Optimal Trajectory Planning for a Differentially Driven Unmanned Ground Vehicle in an Obstacle Ridden Environment

MUAFIRA THASNI K.T
TVE19EEGN12
Guided by: Dr. V R Jisha
Mrs. Gifty Ernestina Benjamin (Sci/Engr 'SF', VSSC)

Mr. Venkateswaran J (Sci/Engr 'SC', VSSC)

Department of Electrical Engineering
College of Engineering Trivandrum

July 15, 2021

Overview

Introduction

- Unmanned Ground Vehicles (UGV)
- Military and civilian applications
- Self driving vehicles with sensors
- Limited or no human interventions
- Path planning or trajectory planning
- Challenges :
    - road condition
    - uncertainty in the vehicle states
    - limited knowledge on environment
    - moving obstacles

Literature Review

Sl No:	PAPER DETAILS	OBJECTIVES	METHODOLOGY	REMARKS
1	Elbanhawi, Mohamed & Simic, Milan & Jazar, Reza. (2013). Autonomous Robot Path Planning: An Adaptive Roadmap Approach. Applied Mechanics and Materials. 373-375.	To do path planning for autonomous robot	Road map based method	Point mass assumption
2	Scan Matching Online Cell Decomposition for Coverage Path Planning in an Unknown Environment, Batsaikhan Dugarjav 2013	To guarantee complete coverage path planning	Cell decomposition	High computational cost, and high execution time (3.5 min) Efficient coverage - path planning, robust
3	The Path Planning of Mobile Robots Based on an Improved A* Algorithm, Lu Chang, Liang Shan,2019	To plan a safe and efficient path for a mobile robot	Improved A* (map compression, Path smoothing)	Optimal Consider robot size Large searching area Feasible and safer path Computational complexity
4	Faster RRT-based Nonholonomic Path Planning in 2D Building Environments Using Skeleton-constrained Path Biasing, Yiqun Dong ,Efe Camci,2017	Nonholonomic path planning in 2 d environment	RRT based method	Probabilistically complete algorithm, Better planning time, path length and Clearance

Lierature Review

SI No:	PAPER DETAILS	OBJECTIVES	METHODOLOGY	REMARKS
5	B. Matebese, D. Withey and M. K. Banda, "Path planning with the Leapfrog method in the presence of obstacles," 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO), Qingdao, 2016, pp. 613-618	To do motion planning with moving obstacle avoidance	Hamilton-Jacobi-Belllman (HJB) differential equations	Difficult for higher dimensions
6	Iyer, S. V., Time optimal trajectory generation for a differerential drive robot, State University of New York at Bu_alo, 2009	To minimize final time	Pontryagin's minimum principle	Suited for linear s/ms
7	A.Albert, L.Imsland,J. Haugen, Numerical Optimal Control Mixing Collocation with Single Shooting: A Case Study,IFAC-PapersOnLine, Volume 49, Issue 7, 2016,	To compare collocation and single shooting methods	Collocation, Single shooting mixed with collocation	Combined approach is faster than the pure collocation method
8	E.Drozdova, Siegbert Hopfgarten, Ev. Lazutkin, Pu Li, Autonomous Driving of a Mobile Robot Using a Combined Multiple-Shooting and Collocation Method, IFAC-PapersOnLine, Volume 49, Issue 15, 2016, Pages 193-198,	To implement a highly efficient numerical approach for online	Multiple-Shooting and Collocation Method	Efficient dynamic optimization better accuracy, less computational effort,

Literature Review

SI No:	PAPER DETAILS	OBJECTIVES	METHODOLOGY	REMARKS
9	F. Fahroo and M. I. Ross, "Direct trajectory optimization by a chebyshev pseudospectral method," Journal of Guidance, Control Dynamics 25(1), 160–166 (2002).	To solve a generic optimal control problem with state and control constraints	Chebyshev's pseudo spectral method	Simple and efficient, Rapid and accurate
10	L. R. Lewis, I. M. Ross and Qi Gong, "Pseudospectral motion planning techniques for autonomous obstacle avoidance," 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, 2007, pp. 5997-6002	To generate a minimum- time trajectory for an autonomous vehicle	Legendre Pseudo spectral method	Optimality is verified for simple and complex environments
11	Z. K. Su, H. L. Wang and P. Yao, "A hybrid backtracking search optimization algorithm for nonlinear optimal control problems with complex dynamic constraints," Neurocomputing. 186, 182–194 (2016).	To find a global optimum trajectory	Gauss pseudo spectral	Fast and accurate robust
12	Mao, R., Gao, H., & Guo, L. (2020). Optimal Motion Planning for Differential Drive Mobile Robots based on Multiple-Interval Chebyshev Pseudospectral Methods. Robotica, 1-20.	To solve the motion planning problem of nonholonomic mobile robots with kinematic and dynamic constraints	Chebyshev's pseudo spectral method,	Single interval is sufficient for good result, Multiple interval for more accurate clearance

Research Gap

- Premature convergence
- Sensitivity of initial guess

Objectives

- To design and implement an optimal trajectory planning algorithm for a differentially driven UGV in minimum time with minimal control effort
- To modify the path in presence of,
  1. Static obstacles
  2. Dynamic obstacles

System description
Vehicle Structure

- Two independent driving wheels on a common axis
- Two DC motors mounted on the wheels
- Change direction by varying relative angular speed of wheels

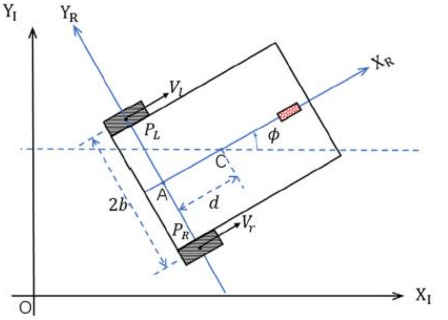


Figure 1: Structure of differential drive ground vehicle represented in Inertial frame

System description
Vehicle kinematics

$$\begin{aligned}
\dot{x} &= v\cos\phi - d\omega\sin\phi \\
\dot{y} &= v\sin\phi - d\omega\cos\phi \\
\dot{\phi} &= \omega \\
\dot{v} &= u_1 \\
\dot{\omega} &= u_2
\end{aligned} \tag{1}$$

Symbols	Description
C	The center of mass point
(x, y)	Coordinate of the vehicle centre of mass
ϕ	The orientation in the inertial frame
d	The distance between point C and A
v	linear velocity
ω	angular velocity

Problem Formulation
Optimal Trajectory Planning Problem

Determine the control inputs to minimize the objective function J

$$\text{Minimize } J = \mathcal{M}\left(\mathrm{X}\left(t_0\right), t_0, \mathrm{X}\left(t_f\right), t_f; \mathrm{P}\right) + \int_{t_0}^{t_f} \mathcal{L}(\mathrm{X}(t), U(t), t; \mathrm{P}) dt \tag{2}$$

Subjected to :

$$\text{Boundary conditions : } \mathrm{X}\left(t_0\right) = \mathrm{X}_0, \mathrm{X}\left(t_f\right) = \mathrm{X}_f \tag{3}$$

$$\text{Control constraints : } \mathrm{U}_{\mathrm{min}} \leq \mathrm{U} \leq \mathrm{U}_{\mathrm{max}} \tag{4}$$

$$\text{State constraints : } \mathrm{X}_{\mathrm{min}} \leq \mathrm{X} \leq \mathrm{X}_{\mathrm{max}} \tag{5}$$

Problem Formulation

Optimal Trajectory Planning Problem

$$\text{Dynamic constraints :} \quad \begin{pmatrix} \dot{x} = v\cos\phi - d\omega\sin\phi \\ \dot{y} = v\sin\phi - d\omega\cos\phi \\ \dot{\phi} = \omega \\ \dot{v} = u_1, \dot{\omega} = u_2 \end{pmatrix} \tag{6}$$

- **Final time minimization :**

$$\text{Minimize } J = \int_{t_0}^{t_f} 1dt = t_f - t_0 = t_f \tag{7}$$

- **Final time minimization with minimal control effort :**

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathrm{U} * \mathrm{U}^{\mathrm{T}})\mathrm{d}t \tag{8}$$

Problem Formulation

Obstacle Ridden Environment

- Obstacle is approximated as circle with radius r_{ob}
- UGV is also assumed as circle with radius r_m
- Maximum distance between centres of obstacle and vehicle during collision

$$d_{collision} = r_m + r_{ob} \tag{9}$$

- Distance between obstacle and vehicle

$$d_{sep} = \sqrt{(x_{ob} - x)^2 + (y_{ob} - y)^2} \tag{10}$$

- Inequality constraint for obstacle avoidance

$$d_{collision} - d_{sep} \leq 0 \tag{11}$$

Methodology
Pseudo Spectral Method

- This optimal control problem (OCP) is solved by using Pseudo spectral method
- Transforms dynamic equations into algebraic form
- Approximate the state using interpolating polynomials
- Optimal control problem to nonlinear programming problem
- Solve numerically using numerical programming solvers
- Four components : domain transformation, interpolation, differentiation, integration

Motivation

- Exponential Convergence
- Coarse grids are enough to give good accuracy
- Less sensitivity towards initial guess

Domain transformation

- Physical domain : $t \epsilon [t_0, t_f]$
- Computational domain: $\tau \epsilon [-1, 1]$

$$\tau = \frac{2}{t_f - t_0} t - \frac{t_f + t_0}{t_f - t_0} \tag{12}$$

Interpolation

- $N + 1$ nodes belongs to [-1,1]
- Interpolating basis polynomial $\varphi_k(\tau)$

$$P(\tau) = f(\tau) = \sum_{k=0}^{N} f(\tau_k) \varphi_k(\tau) \tag{13}$$

Differentiation

- Derivative of $f(\tau)$ can be approximated as

$$\dot{f}(\tau_k) \approx \dot{F}^N(\tau_k) = \sum_{i=0}^{N} D_{ki} f(\tau_i) \tag{14}$$

- D_{ki} : Differentiation matrix of order $(N+1)X(N+1)$

Numerical Quadrature

$$\int_{t_0}^{t_f} f(t)dt = \frac{tf - t_0}{2} \int_{-1}^{1} f(\tau)d\tau \tag{15}$$

$$\int_{-1}^{1} f(\tau) \approx \sum_{k=0}^{N} w_k f(\tau_k) \tag{16}$$

- w_k : predetermined quadrature weights

Descretization Matrices

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

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

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

h = time horizone

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

$$\text{Boundary}, \phi = \begin{bmatrix} \phi_1 \cdots & \phi_{n_\phi} \end{bmatrix}_{1\times n_\phi}$$

Defect Constraints

- For ensure the accuracy of approximation by differentiation matrix

$$\zeta = 0 = \mathrm{D}\Xi - \mathrm{F} \tag{17}$$

Objective Function

$$\sum_{k=0}^{N} w_k \mathcal{L}\left(t_k, \boldsymbol{\xi}\left(t_k\right), \mathrm{u}\left(t_k\right)\right) = \frac{h}{2}\sum_{k=0}^{N} w_k \mathcal{L}\left(\tau_k, \boldsymbol{\xi}\left(\tau_k\right), \mathrm{u}\left(\tau_k\right)\right) \tag{18}$$

$$\mathcal{M}\left(t_0, \boldsymbol{\xi}\left(t_0\right), t_f, \boldsymbol{\xi}\left(t_{N_t}\right)\right) = \mathcal{M}\left(t_0, \boldsymbol{\xi}(-1), t_f, \boldsymbol{\xi}(1)\right) \tag{19}$$

Methodology
Chebyshev PS Method

- Interpolating nodes are the roots of Chebyshev polynomial
- Chebyshev Gauss Lobatto (CGL) nodes

$$\tau_k = -\cos\left(\frac{\pi k}{N}\right) \in [-1, 1] \tag{20}$$

$$T_N = \cos\left(N \cos^{-1}(\tau)\right) \tag{21}$$

$$\begin{gathered}\phi_k(\tau) = \frac{(-1)^{k+1}}{N^2 a_k} \frac{\left(1-\tau^2\right)}{\tau - \tau_k} \dot{T_N}(\tau) \\ \text{where, } a_k = \begin{cases} 2 & k = \{0, \alpha\}\} \\ 1 & \text{otherwise} \end{cases}\end{gathered} \tag{22}$$

- **Motivation :** less optimization error (Lebesgue constant is lesser for CGL nodes) and n degree polynomial is sufficient to give n+1 nodes

- Differentiation with Chebyshev polynomial
- derivatives of $f^N(\tau)$ in terms of $f(\tau)$ at the CGL points τ_k

$$D_{ki} = \begin{cases} \frac{a_k}{a_i}\frac{(-1)^{k+i}}{(\tau_k-\tau_i)} & \text{if } k \neq i \\ -\frac{\tau_k}{2\left(1-\tau_k^2\right)} & \text{if } 1 \leq k = i \leq N-1 \\ \frac{2N^2+1}{6} & \text{if } k = i = 0 \\ -\frac{2N^2+1}{6} & \text{if } k = i = N \end{cases} \tag{23}$$

- Quadrature Weights:

$$\begin{gathered} w_k = \frac{c_k}{N}\left(1 - \sum_{j=1}^{\lfloor N/2 \rfloor} \frac{b_j}{4j^2-1}\cos(2j\tau_k)\right) \\ \text{where: } b_j = \begin{cases} 1 & \text{if } j = N/2 \\ 2 & \text{if } j < N/2 \end{cases}, \quad c_k = \begin{cases} 1 & \text{if } k = \{0, N_t\} \\ 2 & \text{otherwise} \end{cases} \end{gathered} \tag{24}$$

Concept of Nodes

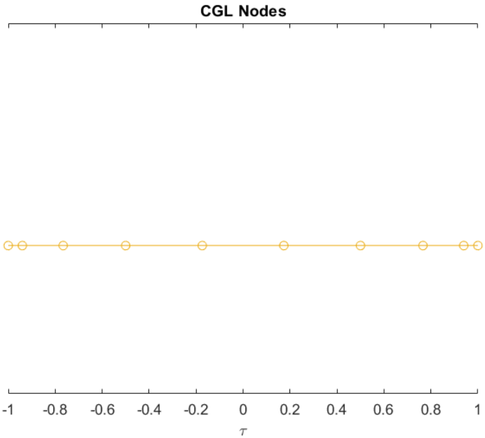


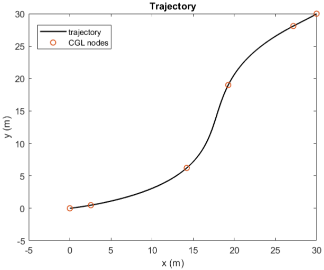
Figure 2: Nodes versus computational time domain

Simulation Parameters

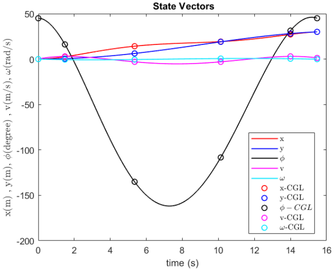
- Dimensions of the vehicle : $r_m = 0.42m$
- starting position : (0,0) and goal position : (30,30)
- Initial conditions : time = 0 s, velocity = 0 m/s
- Final conditions : Maximum final time = 100 s
- Lower bounds: $x = -50$, $y = -50$, $\phi = -\pi rad$, $v = -3m/s$, $\omega = -10rad$, $[u_1, u_2] = [-5, -5]$
- Upper bounds : $x = 50$, $y = 50$, $\phi = \pi rad$, $v = 3m/s$, $\omega = 10rad/s$, $[u_1, u_2] = [5, 5]$
- Initial guess : Linear initial trajectory with v=1 m/s
- FMINCON, Step tolerance = 100 eps,constraint tolerance = 10^{-6}

Simulation Results

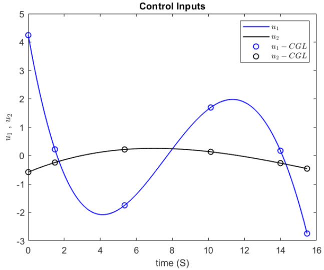
$\phi_0 = 45^0$, $\phi_f = 45^0$, Obstacle free environment



trajectory



state vectors

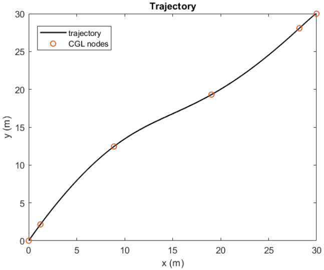


control inputs

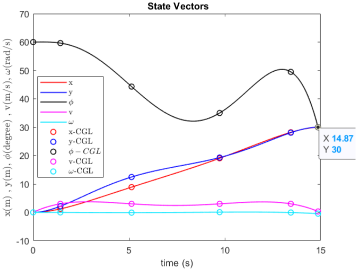
Figure 3: optimum solution obtained with 6 nodes

Simulation Results

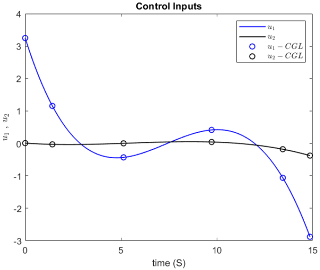
$\phi_0 = 60^0$, $\phi_f = 30^0$, **Obstacle free environment**



trajectory



state vectors

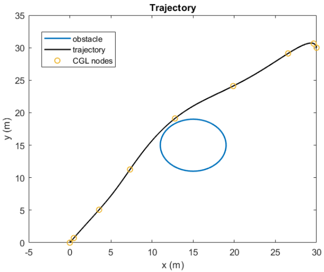


control inputs

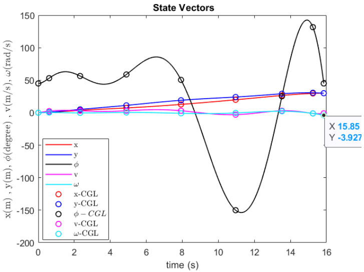
Figure 4: optimum solution obtained with 6 nodes

Simulation Results

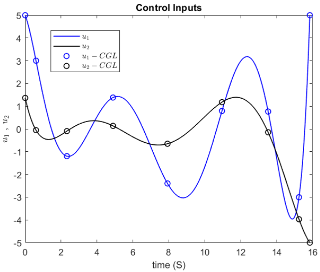
Large obstacle in the middle of the path, $\phi_0 = 45^0$, $\phi_f = 45^0$



trajectory



state vectors

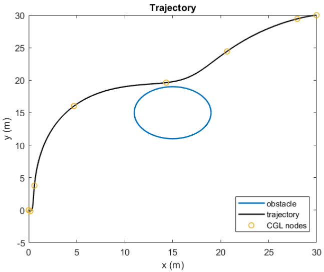


control inputs

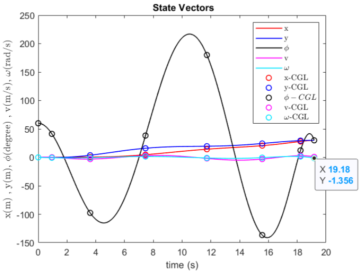
Figure 5: optimum solution obtained with 9 nodes

Simulation Results

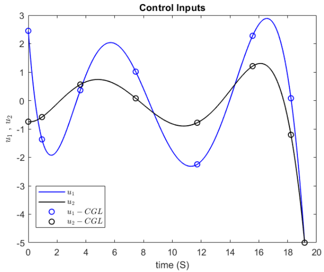
Large obstacle in the middle of the path, $\phi_0 = 60^0$, $\phi_f = 30^0$



trajectory



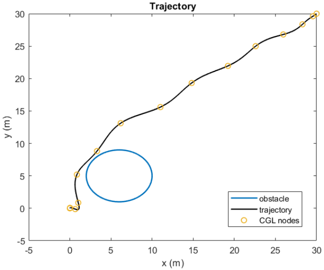
state vectors



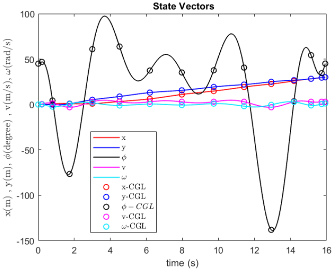
control inputs

Figure 6: optimum solution obtained with 9 nodes

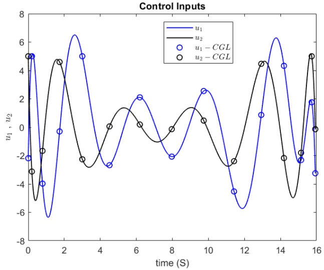
Large obstacle close to end points, $\phi_0 = 45^0$, $\phi_f = 45^0$



trajectory



state vectors

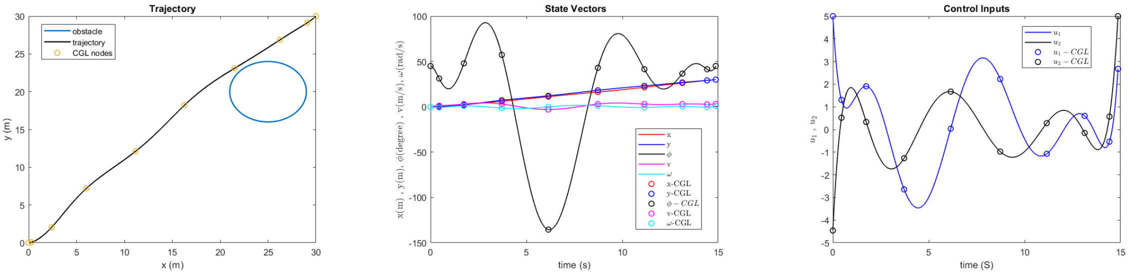


control inputs

Figure 7: optimum solution obtained with 14 nodes

Simulation Results

Large obstacle close to end points, $\phi_0 = 60^0$, $\phi_f = 30^0$

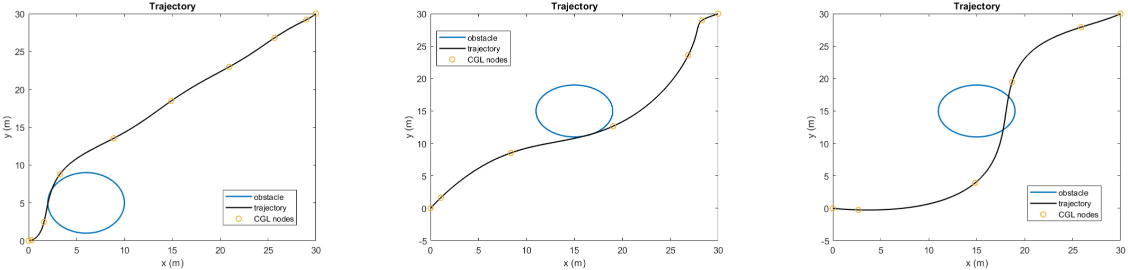


trajectory | state vectors | control inputs

Figure 8: optimum solution obtained with 13 nodes

Simulation Results for Obstacle ridden Space

Large obstacle - Failed cases

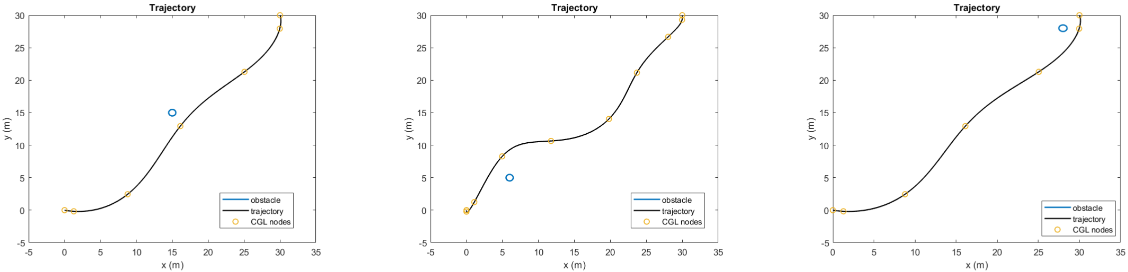


$\phi_0 = 45^0,\ \phi_f = 45^0$ $\qquad$ $\phi_0 = 60^0,\ \phi_f = 30^0$ $\qquad$ $\phi_0 = 60^0,\ \phi_f = 30^0$

Figure 9: Some failed trajectory generations

Small obstacle

$\phi_0 = 45^0$, $\phi_f = 45^0$

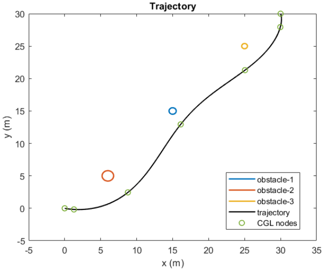


obstacle at midway | obstacle close to start | obstacle close to goal

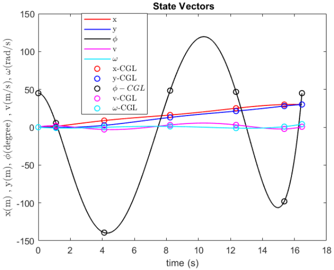
Figure 10: optimum trajectory for different scenarios

Simulation Results for Obstacle Ridden Environment

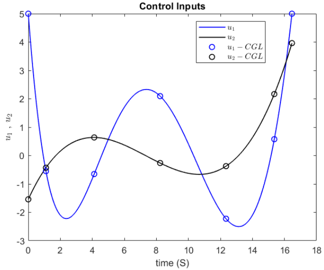
Multiple Obstacles



trajectory



state vectors

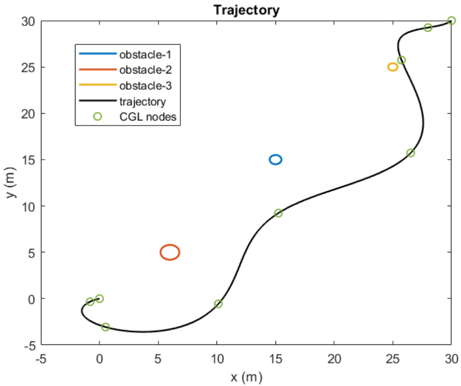


control inputs

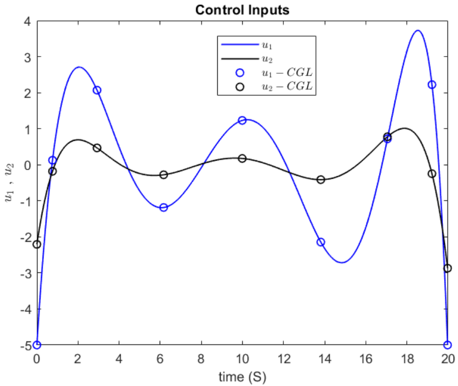
Figure 11: optimum solution obtained with 7 nodes

Simulation Results

Minimum final time with minimal control effort



trajectory



control inputs

Figure 12: optimum solution obtained with 9 nodes

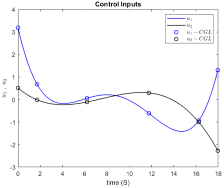
Sensitivity of Number of nodes

Computational cost

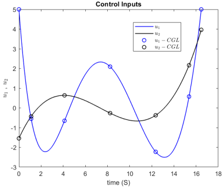
Table 1: Variation of computation time with respect to number of nodes

No. of nodes	6	7	8	9	10	12	25
Computation time	0.558836	1.004668	1.459270	1.300093	1.796466	1.757472	12.986705
Objective function	17.96401	16.46675	17.34132	15.21626	18.23093	17.26984	15.72468

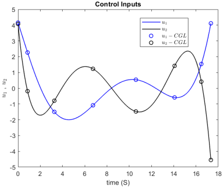
Sensitivity of Number of nodes
Control inputs



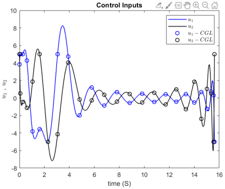
5 nodes



6 nodes



7 nodes



25 nodes

Figure 13: Control inputs for different number of nodes

Modified Formulation for the Optimal Control Problem

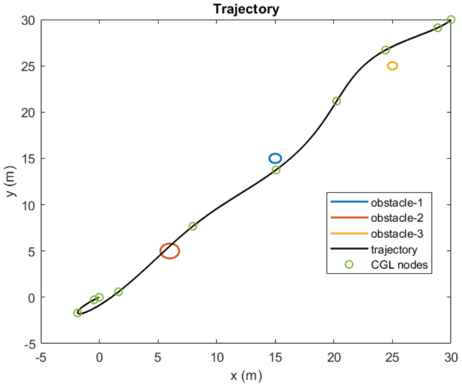
- To ensure maximum possible distance between obstacle and vehicle
- Inverse of distance between vehicle and obstacle is added to the performance index

$$P_i = \frac{1}{(d_i - r_{obi})} \tag{25}$$

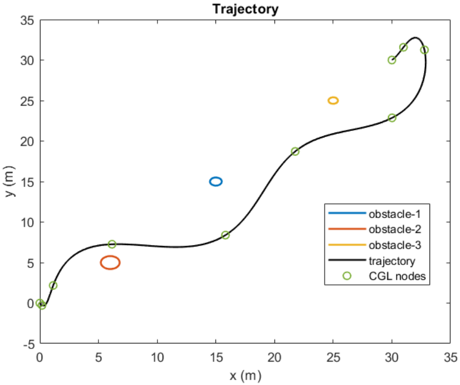
- d_i is the distance between vehicle and i^{th} obstacle
- r_{obi} is the sum of radii of vehicle and i^{th} obstacle

Modified Formulation for the Optimal Control

Simulation Results



failed trajectory before cost function updating

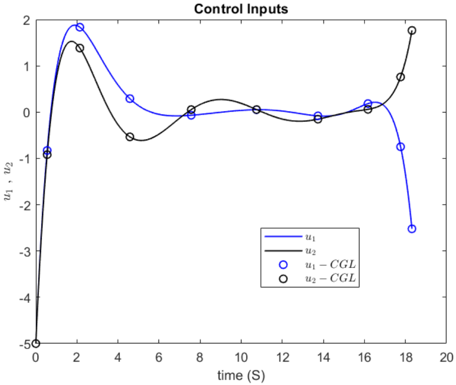


trajectory after cost function updated

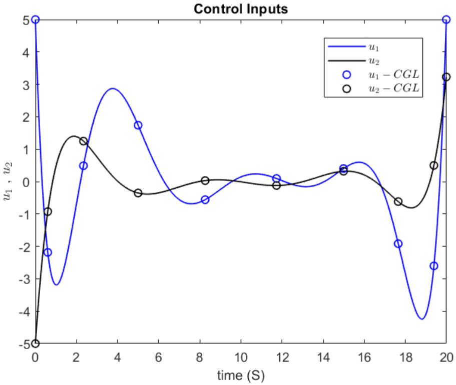
Figure 14: Change in trajectory with modified problem

Modified Formulation for the Optimal Control Problem

Simulation Results



control inputs before cost function updation



control inputs after cost function updated

Figure 15: Change in control inputs for modified problem

Proposed Optimal Trajectory Design
Problem formulation

- Control vectors at goal position are set to a specific value
- Determine the control inputs to,

$$\text{Minimize } J = \int_{t_0}^{t_f} 1 + (\mathrm{U} * \mathrm{U}^{\mathrm{T}}) + \sum_{i=1}^{N_{ob}} P_i \mathrm{d}t \tag{26}$$

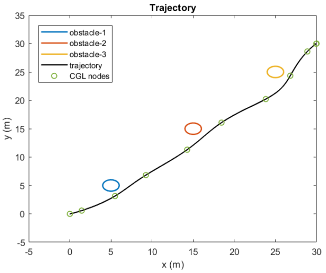
subjected to,

$$\text{Boundary conditions : } \mathrm{X}(t_0) = \mathrm{X}_0, \mathrm{X}(t_f) = \mathrm{X}_f, \mathrm{U}(t_f) = \mathrm{U}_f \tag{27}$$

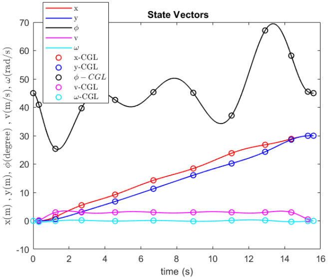
$$\text{Control constraints : } \mathrm{U}_{\min} \leq \mathrm{U} \leq \mathrm{U}_{\max} \tag{28}$$

$$\text{State constraints : } \mathrm{X}_{\min} \leq \mathrm{X} \leq \mathrm{X}_{\max} \tag{29}$$

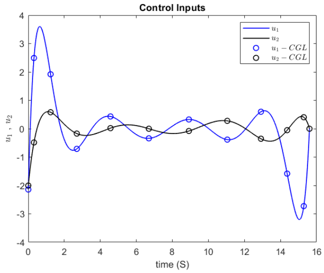
$$\text{Dynamic constraints : } \left(\begin{array}{l} \dot{x} = v \cos\phi - d\omega \sin\phi \\ \dot{y} = v \sin\phi - d\omega \cos\phi \\ \dot{\phi} = \omega \\ \dot{v} = u_1, \dot{\omega} = u_2 \end{array} \right) \tag{30}$$



trajectory



state vectors



control inputs

Figure 16: optimum solution obtained with 12 nodes

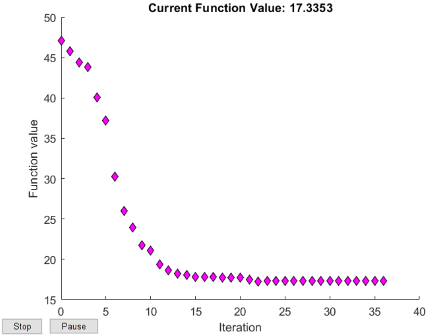


Figure 17: Convergence of function versus iteration

Dynamic Obstacle Environment

Proposed Problem Formulation

- Time dependent path constraint
- Obstacle with uniform velocity
- Distance between vehicle and obstacle

$$d_{sep}(t) = \sqrt{(x_{ob} - x)^2 + (y_{ob} - y)^2} \tag{31}$$

$$d_{collision} = r_m + r_{ob} \tag{32}$$

- Path constraint,

$$d_{collision} - d_{sep}(t) \leq 0 \tag{33}$$

- Obstacle position,

$$x_{ob}(t) = x_{ob}(0) + v_{ob}.t.cos(\alpha) \tag{34}$$

$$y_{ob}(t) = y_{ob}(0) + v_{ob}.t.sin(\alpha) \tag{35}$$

Dynamic Obstacle Environment

Simulation Parameters

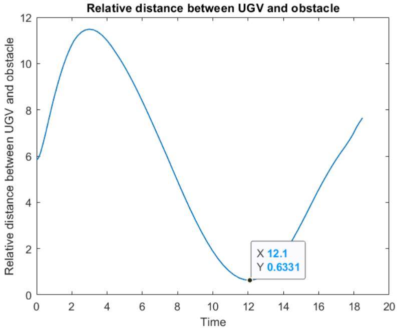
Case 1

- Vehicle : start (0,0) to goal (30,30) with $\phi_0 = \phi_f = \pi/4$
- obstacle : initial position (5,5), with velocity=1.5m/s, direction $\alpha = \pi/4$

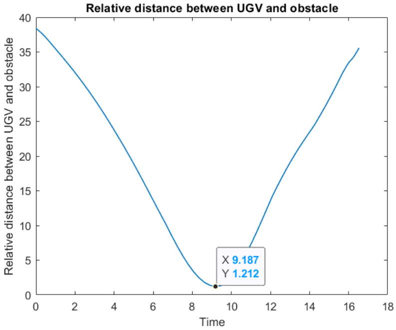
Case 2

- Vehicle : start (0,0) to goal (30,30) with $\phi_0 = \phi_f = \pi/2$
- obstacle : initial position (28,28), with velocity=2m/s, direction $\alpha = 5\pi/4$

Optimal Trajectory Planning with Minimum time



Case 1



Case 2

Figure 18: Relative distance between obstacle and vehicle versus time

- $v_o = 1.5m/s$, $\alpha = 5pi/4$, $x_{ob}, y_{ob} = (20, 20)$, $r_{ob} = 0.5m$,
- $\phi_0 = pi/4$ and $\phi_f = pi/4$

Optimal Trajectory Planning with Minimum Time and Minimal Control Effort

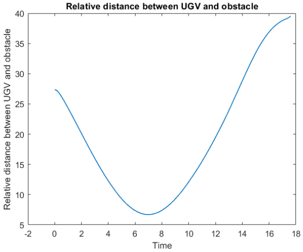


Figure 19: Relative distance between UGV and obstacle

Contributions of the Thesis

- Optimal trajectory planning problem is formulated to optimize final time, control effort and avoid static obstacles
- Optimal trajectory planning problem is formulated for dynamic obstacle avoidance
- Solution using Chebyshev pseudo spectral method

Conclusions

- Optimal path planning while moving to a target point is achieved in minimum time with minimal control effort
- According to the initial and final orientation of the vehicle, different optimal solutions can be formed
- Trajectory is modified for obstacle ridden environment
- Good optimal solutions are obtained for small obstacles
- Conventional or global PS method gives better solution than the multiple interval PS method
- Computational cost increases with number of nodes

Conclusions

- 10 fold improvement in computation time can be done by optimizing the code, using packages like SNOPT,GPOPS etc.
- Further improvement in computation speed can be done by eliminating windows and MATLAB overhead
- Control inputs become more oscillatory as number of nodes increases
- Optimal trajectory problem is modified by including additional term for obstacle avoidance in cost function
- Optimal trajectory with minimum time is designed for moving obstacle scenario

Future Scope

- Combining Chebyshev and Legendre methods may give better results.
  - Legendre basis polynomial give more stability than the Chebyshev basis polynomial.
  - CGL nodes have less optimization error.
- Considering vehicle dynamics

References

- Mao, R., Gao, H., Guo, L. (2020). "Optimal Motion Planning for Differential Drive Mobile Robots based on Multiple-Interval Chebyshev Pseudospectral Methods".*Robotica*, 1-20.

- Chen, Qingyang Sun, Zhenping Liu, Daxue Yuqiang, Fang Xiaohui, Li. (2012)." Local Path Planning for an Unmanned Ground Vehicle Based on SVM". *International Journal of Advanced Robotic Systems*.

- Mohamed, Amr Ren, Jing Sharaf, A. El-Gindy, Moustafa. (2018). "Optimal path planning for unmanned ground vehicles using potential field method and optimal control method". *International Journal of Vehicle Performance*. 4. 1-14.

- Betts, J. T., of numerical methods for trajectory optimization," *Journal of Guidance control and dynamics*, Vol. 21, No. 2, 1998, .

- Almayyahi, Auday, et al. "Motion control design for unmanned ground vehicle in dynamic environment using intelligent controller." *International Journal of Intelligent Computing and Cybernetics* ,2017.

- J. Canny, “A Voronoi Method for the Piano-Movers Problem,” *In: IEEE International Conference on Robotics Automation*,1985, pp. 530–535

- J. R. Shewchuk, “Delaunay refinement algorithms for triangular mesh generation,” Comp. Geom. 22(1–3), 21–74 ,2002.

- H. Zhang, W. Lin, and A. Chen, “Path Planning for the Mobile Robot: A Review,” Symmetry, vol. 10, no. 10, p. 450, Oct. 2018.

- H. Zhang, W. Lin, and A. Chen, “Path Planning for the Mobile Robot: A Review,” Symmetry, vol. 10, no. 10, p. 450, Oct. 2018.
- Atyabi, Adham Powers, David. (2013). Review of classical and heuristic-based navigation and path planning approaches. International Journal of Advancements in Computing Technology (IJACT). 5. 1-14.
- B.K. Patle, D.R.K. Parhi, A. Jagadeesh, Sunil Kumar Kashyap,”Matrix- Binary Codes based Genetic Algorithm for path planning of mobile robot”,Computers Electrical Engineering,Volume 67,2018,Pages 708- 728
- Z. K. Su, H. L. Wang and P. Yao, “A hybrid backtracking search optimization algorithm for nonlinear optimal control problems with complex dynamic constraints,” Neurocomputing. 186, 182–194 (2016).

- M. H. Korayem, M. Nazemizadeh and H. R. Nohooji, “Optimal point-to-point motion planning of nonholonomic mobile robots in the presence of multiple obstacles,” J. Braz. Soc. Mech. Sci. 36(1), 221–232 (2014).
- Bryson, A. E., Applied optimal control: optimization, estimation and control , CRC Press, 1975
- Iyer, S. V., Time optimal trajectory generation for a differential drive robot, State University of New York at Bualo, 2009
- A. Gasparetto, P. Boscariol, A. Lanzutti and R. Vidoni, “Path Planning and Trajectory Planning Algorithms: A General Overview,” Mech. Mach. Sci. 29, 3–27 (2015).

THANK YOU