OPTIMAL TRAJECTORY PLANNING FOR A DIFFERNTIALLY DRIVEN UNMANNED GROUND VEHICLE IN AN OBSTACLE RIDDEN ENVIRONMENT

A PROJECT REPORT

submitted by

MUAFIRA THASNI K.T
(Reg. No. TVE19EEGN12)

to

the APJ Abdul Kalam Technological University
in partial fulfillment of the requirements for the award of the Degree

of

Master of Technology
in
Electrical and Electronics Engineering

with specialisation in

Guidance and Navigation Control

Department of Electrical Engineering
College of Engineering Trivandrum
Thiruvananthapuram - 695 016
KERALA

JULY 2021

DECLARATION

I undersigned hereby declare that the project report entitled "**Optimal Trajectory Planning for a Differentially Driven Unmanned Ground Vehicle in an Obstacle Ridden Environment**", submitted for partial fulfillment of the requirements for the award of degree of Master of Technology in Electrical and Electronics Engineering with specialisation in Guidance and Navigation Control, of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of *Mrs. Gifty Ernestina Benjamin R*, Sci/Engr 'SF', VSSC (External Guide), *Mr. Venkateswaran J*, Sci/Engr 'SC', VSSC (External Co-Guide) and *Dr. Jisha V.R*,Associate Professor (Internal Guide), Department of Electrical Engineering. This submission represents my ideas in my own words and where ideas or words of others have been included. I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Thiruvananthapuram
July 2, 2021

MUAFIRA THASNI K.T

DEPARTMENT OF ELECTRICAL ENGINEERING

COLLEGE OF ENGINEERING TRIVANDRUM

THIRUVANATHAPURAM - 695 016

CERTIFICATE

This is to certify that the report entitled " **Optimal Trajectory Planning for a Differentailly Driven Unmanned Ground Vehicle in an Obstacle Ridden Environment** " submitted by **MUAFIRA THASNI K.T** , **(Reg. No. TVE19EEGN12)** of fourth semester to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electrical and Electronics Engineering with specialisation in Guidance and Navigation Control, is a bonafide record of the project work done by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. Jisha V.R.
Internal Supervisor
Professor
Department of Electrical Engineering
College of Engineering Trivandrum

Mrs. Gifty Ernestina Benjamin R.
External Supervisor
Sci/Engr. 'SF'
Guidance Development Division
VSSC Thiruvananthapuram

Dr. Lekshmi A
Professor and PG Co-ordinator
Department of Electrical Engineering
College of Engineering Trivandrum

Dr. Savier J.S.
Professor and Head
Department of Electrical Engineering
College of Engineering Trivandrum

Acknowledgment

A lot of effort and hard work has been put into this project in course of its development. However, it would not have been possible without the kind support and help of many individuals and other sources. I would like to extend my sincere thanks to all of them. I take this opportunity to express my deep sense of gratitude and sincere thanks to all who helped me to complete this project report successfully.

I am highly thankful to *Dr. Savier J.S.*, Head of the Department, Department of Electrical Engineering and *Dr. Lekshmi A*, PG Co-ordinator, Department of Electrical Engineering for their support and cooperation. I also thank *Dr. P. Sreejaya*, former Head of the Department, Department of Electrical Engineering.

I would like to thank my internal guide, Associate Professor *Dr. Jisha V.R.*, external guides *Mrs. Gifty Ernestina Benjamin R. (VSSC)* and *Mr. Venkateswaran J* for their valuable time and effort have given for me. I thank the project coordinators, Assistant Professor *Dr. Sreeja S.*, Professor *Dr. P. S. Lalpriya*, Associate Professor *Dr. Arlene Davidson R* and Associate Professor *Dr. Asok Kumar A* for the consistent evaluation and support.

I would like to thank all other staff members and friends who helped me in this endeavour.

MUAFIRA THASNI K.T

Abstract

Unmanned Ground Vehicles (UGV)s are used in many military and civilian applications including border patrol, rescue process, surveillance, industrial applications. In such a case it is important to obtain a fast, safe and efficient path for the UGVs. So in this work, optimal trajectory planning is considered. Optimal trajectory planning will make the trajectory design possible with the minimization of time, control effort, etc. The aim of this work is to design an optimal trajectory planning algorithm for a differentially driven UGV with minimal time of travel and minimum control effort. The problem of optimal trajectory planning for differentially driven UGV can be formulated as an optimal control problem. The pseudo spectral methods in which the infinite dimensional optimal control problem is approximated as finite-dimensional nonlinear programming problem is one of the efficient and simple method, and it is used to solve the optimal control problem in this work. This work uses Chebyshev pseudo spectral method where Chebyshev polynomial is used as basis polynomial and Chebyshev Guass Lobatto (CGL) nodes are chosen as interpolation nodes. Different scenarios of environments with static and dynamic obstacles are considered and optimal trajectory design problem is solved for each scenario.

Contents

List of Tables

List of Figures

Abbreviations

UGV	Unmanned Ground Vehicle
PS	Pseudo Spectral
CPS	Chebyshev Pseudo Spectral
OCP	Optimal Control Problem
CGL	Chebyshev Gauss Lobatto
SQP	Sequential Programming Problem

List of Symbols

C	The centre of mass point of vehicle
A	The mid-point of the rear axle
b	The half distance between two driven wheels
d	The distance between point C and A
r	The radius of the driven wheel
ϕ	The orientation in the inertial frame
θ_r	The angular position of the right wheel
θ_l	The angular position of the left wheel
$\mathbf{R}$	The rotational matrix
$\dot{\xi}_I$	Pose of the UGV represented in inertial reference frame
$\dot{\xi}_R$	Pose of the UGV represented in body reference frame
v	Velocity
ω	Angular velocity
$\dot{v}$	Linear acceleration
$\dot{\omega}$	Angular acceleration
$\mathbf{X}$	State matrix
$\mathbf{U}$	Control input matrix
t_0	Initial time
t_f	Final time
N	Order of polynomial
(x_{ob}, y_{ob})	Coordinates of position
(x, y)	Coordinates of vehicle centre of mass
t	time

Chapter 1

Introduction

1.1 Unmanned Ground Vehicle

For the last decade, the development in the research field of unmanned ground vehicles (UGV) is unprecedented because it has a wide range of applications including military and civilian areas [1; 2]. Generally, UGVs are self-driven vehicles that have sensors for seeking information from the environment. Most of the UGVs do not require any human interventions [3]. A differentially driven UGV has two independently driven wheels in the rear and one or more unpowered wheels to balance the chassis. To reach the required goal position from an initial point, the UGV has to follow a predefined path avoiding obstacles. Most of the applications require autonomous navigation of the UGV. To make the UGV autonomous, the first thing is to do path planning. Path planning can be defined as the process of finding a collision-free path for a robot from its initial position to the goal or target point by avoiding collisions with any static obstacles or other agents present in its environment [4]. Obstacle shape may vary for each environment and in most cases, it will not be a smooth geometrical shape. The obstacle's geometry constrains the UGV's local path. Avoiding dynamic obstacles is also a challenging problem in real-time.

A path planner should generate feasible paths which are easily applicable and accurate. The fast solution is also a challenging requirement for autonomous vehicles. Path length also has importance, trajectory planned should be such as only minimum path length has to be covered. The DARPA challenge initiated in 2004 boosted the efforts to achieve fully autonomous ground vehicles capable of offroad driving with the constraint of limited time. Meanwhile, trajectory optimization techniques were introduced to achieve a certain level of optimality[5].The optimum final time, minimizing control effort, optimum error, etc are the challenges that can be envisaged using optimal control theory. Vehicle uncertainty and requirement for smooth trajectory are other challenging aspects in trajectory planning research.

1.2 Objectives

Main objectives of this work are :

1. To design an optimal trajectory for a differentially driven UGV in minimum time with minimal time control effort
2. To modify the path for obstacle ridden environment

1.3 Thesis Outline

The thesis is organized as follows: Literature review is presented in Chapter 2 and in Chapter 3, a description of the system is given. Some preliminary concepts on the formulation of Optimal Control Problem (OCP) and the proposed problem formulation are given in Chapter 4. The methodology used for solving the formulated problem is described in Chapter 5. Simulation results and analysis of the results are given in Chapter 6 and a note on the future scope and conclusion the work is given in Chapter 7

Chapter 2

Literature Review

Unmanned ground vehicle (UGV) technology is getting attracted by many researchers and developers in recent years. It combines different engineering fields including electrical and mechanical engineering, computer science, and robotics. They are supposed to replace human effort in the civilian and military areas. For most of the existing ground vehicles, navigation is an essential step. Generally, navigation includes the procedure of consecutive motion that guides the vehicle from the origin point to the destination point without having any collisions with the obstacles in the environment. These obstacles may be static or dynamic [6]. According to the literature, autonomous ground vehicle navigation problem can be subdivided as follows: [7]

1. World perception: a stage in which the vehicle senses the environment surrounded by it and identify the obstacles and the paths
2. Path planning and path generation: an ordered sequence of intermediate points that must be followed by the vehicle to reach the destination is created by following the data from the previous step.
3. Motion controlling: to make sure that the vehicle follows the current path, movements, and actions of the UGV must be controlled

Trajectory planning problem can be approximated as path planning problem by ignoring the non-holonomic constraints of the vehicle. The path planning problem computes the sequence of valid configurations that moves the vehicle from the initial position to a required goal position while satisfying a set of given constraints. A feasible path is a path that satisfies the constraints but does not consider the optimality or quality of the solution. Initially, the researches were concentrated on generating feasible paths only. Optimal path planning problems considered the quality aspects of the solution by optimizing some objectives.

Reif and John.H proposed feasible but non-optimum path planning for holonomic vehicles. They consider the obstacles as polygons/polyhedra and the resulting path is not constrained by differential constraints [8].In the area of Optimal path planning, initially, researchers were focused on optimizing path length. Visibility graphs were largely used for solving shortest path problems in 2D. Two new techniques for constructing visibility graphs were proposed in Ref: [9]. Those techniques were non-optimum but unlike the existing research findings, both the techniques were simple and easily implementable. There are many path planning algorithms in literature which generate a geometric path from an initial point to the goal position through predefined via points [10]. Road map based methods are most commonly used in which road maps are used to capture the environment. Visibility graph [11], Voronoi diagram[12], Delaunary triangulation[13] and adaptive road-maps [14] are the different type road map based methods. Algorithms like Dijikstra, A* and D* are some graph search method which can give optimal solutions [15; 16; 17]. Graph search and road map based methods create undesirable discretization and hence those methods are difficult to satisfy the nonholonomic constraints. Road map assumes the vehicle as a point mass and so it's practical application becomes difficult. Research in [18] and [19] are based on cell decomposition method in which the structure free space is represented by a number of small cells. So finding appropriate grid size for the cell decomposition is a task. Finding an exact solution for the path planning problem compromises the computational cost. Since autonomy of the vehicle is the main point of interest, most of the research concentrates on the attempt to find the approximate solutions which converge to the exact or optimal solution [20]

Considering the mechanical limitations will change the sense of path planning problem as a trajectory planning problem [10]. Many intelligent methods like Artificial Neural Network (ANN)[21], Genetic Algorithms[22], Particle Swarm Optimization(PSO), Ant Colony Optimization(ACO) etc. are developed in literature. Another mostly used method is Rapidly exploring Random Tree (RRT) method and Probabilistic Roadmap methods. Those helped to achieve solution for higher dimension problems within a short time. RRT considered the kinematic limitations and control constraints with ignoring dynamic constraints. Suboptimal solutions can be obtained uisng this method[23].

For a differentially driven UGV, non holonomic constraints are not negligible. So considering the differential dynamics, these kinds of problems can be formulated as Non-linear Optimal Control problems[24]. Analytical and numerical solutions are developed in the literature for solving such problems. Using Pontryagin's minimum principle[25; 26] is suitable for linear systems with only lower and upper control bounds. Hamilton Jacobi Bellman (HJB) partial differential equations are also used in literature but that make the solving process more complex for higher dimension problems if the attempt is to solve analytically. As a result of the difficulties during analytical attempts to solve nonlinear optimal control problems, many numerical methods are implemented. Direct and Indirect numerical methods are there. Former uses optimize and then discretize technique while the latter uses discretize and then optimize technique. Direct methods need to calculate variations or state the optimality conditions directly. Indirect methods approximate the state and control variables by linear combination of basis functions. Three fundamental components of solving an optimal control problem are [27],

1. Solving differential equations and integrating functions
2. Solving a system of non linear algebraic equations
3. Solving a non linear optimization problem

All numerical methods to solve optimal control problem require solving differential equations and integrating functions. Indirect method combines numerical solution of differential equations and numerical solutions of systems of non linear equations. Direct method combines numerical solution of differential equations and non linear optimization. This is summarized in figure2.1 [27]. Time marching and collocation are two main categories in numerical solu-

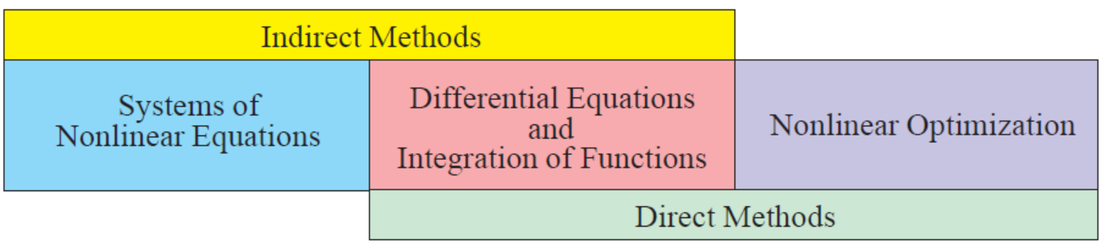


Figure 2.1: The three major components of optimal control and the class of methods that uses each Component

tion methods for differential equations.In time marching, the solution of differential equations at each time step is obtained sequentially using current/or previous information about the solution. Collocation methods are widely used by researchers. Direct collocation is an interpolation method in which it is possible to obtain time histories of both state and control[28]. Hermite Simpson trapezoidal rules are another collocation method which is less accurate because of the lack of robustness in the initial guess. According to the choice for selecting collocation points, three classifications namely Gauss, Radau, Lobatto are there [27]. Gauss collocation points does not include end points, Radau includes atmost one end point and the Lobatto points includes both the end points. Orthogoanal collocation is a subclass of collocation method in which nodes or collocation points are the roots of an orthogonal polynomial. To approximate the definite integral term, orthogonal collocation methods use quadrature rules. Orthogonal collocation is originally introduced by C. D. Boor and B. Schwartz, in 1973 [29]. Chebyshev and Legendre polynomials polynomials can be used to find the collocation points. Accurate quadrature approximation of the integral is is the advantage of orthogonal collocation. Single shooting and multiple shooting which break up the trajectory in to segments are successfully implemnted in several researches [23]. These are sequential methods that parameterize the control. Simultaneous methods that parameterize both state and control are also there in literature.

Gradient based methods and Heuristic methods are commonly used numerical methods for solving non linear problems in literature. Heuristic methods are based on stochastic manner search. Genetic Algorithm and Particle Swarm Optimization are widely used [22; 30]. Gradient based methods generally use two algrithms, Sequential Quadratic Programming (SQP) and Interior Point algorithm.

Computational complexity, computational accuracy and dynamic constraints are the main challenges in trajectory optimisation. Pseudo spectral methods are the global form of orthogonal collocation. State is approximated using a global polynomial. Unlike local collocation, pseudo spectral methods use a fixed number of segments and varying degree polynomials. Exponential convergence is the advantage of pseudo spectral emthods. Chebyshev or Lagrange polynomials are used as basis functions [31; 32]. Fahroo et al [28] proposed a pseudo spectral method in which Chebyshev polynomial is used for interpolation and high degree of accuracy is obtained. The recent work in literature [23] also utilize this method for trajectory optimization problem of a mobile robot to minimize the control effort and avoid obstacles.

This research work tries to overcome some of the main challenges in the trajectory planning including minimum time, minimum control effort and obstacle avoidance. System description is given in Chapter 3 and a discussion on problem formulation is given in Chapter 4

Chapter 3

System description

This chapter presents a brief description of the system structure and kinematics.

3.1 System Description

A differentially driven UGV has two independent driving wheels on a common axis at a distance and a caster mounted on the front, as described in figure 3.1. There are two Direct Current motors mounted on the two driving wheels for operation. The vehicle can thus change its direction by varying the relative rotation speed of its wheels and require no additional steering motion. The symbols in figure 3.1 [23] are explained in table 3.1.

Two coordinate frames are defined as follows:

1. The world-fixed inertial reference frame $X_I O Y_I$ and

2. The robot-fixed reference frame with origin point at the center of axis $X_R A Y_R$.

The trajectory of a vehicle is constrained to the horizontal plane, so the position and orientation of the UGV reference point C in these two frames can be defined as

$$\xi_I = [x_I, y_I, \phi_I]^{\mathrm{T}} \tag{3.1}$$

and

$$\xi_R = [x_R, y_R, \phi_R]^{\mathrm{T}} \tag{3.2}$$

These two frames can be mappped as,

$$\xi_I = \mathbf{R}\xi_R \tag{3.3}$$

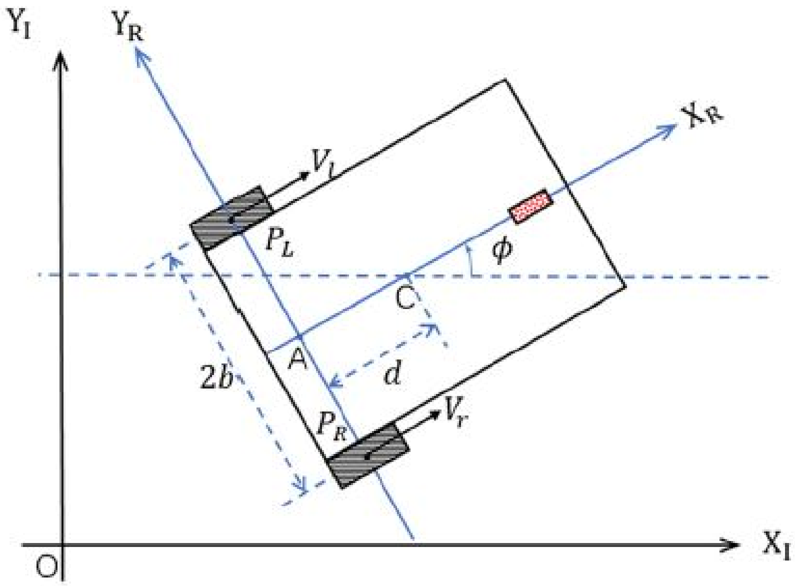


Figure 3.1: Differentially Driven UGV

Symbols	Description
C	The center of mass point
A	The mid-point of the rear axle
b	The half distance between two driven wheels
d	The distance between point C and A
r	The radius of the driven wheel
ϕ	The orientation in the inertial frame
θ_r	The angular position of the right wheel
θ_l	The angular position of the left wheel

Table 3.1: System parameters

where,

$$\mathbf{R} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

Then the motion between frames can be handled by ,

$$\dot{\xi}_I = \mathbf{R}^{-1}\dot{\xi}_R \tag{3.5}$$

Two assumptions are made: no lateral slip constraint and pure rolling constraint. Therefore, the two constraint equations are denoted as:

$$\begin{bmatrix} \dot{y}_R \\ v_r \\ v_l \end{bmatrix} = \begin{bmatrix} d\dot{\phi} \\ r\dot{\theta}_r \\ r\dot{\theta}_l \end{bmatrix} \tag{3.6}$$

$$\Lambda(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0} \tag{3.7}$$

$$\mathbf{\Lambda}(\mathbf{q}) = \begin{bmatrix} -\sin\phi & \cos\phi & -d & 0 & 0 \\ \cos\phi & \sin\phi & b & -r & 0 \\ \cos\phi & \sin\phi & -b & 0 & -r \end{bmatrix} \tag{3.8}$$

$$\mathbf{q} = [x, y, \phi, \theta_r, \theta_l]^{\mathrm{T}}$$

The velocity of point C can be represented as,

$$\dot{\xi}_R = \begin{bmatrix} v & d\omega & \omega \end{bmatrix}^{\mathrm{T}} \tag{3.9}$$

where, linear velocity,

$$v = \frac{r}{2}\left(\dot{\theta}_r + \dot{\theta}_l\right) \tag{3.10}$$

and angular velocity ,

$$\omega = \frac{r}{2b}\left(\dot{\theta}_r - \dot{\theta}_l\right) \tag{3.11}$$

The forward kinematic equations of the ground vehicle can be expressed as [23] :

$$\dot{\xi}_I = \begin{bmatrix} \frac{r}{2}\cos\phi - \frac{dr}{2b}\sin\phi & \frac{r}{2}\cos\phi + \frac{dr}{2b}\sin\phi \\ \frac{r}{2}\sin\phi + \frac{dr}{2}\cos\phi & \frac{r}{2}\sin\phi + \frac{dr}{2b}\cos\phi \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \tag{3.12}$$

Equation (3.12) can be written as,

$$\begin{aligned} \dot{x} &= v\cos\phi - d.\omega\sin\phi \\ \dot{y} &= v\sin\phi - d.\omega\cos\phi \\ \dot{\phi} &= \omega \end{aligned} \tag{3.13}$$

where, x and y are the coordinates of the centre of mass of the vehicle, ϕ is the orientation of the vehicle in inertial reference frame, d is the distance to the centre of mass point from the point of action of force, ie the midpoint of the axle connecting the wheels, v is the linear velocity and ω is the angular velocity of the vehicle.

Vehicles kinematic equations are given as the dynamic constraints to the optimal control problem formulated in coming Chapter. Details of problem formulation is given in the next Chapter.

Chapter 4

Problem Formulation

This chapter describes the optimal trajectory planning problem formulation. In the first section, a general optimal trajectory design problem is formulated. Then the formulation of an optimal control problem in an obstacle ridden environment is given. Proposed problem formulations for static and dynamic environments are given in last two sections.

4.1 General Formulation of Optimal Trajectory Planning Problem

The problem is to design an optimal trajectory for a differentially driven UGV to reach a destination. Hence the system can be transformed into a state-constrained double integrator dynamic system as follows :

$$\begin{aligned} \dot{\mathbf{q}} &= \widehat{\mathbf{S}}(\mathbf{q})\mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{u} \end{aligned} \tag{4.1}$$

$$\mathbf{q} = [x, y, \phi]^{\mathrm{T}} \tag{4.2}$$

$$\mathbf{v} = [v, \omega]^{\mathrm{T}} \tag{4.3}$$

$$\mathbf{x} = [\mathbf{q}, \mathbf{v}]^{\mathrm{T}} \tag{4.4}$$

Then the optimal trajectory design problem can be formulated as a nonlinear optimal control problem as follows,

Determine the control inputs to minimize the objective function J

$$\text{Minimize } J = \mathcal{M}\left(\mathbf{X}\left(t_0\right), t_0, \mathbf{X}\left(t_f\right), t_f\right) + \int_{t_0}^{t_f} \mathcal{L}(\mathbf{X}(t), U(t), t)dt \tag{4.5}$$

Subjected to :

$$\text{Boundary conditions : } \mathbf{X}(t_0) = \mathbf{X}_0, \mathbf{X}(t_f) = \mathbf{X}_f \tag{4.6}$$

$$\text{Control constraints : } \mathbf{U}_{\min} \leq \mathbf{U} \leq \mathbf{U}_{\max} \tag{4.7}$$

$$\text{State constraints : } \mathbf{X}_{\min} \leq \mathbf{X} \leq \mathbf{X}_{\max} \tag{4.8}$$

According to the objectives, cost functions can be defined in terms of Lagrange and Mayer terms as in (4.5) . For minimising the final time, cost function is as in (4.9). For optimizing the control effort, an additional term is introduced as in (4.10).

$$\text{Minimize } J = \int_{t_0}^{t_f} \mathrm{d}t \tag{4.9}$$

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathbf{U} * \mathbf{U}^{\mathrm{T}})\mathrm{d}t \tag{4.10}$$

where, t_0 and t_f are initial and final time, **U** is the control input matrix

4.2 Optimal Trajectory Planning in Obstacle Ridden Environment

In a real scenario, the UGV may encounter obstacles. In this work, obstacles and vehicle are assumed as circles. For representing UGV, circumcircle is taken. A safer zone of 0.5 m width is considered around the circular obstacles. So the inflated radii of obstacles are considered for defining the problem. Let length and breadth of the vehicle be 2B and 2D respectively. Centre of the circumcircle is point of center of gravity and the radius is given by (4.11).

$$r_m = \sqrt{B^2 + D^2} \tag{4.11}$$

,

$$d_{collision} = r_m + r_{ob} \tag{4.12}$$

where, r_m is the radius of the circle considered for vehicle and r_{ob} is that of the obstacle.

Distance between obstacle and vehicle

$$d_{sep} = \sqrt{(x_{ob} - x)^2 + (y_{ob} - y)^2} \tag{4.13}$$

where, x_{ob} and y_{ob} are the coordinates of center of obstacle. Path constraint can be added as an inequality constrained.

Inequality constraint for obstacle avoidance

$$d_{collision} - d_{sep} \leq 0 \tag{4.14}$$

To ensure the obstacle avoidance, inverse of distance between vehicle and obstacle is added to the performance index as (4.16).

$$P_i = \frac{1}{(d_i - r_{obi})} \tag{4.15}$$

d_i is the distance between vehicle and i^{th} obstacle r_{obi} is the sum of radii of vehicle and i^{th} obstacle

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathbf{U} * \mathbf{U}^{\mathrm{T}}) + \sum_{i=1}^{N_{ob}} P_i \mathrm{d}t \tag{4.16}$$

4.3 Proposed Problem Formulation: Optimal Trajectory Planning to Minimise Time and Control Effort in Obstacle Ridden Environment

Proposed optimal trajectory planning problem to move the UGV from a given starting position to a required goal position within minimum time and with minimal control effort in an obstacle ridden environment is formulated as follows.

Determine the control inputs to,

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathbf{U} * \mathbf{U}^{\mathrm{T}}) + \sum_{i=1}^{N_{ob}} P_i \mathrm{d}t \tag{4.17}$$

subjected to,

$$\text{Boundary conditions : } \mathbf{X}(t_0) = \mathbf{X}_0, \mathbf{X}(t_f) = \mathbf{X}_f \tag{4.18}$$

$$\text{Control constraints : } \mathbf{U}_{\min} \leq \mathbf{U} \leq \mathbf{U}_{\max} \tag{4.19}$$

$$\text{State constraints : } \mathbf{X}_{\min} \leq \mathbf{X} \leq \mathbf{X}_{\max} \tag{4.20}$$

$$\text{Dynamic constraints : } \left(\begin{array}{l} \dot{x} = v\cos\phi - d\omega\sin\phi \\ \dot{y} = v\sin\phi - d\omega\cos\phi \\ \dot{\phi} = \omega \\ \dot{v} = u_1, \dot{\omega} = u_2 \end{array} \right) \tag{4.21}$$

$$\text{Path Constraint : } d_{collision} - d_{sep} \leq 0 \tag{4.22}$$

where, N_{ob} is the number of obstacles, t_0, t_f are initial and final time respectively, $\mathbf{U}$ is the control input matrix, P_i is the inverse of the distance between i^{th} obstacle and the vehicle as given in equation (4.15), $\mathbf{X}$ is the state vector matrix, v is the linear velocity, ω is the angular velocity, d is the distance between centre of mass point and the midpoint of axle connecting driving wheels and $d_{collision}$ and d_{sep} are as given in Equations (4.12) and (4.13) respectively.

The kinematic equations of the vehicle are given as dynamic constraints to the optimal control problem. An additional constraint on final values of the control inputs also given to ensure that control inputs reach 0. Then the boundary conditions are given in (4.23)

$$\text{Boundary conditions : } \mathbf{X}(t_0) = \mathbf{X}_0, \mathbf{X}(t_f) = \mathbf{X}_f, \mathbf{U}(t_f) = \mathbf{U}_f \tag{4.23}$$

4.3.1 Optimal Trajectory Planning to Minimise Time and Control Effort in an Environment with Dynamic Obstacle

Under the scope of this study, the movement of vehicle in dynamic environment is also considered. It is assumed that the obstacles are moving with uniform velocity.
Distance between vehicle and obstacle is given by,

$$d_{sep}(t) = \sqrt{(x_{ob} - x)^2 + (y_{ob} - y)^2} \tag{4.24}$$

$$d_{collision} = r_m + r_{ob} \tag{4.25}$$

Path constraint is given as,

$$d_{collision} - d_{sep}(t) \leq 0 \tag{4.26}$$

Coordinates of the pbstacle position are given as,

$$x_{ob}(t) = x_{ob}(0) + v_{ob}.t.cos(\alpha) \tag{4.27}$$

$$y_{ob}(t) = y_{ob}(0) + v_{ob}.t.sin(\alpha) \tag{4.28}$$

Proposed optimal trajectory planning problem to move the UGV from a given starting position to a required goal position within minimum time and with minimal control effort in an environment with dynamic obstacle is formulated as follows. Determine the control inputs to,

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathbf{U} * \mathbf{U}^{\mathrm{T}}) + \sum_{i=1}^{N_{ob}} P_i \mathrm{d}t \tag{4.29}$$

subjected to,

$$\text{Boundary conditions : } \mathbf{X}\left(t_0\right) = \mathbf{X}_0, \mathbf{X}\left(t_f\right) = \mathbf{X}_f, \mathbf{U}\left(t_f\right) = \mathbf{U}_f \tag{4.30}$$

$$\text{Control constraints : } \mathbf{U}_{\min} \leq \mathbf{U} \leq \mathbf{U}_{\max} \tag{4.31}$$

$$\text{State constraints : } \mathbf{X}_{\min} \leq \mathbf{X} \leq \mathbf{X}_{\max} \tag{4.32}$$

$$\text{Dynamic constraints : } \begin{pmatrix} \dot{x} = v\cos\phi - d\omega\sin\phi \\ \dot{y} = v\sin\phi - d\omega\cos\phi \\ \dot{\phi} = \omega \\ \dot{v} = u_1, \dot{\omega} = u_2 \end{pmatrix} \tag{4.33}$$

$$\text{Path Constraint : } d_{collision} - d_{sep} \leq 0 \tag{4.34}$$

where, N_{ob} is the number of obstacles, t_0, t_f are initial and final time respectively, $\mathbf{U}$ is the control input matrix, P_i is the inverse of the distance between i^{th} obstacle and the vehicle as given in equation (4.15), $\mathbf{X}$ is the state vector matrix, v is the linear velocity, ω is the angular velocity, d is the distance between centre of mass point and the midpoint of axle connecting driving wheels and $d_{collision}$ and d_{sep} are as given in Equations (4.25) and (4.24) respectively.

The above formulated problems are solved using Gauss Pseudo Spectral (PS) method which is described in the next chapter. The details of the methodology are explained in Chapter 5.

Chapter 5

Methodology-Chebyshev Pseudo Spectral Method for Solving the Optimal Control Problem

This chapter presents the Pseudo Spectral method used for solving the optimal trajectory design problem formulated in the previous chapter. Initially, a foundation on Pseudo spectral method is given. Then Chebyshev pseudo spectral method is described and then at the end a note on how the problem is solved using Chebyshev pseudo spectral method is given.

5.1 Introduction to Pseudo Spectral Optimal Control

Pseudo spectral methods were originally developed for the solution of partial differential equations and have become a widely applied computational tool in fluid dynamics and also optimal control problem[33]. Pseudo spectral methods are global because they use information over samples of the whole domain of the function to approximate its derivatives at selected points. Using these methods, the state and control functions are approximated as a weighted sum of smooth basis functions, which are often chosen to be Legendre or Chebyshev polynomials in the interval [-1 1], and collocation of the differential-algebraic equations is performed at orthogonal collocation points, which are selected to yield interpolation. One of the main advantages of pseudo spectral methods is the exponential (or spectral) rate of convergence, which is faster. Another advantage is the good accuracy due to the relatively coarse grids. In cases where global collocation is unsuitable (for example, when the solution exhibits discontinuities), multi-domain pseudo spectral techniques have been proposed, where the problem is divided into a number of sub-intervals and global collocation is performed along each sub-interval.

Pseudo spectral methods directly discretize the original optimal control problem to formulate a nonlinear programming problem, which is then solved numerically using a sparse nonlinear programming solver to find approximate local optimal solutions [34]. For differentiation, the derivatives of the state functions at the discretization nodes are easily computed by multiplying a constant differentiation matrix by a matrix with the state values at the nodes. Thus, the differential equations of the optimal control problem are approximated by a set of algebraic equations. The integration in the cost functional of an optimal control problem is approximated by well known Gauss quadrature rules, consisting of a weighted sum of the function values at the discretization nodes. Moreover, as is the case with other direct methods for optimal control, it is easy to represent state and control dependent constraints.

In this thesis, Chebyshev Pseudo Spectral method is used, in which the basis polynomial is the Chebyshev polynomial and interpolation nodes are the roots of Chebyshev Polynomial. Section 5.2 gives the fundamentals of the PS method irrespective of the basis polynomial and choice of nodes. The Chebyshev pseudo spectral method for optimal control problems was originally proposed in 1988 [35]. Fahroo and Ross proposed an alternative method for trajectory optimisation using Chebyshev polynomials [28].

5.2 Foundations of Pseudo Spectral Method

5.2.1 Time Domain Transformation

The physical domain of a problem may be as $t\epsilon[t_0, t_f]$, but in PS methods, this domain is transformed into a computational domain using the equation (5.1) as $t\epsilon[-1, 1]$.

$$\tau = \frac{2}{t_f - t_0}t - \frac{t_f + t_0}{t_f - t_0} \tag{5.1}$$

5.2.2 Interpolation and the Lagrange polynomial

That if $\tau_0, \tau_1, \ldots, \tau_N$ are $N + 1$ distinct numbers and f is a function whose values are given at those numbers, then a unique polynomial $p(\tau)$ of degree at most N exists with,

$$f(\tau_k) = P(\tau_k)\,, \text{ for } k = 0, 1, \ldots, N \tag{5.2}$$

Then polynomial is given by

$$P(\tau) = f(\tau) = \sum_{k=0}^{N} f(\tau_k)\, \varphi_k(\tau) \tag{5.3}$$

where, $P(\tau)$ is known as the Lagrange interpolating polynomial and $\varphi_k(\tau)$ are known as Lagrange basis polynomials. Here N is the order of basis polynomial used.

5.2.3 Differentiation Approximation

Derivative of the function $f(\tau)$ can be approximated by multiplying with Differentiation matrix of order $(N+1)X(N+1)$.

$$\dot{f}(\tau_k) \approx \dot{F}^N(\tau_k) = \sum_{i=0}^{N} D_{ki} f(\tau_i) \tag{5.4}$$

5.2.4 Numerical Quadrature

The transformation in (5.1) changes the integral term as,

$$\int_{t_0}^{t_f} f(t)dt = \frac{tf - t_0}{2} \int_{-1}^{1} f(\tau)d\tau \tag{5.5}$$

Then the integral term in cost function can be approximated as,

$$\int_{-1}^{1} f(\tau) \approx \sum_{k=0}^{N} \omega_k f(\tau_k) \tag{5.6}$$

5.2.5 Discretization Matrices and Optimization Variable Vector

For the sake of simplicity, For simplicity, we denote the derivative function and the time step of the entire time horizon as:

$$\mathbf{f_d}(\tau) \rightleftharpoons \mathbf{f}_{\mathrm{d}}(\tau, \boldsymbol{\xi}(\tau), \mathbf{u}(\tau)) \tag{5.7}$$

$$h = t_f - t_0 \tag{5.8}$$

$$\text{state}, \Xi = \begin{bmatrix} \xi(\tau_0) \\ \vdots \\ \xi(\tau_{N_t}) \end{bmatrix} = \begin{bmatrix} \xi_1(\tau_0) & \cdots & \xi_{n_\xi}(\tau_0) \\ \vdots & \ddots & \vdots \\ \xi_1(\tau_{N_t}) & \cdots & \xi_{n_\xi}(\tau_{N_t}) \end{bmatrix}_{(N_t+1)\times n_\xi} \tag{5.9}$$

$$\text{control}, \mathbf{U} = \begin{bmatrix} \mathbf{u}(\tau_0) \\ \vdots \\ \mathbf{u}(\tau_{N_t}) \end{bmatrix} = \begin{bmatrix} u_1(\tau_0) & \cdots & u_{n_u}(\tau_0) \\ \vdots & \ddots & \vdots \\ u_1(\tau_{N_t}) & \cdots & u_{n_u}(\tau_{N_t}) \end{bmatrix}_{(N_t+1)\times n_u} \tag{5.10}$$

$$\text{Function f}, \mathbf{F} = \frac{h}{2}\begin{bmatrix} \mathrm{f_d}(\tau_0) \\ \vdots \\ \mathrm{f_d}(\tau_{N_t}) \end{bmatrix} = \frac{h}{2}\begin{bmatrix} f_{d_1}(\tau_0) & \cdots & f_{d_{n_\xi}}(\tau_0) \\ \vdots & \ddots & \vdots \\ f_{d_1}(\tau_{N_t}) & \cdots & f_{d_n}(\tau_{N_t}) \end{bmatrix}_{(N_t+1)\times n_\xi} \tag{5.11}$$

$$\text{path constraint, C} = \begin{bmatrix} \mathrm{C}(\tau_0) \\ \vdots \\ \mathrm{C}(\tau_{N_t}) \end{bmatrix} = \begin{bmatrix} C_1(\tau_0) & \cdots & C_{n_C}(\tau_0) \\ \vdots & \ddots & \vdots \\ C_1(\tau_{N_t}) & \cdots & C_{n_C}(\tau_{N_t}) \end{bmatrix}_{(N_t+1)\times n_C} \tag{5.12}$$

$$\text{Boundary}, \phi = \left[\phi_1 \cdots \ \phi_{n_\phi}\right]_{1\times n_\phi} \tag{5.13}$$

where n_ξ, n_u, n_C, and n_ϕ are the number of states, controls, path constraints, and boundary constraints, respectively.

5.2.6 Defect Constraints

To ensure that the approximation for the state derivatives using equation (5.4)is equivalent to the derivative function values given by $f_d(\tau)$, a defect constraint is defined as:

$$\zeta = 0 = \mathrm{D}\Xi - \mathrm{F} \tag{5.14}$$

where, **D** is the differentiation matrix and Ξ is is the matrix of discretized state values, and **F** is the matrix of derivative function values.

Objective Function

The Lagrange and Mayer terms are approximated as follows:

$$\sum_{k=0}^{N} w_k \mathcal{L}\left(t_k, \boldsymbol{\xi}(t_k), \mathbf{u}(t_k)\right) = \frac{h}{2}\sum_{k=0}^{N} w_k \mathcal{L}\left(\tau_k, \boldsymbol{\xi}(\tau_k), \mathbf{u}(\tau_k)\right) \tag{5.15}$$

$$\mathcal{M}\left(t_0, \boldsymbol{\xi}(t_0), t_f, \boldsymbol{\xi}(t_{N_t})\right) = \mathcal{M}\left(t_0, \boldsymbol{\xi}(-1), t_f, \boldsymbol{\xi}(1)\right) \tag{5.16}$$

5.3 Chebyshev Pseudo Spectral Method with CGL Nodes

This section describes the fundamentals required for solving an optimal control problem using Chebyshev Pseudo Spectral (CPS) method. The selection of nodes, Lagrange basis polynomial etc are given in coming sections.

5.3.1 Nodes

For CPS method, nodes are selected as the roots of derivative of Chebyshev polynomial. These nodes are named as Chebyshev Gauss Lobatto (CGL) nodes since it include both the end points according to the quadrature rule [27]. CGL nodes give accurate interpolation since Lebesgue constant for CGL nodes is very less compared with other choice of nodes.
Chebyshev polynomial of order N is given by,

$$T_N = \cos\left(N\cos^{-1}(\tau)\right) \tag{5.17}$$

Thus CGL nodes are given by,

$$\tau_k = -\cos\left(\frac{\pi k}{N}\right) \in [-1, 1] \tag{5.18}$$

These nodes are always between $[-1, 1]$ and contains both the end points.

5.3.2 Interpolating Basis Function

With CGL nodes, Lagrange basis polynomials $\phi_k(\tau)$ in (5.3) is given by,

$$\phi_k(\tau) = \frac{(-1)^{k+1}}{N^2 a_k}\frac{(1-\tau^2)}{\tau - \tau_k}\dot{T_N}(\tau) \tag{5.19}$$

$$\text{where, } a_k = \begin{cases} 2 & k = \{0, \alpha\}\} \\ 1 & \text{otherwise} \end{cases}$$

5.3.3 Differentiation

The differentiation matrix needed in (5.4) for CPS method is given as,

$$D_{ki} = \begin{cases} \frac{a_k}{a_i}\frac{(-1)^{k+i}}{(\tau_k - \tau_i)} & \text{if } k \neq i \\ -\frac{\tau_k}{2\left(1-\tau_k^2\right)} & \text{if } 1 \leq k = i \leq N-1 \\ \frac{2N^2+1}{6} & \text{if } k = i = 0 \\ -\frac{2N^2+1}{6} & \text{if } k = i = N \end{cases} \tag{5.20}$$

5.3.4 Quadrature Weights

The quadrature weights needed in (5.6) is given as,

$$w_k = \frac{c_k}{N}\left(1 - \sum_{j=1}^{\lfloor N/2 \rfloor}\frac{b_j}{4j^2-1}\cos(2j\tau_k)\right) \tag{5.21}$$

$$\text{where: } b_j = \begin{cases} 1 & \text{if } j = N/2 \\ 2 & \text{if } j < N/2 \end{cases}, \quad c_k = \begin{cases} 1 & \text{if } k = \{0, N_t\} \\ 2 & \text{otherwise} \end{cases}$$

5.4 Algorithm to solve Optimal Control Problem

To solve an optimal control problem using CPS method, following steps are followed.

1. Generate a mesh : Chose desired order for polynomial
2. Compute the CGL nodes
3. Compute quadrature weights, Differentiation matrices for CGL nodes
4. Transcribe as a Non linear programming problem
5. Solve Non linear programming problem
6. Untranscribe to discrete optimal control problem
7. Check for error tolerances, If tolerances are not met, go for new mesh with another ordered polynomial and repeat the steps

The problem formulated in previous chapter can be numerically solved using the Chebyshev Pseudo spectral method described in this chapter by making use of any commercial numerical solvers. For this research work, fmincon with SQP algorithm in MATLAB is used to solve the problem and simulation results are given in next chapter.

Chapter 6

Simulation Results and Analysis

Using Chebyshev PS method described in previous chapter, the optimal trajectory design problem is solved in MATLAB with the help of fmincon. Simulations are done for different test cases and optimal solutions are found for each case. Simulation parameters and the results obtained for each case is given in the preceding section of this chapter.

6.1 Simulation Parameters

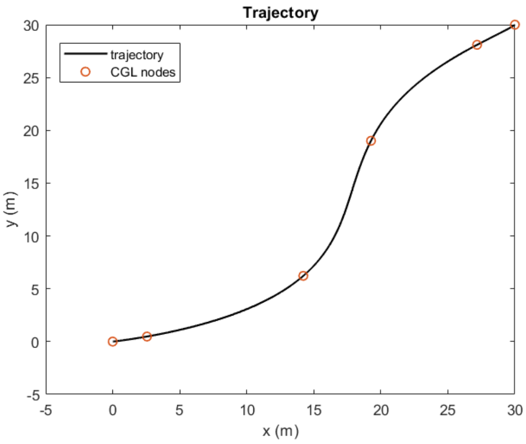
Simulations are done in MATLAB using fmincon with Sequential Quadratic Programming (SQP) algorithm. Initial time is considered to be zero and the vehicle is assumed to be at rest initially. For all the cases, it is supposed that the vehicle has to move from an initial position (0,0) to a final destination (30,30) and it is also supposed that the vehicle should reach it's final destination within a maximum time of 100 s. An initial guess of linear trajectory in which vehicle moves with a velocity of 1 m/s is assumed.

6.2 Obstacle Free Environment

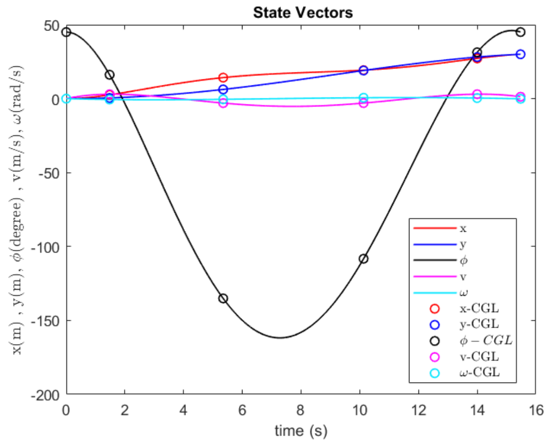
First of all, an optimal trajectory for the minimum final time is simulated considering an obstacle-free environment. Two different combinations of the initial and final orientation of the vehicle have been chosen to understand whether there exists only one optimum solution or more.

Figure 6.1 shows the trajectory obtained, state vectors and control inputs for the combination of initial and final angles as $\phi_0 = 45^0$, $\phi_f = 45^0$ and figure 6.2 is when $\phi_0 = 60^0$, $\phi_f = 30^0$.

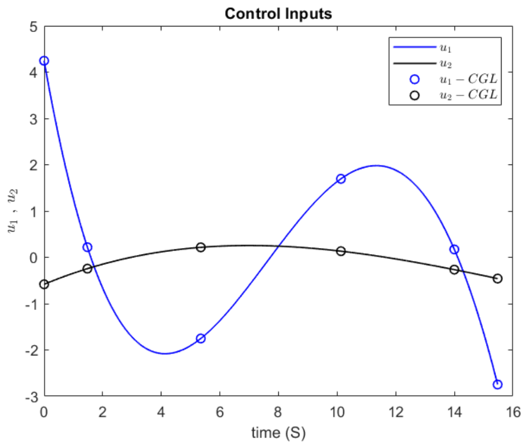
optimal solution is obtained for 6 number of nodes in both cases. Two different trajectories are obtained for moving the vehicle form same initial position to the same goal position as the initial and final orientation changes. So the trajectory obtained is sensitive towards the initial and final orientation of the UGV.



(a) trajectory of the UGV



(b) state vectors

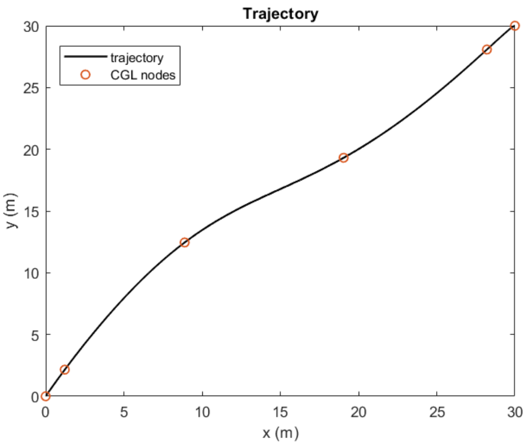


(c) control inputs

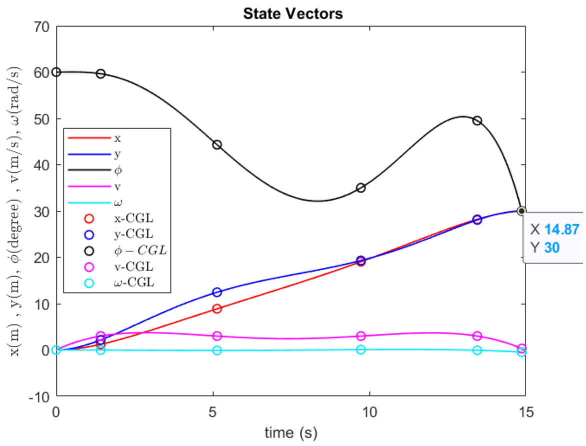
Figure 6.1: optimum solution obtained with 6 nodes, $\phi_0 = 45^0$, $\phi_f = 45^0$

6.3 Environment with Static Obstacles

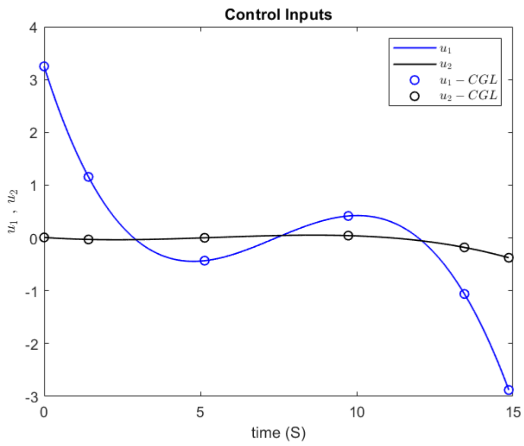
Different scenarios of obstacle ridden environment including large and small obstacles, each placed at different locations are tested and optimum final time solutions are plotted. A large obstacle is considered which has inflated radius greater than 2m and placed close to the end points and in midway of the path. In figure 6.3, obstacle is placed at midway of the trajectory



(a) trajectory of the UGV



(b) state vectors

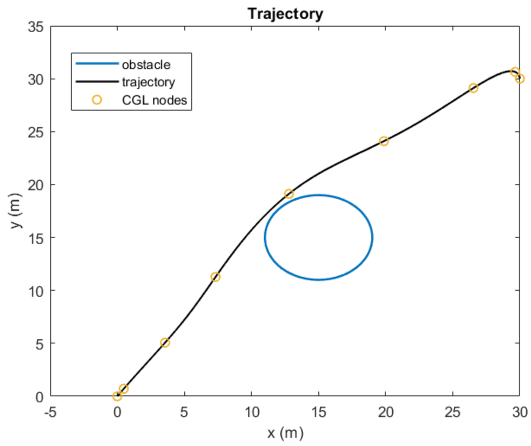


(c) control inputs

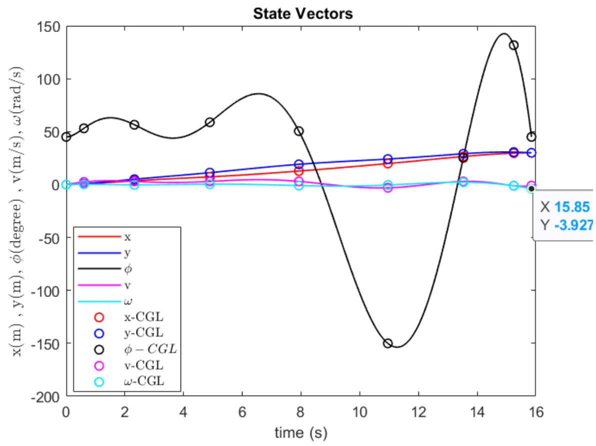
Figure 6.2: optimum solution obtained with 6 nodes, $\phi_0 = 60^0$, $\phi_f = 30^0$

and it is successfully avoided. Corresponding control inputs and state vectors are given. Time taken to reach the goal position is around 16 seconds. Same scenario tested with different combination of initial and final vehicle orientation is given in figure 6.4. A different optimal trajectory is generated. More number of nodes are required for the scenario which contains large obstacle at the end points which is shown in figure 6.5 and figure 6.6. For these cases, control inputs are more oscillatory. While testing different scenarios, the trajectory generated failed to avoid obstacles in many cases. Thus the algorithm is not much suitable for environment with large obstacles.

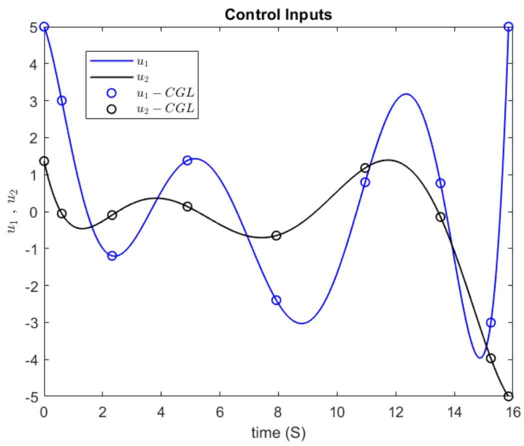
Then after, small obstacles are considered at different positions including close to both the end points and midway of the trajectory. Different trajectories are obtained as given in figure 6.7. Optimum trajectories generated for an environment having three small obstacles (radius <



(a) trajectory of the UGV



(b) state vectors



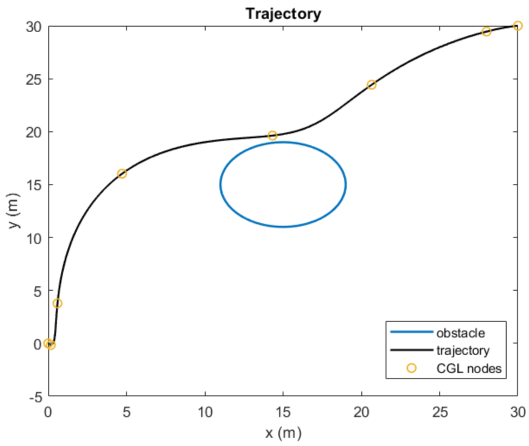
(c) control inputs

Figure 6.3: Large obstacle in the middle of the path, $\phi_0 = 45^0$, $\phi_f = 45^0$

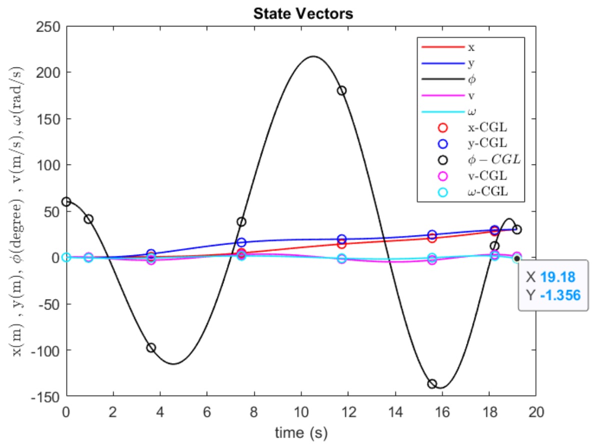
(optimum solution obtained with 9 nodes)

1.5 m) are given in figure 6.8. Since it is clear from the plots that the algorithm works well for small obstacles, small obstacles are considered for the coming sections.

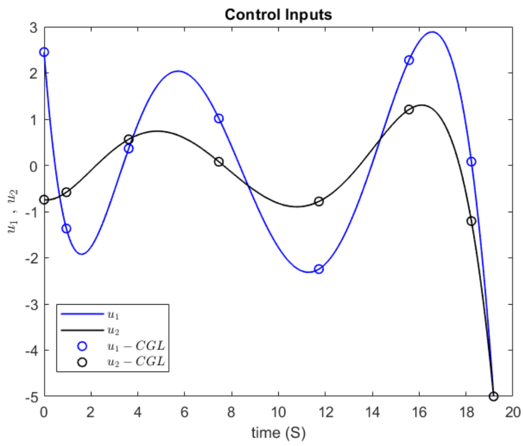
As the algorithm seems to be efficient for time minimization, it is upgraded to design a trajectory with minimal control effort by improving the cost function as in equation (4.10). Optimal trajectory is obtained with 9 nodes. Corresponding trajectory and control inputs are plotted in figure 6.9. Improved capability of the proposed algorithm that formulated as optimal trajectory design problem as in section 4.3 is visualised in figure 6.10 . To ensure stability of the system, constraints are added to control input values at end point as in equation (4.23). Figure: 6.11 shows the optimum trajectory and corresponding control and state vectors plots for the proposed optimal trajectory design problem. Convergence of the objective function is plotted



(a) trajectory of the UGV



(b) state vectors



(c) control inputs

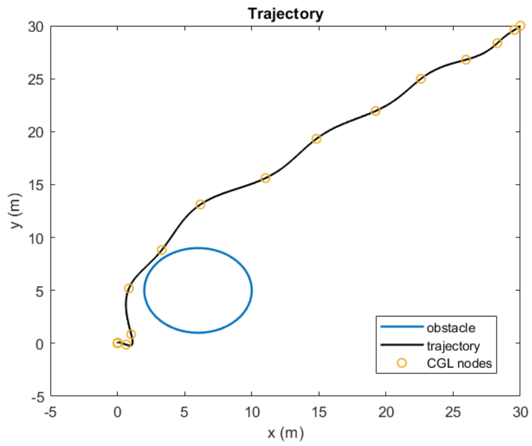
Figure 6.4: Large obstacle in the middle of the path, $\phi_0 = 60^0$, $\phi_f = 30^0$ (optimum solution obtained with 9 nodes)

using MATLAB optimization toolbox function FuncPlot. Exponential/spectral convergence of the PS methods is clear in the figure (figure 6.12).

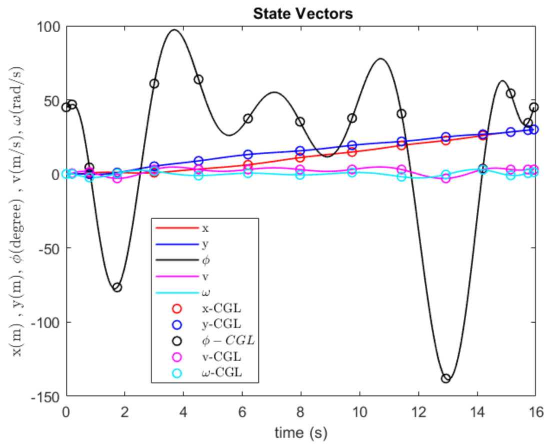
6.4 Environment with Dynamic Obstacle

First, optimal trajectory for the UGV to reach the desired goal position with minimum time while avoiding a dynamic obstacle is simulated. Relative distance between obstacle and the UGV is plotted with respect to the time in figure 6.13. Some snapshots of the animation for obstacle and UGV movement are given in figure 6.14.

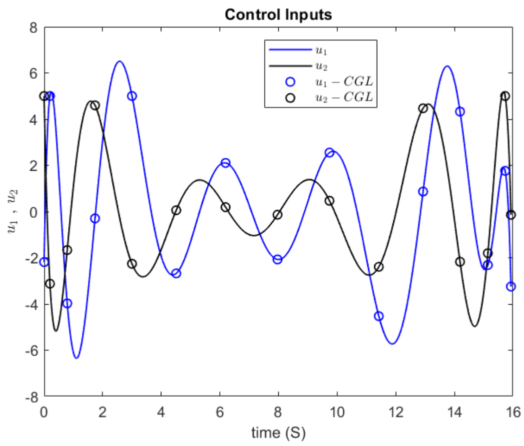
Feasibility of the optimal trajectory design problem proposed in section 4.3.1, relative distance between the vehicle and obstacle with respect to time is plotted 6.15. For all the



(a) trajectory of the UGV



(b) state vectors



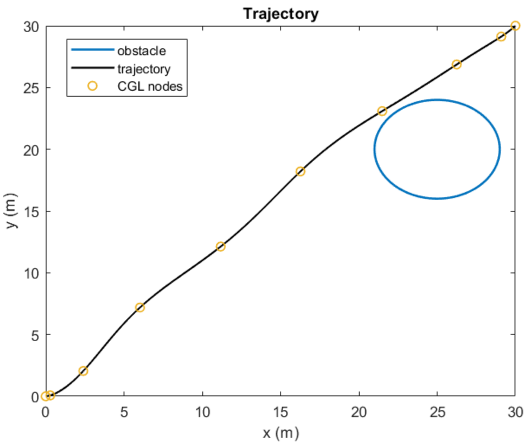
(c) control inputs

Figure 6.5: Large obstacle close to end points, $\phi_0 = 45^0$, $\phi_f = 45^0$ (optimum solution obtained with 14 nodes)

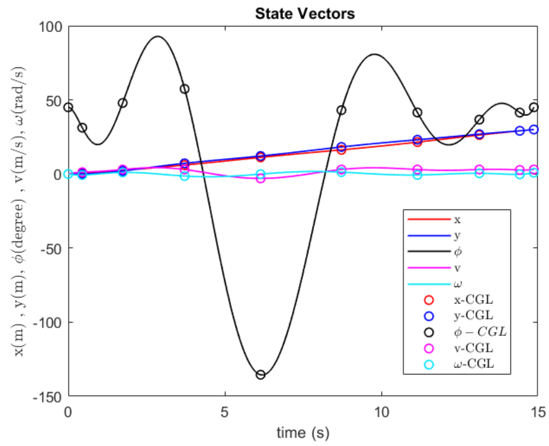
time, there is a distance between UGV and obstacle, so the trajectory will successfully avoid obstacles.

6.5 Sensitivty of Number of Nodes

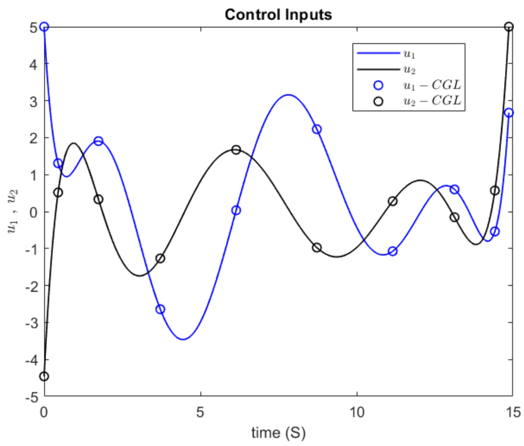
Different test cases are simulated in MATLAB, for all cases solution is converged to a feasible point and local minima is found. More number of nodes are seen to be concentrated on the starting and goal position. So if the configuration space is considered as an obstacle driven one, then the accuracy will be increased as the number of nodes increased. The computational time and the final time obtained for different number of nodes corresponding to some random test



(a) trajectory of the UGV



(b) state vectors

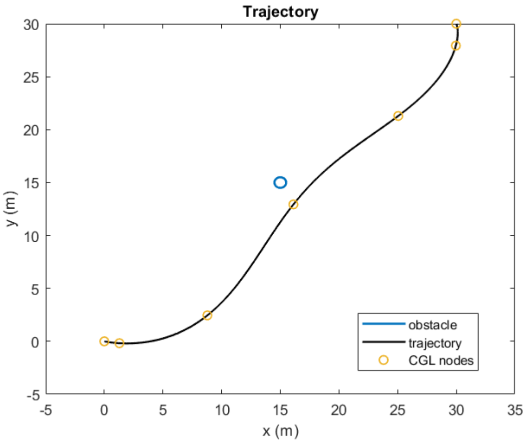


(c) control inputs

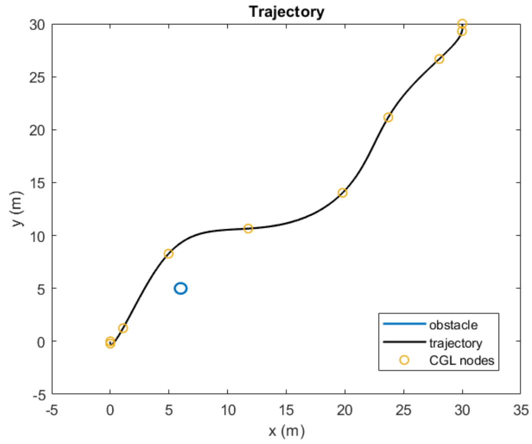
Figure 6.6: Large obstacle close to end points, $\phi_0 = 45^0$, $\phi_f = 45^0$

(optimum solution obtained with 14 nodes)

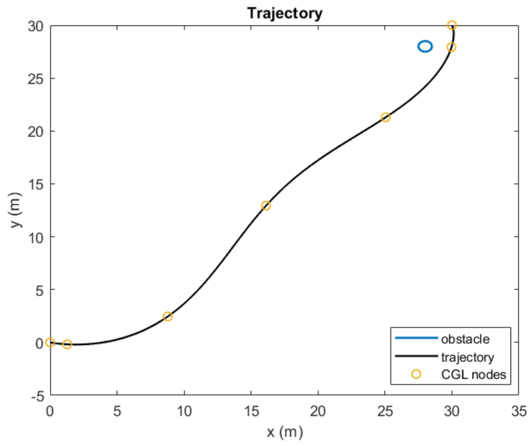
cases are listed in table 6.1. As the number of nodes increases, the control inputs become more oscillatory as shown in figure 6.16



(a) obstacle at midway



(b) obstacle close to start



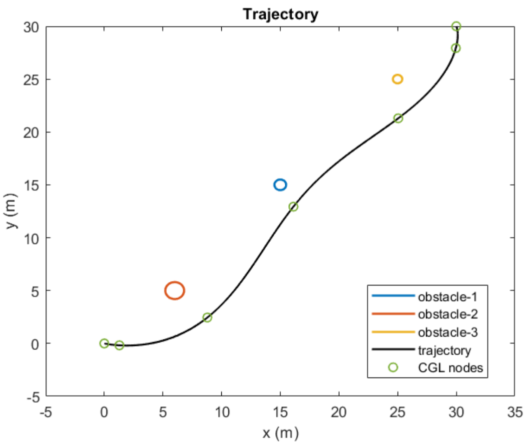
(c) obstacle close to goal

Figure 6.7: optimum trajectories for different scenarios with small obstacles

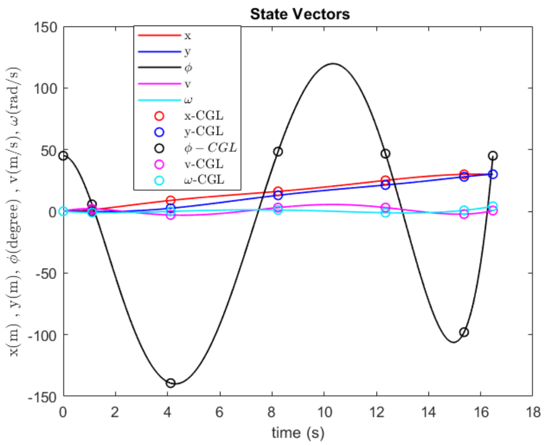
TEST CASES	**4 NODES**	**5 NODES**	**6 NODES**	**8 NODES**	**9 NODES**
Case 1	T=1.61384	T= 0.9264	T=2.9785	T = 2.79	T=3.8339
	f =25.5270	f = 19.137	f =4.8389	f = 6.77	f = 5.9557
Case 2	T=1.17801	T= 1.3104	T=2.8735	T= 3.3956	T=4.66828
	f =5.0666	f = 3.317	f =6.8866	f = 3.263	f =3.24691
Case 3	T=3.77467	T=1.5987	T=3.05884	T= 3.0069	T=4.684330
	f =11.8159	f = 15.195	f =16.3829	f = 12.902	f =14.0898

Table 6.1: Comparison of objective function and computational time

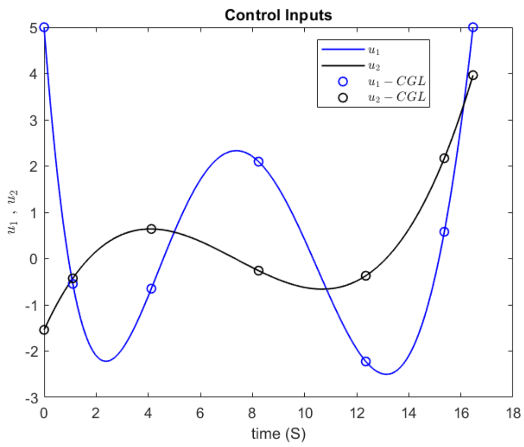
T : computational time, f : objective function or final time



(a) trajectory of the UGV



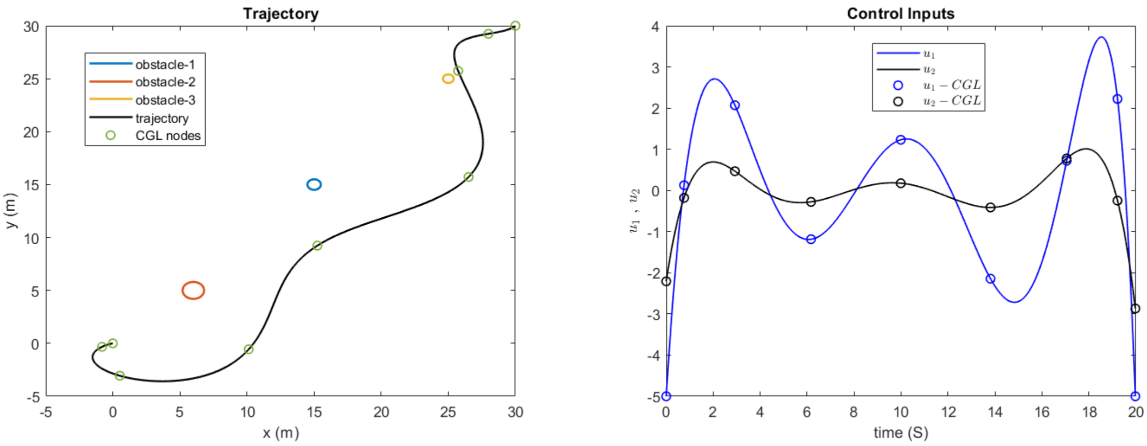
(b) state vectors



(c) control inputs

Figure 6.8: Multiple Obstacles

(optimum solution obtained with 7 nodes)

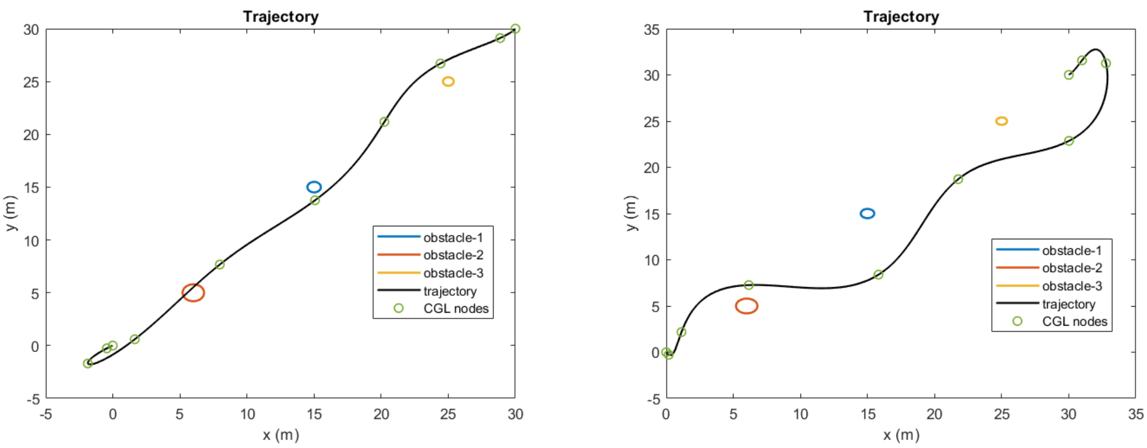


(a) trajectory of the UGV

(b) control inputs

Figure 6.9: Minimum final time with minimal control effort

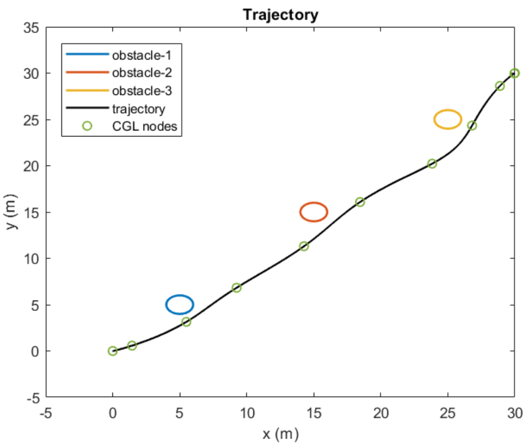
(optimum solution obtained with 9 nodes)



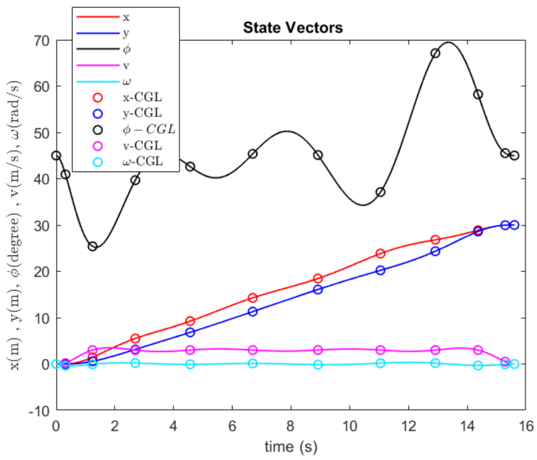
(a) failed trajectory before cost function updation

(b) trajectory after cost function updated

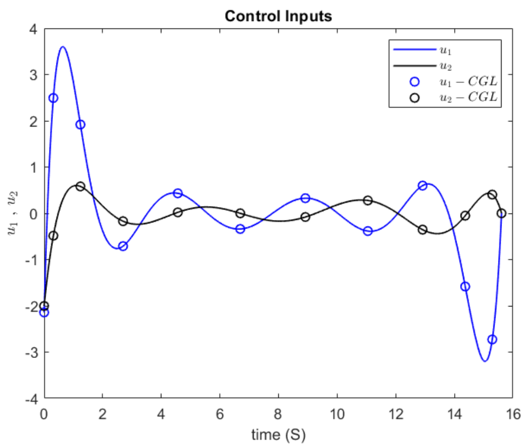
Figure 6.10



(a) trajectory of the UGV



(b) state vectors



(c) control inputs

Figure 6.11: optimum solution obtained with 12 nodes

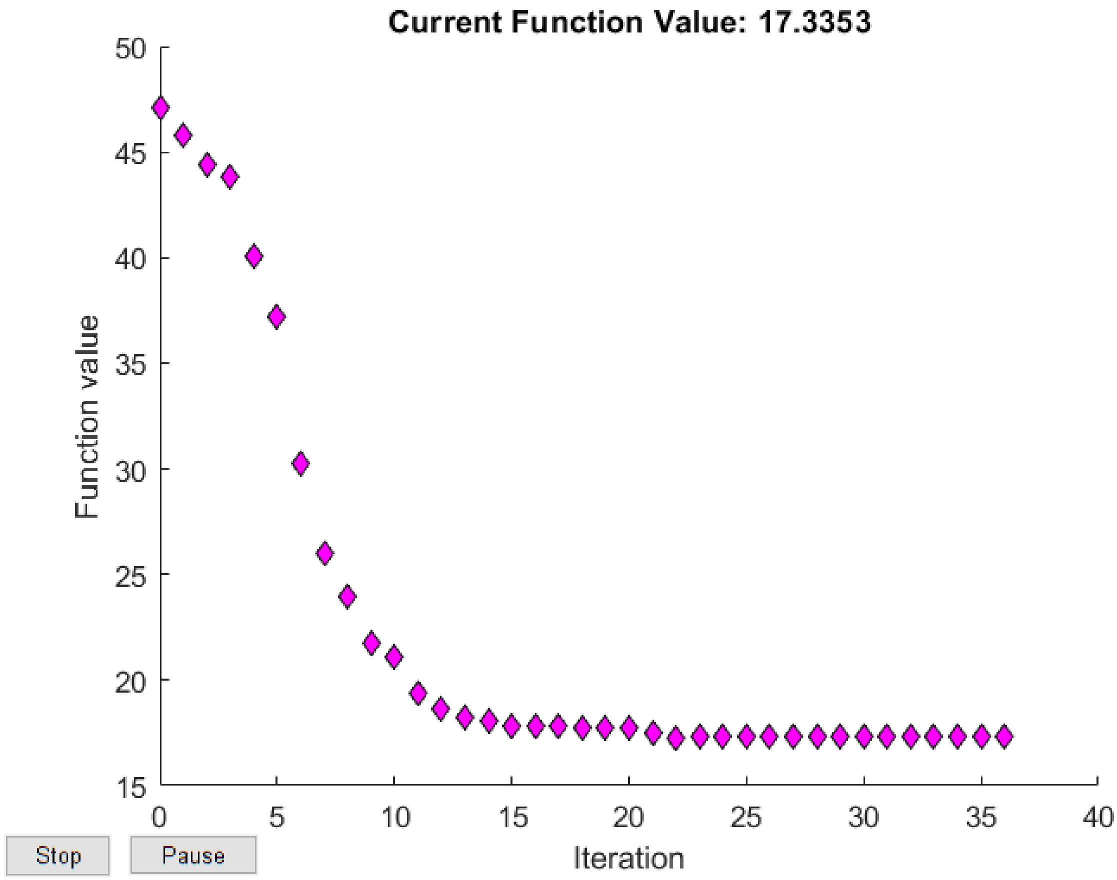


Figure 6.12: Convergence of function versus iteration

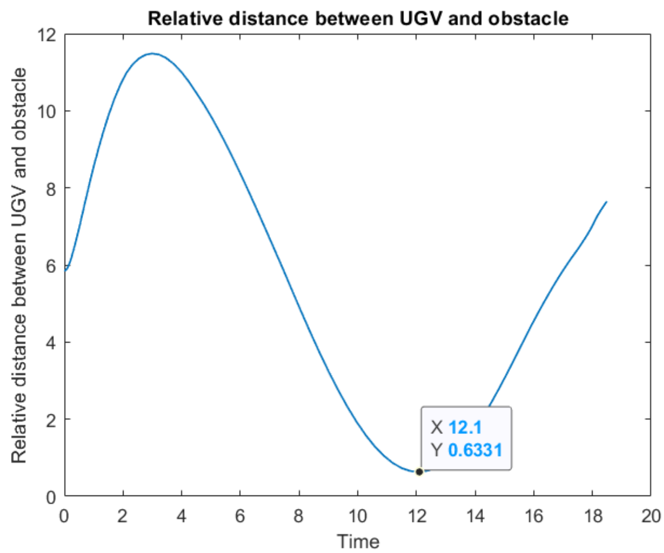
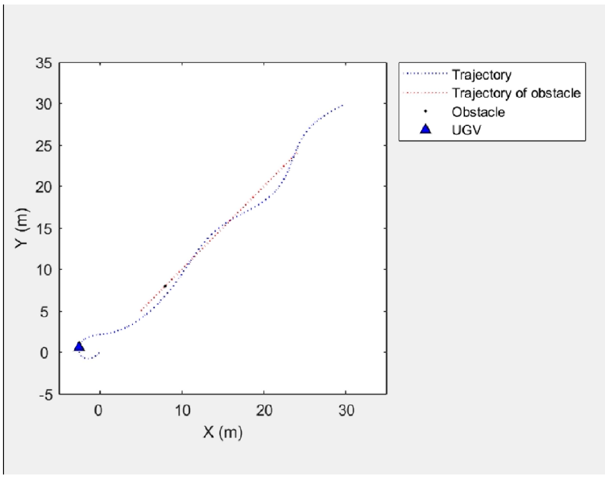


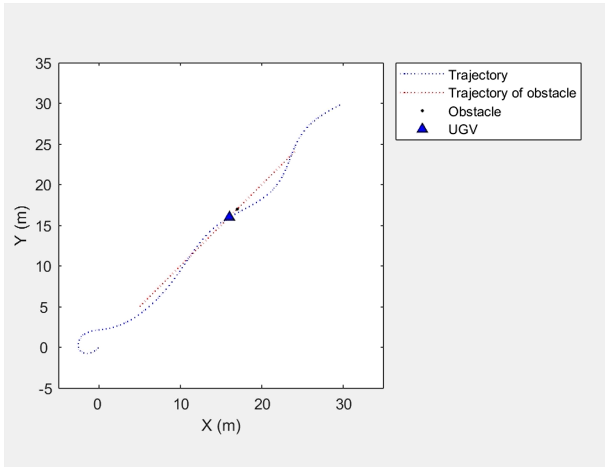
Figure 6.13: Relative distance between obstacle and vehicle versus time

Vehicle : start (0,0) to goal (30,30) with $\phi_0 = \phi_f = \pi/4$

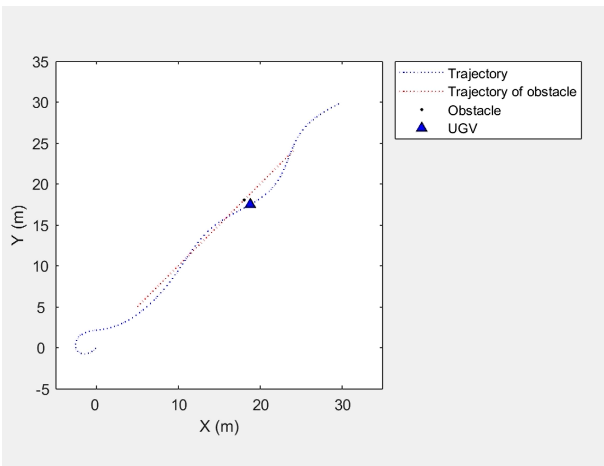
obstacle : initial position (5,5), with velocity=1.5 m/s, direction $\alpha = \pi/4$



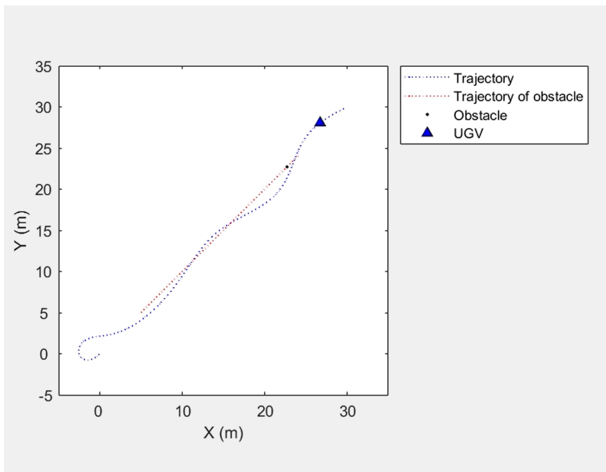
(a)



(b)



(c)



(d)

Figure 6.14: Snapshots from animation for dynamic obstacle avoidance with time minimisation

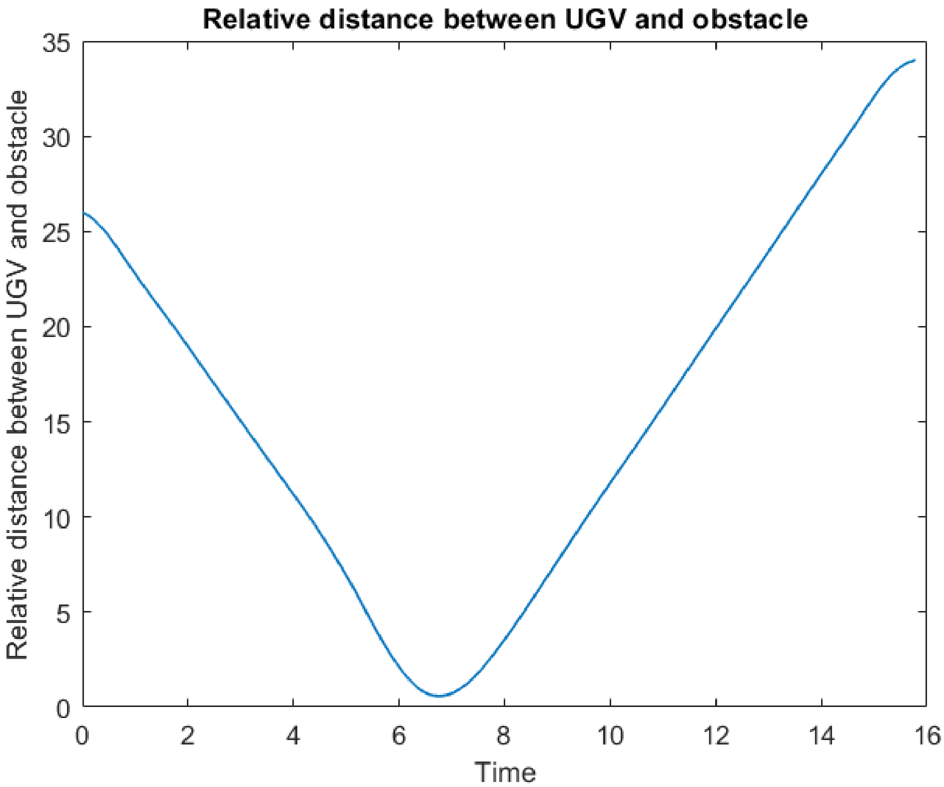
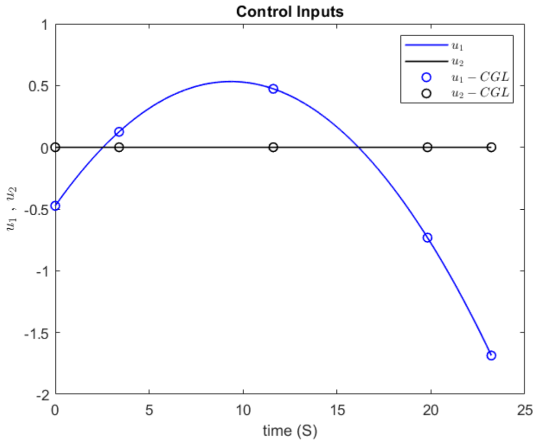
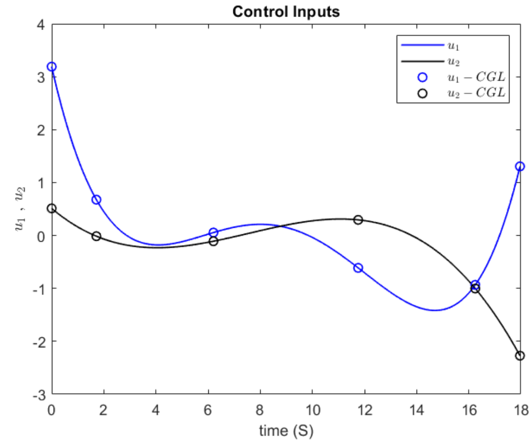


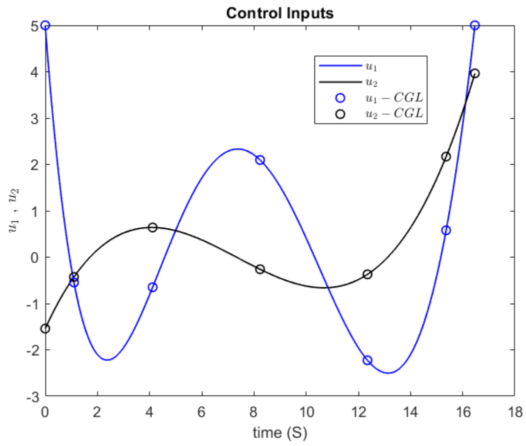
Figure 6.15: Relative distance between UGV and obstacle



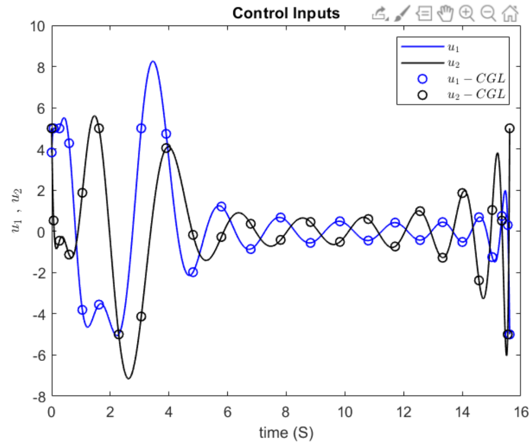
(a) 5 nodes



(b) 6 nodes



(c) 7 nodes



(d) 25 nodes

Figure 6.16: Control inputs for different no of nodes

Chapter 7

Conclusions and Future Scope

This thesis formulates an optimal trajectory design problem for an UGV to optimize final time, control effort and avoid static and dynamic obstacles. The problem is converted as numerical programming problem by using Chebyshev Pseudo Spectral method and thus it is solved by using Sequential Quadratic Programming (SQP) algorithm. MATLAB function fmincon is used for optimization. According to the initial and final orientation of the vehicle, different optimal solutions can be formed. Smooth and fast solutions are obtained for small obstacle ridden environments. Trajectory with minimum time and with minimum control effort is designed. Even if there is an increase in computation cost as the no of nodes increases, it can be further improved by optimizing code and eliminating MATLAB overhead.

7.1 Contribution of the Thesis

In this thesis, optimal trajectory planning problem to move the UGV from a given starting position to a required goal position within minimum time and with minimal control effort in static obstacle ridden environment and dynamic obstacle ridden environment are formulated. These optimal trajectory planning problems are solved using Chebyshev Pseudo Spectral method. Simulation results corresponding to different scenarios are studied.

7.2 Future Scope

CGL nodes are used because of the less optimization error for interpolation using CGL nodes. But, Legendre basis polynomial give more stability than the Chebyshev basis polynomial. Combining these two facts to obtain more stable but accurate solutions can be considered in future as a modification of this work. Vehicle dynamics also can be considered for improving the performance.

References

[1] C. Qingyang, S. Zhenping, L. Daxue, F. Yuqiang, and L. Xiaohui, “Local path planning for an unmanned ground vehicle based on svm,” *International Journal of Advanced Robotic Systems*, vol. 9, no. 6, p. 246, 2012. [Online]. Available: https://doi.org/10.5772/54130

[2] A. Mohamed, J. Ren, A. Sharaf, and M. El-Gindy, “Optimal path planning for unmanned ground vehicles using potential field method and optimal control method,” *International Journal of Vehicle Performance*, vol. 4, pp. 1–14, 01 2018.

[3] Almayyahi and Auday, “Motion control design for unmanned ground vehicle in dynamic environment using intelligent controller,” *International Journal of Intelligent Computing and Cybernetics*, 2017.

[4] X. Xu, Y. Yang, and S. Pan, “Motion planning for mobile robots,” in *Advanced Path Planning for Mobile Entities*, R. Róka, Ed. Rijeka: IntechOpen, 2018, ch. 8. [Online]. Available: https://doi.org/10.5772/intechopen.76895

[5] J. T. Betts, “Survey of numerical methods for trajectory optimization,” *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998. [Online]. Available: https://doi.org/10.2514/2.4231

[6] M. R. Jabbarpour, H. Zarrabi, J. J. Jung, and P. Kim, “A green ant-based method for path planning of unmanned ground vehicles,” *IEEE Access*, vol. 5, pp. 1820–1832, 2017.

[7] D. Shin and S. Singh, “Path generation for robot vehicles using composite clothoid segments,” *Preprints of the IFAC SICICA ’92*, vol. 25, p. 35, 12 1990.

[8] J. H. Reif, “Complexity of the mover’s problem and generalizations,” in *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*, 1979, pp. 421–427.

[9] M. H. Overmars and E. Welzl, "New methods for computing visibility graphs," in *Proceedings of the Fourth Annual Symposium on Computational Geometry*, ser. SCG '88. New York, NY, USA: Association for Computing Machinery, 1988, p. 164–171. [Online]. Available: https://doi.org/10.1145/73393.73410

[10] A. Gasparetto, P. Boscariol, A. Lanzutti, and R. Vidoni, "Path planning and trajectory planning algorithms: A general overview," *Mechanisms and Machine Science*, vol. 29, pp. 3–27, 03 2015.

[11] X.-H. Ni, Z.-Q. Jiang, and W.-X. Zhou, "Degree distributions of the visibility graphs mapped from fractional brownian motions and multifractal random walks," *Physics Letters A*, vol. 373, no. 42, p. 3822–3826, Oct 2009. [Online]. Available: http://dx.doi.org/10.1016/j.physleta.2009.08.041

[12] J. Canny, "A voronoi method for the piano-movers problem," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 530–535.

[13] "Delaunay refinement algorithms for triangular mesh generation," *Computational Geometry*, vol. 22, no. 1, pp. 21 – 74, 2002, 16th ACM Symposium on Computational Geometry. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925772101000475

[14] M. Elbanhawi, M. Simic, and R. Jazar, "Autonomous robot path planning: An adaptive roadmap approach," *Applied Mechanics and Materials*, vol. 373-375, p. 9, 08 2013.

[15] E. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, 1959.

[16] K. Jeddisaravi, R. J. Alitappeh, and F. G. Guimarães, "Multi-objective mobile robot path planning based on a* search," in *2016 6th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2016, pp. 7–12.

[17] "Anytime search in dynamic graphs," *Artificial Intelligence*, vol. 172, no. 14, pp. 1613 – 1643, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000437020800060X

[18] D. Z. Chen, R. J. Szczerba, and J. J. Uhran, "A framed-quadtree approach for determining euclidean shortest paths in a 2-d environment," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 668–681, 1997.

[19] H.-y. Zhang, W.-m. Lin, and A.-x. Chen, “Path planning for the mobile robot: A review,” *Symmetry*, vol. 10, no. 10, 2018. [Online]. Available: https://www.mdpi.com/2073-8994/10/10/450

[20] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, “A survey of motion planning and control techniques for self-driving urban vehicles,” *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[21] A. Atyabi and D. Powers, “Review of classical and heuristic-based navigation and path planning approaches,” *International Journal of Advancements in Computing Technology (IJACT)*, vol. 5, no. 14, 2013.

[22] “Matrix-binary codes based genetic algorithm for path planning of mobile robot,” *Computers Electrical Engineering*, vol. 67, pp. 708 – 728, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S004579061630996X

[23] R. Mao, H. Gao, and L. Guo, “Optimal motion planning for differential drive mobile robots based on multiple-interval chebyshev pseudospectral methods,” *Robotica*, p. 1–20, 2020.

[24] “A hybrid backtracking search optimization algorithm for nonlinear optimal control problems with complex dynamic constraints,” *Neurocomputing*, vol. 186, pp. 182 – 194, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092523121502038X

[25] B. A. E, *Applied optimal control: optimization, estimation and control*, 1st ed. CRC press, 1975.

[26] Iyer and S. V, “Time optimal trajectory generation for a differential drive robot,” *State University of New York at Bu*$_a lo, 2009.$

[27] A. Rao, “A survey of numerical methods for optimal control,” *Advances in the Astronautical Sciences*, vol. 135, 01 2010.

[28] F. Fahroo and I. M. Ross, “Direct trajectory optimization by a chebyshev pseudospectral method,” *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 160–166, 2002. [Online]. Available: https://doi.org/10.2514/2.4862

[29] C. de Boor, "Collocation at gaussian points," *Siam Journal on Numerical Analysis - SIAM J NUMER ANAL*, vol. 10, pp. 582–606, 09 1973.

[30] J.-J. Kim and J.-J. Lee, "Trajectory optimization with particle swarm optimization for manipulator motion planning," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 620–631, 2015.

[31] F. Fahroo and I. M. Ross, "Direct trajectory optimization by a chebyshev pseudospectral method," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 160–166, 2002.

[32] I. M. Ross and F. Fahroo, "Legendre pseudospectral approximations of optimal control problems," in *New Trends in Nonlinear Dynamics and Control and their Applications*, W. Kang, C. Borges, and M. Xiao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 327–342.

[33] C. Canuto, M. Hussaini, A. Quarteroni, and T. Zang, "Spectral methods," 01 2006.

[34] V. M. Becerra, "Psopt optimal control solver user manual," Available at.

[35] J. Vlassenbroeck and R. V. D. A, "Chebyshev technique for solving nonlinear optimal control problems," *IEEE Transactions on Automatic Control*, 1988.