olarak ölçmek için sensör olarak birer adet Küresel Konumlama Sistemi (GPS) cihazı, ataletsel ölçüm birimi (IMU), dijital pusula ve enkoder içermektedir. Tüm bu ölçülen veriler gerçek zamanlı olarak bilgisayarın seri portu üzerinden alınıp buna karşılık gelen gerçek konumu vermek için eşzamanlı olarak bilgisayar üzerinde işlenmektedirler.

Bu çalışmada insansız yer aracının davranışını tanımlamak için sadece kinematik bisiklet modeli kullanılarak araç modellenmiştir. Araç hafif olduğu ve yavaş hareket ettiği için modeldeki dinamik etkiler ihmal edilmiştir. Bisiklet modeli arka tekerleri ve ön tekerleri bir araya getirerek iki tekerlekli tek izli olarak dört teker aracı modelini basitleştirmiştir. Bu basitleştirilmiş kinematik modelde yanal kararlılık üzerine herhangi bir etkiye sahip olmamasından dolayı aracın yanal boyutları ihmal edilmiştir.

İstenilen referans rotaları takip edebilmek için insansız yer aracının konum, hız ve yönelimini kontrol edecek kinematik modele dayalı bir algoritma önerilmiştir. Teorik ve deneysel sonuçlara göre, değiştirilmiş PD kontrolcüyü kullanan önerilen kontrol sistemi istenilen yörünge veya yol takibinde aracın kontrolü işleminde oldukça etkin olduğunu kanıtlamıştır. Kontrol sisteminin etkinliğini kanıtlamak için, hem teorik hem de deneysel çalışmalardan elde edilen sonuçlar literatürdeki mobil robotun kontrolünde kullanılan kayan kip kontrol (SMC) yöntemi ile karşılaştırılmıştır. Karşılaştırmanın sonuçları, önerilen sistemin konum hatası, hız kısıtlaması ve dümenleme açısına göre yüksek bir performansa sahip olduğunu göstermektedir.

Teorik ve deneysel çalışmalarda, yörüngeler veya yollar iki boyutlu uzay eğrileri veya bilindik geometrik eğriler olarak tanımlanmışlardır. Bu eğrileri oluşturmak için kontrol noktaları Google Earth programı ile seçilmişlerdir. Mobil robotun pozisyon ve yönelimindeki hatalar (X_e, Y_e, θ_e) aracın anlık pozisyonu ve ilgili referans eğrisi karşılaştırılarak hesaplanmaktadır. Burada gerçek araç pozisyonu gerçek zamanlı uygulamalar için GPS / IMU entegrasyonu ile belirlenecektir. Tüm bu gerçek zamanlı testler, MATLAB yazılımı üzerine yazılan kullanıcı tanımlı programlar aracılığıyla

gerçekleştirilmektedir. Bu benzetim sonuçları insansız yer aracının tasarımı ve kontrolünde istenilen amaçlara ulaşmada ne denli başarılı olunduğunu göstermektedir.

Anahtar kelimeler: Küresel konumlama sistemi (GPS), Mobil robot, Yol takibi, Yörünge izleme, İnsansız yer aracı.

1. GİRİŞ

Otonom hareketli robotlar (OHR) kendi kendilerine hareket etme ve özel yönlendirme donanımları ya da elektromekanik cihazlara gereksinim duymaksızıniş görecekleri ortamda yönlerini bulabilme kabiliyetine sahip otomatik makinelerdir. Otonom hareketli robotlar giderek yaygınlaşmakta ve hayatın tüm alanlarında kullanıma (ticari, endüstriyel, askeri vb.) girmektedirler. Bu araçların geliştirme süreçleri General Motors, Apple, Google, Intel, Audi, BMW ve daha pek çoklarını içerek 35'ten fazla otomotiv ve teknoloji firmasının katılımıyla daha da hızlanmıştır. 2009'da Google firması kendi kendine giden bir otomobil üretmek maksadıyla kendi kendine sürme teknolojisi geliştirmeye başlamıştır. Google bu teknolojiyi California eyaletinde Toyota'nın araçları ile denemeye başlamış ve aynı yıl kendi kendine giden araçları ile otobanlarda kazasız yarım milyon kilometre yol aldıklarını açıklamıştır.

Otonom hareketli robotlar kendi kendilerine hareket edebilen ve içinde bulundukları ortamda yol alabilen otomatik makinelerdir. Bu hareketli robotlar ortamda belirli bir noktaya sabitlenmezler. Bu robotların seyrüsefer ile ilgili özel yönlendirme donanımlarına ya da elektromekanik aletlere ihtiyaç duymaksızın içinde bulundukları ortamda hareket etmeleri mümkündür. Bazı durumlarda bu hareketli robotların çalışma alanında önceden konumlandırılmış yönlendiriciler ile belirli rotalarda hareket edebilmektedirler.

Kullanım alanlarına ve amaçlarına bağlı olarak bu robotlar farklı donanımlara ve parçalara sahip olabilirler. Bunlardan önemli olan bazıları aşağıda verilmiştir:

1. Kontrol donanımları (bunlar işlemci, mikro-kontrol birimleri ya da kişisel bilgisayarlar yani PC'ler olabilir)

2. Kontrol yazılımları (bunlar farklı yüksek-seviye programlama dilleri (C, C++, Fortran, Pascal vb.) ile yazılabilecek olan ve robotların kontrolünü sağlayan program veya algoritmalardır)
3. Algıyıcılar (sensörler) ve tahrik düzenekleri (aktüatörler) [Tahrik düzenekleri robotları itici gücü sağlayan parçalardır. Algılayıcılar ise hareketli robotun kullanım alanına ve gereksinimlerine göre (dokunma ve ortamı algılama, engellerden kaçınma vb.) işlev gösteren ortam algılayıcılarıdır]

İnsansız Kara Aracı (İKA) malzeme veya insan taşımak ya da başka amaçlarla kara üzerinde hareket edebilen tüm mekanik cihazlara verilen addır. Bu hareket bir sürücüye gereksinim duyulmaksızın robotun karmaşık ortamlarda kendiliğinden ilerleyebileceği şekilde gerçekleşir. İKA'lar genelde ikiye ayrılırlar:

- Uzaktan Kumandalı İKA: Bu tür İKA'lar bir insan tarafından (operatör) uzaktan kontrol edilir.
- Otonom: Bu tür İKA'lar insan müdahalesine gerek duymadan kendi hareketlerini kontrol edebilirler.

Bu araştırma çalışması iki kısımdan oluşmaktadır. İlk kısımda kendi kendine hareket edebilen (otonom) bir İnsansız Kara Aracı (İKA) prototipi tasarlanmıştır. Bu araç dört tekerleği olacak şekilde dizayn edilmiştir. Ön tekerlekler ön dingilin orta kısmına yerleştirilmiş bir DC motor ile aracın yönelimini (dönme açısı)belirlemek amacıyla kullanılmıştır. Yani bu araç ön teker yönelimli bir hareketli robottur. Arka tekerler sabit açılı olup her bir tekerleğe yerleştirilmiş olan birer adet DC motor ile araca hareket kazandırılmaktadır. Hareketli robotun diferansiyel dişli sistemi olmadığından bu iki motorun farklı hızlarda dönmeleri kontrol ünitesi üzerinden yazılımla sağlanmıştır. Bu çalışmada kullanılan DC motorlar 0-24 volt arasında çalışabilmektedir. Voltajın yön ve yük değerleri üzerindeki değişiklikler DC motorların dönme yönü ve hızını doğrudan kontrol etmektedir. Bu doğrultuda düzgün bir sürüş elde etmek için voltaj değişim değerleri göreceli olarak küçük DC voltaj değerleri arasında olmalıdır. Bu üç DC motor ile dönüş ve boylamsal eksenlerde hareket olanağı elde edilmiştir. Sürücüler için kontrol girişleri her bir motora bağlı Arduino mikto-kontrol kartları ile elde edilmiştir. Arduino, PWM çıktı pinleri üzerinden 0 ila 5 volt arasında değişebilen kontrol voltajı imkanı sağlamaktadır. Bilgisayar ve araç prototipi arasındaki tüm iletişim Arduino tarafından sağlanmaktadır.

Çalışmanın diğer kısmı ise aracın kinematik modellemesi ve Yönelim-Takibi ve Yol-İzleme kontrollerinin sağlanmasına yönelik bir kontrol algoritması oluşturmak ve aracın belirli yönelimlerle (ya da rotalarda) hareket ve konumlanmasını ve hızını kabul edilebilir hata payıyla sağlamaya yöneliktir. Aracın konum ve yönelimi, değiştirilmiş bir PD kontrolü şeklinde bir geri-bildirim kontrol mekanizması kullanılarak yapılmaktadır. Bu robotun kontrol sisteminde GPS (Küresel Konumlama Sistemi), bir IMU (Eylemsizlik Ölçüm Birimi), Enkoder ve Dijital pusula (Magnetometre) sensörleri bulunmakta ve bunlar aracın anlık yönelimi ve konumunu tespit etmekte kullanılmaktadır. Bu sistem (GPS ve IMU) açık veya kapalı çok çeşitli ortamlarda iş görebilmektedir.

Bu çalışmada robotun çalışma alanı olarak açık hava ortamları seçilmiştir. GPS ve IMU'lu bu sistem şehir bölgelerinde, vadilerde, tünellerde ve diğer benzer alanlarda iş görebilmektedir. İyi sonuçlar elde etmek için GPS yüksek çözünürlükte kullanılmalı ve sonuçlar önden belirlenmiş rotalarla karşılaştırılmalıdır. GPS/IMU entegrasyonundan gelen veriler aracın anlık gerçek konum ve yönelimini bildirir ve bu veriler önceden belirlenmiş rota (ya da yol) ile kıyaslanarak hatalar tespit edilir (mesafe ve yönelim hataları) ve bunlarla ilgili düzeltme sinyalleri oluşturulur. Bu sırada kodlayıcı tekerlek dönme açısını doğrudan ölçmekte kullanılırken, manyetometre de dünya eksenine oranla aracın yönelimini vermektedir. Bu sinyaller aracın ihtiyaç duyduğu hız ve dönme açısını belirlemekte kullanılır. Hataların telafisi ise bir PC üzerinde kullanıcı tarafından oluşturulan bir program tarafından gerçekleştirilir.

Kara araçları endüstrisi genelde kısmen ya da bütün olarak kendi kendine ilerleyen akıllı araçlar yönünde ilerlemektedir. Bu araçların rol tanımı olarak sürücüye yardımcı olmak belirlenmiştir. Bağımsız seyrüsefer elde etmek içinse bir aracın aşağıdaki özelliklere sahip olması gerekir:

- Verileri işlemek ve aracın konumunu tespit etmek için sensörler ve özel donanımlar (birimler)
- Aracın izleyeceği rota.

- Engellerden kaçınmak için sensörler.

Bu tezde bu özelliklerin ilk ikisi üzerinde durulacak, özellikle de verilerin elde edilmesi ve önceden belirlenen rota ile kıyaslanarak aracın gerçek konum ve yöneliminin tespiti elde edilecektir. Bu noktada bu hedeflerin gerçekleştirilmesi aracın mevcut konumunu tespit etme ve bir sonraki hedefini tespit etme başarısına bağlıdır. Sensör olarak GPS ve IMU kombinasyonu kullanılmaktadır. Önceden belirlenecek rota ve güzergahlar ise Google Earth, GIS vb programlar kullanılarak çizilebilir. Google Earth harita, coğrafya veya bilgi amaçlı kullanılabilen bir programdır. Bu program aracılığıyla, uydu ve hava görüntüleri ve Coğrafı Bilgi Sistemleri (GIS) kullanılarak istenilen bölgenin üç boyutlu haritası elde edilebilmektedir.

Önceden belirlenen yörüngeler veya rotalar, eğriler ya da parametreleri değiştirilerek istenen hale getirilmiş geometrik şekillerdir. Bu eğriler Google Earth'de ilgili kontrol noktaları için gereken koordinatlar girilerek belirlenebilmektedir. Aracın kontrol sistemi aracın konumunu, hızını, yönünü ve yönelimini bilmek zorundadır. Bu bilgiler araç üzerindeki GPS/INS sisteminden sağlanmaktadır(Bradford, P. W. Ve ark., 1996).

Yakın zamanda pek çok araştırmacı hareketli robotlara yönelmiştir. Bu çalışmalar hareketli robot türlerinin incelenmesi, tasarlanmaları ve kontrol edilmeleri, modellenmeleri, GPS kontrolleri, dijital haritalar, ve hareketli sistem donanımları konularında gerçekleştirilmektedir.

Hareketli robotların kontrolünde kullanılabilecek PID, Bulanık Mantık (Fuzzy Logic), Kayma ModuKontrolü gibi çok çeşitli kontrol sistemleri mevcuttur. Yörünge ve rota birbirlerinden bağımsızdırayrı şeylerdir. Yörünge zaman bağımlı iken rota zamandan bağımsızdır. Solea ve Nunes (2007) Kayma Modu Kontrolü kullanan önceden belirlenen bir rotayı izleyebilen otonom bir kendinden tahrikli araç tasarlamıştır. Bu araştırmacılar özel bir kayan yüzey üzerinde yanlamasına ve açısal hataların dahili olarak birbirleri ile ilintili olacağı yeni bir kayma yüzeyi tasarlayıp, hem yanlamasına hem de açısal hataların düzeltilmesi yönünde önemli gelişmeler sağlamışlardır.Kavya, S., C., ve Arun, S., M.(2016) bir Rota-Takip yöntemi ile önceden belirlenmiş bir rota üerinde araç hareketini kontrol ettikleri çalışmalarında, PID ile kontrol edilen iki DC motorlu bir araç önermişlerdir. Bu çalışmalarındaki rota daireseldir.

Bu tezde İKA için hem Yörünge-Takibi hem de Rota-Takibi kontrolü Oransal + Türevsel (PD) ve Kayma Modu kontrolleri kullanılmıştır.Önerilen kontrol yönteminin etkinliğini göstermek için, bu çalışmada literatürde önerilen Kayan Kip Kontrol (SCM) çalışma aracına uygulanmıştır. Sonuçlar önerilen kontrol yönteminin konumsal hata ve özellikledümenlemedavranışı konularında SMC'den daha başarılı olduğunu göstermektedir. Sonuçlar aynı zamanda otonom robotun önceden belirlenmiş yörünge ve rotaları başarı ile takip ettiğini göstermiştir.Çalışmalarda deneysel ve teorik sonuçlar önerilen hareketli robot prototipinin ve kontrol yöntemlerinin başarısı seviyesini göstermektedir.

2. MATERYAL ve YÖNTEM

2.1 Materyal

Bu bölümde İKA'nın CAD programı ile hazırlanmış tasarımı ile hareketli robotun donanımsal özellikleri ve teknolojileri sunulmuştur. Bunların ardından aracın kinematik modeli, basitleştirilmiş şekilede aktarılmıştır. Sonrasında da Yörünge-Takibi ve Rota-Takibi için önerilen yöntemler önerilmiş ve bunların performansları literatürdeki diğer kontrol yöntemleri ile kıyaslanmıştır.

Bir hareketli robot üç ana ve temel kısımdan oluşmaktadır.

A- Mekanik kısım: Bu çalışmada önceden belirlenen bir yörünge ya da rotayı takip etme kabiliyetine sahip kendinden tahrikli bir yer aracı (Otonom İnsansız Kara Aracı: OİKA) tasarlanmıştır. Tasarımda, DC-motor, tekerlekler gibi bazı parçalar doğrudan satın alınırken, şasi, eklem parçaları, aracın kaplaması gibi bazı kısımları ise ihtiyaçlar göz önünde bulundurularak üretilmiştir. Ardından mekanik ve elektronik parçalar da yerleştirilerek araç tamamlanmıştır. Şasinin tamamı kaynaklı eklemlerden oluşmakta olup, geneli 40 mm çaplı olan değişik profil çaplarındaki borulardan imal edilmiştir. Bu hareketli robot dört tekerleğe sahiptir. Arka ve ön tekerleklerin çapı eşit olup, tümü 0.48m çaptadır. Aynı dingildeki tekerlekler arasındaki mesafe 0.93 metre iken, ön ve arka dingil arasındaki mesafe ise 1.10 metredir. Tekerlekler gövdeye spiral yaylı ve titreşim önleyici şok emici sistemlere sahip amortisörler ile bağlanmış, bu sayede hareket esnasında keskin virajlarda bile yol tutuş ve yol kaynaklı sarsıntıların emilimi sağlanmıştır. Süspansiyon sistemi yalnızca aracın dikey hareketinin düzenlenmesini sağlamakla kalmaz, aynı zamanda dönüş esnasında ya da hızlanma ve yavaşlama anlarında tekerleklerin düz bir çizgi halinde yönelimlerini korumalarını da sağlamaktadır.

Arka tekerlekler araca ayrı ayrı bağlanmıştır (birlikte dönmezler) ve saat yönü veya tersine dönerek araca ileri veya geri tahrik sağlamaktadırlar. Bu maksatla tekerleklerde gereken dönme kuvveti her bir arka tekere takılı olan DC motorlar ile sağlanmıştır. Bununla beraber robotun tahrik sisteminin diferansiyel özelliği yoktur. Bu nedenle arkadaki sağ ve sol tekerleğin dönüş esnasında yanal kaymaya maruz kalmadan dönebilmeleri için, her birine değişik açılardaki dönüş süreçlerindeki tekerlek hızları için gerekli dönme hızını kontrol eden kinematik hesaplamalar doğrultusunda voltaj uygulanmaktadır. Öte yandan ön tekerlekler sağa ve sola belirli açılarla dönebilecek ve araca hareket yönünü belirleme imkanı verecek şekilde tasarlanmıştır. Dönme sistemi, kramayer ve pinyonlu bir sistemle DC motorunun dairesel hareketini doğrusal harekete çevirmek suretiyle çalışmaktadır. Bir enkoder ön DC motorunun miline bağlanarak tekerleklerin dairesel hızını ve aynı zamanda dönüş açılarını okuması sağlanmıştır. Motor sürücüleri korumak maksadıyla dönme açısı $\pm 31^{o}$ ile sınırlandırılmıştır. Bunu sağlamak için ilgili dönüş limit açısına ulaşıldığında DC motora giden voltajı kesen ikişer anahtar yerleştirilmiştir.

B- Elektronik kısım: Bu kısımda çalışmada kullanılan elektronik cihazlar açıklanmıştır. Bu cihazlar DC motorlar, DC motor sürücüleri, Arduino mikro-kontrol cihazı, batarya ve elektrik bağlantılardan oluşmaktadır.

Bu çalışmada kullanılan tasarım indirgeme oranı 1:20 olan üç adet DC motor kullanmaktadır. Bu motorlar KORMAS firması tarafından üretilmiş olup, her biri 24 DC volt, 300 Watt, 1200 RPM ve azami 13A akım kapasitesi özelliklerine sahiptir. Bu motorlardan ikisi arka tekerleklere yerleştirilerek aracı hareket ettiren tahrik gücü bunlardan sağlanmıştır. İtme gücü arka tekerleklerde eşit olarak dağıtılmaktadır. Mikro-kontrol cihazı (Arduino MEGA2560) azami 5 V çıkış voltajına sahip olduğundan 24 V gibi çıktıyı elde edememektedir. Bu nedenle gerekli voltajın ve saat yönü ve saatin ters yönü sinyallerinin elde edilmesi için motor sürücü kartları kullanılmıştır. DC motor sürücülü üçüncü DC motor ise ön tekerleklere yerleştirilerek araca ihtiyaç duyduğu dönüş açılarını sağlaması olanağı verilmiştir. Bu DC motor sürücü karları DC motorları çalıştırmak ve bunların mikro-kontrol birimi ile bağlantısını kurmak için gereklidirler. Bu cihazlara kullanarak araca istenilen hız ve yönelim kolaylıkla kazandırılabilmektedir. Mikro-kontrol cihazı ve diğer işlemciler bu motorların ihtiyaç duydukları ağır voltajları kendi başlarına karşılayamamaktadırlar. Bu DC motor sürücü kartları hem ön hem de arka motorların ihtiyaç duyacağı voltajların motorlara ulaştırılmasını sağlamaktadır. Bu motorların ihtiyaç duyduğu voltaj ve akımlar ise toplmamda 35.3 kg ağırlığındaki yeniden şarj edilebilir bataryalar ile sağlanmıştır; bu pillerin her birisi 12 volt ve 120 amper-saat güç sunmaktadır. Bunlar birbirlerine seri olarak bağlanmış ve 24 volt güç sunmaları sağlanmıştır.

C- Algılayıcılar (Sensörler):İnsansız kara taşıtının Yörünge-Takip ve Rota-Takip özelliklerinin gerektirdiği konumlandırma ve yönlendirme kontrolleri için, aracın gerçek konum ve yöneliminin senkronize şekilde bilinmesi gerekmektedir. GPS, IMU, manyetometre ve enkoder gibi sensörler insansız aracın gerçek konum ve yöneliminin ölçülmesi için kullanılmıştır.

GPS mutlak konum koordinatlarını uzun bir süreç içerisinde doğru olarak verirken IMU kısa süreli işlemlerde ivme ve açısal hız bilgisini anlık olarak doğru bir şekilde vermektedir. Bir manyetometre doğrudan Arduino'ya bağlanarak aracın dünya koordinatları eksenine olan açısal durumunu ölçmesi sağlanmıştır. Enkoder ise aracın ön tekerleğinin dümenleme açısını vermektedir. Bu sensör bilgilerinden gelen verile belli bir sistematiğe göre birleştirilerek aracın gerçek konumu ve yönelimi elde edilmektedir. Aracın ölçme sisteminden elde edilen aracın gerçek konumunu tespit etme kabiliyetinin çok yüksek olduğu pratikteki bu ölçüm cihazları üzerine yapılan gerçek zamanlı deneylerle de gözlemlenmiştir. Algılayıcılar, tahrik elemanları ve kontrol yazılımı ile ilgili tüm haberleşmeler bilgisayar üzerinden gerçekleşmektedir.

2.2. Yöntem

Bu tezde otonom bir insansız kara aracının konumsal kontrolleri klasik PD kontrol sistemi üzerinde yapılan az bir değişiklikleelde edilen PD kontrolcü kullanılarak Yörünge-Takibi ve Rota-Takibi uygulamaları için yapılmıştır. Tasarlanan kontrol sisteminin performansının değerlendirilmesi için, önerilen kontrol algoritması ile elde edilen sonuçlar başka bir kontrol yöntemlerinin sonuçları ile karşılaştırılmıştır. Karşılaştırma için literatürde Yörünge Takibi için kullanılan Kayma Modu Kontrol sistemi seçilmiştir. Kontrol sistemi, sekiz şekli, keskin dönüşlü düz çizgiler, ve düzensiz yollar gibi değişik yörünge şekilleri ile test edilmiştir. Düzensiz yollar aynı zamanda rota-takibi yönteminde de denenmiştir. Bu düzensiz yol AutoCAD programındaki Spline komutu kullanılarak oluşturulmuştur. Spline komutu, bir dizi kontrol noktası üzerinden geçen ve düzenli olmayan B-Spline çizimi için kullanılmaktadır. Kontrol noktaları aynı zamanda kontrol çerçevesi üzerindeki bir dizi köşe olarak da tanımlanabilir. Ardından, elde edilen yörünge AutoCAD programından MATLAB’a aktarılmaktadır. Gerçek zamanlı kullanımda ise rota veya yörünge Google Earth haritalarından elde edilmektedir. Bu çalışmada önerilen PD kontrol sistemi ve literatürden alınan SMC sistemi kullanılarak Yörünge-Takibi ve Rota-Takibi uygulamaları denenmiştir. Yörünge, hareket halindeki nesnelerin konumlarının zamanın bir fonksiyonu olarak ifadesidir. Yörünge-Takibi zamanı bir kısıtlayıcı olarak almaktadır. Bunun anlamı, belirli bir zamanda belirli bir konumda bulunmak gerektiğidir. Yani araç, belirli bir zamanda belirlenmiş bir referans noktasında bulunmaya çalışmaktadır.

Bir araç için belirlenecek yörüngelerin daha düzgün ivmelenmelere uygun olması ve ani değişiklikler içermemesi gerekmektedir. Yatay ve boylamsal ivmelenme doğrusal hıza bağlıdır. Yani yörünge planlaması hem eğri planlaması hem de hız profilini göz önünde bulundurmalıdır. Yörünge takibinde karşılaştırma işlemi verilen tüm güzergah noktalarında tekrarlanır ve hatalar tespit edilerek (X_e, Y_e, θ_e), girdilere uygulanacak olan değişiklikler hesaplanır. Değiştirilecek girdiler PD kontrol sistemindeki katsayılar değiştirilerek elde edilebilir.Bu sayede araç rotasında kalır. Bu araştırmada PD katsayıları deneme yanılma yöntemleri ile elde edilmiştir. Ayrıca Y_e için bir limit konmuştur. Bunlar bizim PD üzerinde yaptığımız değişikliklerdir.

Kayma Modu Kontrol sistemi ise doğrusal olmayan bir kontrol sistemi süreci olup, doğrusal olmayan bir sistemin dinamiklerini, sistemin normal yöneliminin kesiti yönünde kaydıran ve devamlı olmayan bir kontrol sinyali kullanmaktadır. Kullanılan geri bildirim kontrol kuralı ise zamanın devamlı bir fonksiyonu şeklinde değildir. Onun yerine, devamlı yapıdan, mevcut konum ve durum uzayına bağlı olarak başka bir yapıya geçebilmektedir. Dolaysıyla, Kayma Modu Kontrolü değişken yapılı bir kontrol yöntemidir. Bu çalışmada kullanılan Kayma Modu Kontrolü literatürdeki iki çalışmadan alınmıştır(Solea, Razvan ve Urbano Nunes, 2007, ve Solea, R. Ve ark.,2009).Bu araştırmacılar iki kayan yüzey hazırlamışlardır: ilki X_e (boylam hataları) içinken, diğeri Y_e (yatay hatalar) ve θ_e (açısal hata) nın kombinasyonu içindir. Yörünge takibinde, aracın daha önceden belirlenen rotayı doğru şekilde takip edebilmesi için, X_e (boylam hataları) ve diğer hata kombinasyonlarının Y_e (yatay hata) ve θ_e (açısal hata)tüm hatalarının aynı anda kontrol edilmesi gerekmektedir.

Rota takibi ise, bir nesneyi (robot kolu, hareketli robot, gemi, hava aracı vs.) girilen bir rotada hareket ettirmek ve hedefe ulaştırmak konuları ile ilgilenen kontrol sistemlerinin tasarımı ile ilgilenmektedir. Rota takibinde, hareketli robotun rotaya yakınsaması yörünge-takibine göre daha isabetlidir. İzlenecek yol,yörüngetakibinde zamana bağlıyken, rota takibinde zamandan bağımsız parametrelerle verilmektedir. Rota takip sürecinde robot ya sabit bir hızla hareket eder ya da planlayıcıdan gelen verilerde belirtilen değişken hızları kullanır. Rota-takibi kullanımı, zamana bağlı hataların, konuma bağlı hatalardan daha az hassas olduğu durumlarda (yani konumsal hatalar, zamanında ulaşmaktan daha önemli olduğunda) daha sık kullanılmaktadır. Rota takibinde, aracın önceden belirlenmiş yolu doğru şekilde takip edebilmesi için, yalnızca θ_e (yönelim hatası) ve Y_e (yatay hatalar)'nin aynı anda kontrolü yeterlidir. Burada önerilen kontrol algoritması gerçek hareketli robot konumu ve istenen yoldaki en yakın referans noktasının konumuna bağımlı olarak çalışmaktadır. Bu karşılaştırma süreci araç hedefine ulaşana kadar devam eder. Direksiyon kontrolü için değer ve yön, $Y_e$ve θ_e 'nin her ikisine birden bağlı olarak belirlenmektedir.

Bu çalışmada, insansız kara aracının davranışlarının tanımlanmasında yalnızca kinematik bisiklet modeli kullanılmıştır. Modelde dinamik etkiler görmezden gelinmiştir, zira önerilen hareketli robothafiftir ve yavaş hareket etmektedir. Bisiklet modeli, arka ve ön tekerlekleri kendi aralarında bir araya getirerek dört tekerli modeli basitleştirmektedir. Bu basitleştirilmiş kinematik modelde aracın yatay dengesine etkileri önemsiz olduğundan aracın yatay boyutları görmezden gelinmiştir.

3- BULGULAR

Bu tez iki kısımdan oluşmaktadır. İlk kısımda insansız bir kara aracı tasarlanırken, ikinci kısımda bu aracın kinematik kontrolü Yörünge-takibi ve Rota-takibi sistemleri kullanılarak modellenmiştir. Bu çalışmada aracın kontrolü teorik olarak klasik PD ve SMC kullanılarak yapılmıştır. Bu robotun gerçek-zamanlı kontrolü gelecekteki başka bir çalışmaya bırakılmıştır.

Önerilen kontrol sisteminin aracın istenen güzergahta isabetle ilerlemesini sağlamakta yüksek seviyede başarılı olduğu görülmüştür. Kontrol sisteminin etkinliğini kanıtlamak amacıyla, elde edilen sonuçlar literatürden alınan Kayma Modu Kontrol sisteminin performansı ile karşılaştırılmıştır. Bu karşılaştırmalar önerilen sistemin konumsal hata, hız kısıtlaması ve dönüş açıları konularında daha başarılı olduğunu göstermektedir.

Robotun kontrolündeki en önemli değişken direksiyon açısıdır. Genel kontrolün başarısı büyük oranda bu değişkenin uygun kontrolüne bağlıdır. Karşılaştırma esnasında, direksiyon açısı konusunda önerdiğimiz sistemin SMC sistemine göre daha başarılı olduğu görülmektedir.Bunun ana nedeni SMC'de görülen dalgalanmalı direksiyon açısı kontrolleridir. Bu durum özellikle düzensiz rotalarda ortaya çıkmaktadır. Aracımızın azami dönüş açısı 31^0 derece olup kontrol birimi tasarlanırken bu durum da hesaplama algoritmalarına dahil edilmiştir. Keskin dönüşlü doğrusal yollarda bu yaklaşımın çok başarılı olduğu, SMC'nin bu rotalarla başa çıkmakta zorlandığı gözlenmiştir. SMC bu rotalarda 90 dereceden büyük açıları hesaplamalarında kullanamadığı için zorlanmaktadır. Sekiz şeklinde yolda ise, her iki yöntem için azami dönüş açısı $\pm 10.5^o$ .olduğundan benzer performans göstermektedirler.

Her iki tür kontrol sistemi için konum hataları sıfıra yaklaşmaktadır. Kimi zamanlarda konumsal hata 2 metreye kadar artabilse de, her iki yöntemin de ortalama hata payı kabul edilebilir düzeyde bulunmuştur.

Araç, azami doğrusal hızı 1.31 m/s olacak şekilde tasarlanmıştır; kontrol birimi tasarlanırken bu durum da göz önüne alınmıştır. Aracın doğrusal hızı, referans hıza neredeyse eşittir.

4. TARTIŞMA ve SONUÇ

Bu çalışma iki kısımdan oluşmaktadır. İlk kısımda prototip bir insansız kara aracı tasarlanmıştır. İkinci kısımda ise bu aracın kontrolü Yörünge-takip ve Rota-takip sistemleri ile sınamaya tabi tutulmuştur. Bu hareketli robot önce bir CAD programında tasarlamış ve ardından bu tasarıma uygun şekilde üretilmiştir. Araçta GPS (Global Positioning System), IMU (Inertial Measurement Unit), enkoder ve dijital pusula algılayıcılara sahiptir.Araç bunları kullanarak konumunu ve yönünü anlık olarak tespit edebilmektedir. Bunlara ek olarak, prototipte DC motorlar, DC motor sürücü kartları, mikro-kontrol çipleri ve kişisel bilgisayar da bulunmaktadır. Prototip tasarlanırken tüm parçalar birbirlerine uyumlu olacak şekilde ve özellikle Arduino'yu bir mikro-kontrol işlemcisi olarak kullanabilecek şekilde tasarlanmıştır. DC Motorlar ve bilgisayar arasındaki iletişim ve algılayıcılardan gelen bilgilerin okunması işleri ile DC motorların tahrik sağlamaları DC motor kontrol kartları ile yapılmaktadır.

İkinci kısımda ise tasarlanan ve üretilen araç önceden belirlenen rotalar üzerinde aracın konum ve hızı kontrol edilmek suretiyle kabul edilebilir hata payları çerçevesinde elde edilmiştir. Bu amaçla iki adet geri bildirim mekanizması araca eklenmiştir. İlki tarafımızca geliştirilen ve biraz modifiye edilmiş klasik PD kontrol mekanizması iken, ikincisi ise literatürdeki iki çalışmadan elde edilen SMC mekanizmasıdır.

Çalışmadan elde edilen sonuçlar aşağıdaki şekilde sıralanabilir.

- Araç dinamik modelin karmaşıklığına gerek olmadan düşük hızlarda kinematik model ile kontrol edilebilmektedir.
- Bu durum mekanik modelde hızlı hareket eden ve keskin dönüşler yapması gereken araçların kayma açısı konusunda tedbirler alması gerektiğini düşündürmektedir. Bunun elde edilmesi için sistemin karmaşıklığının artırılması, belki de dinamik bir model kullanılması gerekmektedir.

- Sonuçlardan yola çıkarak aracın sınırları dahilinde sabit hızla ilerlemenin istenen yörünge veya rotalara uymayı kolaylaştırdığı, değişken hızlı kontrollere göre daha isabetli sonuçlar elde edildiği gözlenmiştir.
- Önerilen kontrol sistemi kabul edilebilir seviyede küçük hatalara sahiptir ve 90 dereceyi aşan keskin dönüşlü yollarda da tercih sebebi olabilecek kadar başarılı görülmektedir. Zira SMC aynı tür yollarda başarı gösterememektedir. Buna ilaveten, SMC ile direksiyon açısında dalgalanmalar sık görülmektedir.

CURRICULUM VITAE

Name	Firas Muhammad Saib M.M. AL-NAQSHBANDI
Year of Birth	1974
Place of Birth	IRAQ - ERBIL
Sex	Male
Marital Status	Single
Nationality	IRAQI
Permanent Address	IRAQ – ERBIL –AZADI
Telephone	009647504728142
Email	firasmsn@yaoo.com firasmsn@gmail.com
Education	B.Sc. in Mechanical Engineering - University of Salahaddin - Erbil / College of Engineering IRAQ-Erbil in 1998.
Employment Record	Ministry Of Higher Education And Scientific Research. Location: IRAQ - Erbil Duration: 2002 to present

UNIVERSITY OF VAN YUZUNCU YIL
THE ISTITUTE OF NATURAL AND APPLIED SCIENCES
THESIS ORIGINALITY REPORT

Date: 20/07/2018...

Thesis Title: **THE DESIGN OF AN UNMANNED GROUND VEHICLE AND ITS TRAJECTORY-TRACKING CONTROL BY USING GPS**

The title of the mentioned thesis, above having total 121 pages with cover page, introduction, main parts and conclusion, has been checked for originality by TURNITIN computer program on the date of 20/07/2018 and its detected similar rate was 2% according to the following specified filtering.

Originality report rules:

- Excluding the Cover page,
- Excluding the Thanks,
- Excluding the Contents,
- Excluding the Symbols and Abbreviations,
- Excluding the Materials and Methods
- Excluding the Bibliography,
- Excluding the Citations,
- Excluding the publications obtained from the thesis,
- Excluding the text parts less than 7 words (Limit match size to 7 words)

I read the Thesis Originality Report Guidelines of VAN Yuzuncu Yil University for Obtaining and Using Similarity Rate for the thesis, and I declare the accuracy of the information I have given above and my thesis does not contain any plagiarism; otherwise I accept legal responsibility for any dispute arising in situations which are likely to be detected.

Sincerely yours.

20/07/2018

Date and signature

Name and Surname: Firas Muhammad Saib M.M. AL-NAQSHBANDI
Student ID#:149101142

Science: The Institute of Natural and Applied Science

Program: Mechanical Engineering.

Statute: M. Sc. ☒ Ph.D. ☐

APPROVAL OF SUPERVISOR
Assist. Prof. Dr. Atilla BAYRAM

APPROVAL OF THE INSTITUTE

Prof. Dr. Suat ŞENSOY
Enstitü Müdürü