Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System

Tamiru Takele Mola

A Thesis Submitted to

The Department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of

Master's in Electrical Power and Control Engineering

Office of Graduate Studies

Adama Science and Technology University

June, 2022.

Adama, Ethiopia

Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System

Tamiru Takele Mola

Advisor:

Prof. Sam Sun MA

A Thesis Submitted to

The Department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of

Master's in Electrical Power and Control Engineering

Office of Graduate Studies

Adama Science and Technology University

June, 2022.

Adama, Ethiopia

DECLARATION

I hereby declare that this Master Thesis entitled “Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System” is my original work. That is, it has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged though citation.

Name of student	Signature	Date

RECOMMENDATION

I, the advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled "Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System" prepared under my guidance by Tamiru Takele submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Power and Control Engineering, postgraduate in Control Engineering.

Therefore, I recommend the submission of revised version of the thesis to the department following the applicable procedures.

________________________	__________________	__________________
Name of Advisor	Signature	Date

APPROVAL SHEET

I, the advisors of the thesis entitled “Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System” and developed by Tamiru Takele, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

____________________ ____________________ ________________

Advisor Signature Date

We, the undersigned, members of the Board of Examiners of the thesis by Tamiru Takele have read and evaluated his thesis entitled “Design of Super Twisting Sliding Mode Controller for Trajectory Tracking Control of Autonomous Ground Vehicle System” and examined the candidate during open defense. This is, therefore, to certify that the thesis has been accepted in partial fulfillment of the requirement of the Degree of Masters in control engineering.

____________________________ _____________________ _________________

Chairperson Signature Date

______________________________ ______________________ __________________

Internal Examiner Signature Date

______________________________ ______________________ __________________

External Examiner Signature Date

Finally, approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

_________________________ ___________________ _____________

Department Head Signature Date

_________________________ ______________________ _______________

School Dean Signature Date

_________________________ ____________________ ____________

Office of Postgraduate Studies, Dean Signature Date

ACKNOWLEDGMENT

First and foremost, I would like to thank my Almighty God for his multiplied peace, health, strength and happiness through his only begotten Son Jesus Christ who saved the world through the cross work.

It gives me great pleasure to express the deepest gratitude to my adviser and mentor Prof. Sam Sun MA, for his heartfelt support and guidance. He has given me nice guidance, inspiration and suggestions throughout doing this thesis work.

Besides my advisor, my thanks are extended to my lovely parents, brothers and sisters for their prayer and best wishes to me all the time; and my friends who helped me more doing this work.

Table of Contents

DECLARATION i

RECOMMENDATION ii

APPROVAL SHEET iii

ACKNOWLEDGMENT iv

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF ACRONYMS xi

LIST OF SYMBOLS xii

ABSTRACT xv

CHAPTER ONE 1

INTRODUCTION 1

1.1 Background 1

1.2 Statement of the Problem 2

1.3 Objectives of the Study 3

1.4 Scope of the Study 3

1.5 Limitations of the Study 4

1.6 Motivation of the Study 4

1.7 Significance of the Study 4

1.8 Thesis Organization 4

CHAPTER TWO 6

LITERATURE REVIEW 6

2.1 Chapter Overview 6

2.2 Brief technical review of AGV 6

2.3 Sensors and their functionality in autonomous vehicle technology 7

2.4 Level of automation in AGV system 10
2.5 Related works 11
CHAPTER THREE 15
METHODOLOGY 15
3.1 Chapter Overview 15
3.2 Materials Used 15
3.3 Methods 15
3.4 Block diagram of proposed research work 16
3.5 Mathematical Modelling of an Autonomous Ground Vehicle System 17
3.6 Actuation Model 27
3.7 System Controller Design 28
3.8 The Proposed Optimization Techniques 37
3.9 Objective Functions 39
3.10 Overall System Structure 40
CHAPTER FOUR 41
RESULTS AND DISCUSSIONS 41
4.1 Chapter Overview 41
4.2 Simulink Modelling 41
4.3 Parameter settings for the proposed STSMC controller 42
4.4 Open Loop Response 45
4.5 Platform testing for closed loop responses 46
CHAPTER FIVE 52
CONCLUSIONS AND RECOMMENDATIONS 52
5.1 Conclusions 52
5.2 Recommendations 52

REFERENCES 53
APPENDIXES.. 59
Appendix A: AGV System Simulink Models 59
Appendix B: Proposed STSMC Controller in Simulink 60
Appendix C: GA Code for FOPID Controllers 61
Appendix D: GA Code for Proposed STSMC 63

LIST OF TABLES

Table 2:1 Summary of some literature reviewed 14
Table 4:1 Autonomous Ground Vehicle parameters used for simulation 44
Table 4:2 GA tuned controller parameters 44
Table 4:3 Boundary of tuning parameters..45
Table 4:4 the error performance analysis of the trajectory $\theta_d = 5\sin(t)$ 49
Table 4:5 System performance measure comparisons 51

LIST OF FIGURES

Figure 2:1 Different sensors on AGV system technology[7] 8
Figure 2:2 Principle of vision sensor system[8] 10
Figure 3:1 Block diagram of the research methodology 16
Figure 3:2 Block diagram of the proposed research work 17
Figure 3:3 Kinematics of lateral vehicle motion 18
Figure 3:4 Two degree of freedom lateral dynamics.. 21
Figure 3:5 Equivalent bicycle model 22
Figure 3:6 Phase portrait of a sliding motion in sliding mode control 29
Figure 3:7 Trajectory of the super-twisting controller. 31
Figure 3:8 Generalized FOPID controller [17] 36
Figure 3:9 the block diagram of fractional order controller 37
Figure 3:10 Overall system structure 40
Figure 4:1 The Simulink model of the proposed system 42
Figure 4:2 Convergence curve of proposed GA 43
Figure 4:3 Convergence curve of proposed PSO 43
Figure 4:4 Open loop response of the proposed system 45
Figure 4:5 the orientation angle and velocity response for a sinusoidal trajectory $\theta_d = 5\sin(t)$ proposed system 46
Figure 4:6 Orientation angle and velocity error for sinusoidal trajectory $\theta_d = 5\sin(t)$ of proposed system 47
Figure 4:7 Orientation angle and velocity error for sinusoidal trajectory $\theta_d = 5\sin(t)$ of FOPID controller 47
Figure 4:8 Orientation angle error for sinusoidal trajectory $\theta_d = 5\sin(t)$ of FOPID controller 48
Figure 4:9 the orientation angle and velocity response for linear trajectory $\theta_d = \pi / 8$ proposed system 49
Figure 4:10 the orientation angle and velocity error response for linear trajectory $\theta_d = \pi / 8$ of proposed system 50

Figure 4:11 the orientation and velocity response for linear trajectory $\theta_d = \pi / 8$ of FOPID controller......50
Figure 4:12 the orientation and velocity error response for linear trajectory $\theta_d = \pi / 8$ of FOPID controller......51

LIST OF ACRONYMS

ACC	Adaptive Cruise Control
AGV	Autonomous Ground Vehicle
ASTM	Adaptive Super Twisting Sliding Mode Controller
CACC	Cooperative Adaptive Cruise Control
CTVE	Center of Excellence for Transport and Vehicle Engineering
FOPID	Fractional Order Proportional Integral Derivative Controller
FOSMC	First Order Sliding Mode Controller
GA	Genetic Algorithm
HOSMC	Higher Order Sliding Mode Controller
ITAE	Integral Time Absolute Error
PID	Proportional Integral Derivative Controller
PSO	Particle Swarm Optimization
SAE	Society of Automotive Engineers
SMC	Sliding Mode Controller
STSMC	Super Twisting Sliding Mode Controller
UAV	Unmanned Aerial Vehicles
UUV	Unmanned Underwater Vehicle

LIST OF SYMBOLS

e_θ	Orientation error
e_v	Velocity error
U1	Control input one
U2	Control input two
T1	Right motor torque
T2	Left motor torque
L	Wheel base of the vehicle
l_f	Front wheel distance of the vehicle
l_r	Rear wheel distance of the vehicle
δ	Wheel steering angle
$\dot{\psi}$	Angular velocity
M	mass
β	Wheel slip angle
F_{yf}	Front wheel lateral tire force
F_{yr}	Rear wheel lateral tire force
a_y	Inertial acceleration
V_x	Longitudinal Velocity
I_z	Z axis momentum
α_f	Front tire slip angle

α_r	Rear tire slip angle
$C_{\alpha r}$	Cornering stiffness of each front tire
$C_{\alpha f}$	Cornering stiffness of each rear tire
θ_{vf}	Front tire velocity angle
θ_{vr}	Rear tire velocity angle
$\dot{\psi}_d$	Desired orientation velocity
T_m	Motor torque
T_l	Load torque
ω	Angular speed of motor
k_b	Back emf
i_a	Motor current
J_m	The motor inertia
e_a	The back emf voltage
v_a	The applied voltage to the motor
S1	Sliding surface for controller one
S2	Sliding surface for controller two
k_p	Proportional gain constant
k_i	Integral gain constant
k_d	Derivative gain constant
x_{1d}	Desired orientation angle for controller one

x_{2d}	Desired orientation velocity for controller two
C1	Sliding constant for controller one
C2	Sliding constant for controller two
K1	Gain constant for STSMC1
K2	Gain constant for STSMC2
u_{eq1}	Equivalent controller one
u_{eq2}	Equivalent controller two
s^{λ}	Integral order controller
$s^{-\mu}$	Derivative order controller
J	Objective cost functions

ABSTRACT

Autonomous Ground Vehicle (AGV) system research has getting a lot of attention in recent decades, within benefit of rapid advances in artificial intelligence technology and rising demand for traffic safety and efficiency. Autonomous Ground Vehicle system is an unstable, a highly nonlinear and time varying system. One of the main technologies in AGV research is trajectory tracking control, which is basic task of AGV. In this research work trajectory tracking control, Super Twisting Sliding Mode (STSMC) controller is designed for Autonomous Ground Vehicle (AGV) system based on Genetic algorithms (GA) and Particle Swarm optimization (PSO) technique. To compare the designed controller GA-based Fractional Order Proportional Integral Derivative (FOPID) controller was designed. Based on the bicycle model, the lateral dynamics model of the vehicle may established, which is further expressed in a state space form. The effectiveness and validation of the proposed method has been verified through different patterns of reference paths using MATLAB–Simulink software package. Also, the robustness of the system may conducted by adding disturbances due to friction of wheels during the vehicle motion. The obtained results of STSMC controllers are assumed to show the advantage and the good tracking performance of the technique in terms of minimizing trajectory tracking error, PSO based STSMC has less $(\approx 0.0006\sec)$ *for orientation angle control and* $(\approx 0.0007\sec)$ *velocity control tracking error performance and the complement of the trajectory following over Fractional Order Proportional Integral Derivative (FOPID) controller. Finally PSO based STSMC controller was acceptable rather than GA based STSMC and GA and PSO based FOPID controller for trajectory tracking control of AGV system.*

Key Words: *Autonomous Ground Vehicle, Trajectory tracking, Super Twisting Sliding Mode Controller, Optimization algorithms.*

CHAPTER ONE

INTRODUCTION

1.1 Background

An Autonomous Ground Vehicle (AGV), also known as an intelligent vehicle, is a ground vehicle that can navigate in structured or unstructured environments without continuous human guidance. Depending on the context in which they operate, autonomous vehicle systems may differ. Flying and aerial vehicles, often known as unmanned aerial vehicles (UAV), are autonomous vehicles that operate above land at a higher height. There are also unmanned and autonomous vehicles operating below sea level which also known as unmanned underwater vehicle (UUV)(Amer et al., 2017). This study focus on the autonomous vehicles that are operating on the ground which also known as autonomous ground vehicle (AGV).

An AGV is able to detect and circumnavigate obstacle and finally reach a goal which can be a point, a region or a sequence of points such as a trajectory. Today's transportation systems are increasingly complex systems with some difficulty in ensuring the control and security of these systems the number of vehicles is growing exponentially and the accomplishment of simple tasks really becomes a defeat with risk for the human being. Autonomous vehicles can solve this problem and act in the place of human beings. greasy to their capacity mobile robots (Car like vehicles or autonomous vehicles) are able to perform many tasks in dangerous places where humans cannot enter, those sites where harmful gases or high temperature are present in a harsh environment to humans and to ensure the delivery of goods at long distances in risky roads with autonomous vehicles. We can save money by performing various routine tasks(Chen, 2007). The trajectory tracking problem of an autonomous ground vehicle (AGV) has taken a considerable attention over the past few years. With the rapid development of autonomous vehicles, autonomous vehicle control plays an important role in the vehicle autonomy research area. So that this goal is to ensure this means that it is necessary to upgrade and optimize autonomous vehicle controllers that solve complicated problems and tackle complicated in variable environments.

The traditional PID controller has been used to control the various industrial processes in the world. There are different methods for adjusting the PID controller parameters according to the variation in the state of the environment and the system among these best-known methods, frequently used in industrial applications, the Ziegler-Nichols method, the genetic algorithm GA, fuzzy logic controller, etc. The PSO optimization technique was another very fashionable method of tuning(Zamani et al., 2009).

Autonomous vehicle control includes longitudinal control and lateral control at the lower level of control. Longitudinal control focuses on the speed control such as the notable adaptive cruise control (ACC) or more advanced cooperative adaptive cruise control (CACC) while lateral control deals with the vehicle steering control and path following(Yousif et al., 2018). This thesis work focuses on investigating autonomous vehicle steering control methods to improve trajectory tracking following performance.

1.2 Statement of the Problem

Autonomous ground vehicles are an important key of industrial automation. Such vehicles could be utilized in different applications such as monitoring, transportation, military and many other potential applications such as in construction, agriculture or planetary exploration. The trajectory tracking problems is one of area of current research. A considerable attention has been paid in recent years to deal with such problems. The autonomous vehicle must be able to track and extract information from its reference trajectory using controllers. Thus, these tracked information are fed to plant and execute its mission within its environment without human intervention. The autonomous ground vehicle must use sensors to perceive its surroundings and plan the motion.

In recent years, with the rapid development of modern computing techniques, artificial intelligence techniques have been applied widely in solving trajectory tracking problems. From the literature survey, it can be clearly seen that several issues like path planning and trajectory tracking problems, low performance on controllers, not using good optimization algorithms have not been addressed yet. Therefore, in this paper, all aforementioned issues have been involved to create a new problem formulation for solving the trajectory tracking problem of AGV by using Sliding Mode Controller (SMC). SMC has good performance in the case of trajectory tracking and disturbance rejection capabilities. However, SMC has a drawback due to its internal chattering problems. This may lead to having different voltage and current distributions to the motors. So, the torques in each actuator will be different and generate an oscillations on the vehicle system. To mitigate this problem a Super Twisting Sliding Mode Controller (STSMC) is proposed to control the orientation and velocity of autonomous ground vehicle.

1.3 Objectives of the Study

1.3.1 General objectives

The general objective of this research work is to design the Super Twisting Sliding Mode controller for trajectory tracking control of Autonomous Ground Vehicle system.

1.3.2 Specific objectives

The specific objectives of this study are:

- To study the mathematical modelling of the Autonomous Ground Vehicle system..
- To control the orientation and velocity of the Autonomous Ground Vehicle system.
- To analyze the performance and robustness of Super Twisting Sliding Mode controller.
- To tune the designed STSMC controller parameters optimally using the GA and PSO optimization techniques.
- To compare the performance results of the proposed controller with FOPID controller from simulation works.
- To make relevant conclusions and recommendations based on the simulation result.

1.4 Scope of the Study

The scope of this research work was determining the mathematical model of the Autonomous Ground Vehicle dynamics and designing GA based STSMC, PSO based STSMC for trajectory tracking control of the orientation and velocity of the vehicle system in comparison with GA and PSO based FOPID controller using the MATLAB/Simulink.

1.5 Limitations of the Study

The research work is limited to simulating the system using MATLAB/Simulink and demonstrating how the system’s various components interact with one other. This is due to the difficulty in obtaining the components for the implementation of an actual system. It also necessitates the extra time and money spent on building the actual prototype implementation.

1.6 Motivation of the Study

Advancement and introduction of impressive technology in autonomous ground vehicle has developed new fields of applications for it. Today Autonomous Ground Vehicles are utilized for area mapping, scene perception, and manipulation in industrial, agricultural, natural and mining situations. Thus, the design of robust control algorithms takes an important role in designing these advanced technology systems.

1.7 Significance of the Study

Intelligent vehicle greatly improves the efficiency of transportation system, significantly reduce the traffic accident rate, and reduce the energy consumption and the exhaust emissions. Such vehicles could be utilized in different applications such as monitoring, transportation, military and many other potential applications such as in construction, agriculture or planetary exploration. Because of the unstable and highly nonlinear nature of the AVG system, designing an efficient controller is essential. So, this research work proposed the robust STSMC controller which stabilizes and solves the tracking performance of this intelligent vehicles.

1.8 Thesis Organization

The thesis is organized as follows:

Chapter one: Introduces the historical background of the AGVs. It also includes a statement of the problem, the detailed objective of the thesis, scope, limitation, motivation, and significance of the thesis.

Chapter two: This chapter discusses the theoretical background and related works related work on AGV system.

Chapter three: This chapter presents the methodology followed, drives the kinematic and dynamic mathematical models, designs the proposed controller and presents genetic algorithm and particle swarm optimization techniques.

Chapter four: Presents and discusses on the proposed results.

Chapter five: Describes the conclusions and recommendations.

CHAPTER TWO

LITERATURE REVIEW

2.1 Chapter Overview

This section provides an overview of previous efforts to design the controllers for Autonomous Ground Vehicles, the sensors and their functionality on AGV system, and the level of automation where the self-driving vehicles follows. The knowledge gained in reviewing different articles and journal papers are extremely useful in the preliminary design of controller for AGV. Recently, many researchers have developed various path tracking control methods considering the non-holonomic constraints of autonomous ground vehicles.

2.2 Brief technical review of AGV

During last decade, autonomous vehicle technology has been a key research and development area in automobile industry. Construction, mining, agricultural equipment, seaborne commercial boats, and unmanned aerial vehicles (UAV) all benefit directly from the innovations created in the automobile industry. Despite the heavy intensity of investment in technology development, it will still take a few decades before an entirely autonomous self-driving vehicle navigates itself through national highways and congested urban cities(Babak et al., 2017).

The main self-driving technology can be divided into four categories. These are sensing, perception, planning and control. Sensing refers to the capability of the autonomous vehicle to collect real-time data from a variety of sensors (i.e., cameras, ultrasonic sonars, LiDARs, and radars) connected with self-driving cars to extract useful information from the environment. The perception module develops the contextual understanding of an environment such as object detection, localization, lane markings, and road signboards based on the information provided by the sensing module. In the next step, the planning module optimizes the information received from the perception module to take decisions in terms of optimal motion planning which leads to the defined goal. Finally, the control module is responsible for the execution of instructions defined by the planning module such as longitudinal (speed, brake) and lateral (steering) motion control(Rasib et al., 2021).

2.3 Sensors and their functionality in autonomous vehicle technology

Different sensors and systems are used for navigation and control of the autonomous vehicle. Prominent sensors such as LIDAR, GPS, radar, vision, ultrasonic and inertial measurement unit, their working principle and usage are discussed in the following sections.

2.3.1 LIDAR

LIDAR (Light Detection and Ranging) transmits a beam of light pulse from a rotating mirror, part of that light will reflect back to the sensor to detect any non-absorbing object or surface. The target distance is calculated as speed of light times the measured time period at multiple angles. It scans the field of view in 3D with a finite spatial resolution and frequency. Using the calculated distance, the sensor constructs a 3D map of the world including the objects around it. LIDAR uses infrared (IR), visible or ultraviolet (UV) waves of the electromagnetic spectrum for different applications and comes in various 1D, 2D and 3D configurations. They are the second most valued sensors, after vision for object detection like a vehicle, pedestrian and obstacles. Their performance is poor in rain, snow, dust and foggy environments. A high-end LIDAR sensor can measure multiple distances per laser pulse which is helpful to see through dust, rain and mostly transparent surfaces such as glass windows and porous object like wire fences. To reduce the signal to noise ratio, a higher power laser generation is desired but in order to prevent damage to the human eye, a laser power of 905nm is used to achieve desired range with low duty cycle.

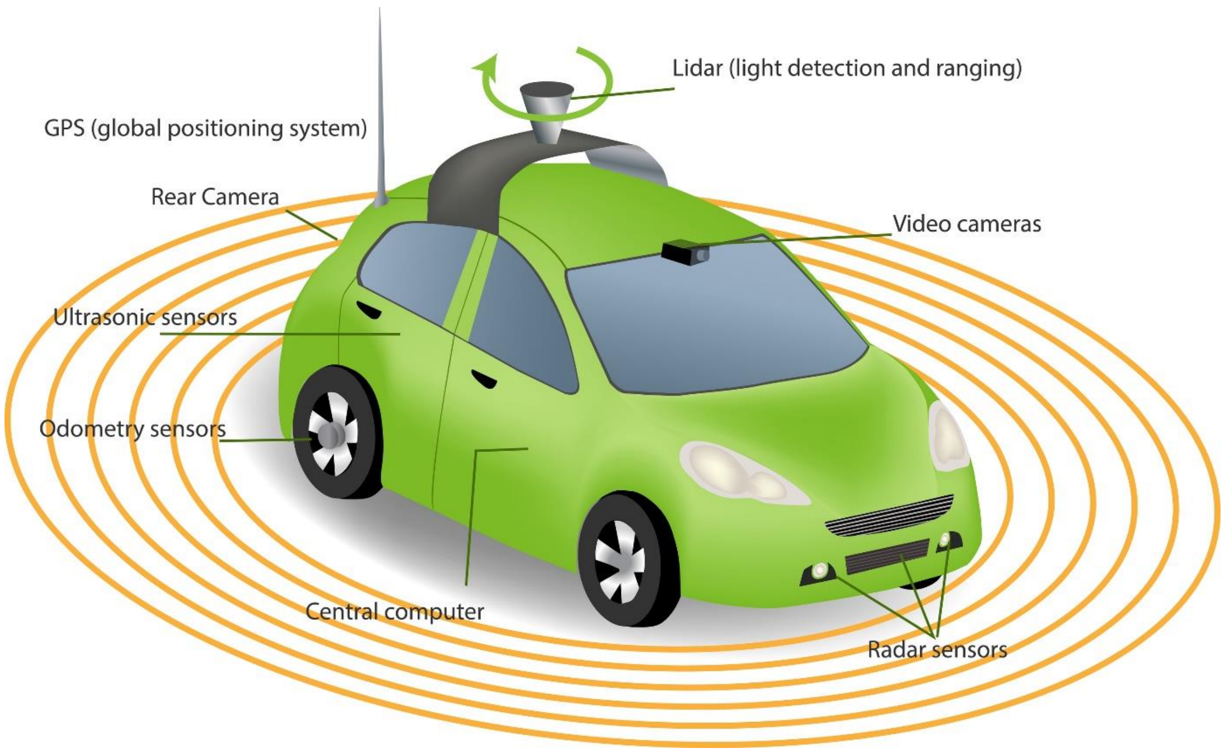


Figure 2:1 Different sensors on AGV system technology(*LIDR Is the Latest Game-Changing Advancement for Autonomous Vehicles - IEEE Innovation at Work*, n.d.)

2.3.2 Radar

Radar (Radio Detection and Ranging) transmits electromagnetic pulses and senses the echoes to detect and track objects. Echoes sensed vary in frequency depending on the speed of the object. Radar can measure the relative distance, velocity, and orientation of the object. In case of monostatic radars in which the transmitter and receiver are located at the same location, range of the target is measured by using the round trip travel time of a pulse, times the speed of light divided by two. They are typically available for short (≈ 30), medium (≈ 60) and long-range (≈ 200 m) distances and range from 3 MHz (Very long range) to 100+ GHz (Short range) frequencies. Radar requires less computing resources compared to vision or LIDAR. Typical application include lane keeping, advance cruise control, object detection.

2.3.3 GPS

Global positioning system (GPS) is used to locate the position of a GPS receiver (x,y,z coordinates) using four or more geostationary satellites. These satellites maintain their position relative to earth and broadcast reference signals. A GPS receiver on earth deciphers the actual location within meters accuracy. However, the so called "differential GPS" can pinpoint a location within centimeter accuracy which is necessary for navigation of autonomous vehicle. GPS consists of three segments: space, control, and user segments. The space segment includes the satellites, control segment manages and controls them and user segment is related to development of user equipment for both military and civil purposes. GPS based navigation application are greatly used to accurately predict the vehicle location with respect to map which is known as localization. However GPS based navigation are inaccurate and can lead to ghosting phenomena.

2.3.4 Vision

Vision system is composed of a camera and image processing unit (Fig 2.2). A typical camera is a combination of focusing lens and array of photo-detectors for each pixel in the field of view (FOV). The array of photo-detectors send pixel information to image processing unit. This unit processes the information based on certain algorithms to detect desired objects. Vision sensors capture more visual information, hence tracking the surrounding environment more effectively than other sensors. They are categorized into mono and stereo types. Mono camera systems are often used for lane marking, lane edge detection, basic object detection, road sign detection and localization. Multiple or Stereo camera systems provide depth for objects detection. Their primary advantage are their low-cost, off-the-shelf components and their software implementation. Their primary disadvantage is handling a full range of ambient and uncontrolled conditions such as lighting, shadowing, reflection, weather, dust, smoke. Also processing data from these sensors in real-time require a large amount of computational resources. Even with these limitations they are extensively used in autonomous vehicles.

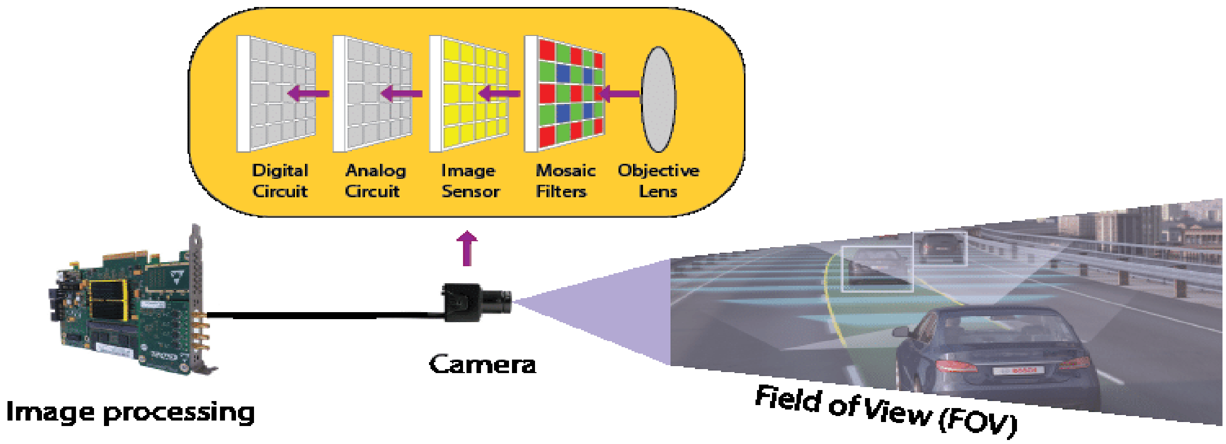


Figure 2:2 Principle of vision sensor system(*Principle of Vision System Figure 7. Inertial Measurement Unit 2.5.... | Download Scientific Diagram*, n.d.)

2.3.5 Inertial Measurement Unit (IMU)

A moving vehicle can experience linear and rotational motions along x, y, and z-axes: lateral, longitudinal, and vertical which can be measured by inertial measurement unit including linear and angular accelerations. This information is used to improve GPS measurements. IMU includes accelerometers and gyroscopes.

2.3.6 Ultrasonic Range

Ultrasonic sensors are short range sensors (typical ≈ 2 m) which send an ultrasonic pulse wave and detect echoes returned from the obstacles using transmitter-receiver pair. These are mainly used in relatively low speed ADAS modules like parking space detection and assistance and obstacle detection during congested traffic conditions. They are reliably detect under any weather conditions like rain, snow and winds. However they have range limitation which restricts them to be used as OEM vehicle sensors.

2.4 Level of automation in AGV system

The vehicle automation can be divided in to several levels as various organizations. The Society of Automotive Engineers (SAE) defines the six level of driving automation from 0 (fully manual) to 5 (fully autonomous) as shown below(Wang et al., 2020):

i. Level 0 (No Automation). All driving tasks are accomplished by the human operator.
ii. Level 1 (Driver Assistance). The human operator controls the vehicle, but driving is assisted with the automation system.
iii. Level 2 (Partial Driving Automation). Combined automated functions are applied in the vehicle, but the human operator still monitors the environment and controls the driving process.
iv. Level 3 (Conditional Driving Automation). The human operator must be prepared to operate the vehicle anytime when necessary.
v. Level 4 (High Driving Automation). The automation system is capable of driving automatically under given conditions, and the human driver may be able to operate the vehicle.
vi. Level 5 (Full Driving Automation). The automation system is capable of driving automatically under all conditions, and the human driver may be able to control the vehicle.

2.5 Related works

A kinematic controller of the mobile platform is derived to obtain a desired velocity by designing a Lyapunov function(Peng et al., 2014). In addition, a robust adaptive tracking controller was proposed. According to the Lyapunov stability theory, the derived robust adaptive controller guarantees global stability of the closed-loop system.

A new control approach is developed for trajectory tracking of mobile robots(Zhang et al., 2014). This control approach was designed based on a global finite-time control law of the angular velocity. The simulation results demonstrated a good convergence and performance to stabilize the angle error of mobile robots.

The authors in (Zhuo-Yun et al., 2020) have proposed an adaptive trajectory tracking method based on the kinematics model in order to solve the non-holonomic restriction problem of trajectory tracking of the wheeled mobile robot. This method is based on the idea of the artificial field to improve the effect of the trajectory tracking.

A classical PID controller is developed for path tracking problem of a mobile robot in (Normey-Rico et al., 2001). A new PID tuning method was proposed based on basic control tools, which takes into account the robustness of the closed-loop system.

The authors in reference (Lee & Chang, 2010) presented two optimization algorithms for optimizing a FOPID controller. These algorithms are an electromagnetism and evolutionary algorithm. Hence, a combination of the two algorithms was introduced to take the advantages of both algorithms and reduce the computation complexity of the electromagnetism algorithm.

Authors in (P. Zhao et al., 2012) presented and discussed the control system architecture of the autonomous vehicle, called “intelligent pioneer”, and the path tracking and stability of motion to effectively navigate in unknown environments based on adaptive PID control.

The experimental implementation of trajectory control of the mobile robots is presented in (García-Sánchez et al., 2017) considering the dynamics of their subsystems. It takes account of subsystem dynamics which helps modelling mechanism.

The Particle Swarm Optimization approach is used in (Maiti et al., 2008) to create fractional order PID controllers that outperform its integer order counterparts. Controller synthesis is based on required peak overshoot and rise time specifications. To get the best set of controller parameters, the characteristic equation is minimized.

In (Hoang & Pham, 2018) an Adaptive Super Twisting Sliding Mode (ASTSM) Control algorithm is applied to adjust attitudes of the quadcopter under harsh conditions such as nonlinear, strong coupling, high uncertainties and disturbances. It addresses the modelling and control aspects for quadcopter or drone unmanned aerial vehicles (UAVs). Simulation results show that the Adaptive Super Twisting Sliding Mode controller can achieve robust operation with disturbance rejection, parametric variation adaptation as well as chattering attenuation.

The PID Trajectory Tracking Control of Autonomous Ground Vehicle Based on Genetic Algorithm based on the bicycle model, the lateral dynamics model of the vehicle is established, is proposed in (B. Zhao et al., 2019), in which PID parameters are optimized by genetic algorithms. The simulation results show that the designed PID controller has good tracking performance even in the presence of external disturbance.

In (Dadras, 2017) Path Tracking Using Fractional Order Extremum Seeking Controller for Autonomous Ground Vehicle utilizes a Fractional Order Extremum Seeking Controller (FO-ESC) in order to control a non-holonomic autonomous ground vehicle to track the behavior of the predefined reference path.

In general, from the above literature reviewed and summary in table 2.1, most of their results track the behavior of the predefined reference path and trajectory with maximum tracking error for the Autonomous Ground Vehicle. Some of them checked the tracking performance even in the presence of external disturbance and they also used optimization technique. Even if, their work tracks the behavior of the predefined reference path, their tracking performance and error minimization technique is not such much optimized. Therefore, this research work is proposed to minimize the trajectory tracking error performance, and to control the orientation angle and velocity of the AGV using STSMC controller with genetic algorithm and particle swarm optimization techniques in comparison with fractional order PID controller.

Table 2:1 Summary of some literature reviewed

References	Techniques	Contributions	Strengths	Limitations
[12]	FOPID controller	It proposes a design method for a robust fractional-order PID (FOPID) controller for time-delay systems	Robust stability is analyzed for the Bode's model in terms of gain and phase margins to help the parameter tuning.	Its five parameters are reduced to one in the Bode shaping by data fitting between a parametric model and the real plant
[18]	Adaptive Super twisting SMC	It addresses the modelling and control aspects for quadcopter or drone	It ensures parametric variation adaptation	Adjusting and tuning attitude of quadcopter without optimization technique is hard
[19]	GA-based PID	It designs PID controller for trajectory tracking control for AGV systems	It uses GA optimization technique to tune the controller parameters	Its selection of objective function value is not clear
[20]	Fractional Order Extremum Seeking	It proposes path tracking control for autonomous vehicle using FO-ESC controller	It uses Extremum seeking control approach to optimize the control parameters	The model it was used is unicycle dynamics

CHAPTER THREE

METHODOLOGY

3.1 Chapter Overview

This chapter deals with the methods used to complete this research work and the materials used in the study.

3.2 Materials Used

In this research work, software such as MATLAB/2017a, Microsoft office 2016, and dia.io Diagrams and MathType 7.0 equation are used. MATLAB is the computing software which consists of technical toolboxes and Simulink. Microsoft office is used for editing the thesis documentation. MathType 7.0 is used for writing mathematical formulas and equations in Microsoft offices word and PowerPoint presentation tools. dia .io Diagrams is used to draw block diagrams and flowcharts.

3.3 Methods

In order to meet the above mentioned objectives of this research work, the following formal procedures was followed.

1. Literature Review: The work of this thesis work is necessarily based on what has already been done by other scholars related to this selected topic. So looking into the works of these scholars and getting necessary and important information related to the topic is essential. Different literatures were referred from different available sources such as books, papers, journals, internet, etc. Therefore the Literature review includes searching and reading books, articles, papers, journals, simulation tools from any sources that are related to my selected topic.
2. Based on the reviewed data, the system modelling and design was done using MATLAB/SIMULINK.
3. After analyzing the designed system, the whole work of the project was concluded by suggesting implied recommendation and further work to be done on the area.

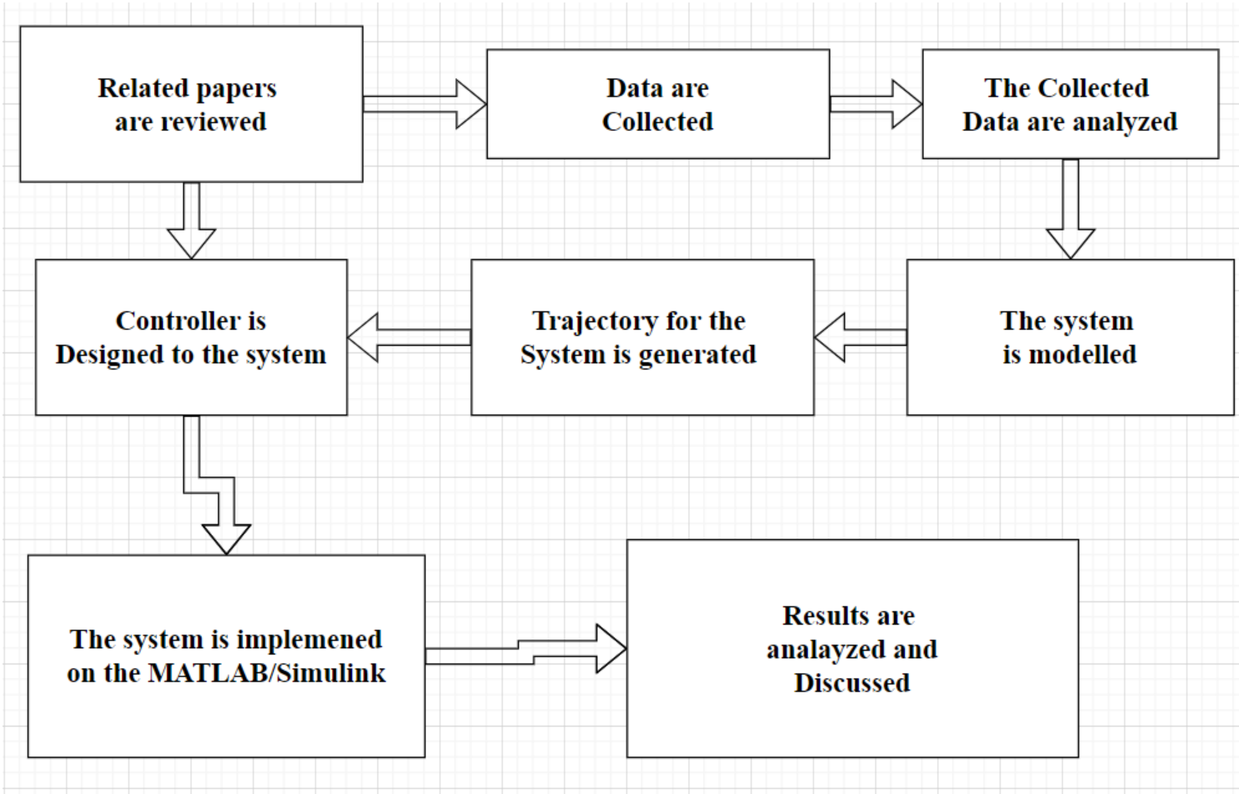


Figure 3:1 Block diagram of the research methodology

3.4 Block diagram of proposed research work

The block diagram of the control system for trajectory tracking of autonomous ground vehicle is depicted in Figure 3:2 below. Two STSMC controllers are used for driving the autonomous ground vehicle's wheels separately. The first controller receives the difference between the desired generated trajectory and actual instantaneous trajectory as an input. Therefore, the vehicle must change its orientation frequently to track the desired trajectory. The output is for controlling the right wheel. The second controller experiences the difference between the desired and actual velocity as input. The desired velocity was represented by a constant value. The outputs of both controllers are fed to the actuators of the autonomous ground vehicle for controlling the motion and tracking the given trajectories.

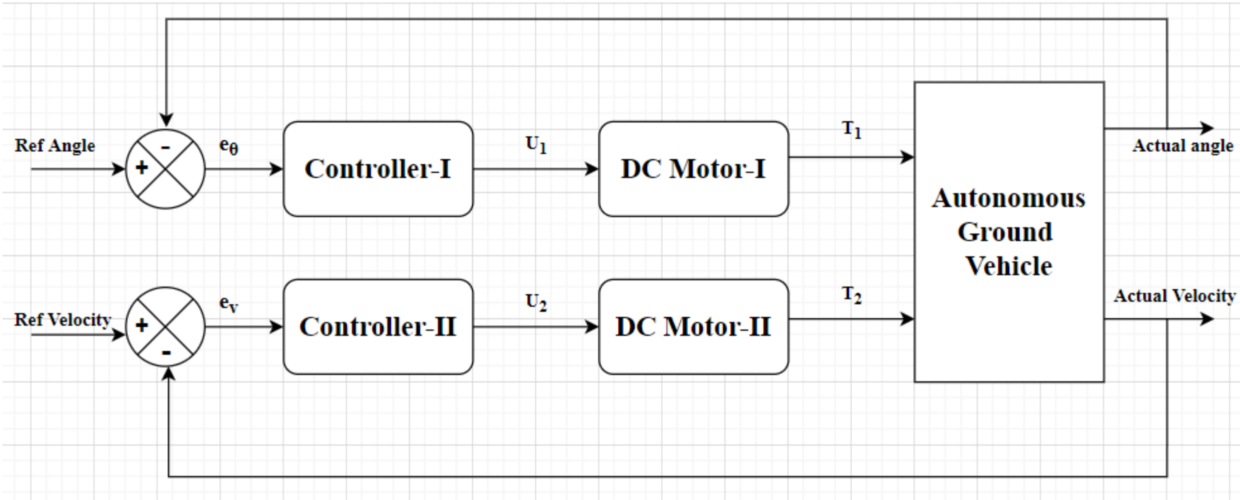


Figure 3:2 Block diagram of the proposed research work

3.5 Mathematical Modelling of an Autonomous Ground Vehicle System

Vehicle system development is complicated and requires knowledge from various areas, such as mechanical design, electronics development and control systems (Kamalakkannan, 2017). The following sections are involved on deriving the kinematic and dynamic models of autonomous vehicle system, since accurate modelling of the kinematic and/or dynamics behaviors of a vehicle, are required to achieve reasonable simulation results. Since, the research focuses on four wheel steering vehicles; where the front and the rear wheels are steered in opposite directions when the vehicle turns and the front wheels and the rear wheels will not share a common instantaneous center of rotation (ICR) due to the Ackermann mechanisms (J. S. Zhao et al., 2013). Instead, the vehicle turns about the ICR determined by the axis of two imaginary wheels, which is known as bicycle model.

3.5.1 Kinematic Model of Lateral Vehicle Motion

A kinematic model for vehicle's lateral motion can be developed using the assumptions described below. Without taking into account the forces that affect the motion, such a model provides a mathematical description of the vehicle motion. The equations of motion are solely dependent on system's geometric relationships.

Consider a bicycle model of the vehicle as shown in Figure 3.3 (Ling, 2006) below. In the bicycle model, the two left and right front wheels are represented by one single wheel at point

A. Similarly the rear wheels are represented by one central rear wheel at point B. The center of gravity (c.g.) of the vehicle is at point C. The distances of points A and B from the c.g. of the vehicle are ℓ_f and ℓ_r respectively.

The wheelbase of the vehicle is $L = \ell_f + \ell_r$.

The vehicle is assumed to have planar motion. Three coordinates are required to describe the motion of the vehicle: X, Y and ψ. (X, Y) are inertial coordinates of the location of the c.g. of the vehicle while ψ describes the orientation of the vehicle. The velocity at the c.g. of the vehicle is denoted by V and makes an angle β with the longitudinal axis of the vehicle. The angle β is called the slip angle of the vehicle. The front wheel steering angle is δ.

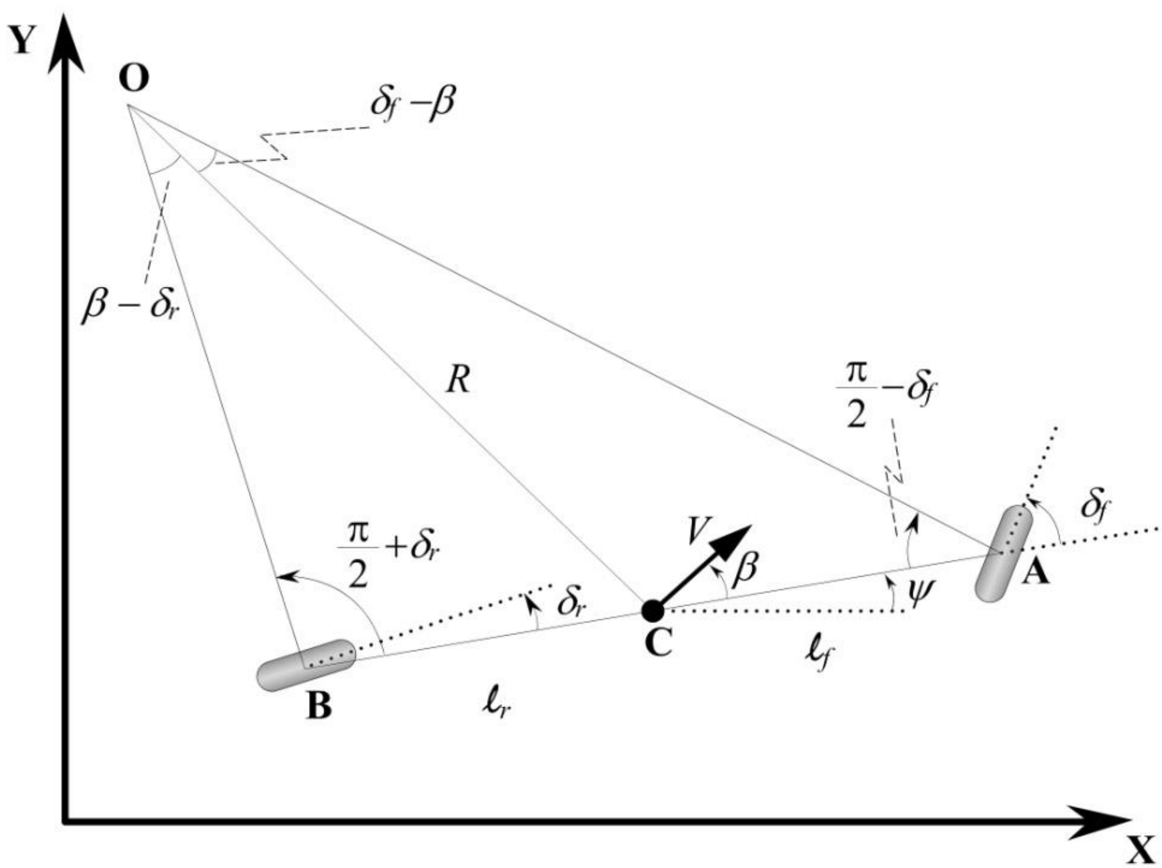


Figure 3:3 Kinematics of lateral vehicle motion

Assumptions

The major assumption used in the development of the kinematic model is that the velocity vectors at point A is in the direction of the orientation of the front wheels. In other words, the velocity vector at the front wheel makes an angle δ with the longitudinal axis of the vehicle.

At low speeds, the lateral force generated by the tires is small. In order to drive on any circular road of radius R, the total lateral force from both tires is $\frac{mV^2}{R}$ which varies quadratically with the speed and is small at low speeds.

The point O is the instantaneous rolling center for the vehicle. The point O is defined by the intersection of lines AO and BO which are drawn perpendicular to the orientation of the two rolling wheels.

The radius of the vehicle's path R is defined by the length of the line OC which connects the center of gravity C to the instantaneous rolling center O. The velocity at the c.g. is perpendicular to the line OC. The direction of the velocity at the c.g. with respect to the longitudinal axis of the vehicle is called the slip angle of the vehicle β. The angle ψ is called the heading angle of the vehicle. By applying some basic geometry and trigonometry;

$$\tan\delta = \frac{\ell_f + \ell_r}{\bar{R}} \quad (3.1)$$

$$\tan\beta = \frac{\ell_r}{\bar{R}} = \frac{\ell_r}{\ell_f + \ell_r}\tan\delta \quad (3.2)$$

$$\text{Where } \bar{R} = \frac{\ell_f + \ell_r}{\tan\delta} \quad (3.3)$$

$$\cos\beta = \frac{\bar{R}}{R} = \frac{1}{\ell_f + \ell_r}\tan\delta \quad (3.4)$$

If we assume that the radius of the vehicle path changes slowly due to low vehicle speed, then the rate of change of orientation of the vehicle (i.e. ψ) must be equal to the angular velocity of the vehicle. Since the angular velocity of the vehicle is $\frac{V}{R}$, it follows that

$$\dot{\psi} = \frac{V}{R} \tag{3.5}$$

Using Eq. (3.5), Eq. (3.4) can be re-written as

$$\dot{\psi} = \frac{V\cos(\beta)}{\ell_f + \ell_r} \tan\delta \tag{3.6}$$

The overall equations of motion are therefore given by

$$\dot{X} = V\cos(\psi + \beta) \tag{3.7}$$

$$\dot{Y} = V\sin(\psi + \beta) \tag{3.8}$$

$$\dot{\psi} = \frac{V}{\ell_f + \ell_r} \cos\beta \tan\delta \tag{3.9}$$

The Eq. (3.7), (3.8), (3.9) are nonlinear differential equations for the three states of vehicle namely X, Y and ψ , which has two inputs V and δ .

The slip angle β can be obtained as

$$\beta = \tan^{-1}\left(\frac{\ell_r}{\ell_f + \ell_r} \tan\delta\right) \tag{3.10}$$

The model equations (3.7), (3.8), (3.9) are referred to as vehicle's kinematic model since it describes the velocities but not forces or torques that have effect on the velocity. In the next section, the dynamic model will be analyzed.

3.5.2 Bicycle Model of Lateral Vehicle Dynamics

A bicycle model of the vehicle with two degrees of freedom is considered, as shown in figure 3.4 (Ling, 2006) and figure 3.5 (P. Zhao et al., 2012). The two degrees of freedom are represented by the vehicle lateral position y and the vehicle yaw angle ψ. The vehicle lateral position is measured along the lateral axis of the vehicle to the point O which is the center of rotation of the vehicle. The vehicle yaw angle is measured with respect to the global X axis. The longitudinal velocity of the vehicle at the center of gravity is denoted by V_x.

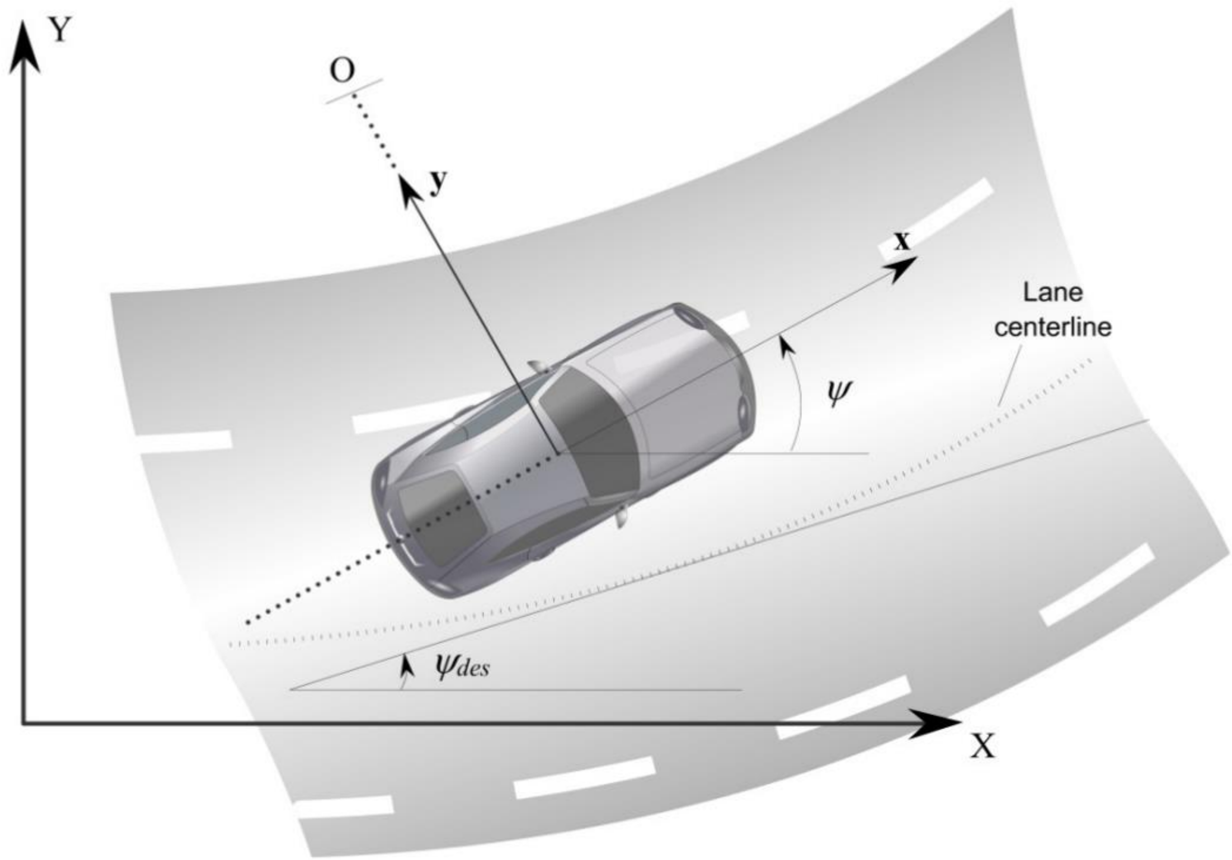


Figure 3:4 Two degree of freedom lateral vehicle dynamics

Applying Newton's second law for motion along the y axis

$$ma_y = F_{yf} + F_{yr} \tag{3.11}$$

Where $a_y = \left(\frac{d^2y}{dt^2}\right)$ is the inertial acceleration of the vehicle at the c.g. in the direction of the y axis F_{yf} and F_{yr} are the lateral tire forces of the front and rear wheels respectively.

Two terms contribute to a_y: the acceleration $\ddot{y}$ which is due to motion along the y axis and the centripetal acceleration $V_x\dot{\psi}$. Hence

$$a_y = \ddot{y} + V_x \dot{\psi} \tag{3.12}$$

Substituting from Eq. (3.12) into Eq. (3.11), the equation for the lateral translational motion of the vehicle is obtained as

$$m(\ddot{y} + V_x \dot{\psi}) = F_{yf} + F_{yr} \tag{3.13}$$

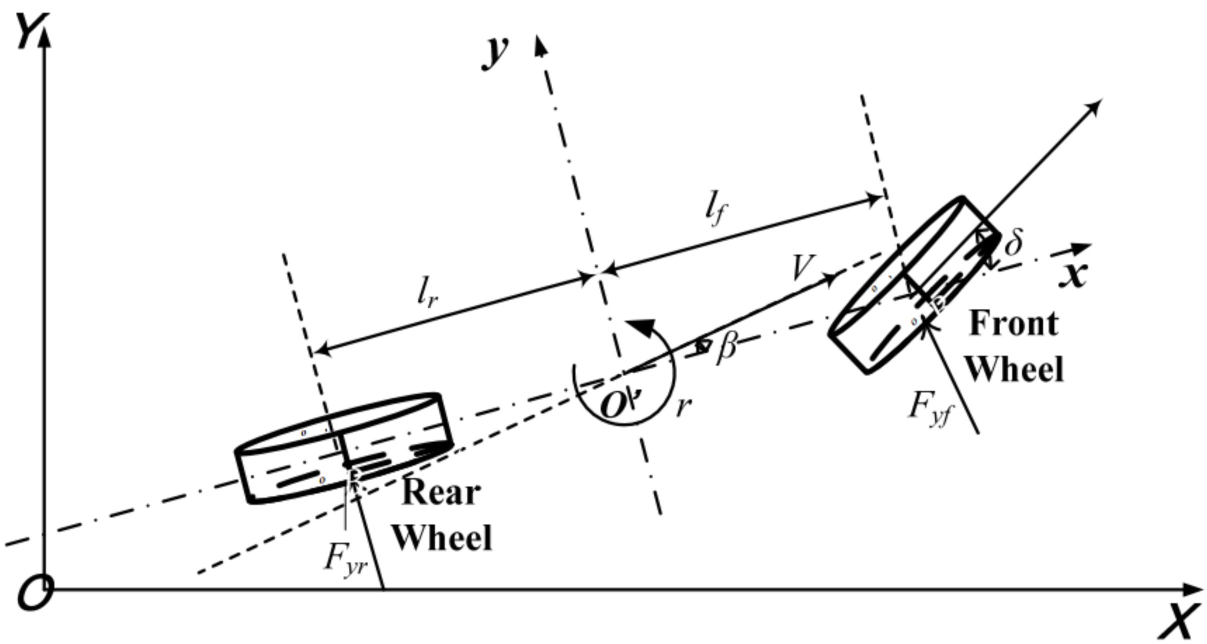


Figure 3:5 Equivalent bicycle model

Moment balance about the z axis yields the equation for the yaw dynamics as

$$I_z \ddot{\psi} = \ell_f F_{yf} - \ell_r F_{yr} \tag{3.14}$$

Where ℓ_f and ℓ_r are the distances of the front tire and the rear tire respectively from the c.g. of the vehicle.

To model the lateral tire forces F_{yf} and F_{yr} that act on the vehicle; the lateral tire force of a tire is proportional to the "slip-angle" for small slip-angles. The slip angle of a tire is defined as the angle between the orientation of the tire and the orientation of the velocity vector of the wheel. the slip angle of the front wheel is

$$\alpha_f = \delta - \theta_{vf} \tag{3.15}$$

Where θ_{vf} is the angle that the velocity vector makes with the longitudinal axis of the vehicle and δ is the front wheel steering angle, the rear slip angle is similarly given by

$$\alpha_r = -\theta_{vr} \tag{3.16}$$

The lateral tire force for the front wheels of the vehicle can therefore be written as

$$F_{yf} = 2C_{\alpha f}\left(\delta - \theta_{vf}\right) \tag{3.17}$$

Where the proportionality constant $C_{\alpha f}$ is called the cornering stiffness of each front tire, δ is the front wheel steering angle and θ_{vf} is the front tire velocity angle.

Similarly the lateral tire for the rear wheels can be written as

$$F_{yr} = 2C_{\alpha r}\left(-\theta_{vr}\right) \tag{3.18}$$

Where $C_{\alpha r}$ the cornering stiffness of each rear is tire and θ_{vr} is the rear tire velocity angle.

The following relations can be used to calculate θ_{vf} and θ_{vr}:

$$\tan\left(\theta_{vf}\right) = \frac{V_y + \ell_f \dot{\psi}}{V_x} \tag{3.19}$$

$$\tan\left(\theta_{vr}\right) = \frac{V_y - \ell_r \dot{\psi}}{V_x} \tag{3.20}$$

Using small angle approximation and using the notation $V_y = \dot{y}$

$$\theta_{vf} = \frac{\dot{y} + \ell_f \dot{\psi}}{V_x} \tag{3.21}$$

$$\theta_{vr} = \frac{\dot{y} - \ell_f \dot{\psi}}{V_x} \tag{3.22}$$

Substituting from Eqs. (3.15), (3.16), (3.21) and (3.22) into Eqs. (3.13) and (3.14), the state space model can be written as

$$\frac{d}{dt}\begin{bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{\alpha f} + 2C_{\alpha r}}{mV_x} & 0 & -V_x - \frac{2C_{\alpha f}\ell_f - 2C_{\alpha r}\ell_r}{mVx} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2\ell_f C_{\alpha f} - 2\ell_r C_{\alpha r}}{I_z V_x} & 0 & -\frac{2\ell_f{}^2 C_{\alpha f} + 2\ell_r{}^2 C_{\alpha r}}{I_z V_x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2\ell_f C_{\alpha f}}{I_z} \end{bmatrix}\delta \tag{3.23}$$

When the objective is to develop a steering control system for automatic lane keeping, which is a trajectory control, it is useful to utilize a dynamic model in which the state variables are in terms of position and orientation error with respect to the road.

Hence the lateral model developed in above equations will be re-defined in terms of the following state error variables:

x_1, the distance of the c.g. of the vehicle from the center line of the lane.

x_2, the orientation of the vehicle with respect to the road. Consider a vehicle traveling with constant longitudinal velocity V_x on a road of constant radius. Again, assume that the radius is large so that the same small angle assumptions as in the previous section can be made. Define the rate of change of the desired orientation of the vehicle as

$$\dot{\psi}_d = \frac{V_x}{R} \tag{3.24}$$

The desired acceleration of the vehicle can then be written as

$$\frac{{V_x}^2}{R} = V_x \dot{\psi}_d \tag{3.25}$$

Define $\ddot{x}_1$ and x_2 as follows

$$\ddot{x}_1 = \left(\ddot{y} + V_x \dot{\psi}\right) - \frac{{V_x}^2}{R} = \ddot{y} + V_x \left(\dot{\psi} - \dot{\psi}_d\right) \tag{3.26}$$

And

$$x_2 = \psi - \psi_d \tag{3.27}$$

Define

$$\dot{x}_1 = \dot{y} + V_x \left(\psi - \psi_d\right) \tag{3.28}$$

Eq. (3.28) is consistent with Eq. (3.26) if the velocity is constant. Hence the approach taken is to assume the longitudinal velocity is constant and obtain a LTI model.

Substituting from Eqs. (3.27) and (3.28) into (3.13) and (3.14), we find

$$m\ddot{x}_1 = \dot{x}_1\left[-\frac{2}{V_x}C_{\alpha f} - \frac{2}{V_x}C_{\alpha r}\right] + x_2\left[2C_{\alpha f} + 2C_{\alpha r}\right] + \dot{x}_2\left[-\frac{2C_{\alpha f}\ell_f}{V_x} + \frac{2C_{\alpha r}\ell_r}{V_x}\right] \\ +\dot{\psi}_d\left[-\frac{2C_{\alpha f}\ell_f}{V_x} + \frac{2C_{\alpha r}\ell_r}{V_x}\right] + 2C_{\alpha f}\delta \tag{3.29}$$

And

$$I_z\ddot{x}_2 = 2C_{\alpha f}\ell_f\delta + \dot{x}_1\left[-\frac{2C_{\alpha f}\ell_f}{V_x} + \frac{2C_{\alpha r}\ell_r}{V_x}\right] + x_2\left[2C_{\alpha f}\ell_f - 2C_{\alpha r}\ell_r\right] \\ +\dot{x}_2\left[-\frac{2C_{\alpha f}\ell_f{}^2}{V_x} - \frac{2C_{\alpha r}\ell_r{}^2}{V_x}\right] - I_z\ddot{\psi}_d + \dot{\psi}_d\left[-\frac{2C_{\alpha f}\ell_f{}^2}{V_x} - \frac{2C_{\alpha f}\ell_r{}^2}{V_x}\right] \tag{3.30}$$

The state space model in tracking error variables is therefore given by

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{\alpha f} + 2C_{\alpha r}}{mV_x} & \frac{2C_{\alpha f} + 2C_{\alpha r}}{m} & \frac{-2C_{\alpha f}\ell_f + 2C_{\alpha r}\ell_r}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2C_{\alpha f}\ell_f - 2C_{\alpha r}}{I_zV_x} & \frac{2C_{\alpha f}\ell_f - C_{\alpha r}\ell_r}{I_z} & -\frac{2C_{\alpha f}\ell_f{}^2 + 2C_{\alpha r}\ell_r{}^2}{I_zV_x} \end{bmatrix}\begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix} \\ + \begin{bmatrix} 0 & 0 \\ \frac{2C_{\alpha f}}{m} & -\frac{2C_{\alpha f}\ell_f - 2C_{\alpha r}\ell_r}{mV_x} - V_x \\ 0 & 0 \\ \frac{2C_{\alpha f}\ell_f}{I_z} & -\frac{2C_{\alpha f}\ell_f{}^2 + 2C_{\alpha r}\ell_r{}^2}{I_zV_x} \end{bmatrix}\begin{bmatrix} \delta \\ \dot{\psi}_d \end{bmatrix} \tag{3.31}$$

Note that the lateral dynamics model shown above is a function of the longitudinal vehicle speed which has been assumed to be constant.

3.5.3 State space equation for system Dynamics

The obtained lateral and yaw dynamics equations of the system can also represented in state space form. There are four states of the system dynamics in equation (3.31), and the state vector can be defined as;

$$x = \left[x_1 \;\; x_2 \;\; x_3 \;\; x_4\right]^T \tag{3.32}$$

The position errors and their rates are used for lateral dynamics equation and the orientation errors and their rates are used for yaw dynamics equation. Thus, the states can be written as:

$$x = \begin{bmatrix} x_1 & \dot{x}_1 & x_2 & \dot{x}_2 \end{bmatrix}^T \qquad (3.33)$$

In order to form the state space model of the system the control input vector should also be defined. Therefore control vector u can be represented as:

$$u = \begin{bmatrix} u_1 \ u_2 \end{bmatrix}^T \qquad (3.34)$$

Since the system has both steering angle, δ and velocity, $\dot{\psi}_d$ inputs, the control input state is written as:

$$u = \begin{bmatrix} \delta & \dot{\psi}_d \end{bmatrix}^T \qquad (3.35)$$

From equation (3.31), representing $\dot{x} = Ax + Bu$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & 0 & 1 \\ 0 & a_{42} & a_{43} & a_{44} \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & 0 \\ b_{41} & b_{42} \end{bmatrix} \qquad (3.36)$$

Therefore the final state space representation of the system becomes:

$$\dot{x}_1 = \dot{x}_1 \qquad (3.37)$$

$$\ddot{x}_1 = a_{22}\dot{x}_1 + a_{23}x_2 + a_{24}\dot{x}_2 + b_{21}\delta + b_{22}\dot{\psi}_d \qquad (3.38)$$

$$\dot{x}_2 = \dot{x}_2 \qquad (3.39)$$

$$\ddot{x}_2 = a_{42}\dot{x}_1 + a_{43}x_2 + a_{44}\dot{x}_2 + b_{41}\delta + b_{42}\dot{\psi}_d \qquad (3.40)$$

where, $a_{22} = \left[\dfrac{-2C_{af} + 2C_{ar}}{mV_x}\right]$, $a_{23} = \left[\dfrac{2C_{af} + 2C_{ar}}{m}\right]$, $a_{24} = \left[\dfrac{-2C_{af}l_f + 2C_{ar}l_r}{mV_x}\right]$

$a_{42} = \left[\dfrac{-2C_{af}l_f - 2C_{ar}}{I_z V_x}\right]$, $a_{43} = \left[\dfrac{2C_{af}l_f - C_{ar}l_r}{I_z}\right]$, $a_{44} = \left[\dfrac{-2C_{af}l_f{}^2 + 2C_{ar}l_r{}^2}{I_z V_x}\right]$

$$b_{21} = \left[\frac{2C_{af}}{m}\right], \; b_{22} = \left[\frac{-2C_{af} l_f - 2C_{ar} l_r}{mV_x} - V_x\right]$$

$$b_{41} = \left[\frac{2C_{af} l_f}{I_z}\right], \; b_{42} = \left[\frac{-2C_{af} l_f^2 + 2C_{ar} l_r^2}{I_z V_x}\right]$$

3.6 Actuation Model

An actuator is an electrical motor that drives a mechanical part of a robotic mechanism. The actuator receives a control signal directly from a control system to drive wheels into a specified motion. A DC motor is used as an actuator in this work. The model of DC motor is given in the following equations:

$$T_m = k_b I_a \qquad (3.41)$$

$$E_a = k_b \omega = k \frac{d\theta}{dt} \qquad (3.42)$$

Applying newton's law and Kirchhoff's law

$$J \frac{d^2\theta}{dt^2} + b \frac{d\theta}{dt} = k_b I_a \qquad (3.43)$$

$$L \frac{d\theta}{dt} + RI_a = V - k \frac{d\theta}{dt} \qquad (3.44)$$

Using Laplace transform we can get

$$[Js^2 + bs]\theta(s) = k_b I_a \qquad (3.43a)$$

$$LsI_a(s) + RI_a(s) = V(s) - k_b s\theta(s) \qquad (3.43b)$$

Where s is Laplace operator, from (3.43a) and (3.43b)

$$[Js^2 + bs]\theta(s) = k_b \left[\frac{V(s) - k_b s\theta(s)}{R + Ls}\right] \qquad (3.43c)$$

The open loop transfer function $G_\delta(s)$ and $G_V(s)$ for dc motor relating the input voltage and output orientation angle and velocity respectively.

$$G_\delta(s) = \frac{\theta(s)}{V(s)} = \frac{k_b}{[s(R + Ls)(Js + b) + k_b^2]} \qquad (3.44a)$$

$$G_V(s) = \frac{\omega(s)}{V(s)} = \frac{k_b}{[(R + Ls)(Js + b) + k_b^2]} \qquad (3.44b)$$

Where

ω=the angular speed of the motor,

k_b = back emf constant

T_m =The motor torque,

T_L =Load torque,

J =The motor inertia,

E_a =The back e.m.f. voltage,

I_a =The motor current,

V_a =The applied voltage to the motor.

3.7 System Controller Design

3.7.1 Sliding Mode control

Sliding mode control technique is perceived as a robust control tool for the complex higher order nonlinear systems under the effect of parametric uncertainties and external perturbations. Sliding mode operation was first derived in variable structure systems in the period of 1950s in former Soviet Union. In the early design stage, the variable structure system was unable to pay attention in the control era. Owing to the execution complexity particularly the chattering issues with sensors, actuators and switching systems. But there are some key advantages to this technique as it alters the dynamic behavior of system by selecting appropriate switching function which makes the closed loop response insensitive to matched uncertainty in the system. Such promising feature makes an attention to the researchers for contribution in the development of SMC. In recent development, SMC design approaches to the key issues like elimination of chattering, compensation of effect of unstructured dynamics, adaptability in uncertain system and enhancement in dynamic performance of closed loop system. The key advantage of selecting SMC controller over the linear control like PID, as SMC provides an existence of stability and robustness performance in many uncertain systems where PID fails in the case of uncertain environment. SMC is a robust nonlinear control scheme, with a powerful capability to reject disturbances and plant uncertainties. This control scheme works on the principle of continuously altering the configuration of the controller to keep the state variables on the sliding manifold(Alam et al., 2018; Gambhire et al., 2021).

SMC synthesis has two separated stages which is sliding surface design and control input design. It is established that the most crucial step in designing the SMC is the construction of the sliding surface which is expected to response the desired control specifications and performances. The states trajectories are forced to be reached and stayed on the sliding surface as depicted in Fig. below(Ghazali et al., 2011).

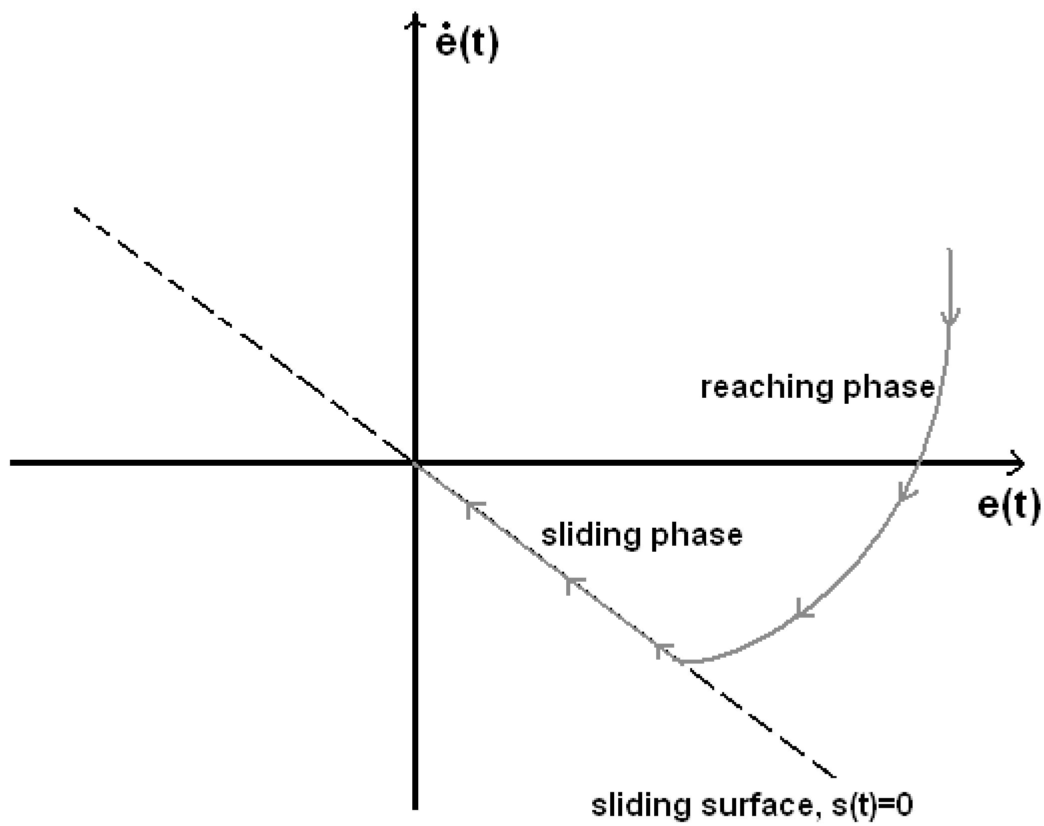


Figure 3:6 Phase portrait of a sliding motion in sliding mode control

The control signal in SMC design consists of equivalent control and switching control where the control action is corresponding with the sliding phase and reaching phase.

To design sliding mode control (SMC) we should choose the sliding surfaces. The surface may include the surface like PD, PI and PID.

As it is common Sliding Mode Theory (Utkin, 1992), we will consider the scalar discontinuous control action $u = u(s,t)$ at time t as

$$u(s,t) = -K(t)\,\mathrm{sign}(s) \tag{3.45}$$

$$\mathrm{sign}(s) := \begin{cases} 1 \;\text{ if } s > 0 \\ -1 \;\text{ if } s<0 \\ \in [-1\ 1] \text{ if } s=0 \end{cases} \tag{3.46}$$

3.7.2 Design of super twisting sliding mode control

As mentioned before, the main disadvantage of first-order sliding mode control (FOSMC) is a chattering problem. This effect consists of the oscillation of the system variables around the sliding surface, which causes a discontinuous control signal. This effect can disturb or damage the physical system. One of the interesting ways to eliminate or reduce chattering effect is using HOSM. In this technique, the higher order time derivatives of sliding surface keep to zero(Shtessel et al., 2014). As a result, this action mitigates the chattering effect. But, the increasing information demand is the main problem in the implementation of HOSM algorithms. Generally, r-th order sliding controller requires the knowledge of (r-1)-th order of time derivative. Thus, a r-th order sliding mode is determined by

$$s = \dot{s} = \ddot{s} = \ldots = s^{(r-1)} = 0 \tag{3.47}$$

Hence, the second-order sliding mode controller is simple. In fact, it needs the knowledge of the first-order time derivative of the sliding surface. These controllers require real-time measurement of $\dot{s}$ or at least of $\mathrm{sign}(\dot{s})$.

Among several second-order sliding mode controllers, the super-twisting controller has a good feature because it does not need the knowledge of $\dot{s}$. The super-twisting controller can be used instead of the first-order (conventional) sliding mode using the same available information. Figure 3.7 shows the phase portrait in the super-twisting controller(Nasiri et al., 2019).

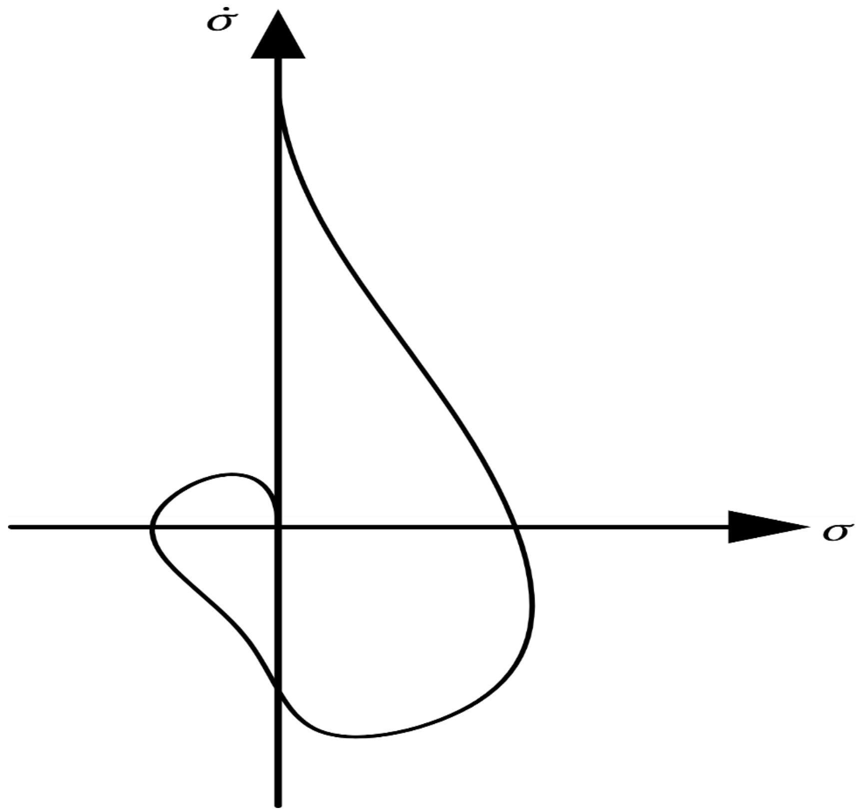


Figure 3:7 Trajectory of the super-twisting controller.

For this system controller design choosing the PD surface as:

$$S_1 = C_1 e_1 + \dot{e}_1 \tag{3.48}$$

Where C1 is a sliding constant and e1 and ė1 are the error and error derivative respectively. Since $u(t) = k_p e(t) + k_d \frac{d}{dt} e(t)$ for PD surface.

Let us define the steering angle tracking errors. The steering tracking error can be determined as a difference between the trajectory and actual angle of AGV.

$$e_1 = x_{1d} - x_1 \tag{3.49}$$

$$\dot{e}_1 = \dot{x}_{1d} - \dot{x}_1 \tag{3.50}$$

$$\ddot{e}_1 = \ddot{x}_{1d} - \ddot{x}_1 \tag{3.51}$$

In sliding mode phase, the tracking error will converge to equilibrium point and the sliding surface is supposed to be $s(t) = \dot{s}(t) = \ddot{s}(t) = 0$ where in this situation, the tracking error is trapped in the sliding surface. The equivalent control is obtained when $\ddot{s}(t) = 0$.

Differentiating the surface S1 of eq. (3.48)

$$\dot{S}_1 = C_1\dot{e}_1 + \ddot{e}_1 \tag{3.52}$$

Inserting eq. (3.51) and (3.50) in eq. (3.52)

$$\dot{S}_1 = C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - \ddot{x}_1\right) \tag{3.53}$$

Inserting eq. (3.35) in eq. (3.50)

$$\dot{S}_1 = C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2 - b_{21}\delta - b_{22}\psi_d\right) \tag{3.54}$$

To verify the stable convergence behavior of nonlinear controller, Lyapunov theorem as a well-known approach is used in stability analysis (Dai et al., 2019; Hu et al., 2021; Wu et al., 2019). Lyapunov function can be chosen to prove the stability as in Eq.(3.52) with its time derivative given in Eq.(3.53).

$$V = \frac{1}{2}s^2 \tag{3.55}$$

$$\dot{V} = s\dot{s} \tag{3.56}$$

With V=0 and V>0 for $s \neq 0$. To guarantee the trajectory move from reaching phase to sliding phase and ensure the stability, it is necessary to follow the reaching condition:

$$\dot{V} < 0,\ \text{for } s \neq 0,\ \dot{s} \neq 0 \tag{3.57}$$

For $S_1\dot{S}_1 \leq 0$ eq. (3.54) becomes

$$C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2 - b_{21}\delta - b_{22}\psi_d\right) = 0 \tag{3.58}$$

$$C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right) = b_{21}\delta + b_{22}\dot{\psi}_d \tag{3.59}$$

Letting $b_{22} = 0$ eq. (3.59) becomes

$$C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right) = b_{21}\delta \tag{3.60}$$

$$\delta = \frac{C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right)}{b_{21}} \tag{3.61}$$

Letting $b_{21} = 0$ eq. (3.59) becomes

$$C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right) = b_{22}\dot{\psi}_d \tag{3.62}$$

$$\dot{\psi}_d = \frac{C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right)}{b22} \quad (3.63)$$

Now letting $u_{eq1} = \delta + \dot{\psi}_d$ (3.64)

$$u_{eq1} = \left[\frac{b_{21} + b_{22}}{b_{21}b_{22}}\right]\left(C_1\left(\dot{x}_{1d} - \dot{x}_1\right) + \left(\ddot{x}_{1d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2\right)\right) \quad (3.65)$$

For second controller, choosing the sliding surface two

$$S_2 = C_2 e_2 + \dot{e}_2 \quad (3.66)$$

Let us also define the velocity tracking errors as

$$e_2 = x_{2d} - x_2 \quad (3.67)$$

$$\dot{e}_2 = \dot{x}_{2d} - \dot{x}_2 \quad (3.68)$$

$$\ddot{e}_2 = \ddot{x}_{2d} - \ddot{x}_2 \quad (3.69)$$

Differentiating the surface S2 of eq. (3.66)

$$\dot{S}_2 = C_2 \dot{e}_2 + \ddot{e}_2 \quad (3.70)$$

Inserting eq. (3.68) and (3.69) in eq. (3.70)

$$\dot{S}_2 = C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - \ddot{x}_2\right) \quad (3.71)$$

Inserting eq. (3.40) in eq. (3.71)

$$\dot{S}_2 = C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{42}\dot{x}_1 - a_{43}x_2 - a_{44}\dot{x}_2 - b_{41}\delta - b_{42}\dot{\psi}_d\right) \quad (3.72)$$

For $S_2\dot{S}_2 \le 0$ eq. (3.72) becomes

$$C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{22}\dot{x}_1 - a_{23}x_2 - a_{24}\dot{x}_2 - b_{41}\delta - b_{42}\dot{\psi}_d\right) = 0 \quad (3.73)$$

$$C_1\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{42}\dot{x}_1 - a_{43}x_2 - a_{44}\dot{x}_2\right) = b_{41}\delta + b_{42}\dot{\psi}_d \quad (3.74)$$

Letting the parameter $b_{42} = 0$ in eq. (3.74) becomes

$$C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{42}\dot{x}_1 - a_{43}x_2 - a_{44}\dot{x}_2\right) = b_{41}\delta \quad (3.75)$$

$$\delta = \frac{C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{42}\dot{x}_1 - a_{43}x_2 - a_{44}\dot{x}_2\right)}{b_{41}} \quad (3.76)$$

Letting the parameter $b_{41} = 0$ in eq. (3.74) becomes

$$C_2\left(\dot{x}_{2d}-\dot{x}_2\right)+\left(\ddot{x}_{2d}-a_{42}\dot{x}_1-a_{43}x_2-a_{44}\dot{x}_2\right)=b_{42}\dot{\psi}_d \tag{3.77}$$

$$\dot{\psi}_d=\frac{C_2\left(\dot{x}_{2d}-\dot{x}_2\right)+\left(\ddot{x}_{2d}-a_{42}\dot{x}_1-a_{43}x_2-a_{44}\dot{x}_2\right)}{b42} \tag{3.78}$$

Now letting $u_{eq2}=\delta+\dot{\psi}_d$ (3.79)

$$u_{eq2}=\left[\frac{b_{41}+b_{42}}{b_{41}b_{42}}\right]\left(C_2\left(\dot{x}_{2d}-\dot{x}_2\right)+\left(\ddot{x}_{2d}-a_{42}\dot{x}_1-a_{43}x_2-a_{44}\dot{x}_2\right)\right) \tag{3.80}$$

In SMC, the control law u is a combination of two types of control given by:

$$U=u_{eq}+u_c \tag{3.81}$$

Where u_{eq} is the equivalent controller used to control the system states during the sliding surface. Eq. (3.65) and (3.80) are used to obtain the equivalent control. Using control action input for STSMC as given by(Humaidi & Hasan, 2019; Ouchen et al., 2021; Y. Zhao et al., 2018);

$$u_c=-K_1\,|\sqrt{s}|\,\mathrm{sgn}(s)-K_2\int_0^t \mathrm{sgn}(s)dt \tag{3.82}$$

Therefore the controller for STSMC1 designed in order to control the orientation of the AGV system as:

$$U_1=u_{eq1}+u_c \tag{3.83}$$

Inserting eq. (3.65) and (3.82) in eq. (3.83) becomes,

$$U_1=\left[\frac{b_{21}+b_{22}}{b_{21}b_{22}}\right]\left(C_1\left(\dot{x}_{1d}-\dot{x}_1\right)+\left(\ddot{x}_{1d}-a_{22}\dot{x}_1-a_{23}x_2-a_{24}\dot{x}_2\right)\right)+\left(-K_1\,|\sqrt{s}|\,\mathrm{sgn}(s)-K_2\int_0^t \mathrm{sgn}(s)dt\right) \tag{3.84}$$

Therefore the controller for STSMC2 designed in order to control the velocity of the AGV system as:

$$U_2 = u_{eq2} + u_c \tag{3.85}$$

Inserting eq. (3.80) and (3.82) in eq. (3.85) becomes,

$$U_2 = \left[\frac{b_{41} + b_{42}}{b_{41}b_{42}}\right]\left(C_2\left(\dot{x}_{2d} - \dot{x}_2\right) + \left(\ddot{x}_{2d} - a_{42}\dot{x}_1 - a_{43}x_2 - a_{44}\dot{x}_2\right)\right) + \left(-K_1 |\sqrt{s}| \operatorname{sgn}(s) - K_2 \int_0^t \operatorname{sgn}(s)\, dt\right) \tag{3.86}$$

Where C1, C2, K1 and K2 are constant gain parameters.

3.7.3 Fractional order PID Controller

Fractional calculus is a mathematical topic which studies the ability of taking real number power of both the differential and integration operators. It is necessary to understand the theory of fractional calculus in order to realize the significance of fractional order integration and derivation. Fractional calculus is the branch of calculus that generalizes the derivative or integral of a function to non-integer order, allowing calculations such as deriving a function to 1/2 order. There are several definitions to describe the fractional derivative. The firmly established definitions are Grunwald–Letnikov definition and the Riemann–Liouville definition. The most frequently used definition in fractional-order calculus is the Riemann– Liouville definition, in which the fractional-order integration is defined as follows(Al-Mayyahi et al., 2016):

$$aD_t^{\alpha} = \begin{cases} \dfrac{d^{\alpha}}{dt^{a}} \alpha > o \\ \alpha = 0 \\ \int_{\alpha}^{t} (d\tau)^{-\alpha} \alpha < 0 \end{cases} \tag{3.87}$$

Where a and *t* are the limits of fractional differentiation, α is the calculus order.

The Laplace transform of the fractional derivative of *f*(*t*) is given by:

$$L\{D^{\alpha} f(t)\} = s^{\alpha} F(s) - [D^{\alpha-1} f(t)]_{t=0} \tag{3.88}$$

Where $F(s)$ is the Laplace transform of $f(t)$.

The Laplace transform of the fractional integral of $f(t)$ is given as :

$$L\{D^{-\alpha} f(t)\} = s^{-\alpha} F(s) \tag{3.89}$$

Fractional-order proportional-integral-derivative (FOPID) controllers have received considerable attention in the last years both from academic and industrial point of view. In fact, in principle, they provide more flexibility in the controller design, with respect to the standard PID controllers, because they have five parameters to select (instead of three). However, this also implies that the tuning of the controller can be much more complex. The integral–differential equation defining the control action of a fractional-order PID controller is given by

$$u(t) = k_p e(t) + k_i D_t^{-\lambda} e(t) + k_d D_t^{\mu} e(t) \tag{3.90}$$

The transfer function of FOPID is attained by taking Laplace transform as follows:

$$U(s) = k_p + k_i s^{-\lambda} + k_d s^{\mu} \tag{3.91}$$

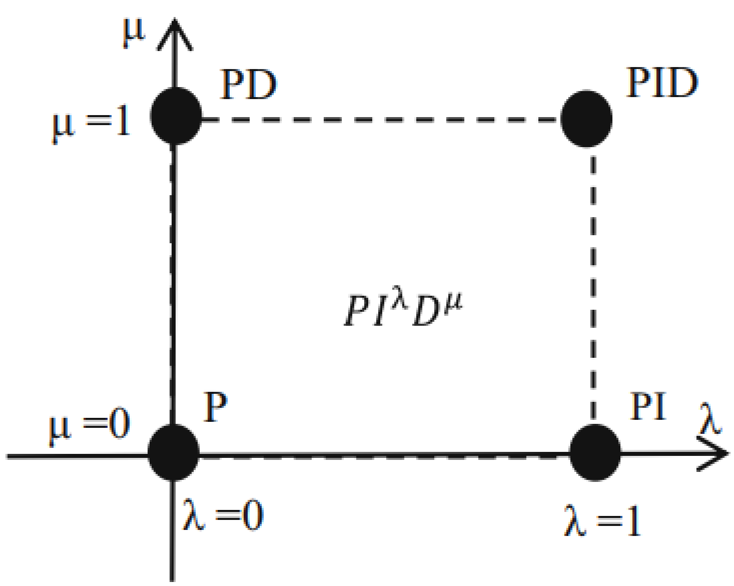


Figure 3:8 Generalized FOPID controller (Maiti et al., 2008)

The fractional order PID controller generalizes the integer order PID controller and expands it from point to plane as shown in above figure. This expansion adds more flexibility to controller design and we can control our real world processes more accurately.

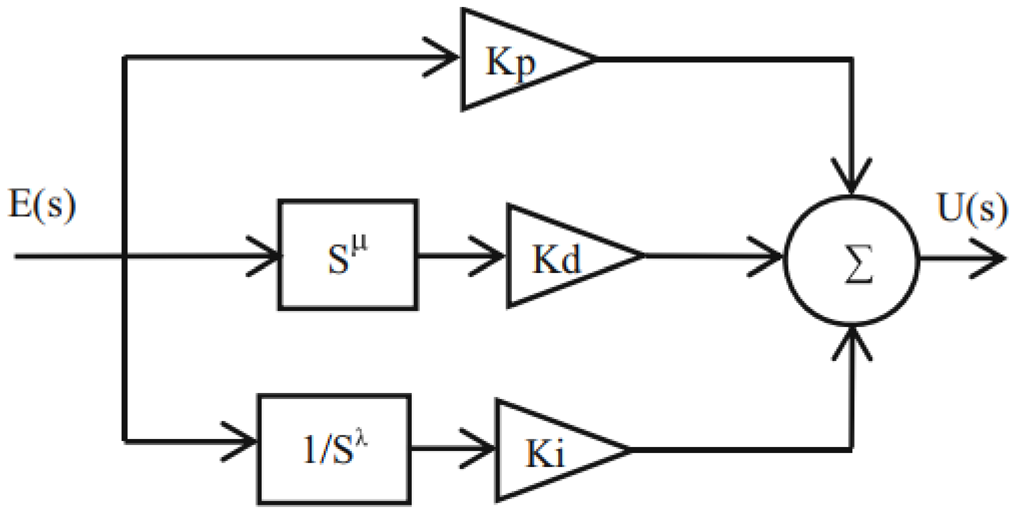


Figure 3:9 the block diagram of fractional order controller

In above figure 3.9 $E(s)$ represents Laplace transform of $e_1(t)$ and $e_2(t)$ which are orientation angle and velocity error respectively and $U(s)$ is U_1 and U_2 which are orientation angle and velocity controllers. Thus in addition to designing the parameters, k_p , k_i and k_d FOPID is used to design the orders λ, μ of integral and derivative controllers. The orders λ, μ can assume any real numbers, not essentially be integers. The FOPID generalizes the traditional integer order PID and extends it from point to plane (Cao & Cao, 2006).

3.8 The Proposed Optimization Techniques

Choosing suitable optimization techniques is one of criteria in robust controller design. In following section both Genetic algorithm and Particle Swarm optimization techniques which are used to optimize the controllers gain are briefly introduced.

3.8.1 Genetic Algorithms (GA)

Genetic Algorithm (GA) may be attributed as method for optimizing the search tool for difficult problems based on genetics selection principle. In additions to Optimization it also serves the purpose of machine learning and for Research and development. It is analogous to biology for chromosome generation with variables such as selection, crossover and mutation together constituting genetic operations which would be applicable on a random population initially.GA aims to yield solutions for the consecutive generations. Genetic algorithm is of great use in order to find out shorted path by application of path encoding in graph converting into chromosomes.

The extent of success in individual production is directly in proportion to fitness of solution which is represented by it, thereby ensuring that quality in successive generations will be better. The process is concluded once an GA is most suitable for the issues that need optimization associated with some computable system.. John Holland may be regarded as funding father of original genetic algorithm and is attributed to year 1970's as funding date .Additionally a random search method represented by Charles Darwin for a defined search space in order to effetely solve a problem(Deemyad et al., 2020; Lambora et al., 2019; Mirjalili et al., 2020).

The algorithm generates a "population" of possible solutions to the problem and allows them to "evolve" over time to find better and better ones. The algorithm ends when the population has attained a suitable fitness level or a maximum number of generations has been produced. If the method has finished due to the maximum number of generations, a good solution may or may not have been found(Farahmandrad et al., 2020).

3.8.2 Particle Swarm Optimization

Particle Swarm Optimization is an algorithm capable of optimizing a non-linear and multidimensional problem which usually reaches good solutions efficiently while requiring minimal parameterization (Clerc, 2010).

Over the ages, nature has constantly been a rich source of inspiration for science, with much still to discover about and learn from. Swarm Intelligence (SI), a major branch of artificial intelligence, was rendered to model the collective behavior of social swarms in nature. Ultimately, Particle Swarm Optimization algorithm (PSO) is arguably one of the most popular SI paradigms. Over the past two decades, PSO has been applied successfully, with good return as well, in a wide variety of fields of science and technology with a wider range of complex optimization problems, thereby occupying a prominent position in the optimization field. However, through in- depth studies, a number of problems with the algorithm have been detected and identified; e.g., issues regarding convergence, diversity, and stability.

Consequently, since its birth in the mid-1990s, PSO has witnessed a myriad of enhancements, extensions, and variants in various aspects of the algorithm, specifically after the twentieth century, and the related research has therefore now reached an impressive state. Swarm algorithms are increasingly used to optimize different types of problems in various fields. Without many technical assumptions, these algorithms are often, with easy implementation, able to effectively solve complex optimization problems. For example, it is not crucial to have an objective function that is differentiable or separable. Due to their flexibility, these algorithms have the potential to tackle various real-world optimization problems increasingly in different disciplines, especially engineering and computer science (Houssein et al., 2021).

3.9 Objective Functions

In order to obtain the satisfactory dynamic and steady-state characteristics while avoiding excessive controller output, a performance function needs to be defined. The most important step in using any optimization technique is to select the objective functions that will be used to assess each population's fitness (particle). Since the proposed autonomous ground vehicle system is multiple input multiple output system, the cost function which minimizes the error of the system is the sum of two objective cost functions (Al-Mayyahi et al., 2016). Which is written as:

$$J = ITAE = \int_0^{\tau} t\left(|e_1(t)| + |e_2(t)|\right) d\tau \tag{3.92}$$

Whereas $e_1(t) = \delta_{ref} - \delta_{act}$, and $e_2(t) = \dot{\psi}_{dref} - \dot{\psi}_{dact}$. Where δ orientation is angle and $\dot{\psi}_d$ is velocity.

3.10 Overall System Structure

The overall system structure of the proposed system is shown in the figure 3.8 below.

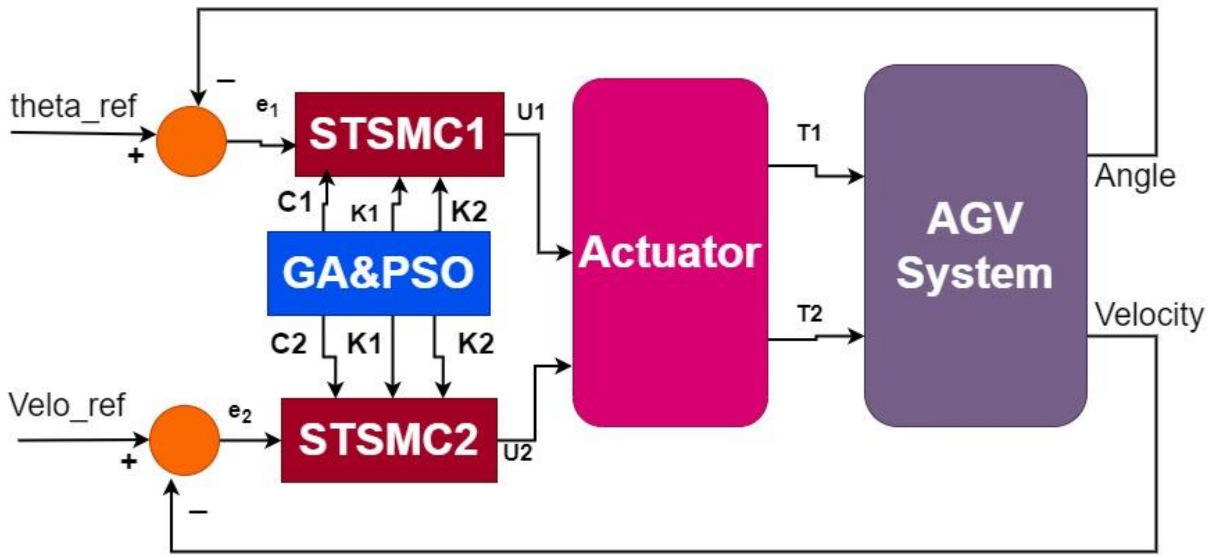


Figure 3:10 Overall system structure

CHAPTER FOUR

RESULTS AND DISCUSSIONS

4.1 Chapter Overview

This chapter includes the result and analysis of the proposed super twisting sliding mode control system of the Autonomous ground vehicle. The simulation results of the proposed system are compared with the results of Fractional order PID controller to fulfill the specific objectives of the research. The simulation results explained in this chapter are performed for the same values of the system parameters of some literatures work. In this work the dynamic model of the vehicle is used as a plant model. Based on dynamic model of the vehicle the MATLAB Simulink is created to simulate the system. The parameters used for the Matlab simulations are taken the same parameters for a passenger sedan vehicle as in (Ling, 2006) which is given as in table 4.1.

4.2 Simulink Modelling

MATLAB Simulink is a software package which is used to model, simulate, and analyze dynamic systems. Simulink has the advantage of being capable of complex dynamic system simulations, graphical environment with visual real time programming and broad selections of toolboxes. The dynamic mathematical equations of vehicle presented in chapter three was used to model AGV and proposed controller was designed in MATLAB Simulink R2017a environment. Figure 4.1 shows the complete Simulink model of proposed system.

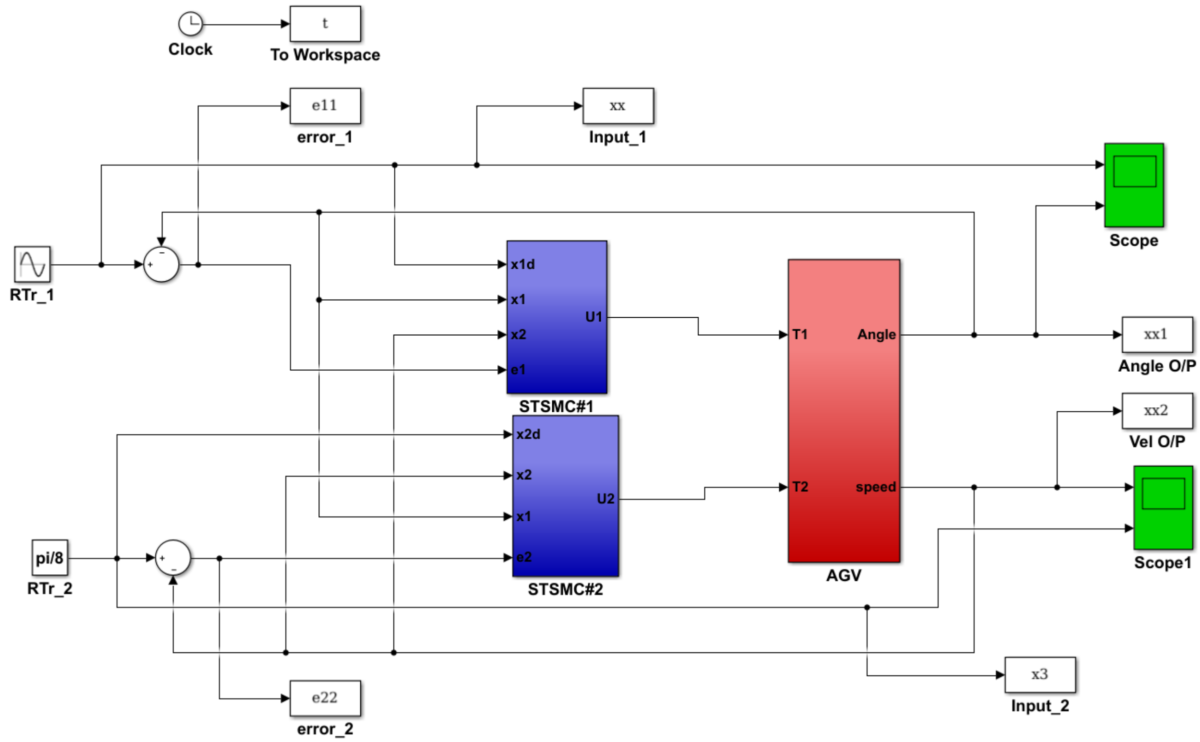


Figure 4:1 The Simulink model of the proposed system

4.3 Parameter settings for the proposed STSMC controller

It deal with the design of proposed controller and its optimal tuning of the four parameters C1, C2, K1 and K2. Tuning the parameters was implemented by GA tuning m-file code on MATLAB 2017a, using the following parameters.

- Number of Population: 20
- Number of Generations: 20
- Lower Bound(LB): [0.001 0.001 -100 -100]
- Upper Bound(UB): [1 1 100 100]

Figure 4.2 depicts the convergence curve for the best and mean fitness values of the GA tuning.

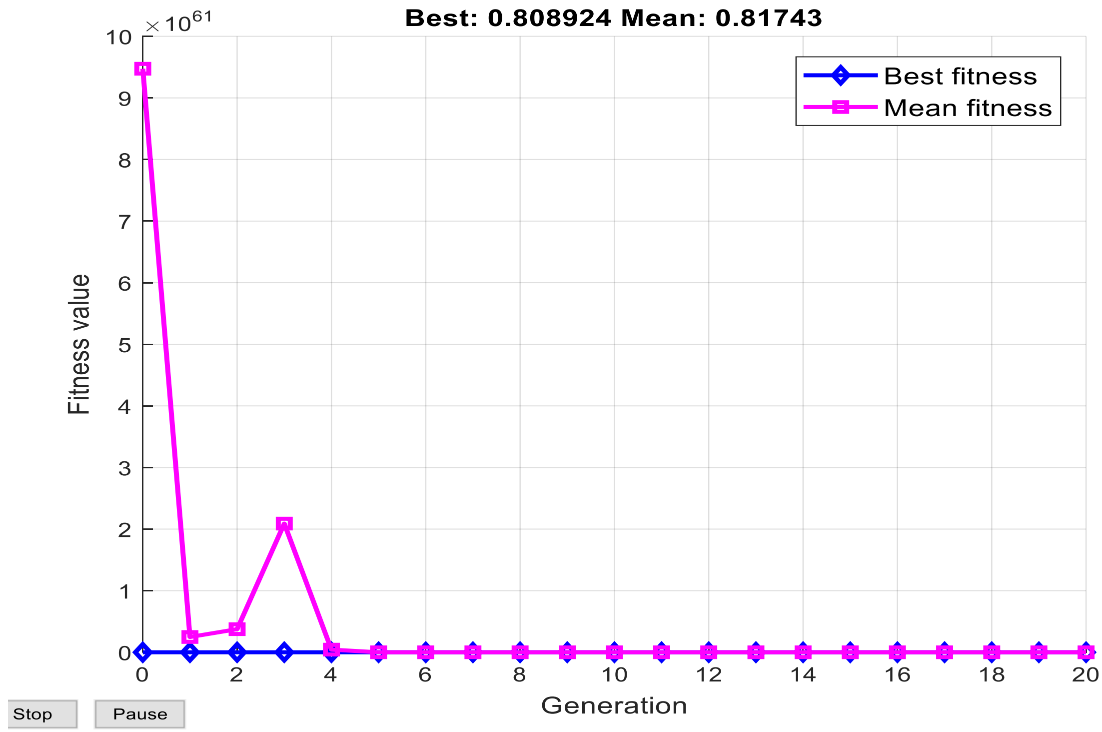


Figure 4:2 Convergence curve of proposed GA

The convergence curve has the best fitness value of 0.808924 for GA and best function value for PSO is 0.171914 and Table 4.2 lists the result of optimally tuned parameters.

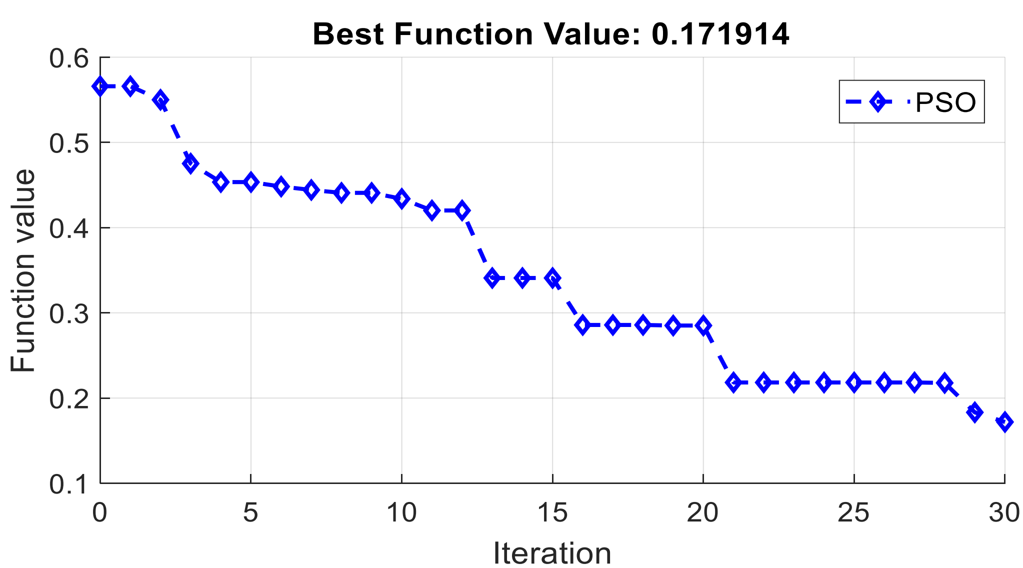


Figure 4:3 Convergence curve of proposed PSO

Table 4:1 Autonomous Ground Vehicle parameters used for simulation

Vehicle Parameters	**Description**	**Value**	**Unit**
M	Mass of the vehicle	1573	Kg
I	Yaw moment of inertia	2873	Kgm^2
Lf	Distance from vehicle c.g to the front axis center	1.1	m
Lr	Distance from vehicle c.g to the front axis center	1.58	m
Vx	Longitudinal velocity of the vehicle	30	m/s
Caf	Cornering stiffness of front tire	80000	N/rad
Car	Cornering stiffness of rear tire	80000	N/rad

Table 4:2 GA tuned controller parameters

Controllers	Parameters name	Parameter symbol	GA tuned Values	PSO tuned values
STSMC	SMC Gain1	C1	0.876	1
	SMC Gain2	C2	0.999	0.9998
	STSMC Gain1	K1	-64.73	-0.7643
	STSMC Gain2	K2	-94.98	-94.8393
FOPID	Proportional Gain	Kp	220.85	231.609
	Integral Gain	Ki	310.02	448.86
	Derivative Gain	Kd	180.04	261.86
	Integral order Gain	λ	0.684	0.553
	Derivative order Gain	μ	0.745	0.989
Computational or elapsed time			2352.36sec	4063.89sec

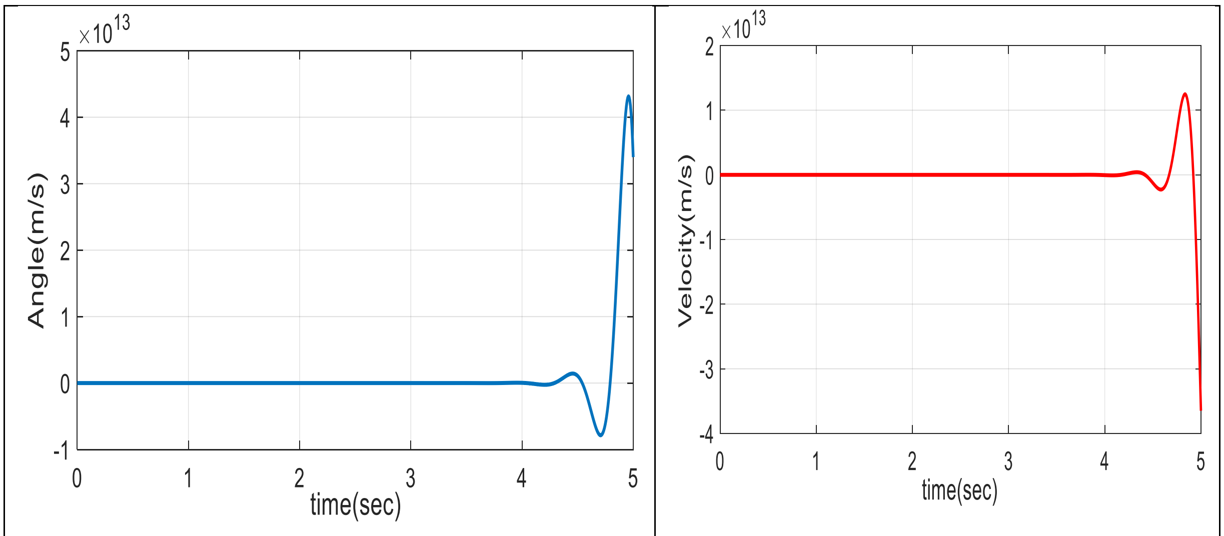
The Genetic Algorithm (GA) and PSO are proposed for adjusting the control parameters with respect to proposed cost function J. The tuning parameters and tuned values are specified in the table 4.2, the computational time which depends on computer processor type, Matlab version and step size has been checked for GA-based and PSO-based STSMC controllers. By comparing execution time PSO is slower than GA.

Table 4:3 Boundary of tuning parameters

Boundary	STSMC				FOPID				
	C1	C2	K1	K2	Kp	Ki	Kd	Lm	Mu
Lower boundary	0.0001	0.0001	-100	-100	0.0001	0.0001	0.0001	0.0001	0.0001
Upper boundary	1	1	100	100	500	500	500	0.9	0.9

4.4 Open Loop Response

For a given step and constant input the system output is highly nonlinear and unstable. The open loop response of the proposed system is shown as in figure 4.2 below:



(a) open loop response for step input angle

(b) open loop response for constant input velocity

Figure 4:4 Open loop response of the proposed system

4.5 Platform testing for closed loop responses

After implementing autonomous ground vehicle's model and the two STSMC controllers have been implemented, the platform is simulated and examined through different case studies to verify the response of the proposed methodology. Each case considers a different path as follows:

Test Case-1

In this case, a sinusoidal trajectory was generated as reference trajectory. Reference trajectory input is $\theta_d = 5\sin(t)$. This trajectory was compared with the actual trajectory to find the orientation and velocity error. Hence, the error was fed into the STSMC controller. The output of the STSMC1 controller is presented the control action that was connected the motor of the left wheel. The second input represents the desired velocity. This velocity was compared with the actual velocity of the vehicle. The error produced by comparing the velocity was inputted to the second STSMC2 controller. The output of the velocity controller was connected the motor of the right wheel. After running the simulation, the relationship between the desired and actual trajectory is shown in Figure 4.5 below. As shown in the Figure 4.5 for sinusoidal reference trajectory the system tracks the trajectory nicely. The error between the desired and the actual orientation and velocity are depicted in Figure 4.6.

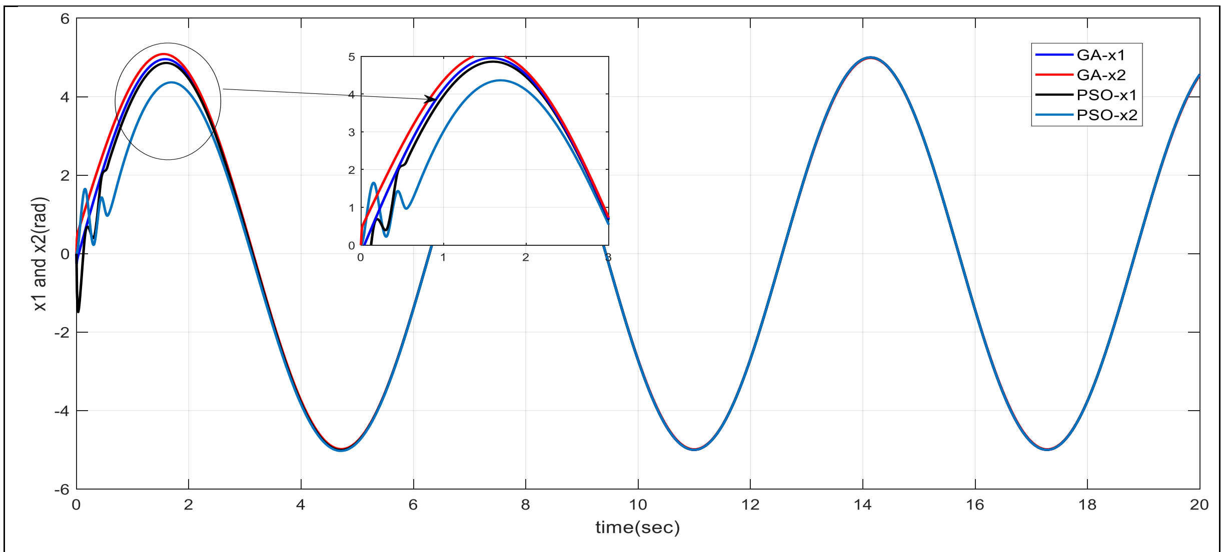
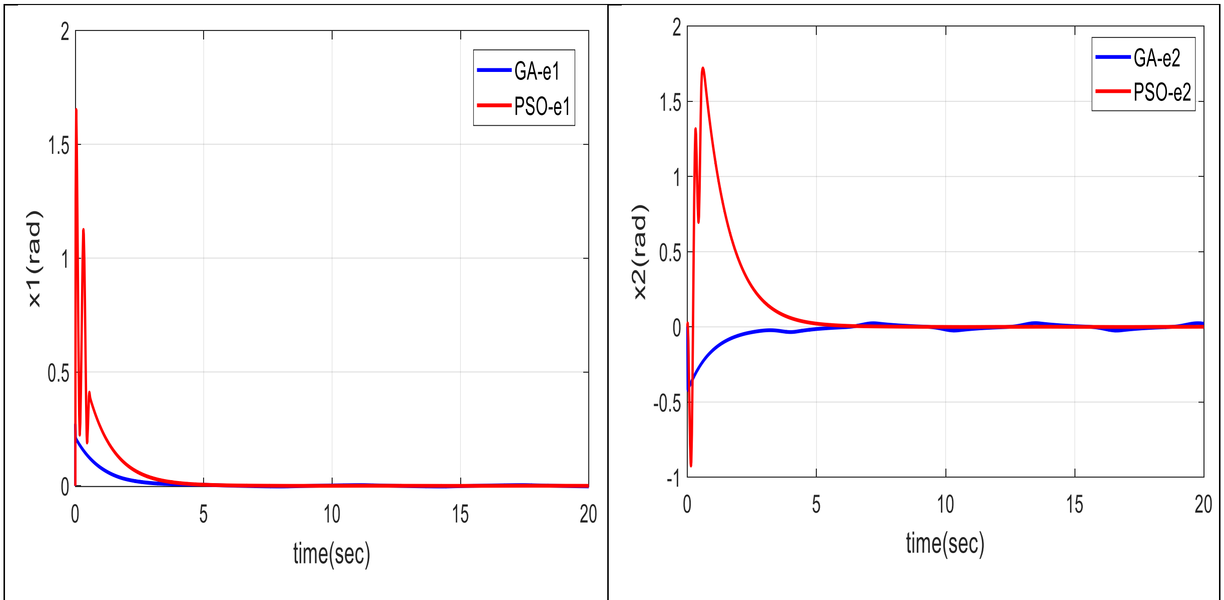


Figure 4:5 the orientation angle and velocity response for a sinusoidal trajectory $\theta_d = 5\sin(t)$ proposed system



(a) Orientation angle error (b) Velocity error

Figure 4:6 Orientation angle and velocity error for sinusoidal trajectory $\theta_d = 5\sin(t)$ of proposed system

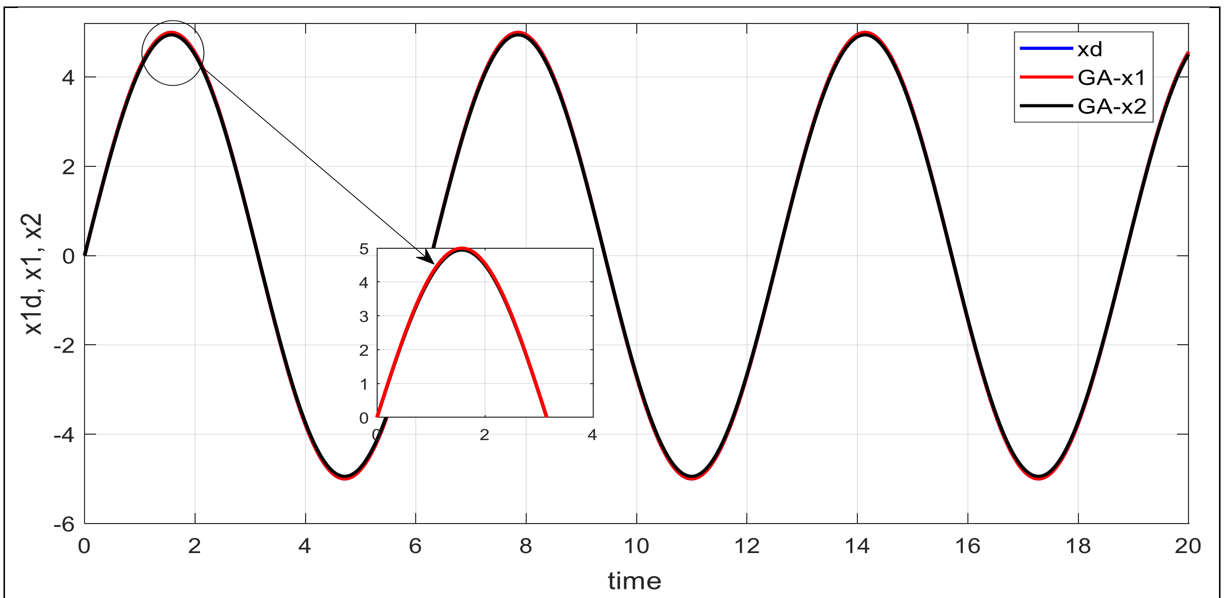


Figure 4:7 Orientation angle and velocity response for sinusoidal trajectory $\theta_d = 5\sin(t)$ of FOPID controller

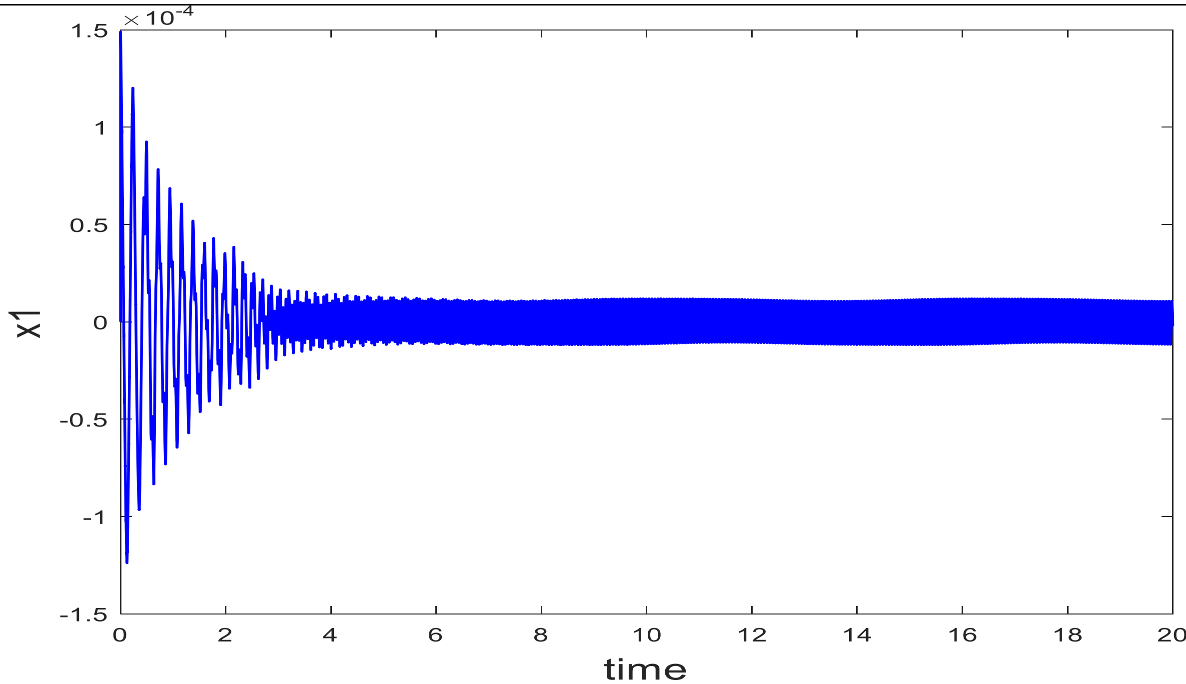
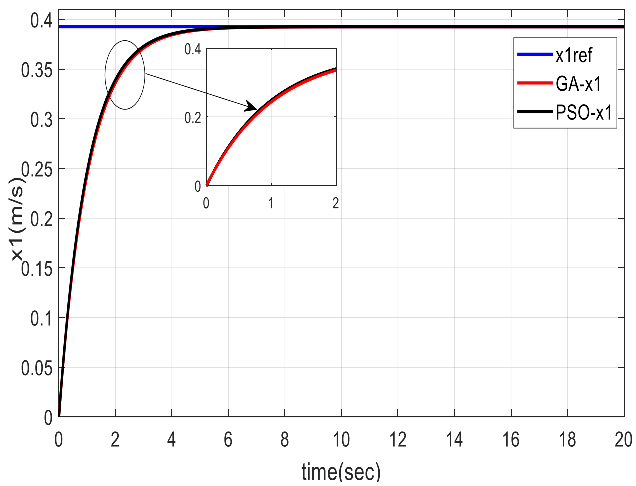


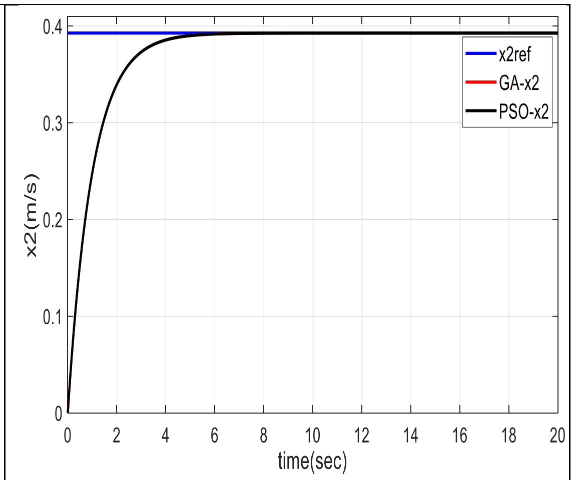
Figure 4:8 Orientation angle error for sinusoidal trajectory $\theta_d = 5\sin(t)$ of FOPID controller

Test Case-2

In response to another case and to show the adaptation of the proposed controller, a linear trajectory was created. To ensure a linear trajectory, a constant orientation should be generated using $\theta_d = \pi / 8$ for interval $0 \le t \le 20[sec]$. Similarly, the relationship between desired and actual trajectory is shown in figure 4.8 and 4.9. The error between the desired and the actual orientation and velocity are depicted in Figure 4.10 and 4.11.



(a) Orientation angle response



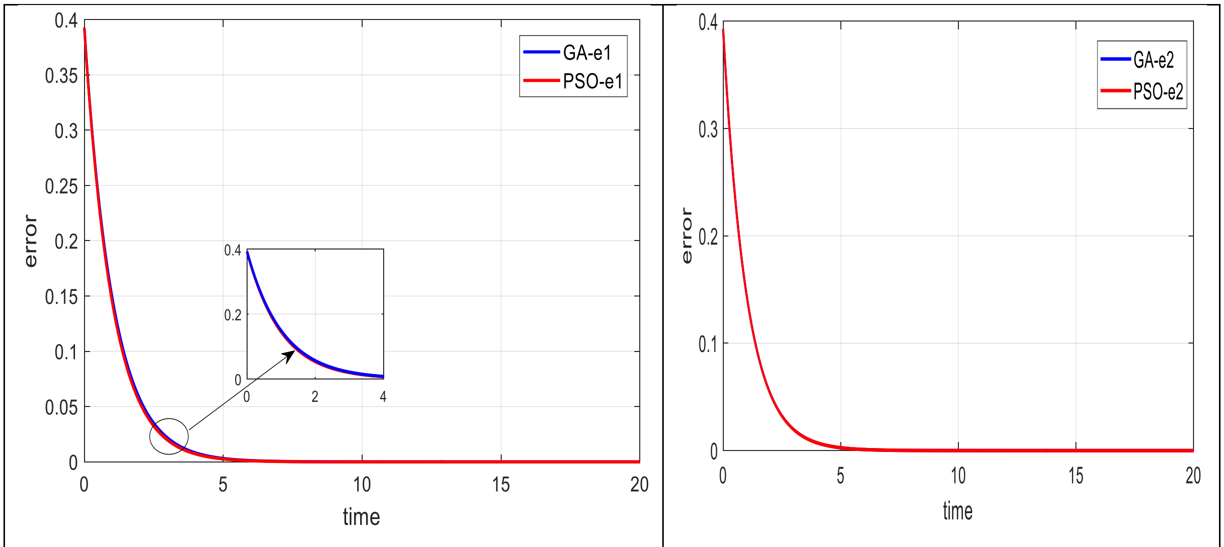
(b) Velocity response

Figure 4:9 the orientation angle and velocity response for linear trajectory $\theta_d = \pi / 8$ proposed system

Table 4:4 the error performance analysis of the trajectory $\theta_d = 5\sin(t)$

Controllers	Optimization	Angle error	Velocity error
STSMC	GA	0.0026	0.0212
	PSO	0.0006	0.0007
FOPID	GA	0.0031	0.0538
	PSO	0.0011	0.0055

As shown as in table 4.4 as depicted in figure 4.6 the error performance, we can say that the proposed STSMC controller has better error performance for PSO optimazation technique than GA based for both orientation angle and velocity control. And the error performance of proposed STSMC controller is much better than GA and PSO based FOPID controller for angle and velocity control. Adding disturbances due to friction of wheels during vehicle motion which is formed by varying the cornering stiffness value $C_{\alpha f} = C_{\alpha r}$=40000 N/rad and mass of 1500kg has the orientation angle and velocity error 0.0004 and 0.0008sec respectively for sinusoidal trajectory of proposed PSO-based STSMC controllers.



(a) Orientation angle error (b) Velocity error

Figure 4:10 the orientation angle and velocity error response for linear trajectory $\theta_d = \pi / 8$ of proposed system

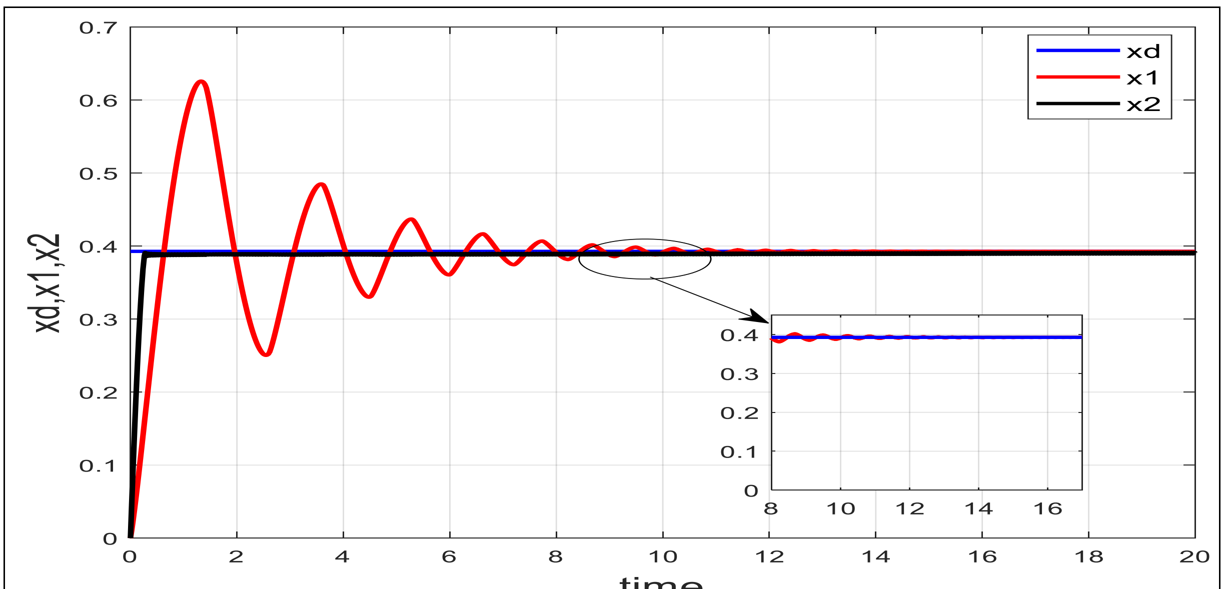


Figure 4:11 the orientation and velocity response for linear trajectory $\theta_d = \pi / 8$ of FOPID controller

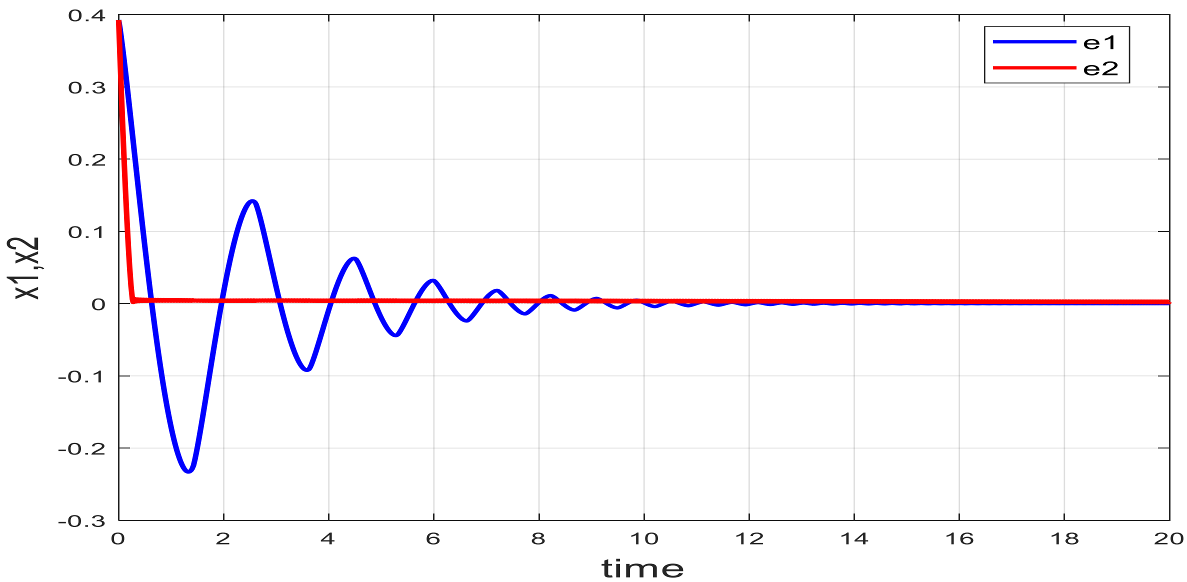


Figure 4:12 the orientation and velocity error response for linear trajectory $\theta_d = \pi / 8$ of FOPID controller

As depicted in figure 4.10 for a linear trajectory of an orientation and velocity error response highly converges to zero for both GA and PSO based optimization technique of proposed STSMC controller. Compared to the orientation angle and velocity response figure 4.11 of FOPID to the response of proposed system in a figure 4.9, it takes time to track the given linear reference trajectory and its error converges to zero after 0.0024 sec time which is sluggish compared to STSMC error convergence time. As shown in table 4.5 PSO- based STSMC had shortest rise time, settling time and smallest maximum overshoot values compared to GA- based STSMC and GA- based FOPID controllers.

Table 4:5 System performance measure comparisons

Performance measures	Designed Controllers		
	PSO- based STSMC	GA- based STSMC	GA- based FOPID
Rise time	0.0313	0.1653	0.2001
Settling time	0.21673	0.39025	0.6821
Max. overshoot	0.00821	0.0295	0.8645

CHAPTER FIVE

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In this thesis work the application of STSMC to autonomous ground vehicle systems were introduced. In this work a simplified 2-DOF bicycle model of a vehicle was used to model the vehicles' lateral dynamics. Its modeled lateral dynamics had been controlled with two different control algorithms. The two control algorithms used to test the trajectory tracking control of the autonomous ground vehicle systems were a fractional order PID controller and super twisting sliding mode controller.

To verify the performance and efficiency of the controller a simulation were obtained using MATLAB/Simulink. To optimize and tune the controllers gain the genetic algorithm and particle swarm optimization techniques were used.

Finally, the obtained results shows that the system tracks the desired trajectory with very small tracking error. In comparison with GA and PSO-based fractional order PID controller, the proposed PSO-based STSMC robustly tracks the given irregular reference environments, it had high system performance measure values 0.0313, 0.21673 and 0.00821sec of rise time, settling time and maximum overshoot respectively.

After observing the simulation results, it can be concluded that the AGV system was stabilized by controlling an orientation steering angle and velocity by using STSMC.

5.2 Recommendations

For future researchers, this research work recommends the following:

- Establishing the more recent advanced algorithms such as machine learning the researcher could study stability and performance analysis of AGV system tracking control is recommended.
- It is recommended to practice and implement the proposed system in CTVE (Center of Excellence for Transport and Vehicle Engineering) in ASTU.

Acknowledgements: The research for this paper was financially supported by Adama Science and Technology University under the grant number of **ASTU/SM-R/539/22,** Adama, Ethiopia.

REFERENCES

Al-Mayyahi, A., Wang, W., & Birch, P. (2016). Design of Fractional-Order Controller for Trajectory Tracking Control of a Non-holonomic Autonomous Ground Vehicle. *Journal of Control, Automation and Electrical Systems*, *27*(1), 29–42. https://doi.org/10.1007/s40313-015-0214-2

Alam, W., Mehmood, A., Ali, K., Javaid, U., Alharbi, S., & Iqbal, J. (2018). Nonlinear control of a flexible joint robotic manipulator with experimental validation. *Journal of Mechanical Engineering*, *64*(1), 47–55. https://doi.org/10.5545/sv-jme.2017.4786

Amer, N. H., Zamzuri, H., Hudha, K., & Kadir, Z. A. (2017). Modelling and Control Strategies in Path Tracking Control for Autonomous Ground Vehicles: A Review of State of the Art and Challenges. *Journal of Intelligent and Robotic Systems: Theory and Applications*, *86*(2), 225–254. https://doi.org/10.1007/s10846-016-0442-0

Babak, S. J., Hussain, S. A., Karakas, B., & Cetin, S. (2017). Control of autonomous ground vehicles: A brief technical review. *IOP Conference Series: Materials Science and Engineering*, *224*(1). https://doi.org/10.1088/1757-899X/224/1/012029

Cao, J. Y., & Cao, B. G. (2006). Design of fractional order controllers based on particle swarm optimization. *2006 1st IEEE Conference on Industrial Electronics and Applications, December 2006*. https://doi.org/10.1109/ICIEA.2006.257091

Chen, Q. (2007). *Studies in Autonomous Ground Vehicle Control Systems : Structure and Algorithms*.

Clerc, M. (2010). Particle Swarm Optimization. *Particle Swarm Optimization*, *May 2011*. https://doi.org/10.1002/9780470612163

Dadras, S. (2017). Path Tracking Using Fractional Order Extremum Seeking Controller for Autonomous Ground Vehicle. *SAE Technical Papers*, *2017-March*(March). https://doi.org/10.4271/2017-01-0094

Dai, M., Qi, R., & Cheng, X. (2019). Super-twisting sliding mode control design for electric dynamic load simulator. *Chinese Control Conference, CCC*, *2019-July*(1), 3078–3083.

https://doi.org/10.23919/ChiCC.2019.8865725

Deemyad, T., Moeller, R., & Sebastian, A. (2020). Chassis Design and Analysis of an Autonomous Ground Vehicle (AGV) using Genetic Algorithm. *2020 Intermountain Engineering, Technology and Computing, IETC 2020*. https://doi.org/10.1109/IETC47856.2020.9249180

Farahmandrad, M., Ganjefar, S., Talebi, H. A., & Bayati, M. (2020). Design of higher-order sliding mode controller based on genetic algorithm for a cooperative robotic system. *International Journal of Dynamics and Control*, *8*(1), 269–277. https://doi.org/10.1007/s40435-019-00516-x

Gambhire, S. J., Kishore, D. R., Londhe, P. S., & Pawar, S. N. (2021). Review of sliding mode based control techniques for control system applications. *International Journal of Dynamics and Control*, *9*(1), 363–378. https://doi.org/10.1007/s40435-020-00638-7

García-Sánchez, J. R., Silva-Ortigoza, R., Tavera-Mosqueda, S., Márquez-Sánchez, C., Hernández-Guzmán, V. M., Antonio-Cruz, M., Silva-Ortigoza, G., & Taud, H. (2017). Tracking control for mobile robots considering the dynamics of all their subsystems: Experimental implementation. *Complexity*, *2017*. https://doi.org/10.1155/2017/5318504

Ghazali, R., Sam, Y. M., Rahmat, M. F., Hashim, A. W. I. M., & Zulfatman, . (2011). Performance Comparison between Sliding Mode Control with PID Sliding Surface and PID Controller for an Electro-hydraulic Positioning System. *International Journal on Advanced Science, Engineering and Information Technology*, *1*(4), 447. https://doi.org/10.18517/ijaseit.1.4.91

Hoang, V. T., & Pham, Q. H. (2018). Adaptive super-twisting second-order sliding mode for attitude control of quadcopter UAVs. *ArXiv*.

Houssein, E. H., Gad, A. G., Hussain, K., & Suganthan, P. N. (2021). Major Advances in Particle Swarm Optimization: Theory, Analysis, and Application. *Swarm and Evolutionary Computation*, *63*(February), 100868. https://doi.org/10.1016/j.swevo.2021.100868

Hu, J., Zhang, Y., & Rakheja, S. (2021). Adaptive Trajectory Tracking for Car-like Vehicles with Input Constraints. *IEEE Transactions on Industrial Electronics*, *0046*(c). https://doi.org/10.1109/TIE.2021.3068672

Humaidi, A. J., & Hasan, A. F. (2019). *(PDF) Particle swarm optimization–based adaptive super-twisting sliding mode control design for 2-degree-of-freedom helicopter*. https://doi.org/10.1177/0020294019866863

Kamalakkannan, B. (2017). *Modelling and Simulation of Vehicle Kinematics and Dynamics*. 1–65.

Lambora, A., Gupta, K., & Chopra, K. (2019). Genetic Algorithm- A Literature Review. *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, *1998*, 380–384. https://doi.org/10.1109/COMITCon.2019.8862255

Lee, C. H., & Chang, F. K. (2010). Fractional-order PID controller optimization via improved electromagnetism-like algorithm. *Expert Systems with Applications*, *37*(12), 8871–8878. https://doi.org/10.1016/j.eswa.2010.06.009

LIDR is the Latest Game-Changing Advancement for Autonomous Vehicles - IEEE Innovation at Work. (n.d.).

Ling, F. F. (2006). Mechanical Engineering Series. In *Control*.

Maiti, D., Biswas, S., & Konar, A. (2008). Design of a fractional order PID controller using particle swarm optimization technique. *ReTIS-08*, 1–5.

Mirjalili, S., Song Dong, J., Sadiq, A. S., & Faris, H. (2020). Nature-inspired Optimizers. In *Studies in Computational Intelligence* (Vol. 811).

Nasiri, M., Mobayen, S., & Zhu, Q. M. (2019). Super-Twisting Sliding Mode Control for Gearless PMSG-Based Wind Turbine. *Complexity*, *2019*. https://doi.org/10.1155/2019/6141607

Normey-Rico, J. E., Alcalá, I., Gómez-Ortega, J., & Camacho, E. F. (2001). Mobile robot path tracking using a robust PID controller. *Control Engineering Practice*, *9*(11), 1209–1214. https://doi.org/10.1016/S0967-0661(01)00066-1

Ouchen, S., Benbouzid, M., Blaabjerg, F., Betka, A., & Steinhart, H. (2021). Direct Power Control of Shunt Active Power Filter Using Space Vector Modulation Based on Supertwisting Sliding Mode Control. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *9*(3), 3243–3253. https://doi.org/10.1109/JESTPE.2020.3007900

Peng, J., Yu, J., & Wang, J. (2014). Robust adaptive tracking control for nonholonomic mobile manipulator with uncertainties. *ISA Transactions*, *53*(4), 1035–1043. https://doi.org/10.1016/j.isatra.2014.05.012

Principle of Vision System Figure 7. Inertial Measurement Unit 2.5.... | Download Scientific Diagram. (n.d.).

Rasib, M., Butt, M. A., Khalid, S., Abid, S., Raiz, F., Jabbar, S., & Han, K. (2021). *Are Self-Driving Vehicles Ready to Launch ? An Insight into Steering Control in Autonomous Self-Driving Vehicles*. *2021*.

Shtessel, Y., Edwards, C., Fridman, L., & Levant, A. (2014). Sliding mode control and observation. In *Sliding Mode Control and Observation*. https://doi.org/10.1007/978-0-8176-4893-0

Utkin, V. I. (1992). Sliding Modes in Control and Optimization. In *Sliding Modes in Control and Optimization*. https://doi.org/10.1007/978-3-642-84379-2

Wang, J., Zhang, L., Huang, Y., & Zhao, J. (2020). Safety of Autonomous Vehicles. *Journal of Advanced Transportation*, *2020*(i). https://doi.org/10.1155/2020/8867757

Wu, Y., Wang, L., Zhang, J., & Li, F. (2019). Path Following Control of Autonomous Ground Vehicle Based on Nonsingular Terminal Sliding Mode and Active Disturbance Rejection Control. *IEEE Transactions on Vehicular Technology*, *68*(7), 6379–6390. https://doi.org/10.1109/TVT.2019.2916982

Yousif, N., Cole, J., Rothwell, J. C., Diedrichsen, J., Zelik, K. E., Winstein, C. J., Kay, D. B., Wijesinghe, R., Protti, D. A., Camp, A. J., Quinlan, E., Jacobs, J. V, Henry, S. M., Horak, F. B., Jacobs, J. V, Fraser, L. E., Mansfield, A., Harris, L. R., Merino, D. M., … Dublin, C. (2018). Control System Design for Autonomous Vehicle Path Following and Collision Avoidance Dissertation. *Journal of Physical Therapy Science*, *9*(1), 1–11.

Zamani, M., Karimi-Ghartemani, M., Sadati, N., & Parniani, M. (2009). Design of a fractional order PID controller for an AVR using particle swarm optimization. *Control Engineering Practice, 17*(12), 1380–1387. https://doi.org/10.1016/j.conengprac.2009.07.005

Zhang, Y., Liu, G., & Luo, B. (2014). Finite-time cascaded tracking control approach for mobile robots. *Information Sciences*, *284*(June), 31–43. https://doi.org/10.1016/j.ins.2014.06.037

Zhao, B., Wang, H., Li, Q., Li, J., & Zhao, Y. (2019). PID Trajectory Tracking Control of Autonomous Ground Vehicle Based on Genetic Algorithm. *Proceedings of the 31st Chinese Control and Decision Conference, CCDC 2019*, 3677–3682. https://doi.org/10.1109/CCDC.2019.8832531

Zhao, J. S., Liu, X., Feng, Z. J., & Dai, J. S. (2013). Design of an Ackermann-type steering mechanism. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, *227*(11), 2549–2562. https://doi.org/10.1177/0954406213475980

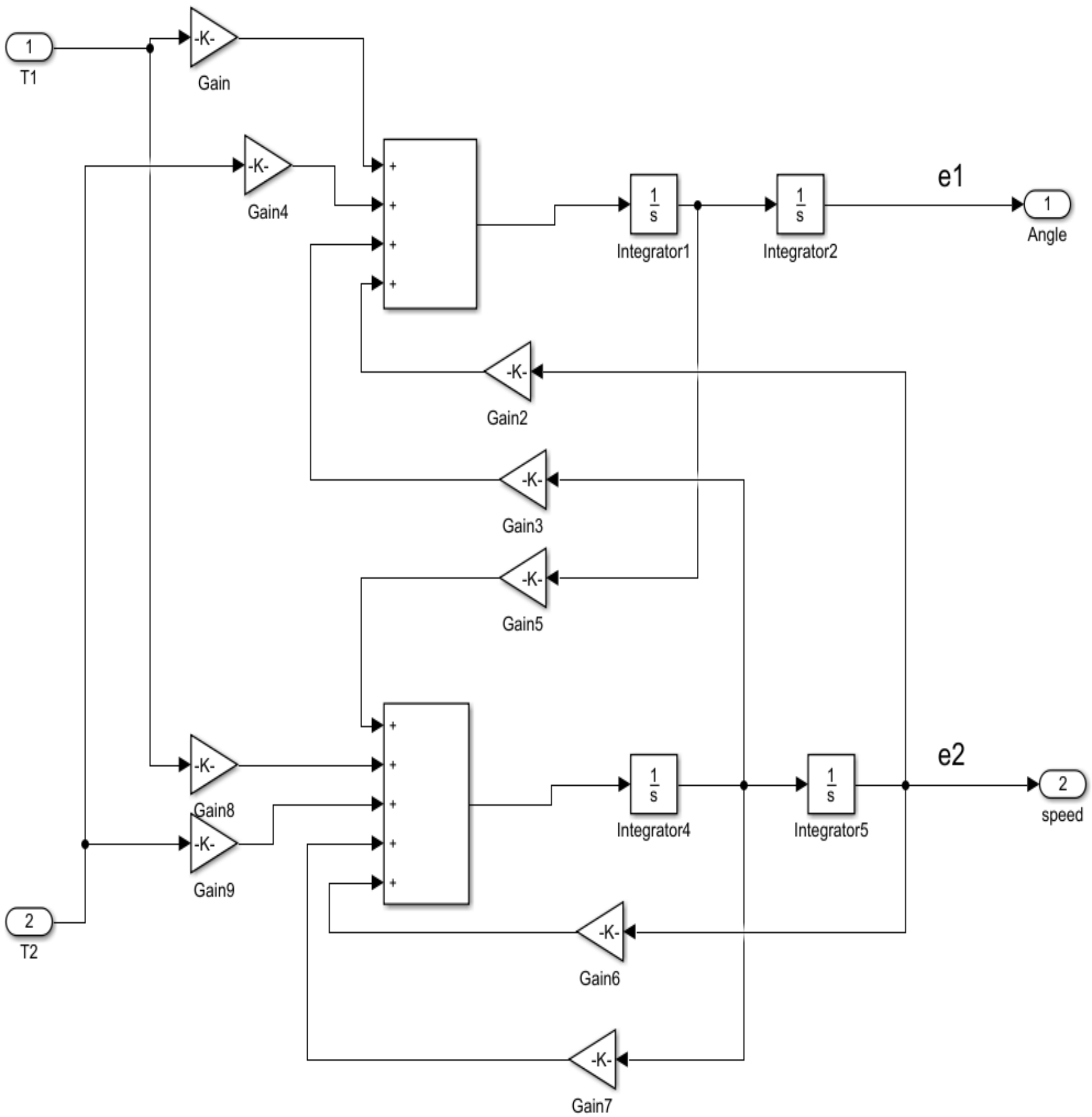
Zhao, P., Chen, J., Song, Y., Tao, X., Xu, T., & Mei, T. (2012). Design of a control system for an autonomous vehicle based on adaptive-PID. *International Journal of Advanced Robotic Systems*, *9*, 1–11. https://doi.org/10.5772/51314

Zhao, Y., Huang, P., & Zhang, F. (2018). Dynamic modeling and Super-Twisting Sliding Mode Control for Tethered Space Robot. *Acta Astronautica*, *143*, 310–321. https://doi.org/10.1016/j.actaastro.2017.11.025

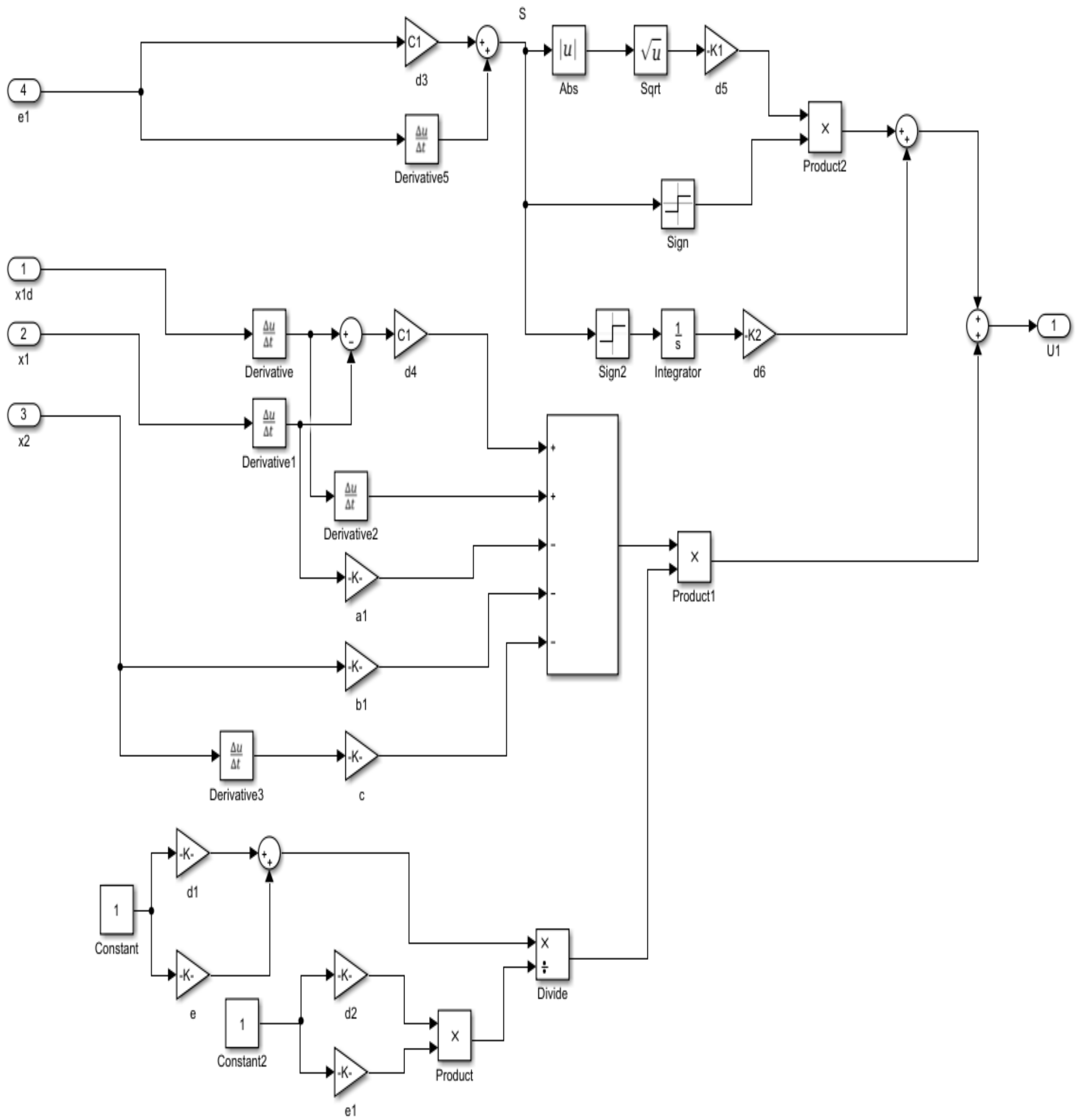
Zhuo-Yun, N., Yi-Min, Z., Qing-Guo, W., Rui-Juan, L., & Lei-Jun, X. (2020). Fractional-Order PID Controller Design for Time-Delay Systems Based on Modified Bode's Ideal Transfer Function. *IEEE Access*, *8*, 103500–103510. https://doi.org/10.1109/ACCESS.2020.2996265

APPENDIXES

Appendix A: AGV System Simulink Models



Appendix B: Proposed STSMC Controller in Simulink



Appendix C: GA Code for FOPID Controllers

```
function Tune_comp_sco
close all
clc
options= optimoptions('ga','PlotFcn',@gaplotbestf);
options.PopulationSize= 20;
options.MaxGenerations= 20;  % Increase value if needed
nvar= 10; % Number of parameters
LB= [0.0001 0.0001 0.0001 0.0001 0.0001 0.0001 0.0001 0.0001 0.0001 0.0001]; % Lower
bound of tuning parameters
UB= [500 500 500 0.9 0.9 500 500 500 0.9 0.9 ]; % Upper bound of tuning parameters
[x,fval]= ga(@EvalObj,nvar,[],[],[],[],LB,UB,[],options)
% Updates Kp, Ki, Kd, Lm and U in simulink model workspace for verification
simulink_model= 'fopid_tune';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('Kp', x(1));
gains.assignin('Ki', x(2));
gains.assignin('Kd', x(3));
gains.assignin('Lm', x(4));
gains.assignin('U', x(5));
gains.assignin('Kp2', x(6));
gains.assignin('Ki2', x(7));
gains.assignin('Kd2', x(8));
gains.assignin('Lm2', x(9));
gains.assignin('U2', x(10));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EvalObj measures a performance index from the simulink model for tuning
% obj= EvalObj(x) returns the objective function value obj calculated using
function obj= EvalObj(x)
```

```
% Updates the values of ,Kp, Ki , Kd , Lm and U in Simulink model workspace
simulink_model= 'fopid_tune';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('Kp', x(1));
gains.assignin('Ki', x(2));
gains.assignin('Kd', x(3));
gains.assignin('Lm', x(4));
gains.assignin('U', x(5));
gains.assignin('Kp2', x(6));
gains.assignin('Ki2', x(7));
gains.assignin('Kd2', x(8));
gains.assignin('Lm2', x(9));
gains.assignin('U2', x(10));
% Simulates the simulink model to get the outputs
simOut= sim(simulink_model,'SaveOutput','on');
% Retrieves time t and error e
t= simOut.find('t');
e1= simOut.find('e1');
e2= simOut.find('e2');
% Calculate a performance index using the trapezoidal integration rule
n= length(t);
ee1= abs(e1);
ee2= abs(e2);
obj= 0;
for i= 2:n
    stepsize= t(i)-t(i-1);
       obj= obj+0.5*t(i)*(ee1(i-1)+ee1(i)+ee2(i-1)+ee2(i))*stepsize; % ITAE
end
```

Appendix D: GA Code for Proposed STSMC

```
function Tune_SMC
close all
clc
options= optimoptions('ga','PlotFcn',@gaplotbestf);
options.PopulationSize= 20;
options.MaxGenerations= 20;  % Increase value if needed
nvar= 4; % Number of parameters
LB= [0.0001 0.0001 -100 -100]; % Lower bound of tuning parameters
UB= [1 1 100 100]; % Upper bound of tuning parameters
[x,fval]= ga(@EvalObj,nvar,[],[],[],[],LB,UB,[],options)
% Updates C1, C2, K1 and K2 in simulink model workspace for verification
simulink_model= 'SMC';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('C1', x(1));
gains.assignin('C2', x(2));
gains.assignin('K1', x(3));
gains.assignin('K2', x(4));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EvalObj measures a performance index from the simulink model for tuning
% obj= EvalObj(x) returns the objective function value obj calculated using
function obj= EvalObj(x)
% Updates C1, C2, K1 and K2 in simulink model workspace for verification workspace
simulink_model= 'SMC';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('C1', x(1));
gains.assignin('C2', x(2));
gains.assignin('K1', x(3));
```

```
gains.assignin('K2', x(4));
% Simulates the simulink model to get the outputs
simOut= sim(simulink_model,'SaveOutput','on');
% Retrieves time t and error e
t= simOut.find('t');
e1= simOut.find('e1');
e2= simOut.find('e2');
% Calculate a performance index using the trapezoidal integration rule
n= length(t);
ee1= abs(e1);
ee2= abs(e2);
obj= 0;
for i= 2:n
   stepsize= t(i)-t(i-1);
      obj= obj+0.5*t(i)*(ee1(i-1)+ee1(i)+ee2(i-1)+ee2(i))*stepsize; % ITAE
end
```