

Control Barrier Function based Energy optimal Obstacles Avoidance for Point-to-Point Maneuvers

Youngjin Kim ^{*,1} Chaozhe R. He ^{*,2} Tarunraj Singh ^{*,3}

** Department of Mechanical Engineering & Aerospace Engineering,
University at Buffalo NY - 14260, USA*

Abstract: This paper focuses on developing a motion planning algorithm for static obstacle avoidance for a kinematic unicycle robot undergoing an energy-optimal point-to-point maneuver. The standard kinematic model is redefined in the geometric center space, motivated by the feedback linearization technique, resulting in a reduced order kinematic model. The proposed optimal motion planning approach is decomposed into two sequential stages: pre-planning and re-planning. In the pre-planning stage, an obstacle-free point-to-point optimal control problem is formulated and solved. Utilizing the solution from the optimal control problem, a perturbation controller is introduced which incorporates the nominal optimal control as a feedforward controller and a feedback tracking controller. In the second stage, the control barrier function method is employed to account for safety requirements, resulting in a minimum intervention control and solved in a point-wise optimization framework that accounts for the obstacles. The safety constraints are used as a quantitative metric to trigger trajectory re-planning, ultimately resulting in a nearly optimal control and trajectory.

Keywords: Motion/Trajectory Planning, Optimal Control, Nonlinear Control, Control Barrier Function, Wheeled Mobile Robot

1. INTRODUCTION

The three state kinematics model of mobile ground robotic systems has been widely studied. Although the model excludes the dynamics of the system, it has served as a benchmark model to realize solutions for autonomous ground vehicle applications (Gao et al. (2020); Werling and Groll (2009)). Due to the complexity of the real-world environment and computational limitations of onboard computers, developing efficient motion planning algorithms for such a system is a fundamental challenge in achieving autonomous navigation. The task of finding the optimal trajectories for the robots traversing from one point to another while avoiding obstacles is the main focus of this paper.

Collision-free path planning is one of the most critical problems in autonomous robotic applications. There is a plethora of literature, including pioneering works such as grid-based algorithms like Dijkstra (Dijkstra (1959)) and A* (Hart et al. (1968)), continuous path planning methods such as artificial potential field approach (Hwang et al. (1992)), and sample-based search algorithm (LaValle and Kuffner Jr (2001)). Nevertheless, the aforementioned classical path planning algorithms offer the benefit in achieving a collision-free path, however, they often disregard the dynamics and kinematics of the robot. Consequently, the

planned path may be infeasible to implement on real-world robotic systems.

The obstacle avoidance problem can be formulated as a pure state-constrained optimal control problem. The first principle-based approach to handling pure state constraints was introduced in (Bryson Jr et al. (1963)). Finding a collision-free path often requires solving a nonconvex optimization problem, where feasibility in obtaining the optimal path and control may not be guaranteed. To alleviate the feasibility issue, the minimum-penetration trajectory generation algorithm is introduced in (Zhang et al. (2020)), where penetration allowance is measured through local linearization. The Hamilton-Jacobi-Bellman principle is explored in (Sundar and Shiller (1997)) to find the shortest collision-free path to the goal. Since the objective of the problem is finding the shortest path, the authors take advantage of geometry as a tool to measure the distance between the current position and the goal to formulate the cost to the goal. However, in this article, our objective is to find a minimum energy collision-free path. Solving the resulting optimal control based on intuition may not be applicable, since following the shortest path does not necessarily result in energy optimality (Kim and Singh (2021)). Solving optimal control problems with pure state constraints, is in general a very challenging problem. It is even more difficult to solve than the problems with both control and state constraints. This is because of the nature of the pure state constraints, where the constraints do not explicitly depend on control inputs (Chachuat (2007)).

¹ Y. Kim: PhD Student, E-mail: ykim35@buffalo.edu.

² C. R. He: Assistant Professor, E-mail: chaozheh@buffalo.edu.

³ T. Singh: Professor, Corresponding Author. E-mail: tsingh@buffalo.edu.

Given a model of the robot, obstacle avoidance problems can also be reformulated by enforcing safety requirements, and by monitoring the control actions for the safety requirement and only intervening when such requirements are compromised. Safety requirements can be specified as *set forward invariance constraints*, or by requiring the states of a system to remain within a prescribed set (Konda et al. (2020)). The notion of *barrier function* was proposed as a tool for checking the invariance of a set given a model of the system dynamics (Prajna and Jadbabaie (2004)). This notion has been recently adapted to the context of control synthesis, yielding control barrier functions (CBFs). CBFs have been demonstrated as a powerful tool for constructively synthesizing controllers that achieve set invariance and thus provide safety assurance (Ames et al. (2017, 2019)). Control synthesis with CBFs derives conditions directly from safety requirements and uses convex optimization to produce safety-filters that minimally modify a purposely-designed controller to ensure safety. It has been successfully implemented on real-world control systems, including mobile robot (Xu et al. (2017)), legged robots (Grandia et al. (2021); Csomay-Shanklin et al. (2021)), autonomous aerial vehicles (Molnar et al. (2021)), and connected automated vehicles (Alan et al. (2023)).

Traditionally, CBFs are used as safety filters to ensure the safety of the system by intervening in the nominal controls. The major contribution of the work is that we extend its utility as a quantitative metric to determine the activation of the re-planning strategy to address safety concerns and performance requirements. The kinematics of the unicycle model are considered in this work to illustrate this fundamental contribution. This model has been recognized as a nonholonomic system, which restricts instantaneous lateral motion. To alleviate the constraints in the control design process, the second contribution of this work, inspired by (De Luca et al. (2002)), is applying the feedback linearization technique to transform the model into a simple 2-dimensional single integrator. The reduced order model permits intuitive construction of the safety constraints from the geometry of obstacles. Furthermore, without modifying the standard CBFs architecture, the constraints can be directly applied to obtain safety function.

The remainder of the paper is organized as follows. We first describe the kinematics model of the robot considered in this work in Section 2. We then delve into the main contribution where motion planning problem formulation and detailed description of the proposed dual-stage optimal trajectory tracking control algorithm for collision-free navigation is introduced in Section 3. Subsequently, Section 4 showcases the numerical simulation results. Lastly, the concluding remarks and plans future work are presented in Section 5.

2. KINEMATIC OF UNICYCLE ROBOT

In this work, we examine the kinematics of a unicycle robot as a governing model to demonstrate the proposed motion planning algorithm. The model is expressed as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \quad (1)$$

where V and ω are the linear and angular velocities respectively and state x , y , and θ represent the position and orientation of the robot, measured from the midpoint \mathcal{P} between the two wheels.

Assuming that the geometric center of the robot is offset by a distance L (where $|L| > 0$), the original states can be rewritten as:

$$\begin{bmatrix} x_c \\ y_c \\ \theta_c \end{bmatrix} = \begin{bmatrix} x + L \cos(\theta) \\ y + L \sin(\theta) \\ \theta \end{bmatrix} \quad (2)$$

and the kinematics of the model for the virtual actuation point \mathcal{C} can be derived as:

$$\underbrace{\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix}}_{\dot{\mathbf{x}}_c} = \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}_{\mathbf{U}} \quad (3)$$

where $\mathbf{U} = [u_1, u_2]$ are the virtual controls at point \mathcal{C} . It has been acknowledged that the standard kinematic model (1) has a major restriction in which the system is inherently nonholonomic, consequently, instantaneous lateral motion cannot be achieved. However, by applying a simple transformation, the proposed geometric center model is not restricted by the kinematic constraints.

Lastly, the virtual controls map to real control V and ω by:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -L \sin(\theta) \\ \sin(\theta) & L \cos(\theta) \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix}. \quad (4)$$

Since L is strictly greater than zero, Equation (4) allows the virtual controls \mathbf{U} to be mapped back to the actual controls V and ω without encountering singularity issues. Figure 1 illustrates the difference between the standard kinematic model (around \mathcal{P}) and the geometric center model (around \mathcal{C}). Since Equation (1) corresponds to a

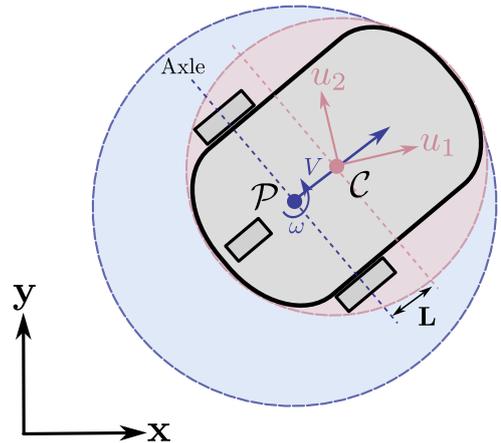


Fig. 1. Schematic of the kinematics of unicycle robot

point mass model, the obstacle avoidance problem requires inflating the obstacle bounds with the radius of a circle centered at a point whose coordinates form the reference for the trajectory generation problem. Clearly the point \mathcal{P} results in a large enclosing circle (Blue) relative to the circle enclosing the point \mathcal{C} , providing another motivation for the selection of the point \mathcal{C} to serve as the reference center.

3. REFERENCE TRAJECTORY TRACKING CONTROL WITH OBSTACLE AVOIDANCE

3.1 Pre-Planning Stage: Optimal Control Problem (OCP)

Minimizing overall energy consumption while navigating complex environments can be easily motivated by the need to prolong service time which significantly influences the capabilities and performance of the robotic system. Here, the integral of the total kinetic energy of the system is considered as a metric to measure the energy consumption of the vehicle:

$$\mathcal{J} = \int_0^{t_f} \alpha \frac{\mathcal{M}V^2}{2} + \beta \frac{\mathcal{I}\omega^2}{2} dt \quad (5)$$

where \mathcal{M} and \mathcal{I} represent mass and inertia of the system, respectively. And α and β are the weighting parameter that can be selected by user to adjust contribution. Lastly, t_f is a terminal time that must be specified to avoid degenerate solutions. For the demonstration purposes, we selected $\alpha = \frac{1}{\mathcal{M}}$ and $\beta = \frac{1}{\mathcal{I}}$, resulting in:

$$\mathcal{J} = \int_0^{t_f} \frac{V^2 + \omega^2}{2} dt \quad (6)$$

To obtain the optimal controls for a point-to-point transition, the initial and terminal states are prescribed as:

$$[x, y, \theta]^T(t=0) = [0, 0, 0]^T \quad (7)$$

$$[x, y, \theta]^T(t=t_f) = [x_f, y_f, \theta_f]^T. \quad (8)$$

With the given objective, the optimal control problem can be formulated as:

$$\min_{V, \omega} \mathcal{J} = \int_0^{t_f} \frac{V^2 + \omega^2}{2} dt \quad (9)$$

Subject to :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \quad (10)$$

$$[x, y, \theta]^T(t=0) = [0, 0, 0]^T \quad (11)$$

$$[x, y, \theta]^T(t=t_f) = [x_f, y_f, \theta_f]^T. \quad (12)$$

The proposed problem can be easily solved using the standard numerical shooting method and also in closed form as derived in (Kim and Singh (2021)). The resulting solution will serve as a reference trajectory for the states and the controls.

3.2 Re-Planning Stage: Control Barrier Function (CBF)

Using the point-to-point optimal trajectories, the controls include a state feedback perturbation control in conjunction with the nominal control as a feedforward component:

$$u_1 = -K_1(x_c - x_c^d) + \dot{x}_c^d \quad (13)$$

$$u_2 = -K_2(y_c - y_c^d) + \dot{y}_c^d \quad (14)$$

where the superscript $(.)^d$ refers to desired trajectories that can be obtained from the solution of the optimal control problem. Note that the reference trajectories and controls can be converted one-to-one using Equation (2) and (4). K_1 and K_2 denote the gains of the feedback control. Considering the geometric center model (3), the error dynamics of the system can be written as:

$$\dot{e}_x + K_1 e_x = 0 \quad (15)$$

$$\dot{e}_y + K_2 e_y = 0 \quad (16)$$

where: $e_x = x_c - x_c^d$ and $e_y = y_c - y_c^d$

which is stable as long as $K_1, K_2 > 0$. This implies asymptotic convergence of the closed-loop system is guaranteed.

The reference trajectories are generated assuming the robot is navigating in an obstacle-free environment. However, to account for the obstacle and ensure safe maneuvering, the following safety requirements need to be fulfilled:

$$(x_c - O_x)^2 + (y_c - O_y)^2 - r^2 \geq 0 \quad (17)$$

where O_x and O_y denote the location of the obstacle in Cartesian coordinates. Here we model the obstacle as a circle with radius r , and r is inflated to provide safety margins that account for the dimension of the robot as the kinematic model assumes the robot as a point mass.

The kinematic model (3) can be rewritten in control affine form as:

$$\underbrace{\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix}}_{\dot{\mathbf{X}}_c} = \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{f(\mathbf{X}_c)} + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{g(\mathbf{X}_c)} \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}_{\mathbf{U}}. \quad (18)$$

Define the control barrier function candidate:

$$h(\mathbf{X}_c) = (x_c - O_x)^2 + (y_c - O_y)^2 - r^2 \quad (19)$$

which is a continuously differentiable function $\mathbb{R}^2 \rightarrow \mathbb{R}$. The safe set $\Omega \subset \mathbb{R}^2$ defined as the 0-superlevel set of h yielding:

$$\Omega = \{\mathbf{X}_c \in \mathbb{R}^2 : h(\mathbf{X}_c) \geq 0\} \quad (20)$$

$$\partial\Omega = \{\mathbf{X}_c \in \mathbb{R}^2 : h(\mathbf{X}_c) = 0\} \quad (21)$$

$$\text{Int}(\Omega) = \{\mathbf{X}_c \in \mathbb{R}^2 : h(\mathbf{X}_c) > 0\} \quad (22)$$

where $\partial\Omega$ and $\text{Int}(\Omega)$ are the *boundary* and *interior*, respectively, of the set Ω . To fulfill safety requirement (17) is equivalent to guarantee the set Ω is forward invariant.

Following (Alan et al. (2023)), the condition of h being a valid control barrier function is given by:

$$\sup_{\mathbf{u} \in \mathbb{R}^2} L_f h(\mathbf{X}_c) + L_g h(\mathbf{X}_c) \mathbf{U} + \alpha(h(\mathbf{X}_c)) > 0 \quad (23)$$

or equivalently:

$$L_g h(\mathbf{X}_c) = \mathbf{0} \longrightarrow L_f h(\mathbf{X}_c) + \alpha(h(\mathbf{X}_c)) > 0 \quad (24)$$

where $\alpha(\cdot)$ denotes a *extended class* \mathcal{K}_∞^e which is continuous strictly increasing function defined on $(-\infty, \infty)$ with $\alpha(0) = 0$, $\lim_{r \rightarrow \infty} \alpha(r) = \infty$ and $\lim_{r \rightarrow -\infty} \alpha(r) = -\infty$. In this work, we pick a specific *class* \mathcal{K}_∞^e function in the form of:

$$\alpha(h(\mathbf{X}_c)) = \gamma h(\mathbf{X}_c) \quad (25)$$

where $\gamma > 0$.

Considering the affine system model (18), the Lie derivative of $h(\mathbf{X}_c)$ along f and g is given by:

$$L_f h(\mathbf{X}_c) = \frac{\partial h}{\partial \mathbf{X}_c} f(\mathbf{X}_c) = 0 \quad (26)$$

$$\begin{aligned} L_g h(\mathbf{X}_c) &= \frac{\partial h}{\partial \mathbf{X}_c} g(\mathbf{X}_c) \\ &= [2(x_c - O_x), 2(y_c - O_y)] \neq \mathbf{0}, \quad \forall \mathbf{X}_c \in \Omega. \end{aligned} \quad (27)$$

Note that $\mathbf{X}_c = [O_x, O_y] \notin \Omega$. This implies that the proposed candidate barrier function is a valid control barrier function.

One may consider Equation (19) as a candidate barrier function for the original kinematics model (1). With the standard kinematic model, the Lie derivative of the candidate barrier function along g is given by:

$$L_g h(\mathbf{X}) = [2(x - O_x) \cos(\theta) + 2(y - O_y) \sin(\theta), 0] \quad (29)$$

when $\theta = \arctan\left(-\frac{(x-O_x)}{(y-O_y)}\right)$, yields:

$$L_g h(\mathbf{X}) = 0 \longrightarrow L_f h(\mathbf{X}) + \alpha(h(\mathbf{X})) \geq 0 \not\approx 0 \quad (30)$$

This illustrates the fact that h is not a valid control barrier function for the original kinematics model.

The goal of maintaining the performance of the nominal controller $\mathbf{U}_n = [u_1, u_2]$ while ensuring the safety of the closed-loop system motivates an optimization-based safety-critical controller \mathbf{U}_{QP} defined as:

$$\mathbf{U}_{\text{QP}} = \min_{\mathbf{U}=(u_1, u_2)} \frac{1}{2} \|\mathbf{U} - \mathbf{U}_n\|^2 \quad (31)$$

Subject to :

$$L_f h(\mathbf{X}_c) + L_g h(\mathbf{X}_c)\mathbf{U} + \gamma h(\mathbf{X}_c) \geq 0 \quad (32)$$

This formulation, denoted as CBF-QP, can be synthesized through a point-wise optimization scheme. The feasibility of controls is guaranteed as control is assumed unbound in this study.

When the robot is distant from the obstacle, the given QP solution coincides with the nominal controls \mathbf{U}_n , while at every instant when safety conditions are violated, the QP solution results in the minimal intervention that regulates the nominal controls to navigate safely around the obstacles. However, since the nominal controls are defined by the pre-planned trajectory where the obstacle is not considered, it can require a large demand for controls to track back to the original reference trajectories after the robot avoids the obstacle. To ensure energy-efficient maneuvering while guaranteeing safe navigation, we propose to re-plan for a new trajectory whenever:

$$L_f h(\mathbf{X}_c) + L_g h(\mathbf{X}_c)\mathbf{U} + \gamma h(\mathbf{X}_c) \leq \epsilon \quad (33)$$

where ϵ is a small positive number. Hence, if the safety condition is less than or equal to the threshold ϵ , the optimal trajectories are updated by solving the following OCP:

$$\min_{V, \omega} \mathcal{J} = \int_{t_r}^{t_f} \frac{V^2 + \omega^2}{2} dt \quad (34)$$

Subject to :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \quad (35)$$

$$[x, y, \theta]^T(t = t_r) = [x(t_r), y(t_r), \theta(t_r)]^T \quad (36)$$

$$[x, y, \theta]^T(t = t_f) = [x_f, y_f, \theta_f]^T \quad (37)$$

where t_r is time when Equation (33) is triggered, and $[x(t_r), y(t_r), \theta(t_r)]$ represents the states at that time instant. Note that it is possible to impose the threshold bound on $h(\mathbf{X}_c)$, but this could lead to overly conservative solutions. The complete motion planning strategy is summarized in Algorithm (1).

Note that re-planning occurs immediately after the intervention takes place. Whenever $\mathbf{U}_{\text{QP}} = \mathbf{U}_n$, which implies that the nominal control guarantees closed-loop safety, then re-planning is not necessary.

Algorithm 1 OCP-CBF Replanning Algorithm

Initialize: t_f, x_f, y_f , and θ_f

Pre-Planning: $[\mathbf{X}_c^d, \mathbf{U}^d]$ = Solve OCP (9)-(12)

while $t \neq t_f$ **do**

Minimum Intervention Control:

\mathcal{U} = Solve CBF-QP (31)-(32)

if $L_f h(\mathbf{X}_c) + L_g h(\mathbf{X}_c)\mathcal{U} + \alpha(h(\mathbf{X}_c)) \leq \epsilon$ **then**

Re-Planning: $[\tilde{\mathbf{X}}_c^d, \tilde{\mathbf{U}}^d]$ = Solve OCP (34)-(37)

Update: $[\mathbf{X}_c^d, \mathbf{U}^d] = [\tilde{\mathbf{X}}_c^d, \tilde{\mathbf{U}}^d]$

else

Continue

end if

end while

4. RESULTS AND DISCUSSION

We demonstrate that the robot can track the reference trajectories while avoiding the obstacle by leveraging the control barrier function and utilizing safety conditions as quantitative guidance to re-plan the optimal trajectories. This section presents the main result of the proposed idea.

To rapidly generate energy optimal trajectories, an algorithmic differentiation-based software CasADi by Andersson et al. (2019) is used, and the CBF-QP problem is solved using MATLAB quadprog. The simulation results are obtained using the following initial parameters:

$$[t_f, L, K_1, K_2, \gamma, \epsilon] = [20, 0.05, 10, 10, 1, 10^{-5}] \quad (38)$$

$$[x, y, \theta](t = t_f) = [1, 1, 0] \quad (39)$$

$$[O_x, O_y, r] = [0.6, 0.4, 0.2] \quad (40)$$

The control gains are carefully selected such that both methods ensure reaching the desired terminal position within the specified maneuver time. The corresponding simulation results are shown in Figure 2.

In the figure, the pre-defined paths are shown in green, which provides the initial guidance to maintain energy optimal maneuver. The path and controls resulting from scenarios without and with re-planning strategy are represented in blue and purple, respectively. The energy costs for the two scenarios are $\mathcal{J}_{w/o} = 0.799$ and $\mathcal{J}_{re} = 0.279$, confirming that the proposed re-planning strategy significantly reduces energy consumption. In other words, converging back to the reference path may not be the best near-optimal solution.

To benchmark the proposed design, the exact energy-optimal obstacle avoidance control problem is formulated:

$$\min_{V, \omega} \mathcal{J} = \int_0^{t_f} \frac{V^2 + \omega^2}{2} dt$$

Subject to :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix}$$

$$(x - O_x)^2 + (y - O_y)^2 - r^2 \geq 0$$

$$[x, y, \theta]^T(t = 0) = [0, 0, 0]^T$$

$$[x, y, \theta]^T(t = t_f) = [x_f, y_f, \theta_f]^T.$$

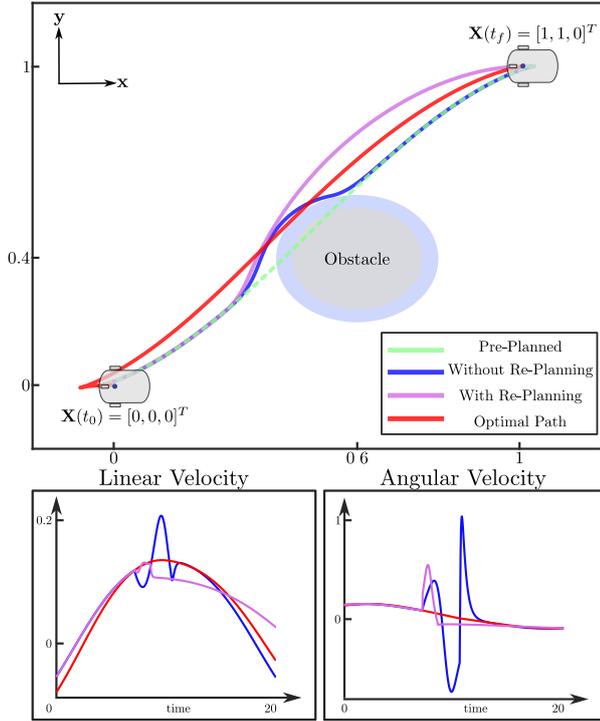


Fig. 2. Robot trajectories and controls: comparison between the re-planning method and without re-planning

The optimal solutions are shown via the red line (Fig. 2), and the corresponding optimal cost is $\mathcal{J}^* = 0.182$.

To investigate the impact of γ on energy consumption, a parametric study is conducted by increasing γ from 0.1 to 5, which is illustrated in Figure 3. The results confirm that leveraging the re-planning mechanism results in a better sub-optimal solution regardless of the γ value.

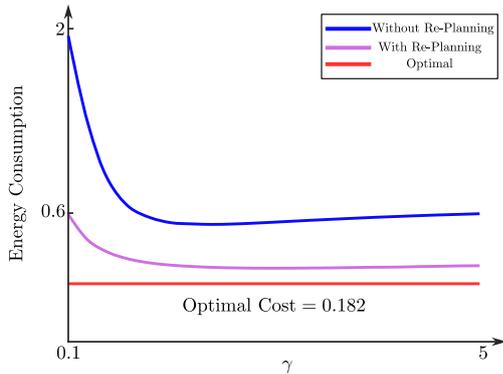


Fig. 3. Energy consumption with respect to γ

The major cost reduction is contributed by re-planning can be examined by evaluating the safety condition over time, where the results are shown in Figure 4.

In the given scenario, we assumed that the obstacle is farther away from the initial states, allowing the minimum intervention controls to resemble the baseline controls, resulting in the safety condition based on pre-planned (green), without re-planning (blue), and with re-planning (purple) to coincide. Since the pre-planned path is generated without consideration of the obstacle, the QP so-

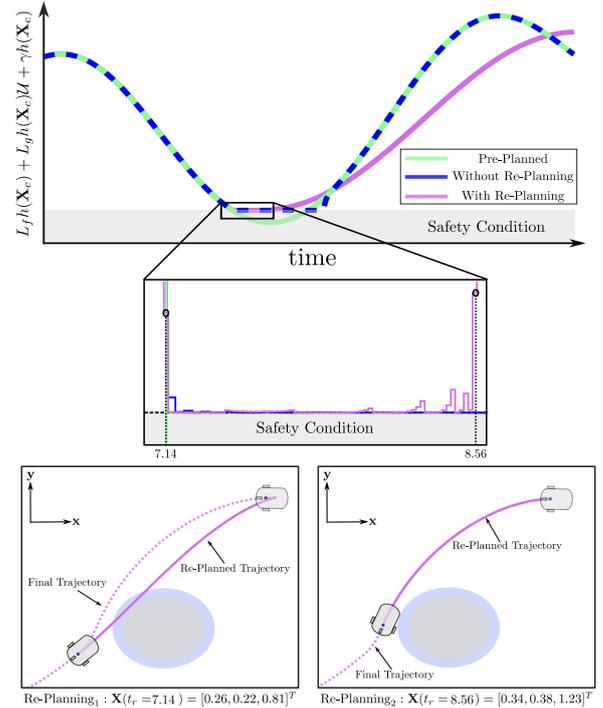


Fig. 4. Evolution of safety condition over time and re-planning results

lutions continuously regulate the baseline controls until the safety condition is satisfied. However, the proposed re-planning method provides better reference trajectories for the next step adapted from the previous states. If the OCP finds a energy optimal path that is collision-free, the minimum intervention controls revert to the baseline controls. Figure 4 also includes the first ($t_r = 7.14$) and last ($t_r = 8.56$) time when re-planning occurs, confirming that when $t_r = 8.56$ the OCP provides a collision-free path to goal, and revealing that the final trajectories (dashed-line) coincided with the final re-planned path.

The proposed algorithm can be extended to complex maneuver scenarios where multiple obstacles exist in the environment. The configuration of the obstacles is shown in Table 1, and corresponding simulation results are shown in Figure 5.

Table 1. Obstacle Configuration

	O_x	O_y	r
Obstacle 1	0.5	0.2	0.2
Obstacle 2	1.4	1.1	0.3
Obstacle 3	1.8	1.5	0.1
Obstacle 4	2.4	1.4	0.4

The re-planning method yields an energy cost of $\mathcal{J}_{re} = 0.94$, while without re-planning, the cost increases to $\mathcal{J}_{w/o} = 6.32$. Compared to the optimal cost of $\mathcal{J}^* = 0.52$, the proposed re-planning method shows substantial improvement in terms of energy savings.

5. CONCLUSION

This article introduces an algorithmic approach to solve state-constrained optimal control problems utilizing the

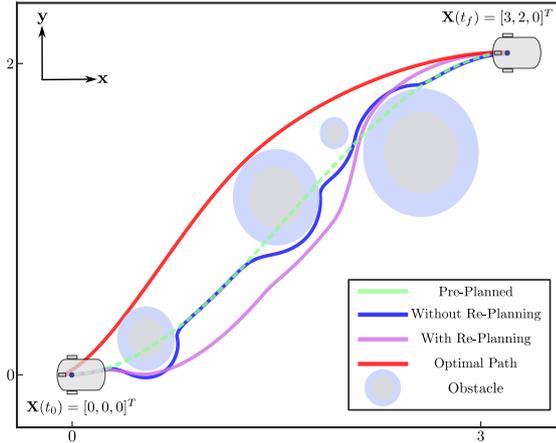


Fig. 5. Robot trajectory with multiple obstacles

control barrier function and optimal control. The control barrier function is a passive approach that handles state constraints by regulating the baseline control in a minimum intervention scheme but can compromise the overall performance. In contrast, solving the obstacle constrained optimal control ensures optimality, but solving state-constrained optimal control problems are generally computationally expensive, and depending on the complexity of the problem, feasibility of obtaining real-time optimal solutions is uncertain. To address the limitations of both methods, we introduced a re-planning mechanism based on control barrier functions that ensures safe maneuvering while accomplishing nearly optimal performance. The energy-optimal obstacle avoidance problem is formulated as a benchmark scenario to demonstrate the proposed algorithm, confirming that the proposed re-planning strategy maintains near-energy optimality while satisfying the safety requirement. Future work will use the proposed algorithm to consider both state and input constraints, and validate in experimental testing.

REFERENCES

- Alan, A., Taylor, A.J., He, C.R., Ames, A.D., and Orosz, G. (2023). Control barrier functions and input-to-state safety with application to automated vehicles. *IEEE Transactions on Control Systems Technology*.
- Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). Control barrier functions: Theory and applications. In *European Control Conference (ECC)*, 3420–3431. IEEE.
- Ames, A.D., Xu, X., Grizzle, J.W., and Tabuada, P. (2017). Control barrier function based quadratic programs for safety critical systems. *Transactions on Automatic Control*, 62(8), 3861–3876.
- Andersson, J.A., Gillis, J., Horn, G., Rawlings, J.B., and Diehl, M. (2019). Casadi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11, 1–36.
- Bryson Jr, A.E., Denham, W.F., and Dreyfus, S.E. (1963). Optimal programming problems with inequality constraints. *AIAA journal*, 1(11), 2544–2550.
- Chachuat, B. (2007). Nonlinear and dynamic optimization: From theory to practice.
- Csomay-Shanklin, N., Cosner, R.K., Dai, M., Taylor, A.J., and Ames, A.D. (2021). Episodic learning for safe bipedal locomotion with control barrier functions and projection-to-state safety. *Proceedings of Machine Learning Research (PMLR)*, 144, 1041–1053.
- De Luca, A., Oriolo, G., and Vendittelli, M. (2002). Control of wheeled mobile robots: An experimental overview. *RAMSETE: articulated and mobile robotics for services and technologies*, 181–226.
- Dijkstra, E.W. (1959). A note on two problems in connexion with graphs. In *Edsger Wybe Dijkstra: His Life, Work, and Legacy*, 287–290.
- Gao, H., Zhu, J., Li, X., Kang, Y., Li, J., and Su, H. (2020). Automatic parking control of unmanned vehicle based on switching control algorithm and backstepping. *IEEE/ASME Transactions on Mechatronics*, 27(3), 1233–1243.
- Grandia, R., Taylor, A.J., Ames, A.D., and Hutter, M. (2021). Multi-layered safety for legged robots via control barrier functions and model predictive control. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 8352–8358. Xi’an, China.
- Hart, P.E., Nilsson, N.J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107. doi:10.1109/TSSC.1968.300136.
- Hwang, Y.K., Ahuja, N., et al. (1992). A potential field approach to path planning. *IEEE transactions on robotics and automation*, 8(1), 23–32.
- Kim, Y. and Singh, T. (2021). Minimum energy control of a unicycle model robot. *Journal of Dynamic Systems, Measurement, and Control*, 143(10), 101003.
- Konda, R., Ames, A.D., and Coogan, S. (2020). Characterizing safety: Minimal control barrier functions from scalar comparison systems. *Control Systems Letters*, 5(2), 523–528.
- LaValle, S.M. and Kuffner Jr, J.J. (2001). Randomized kinodynamic planning. *The international journal of robotics research*, 20(5), 378–400.
- Molnar, T.G., Cosner, R.K., Singletary, A.W., Ubellacker, W., and Ames, A.D. (2021). Model-free safety-critical control for robotic systems. *IEEE robotics and automation letters*, 7(2), 944–951.
- Prajna, S. and Jadbabaie, A. (2004). Safety verification of hybrid systems using barrier certificates. In *International Workshop on Hybrid Systems: Computation and Control (HSCC)*, 477–492. Springer.
- Sundar, S. and Shiller, Z. (1997). Optimal obstacle avoidance based on the hamilton-jacobi-bellman equation. *IEEE Transactions on Robotics and Automation*, 13(2), 305–310. doi:10.1109/70.563653.
- Werling, M. and Groll, L. (2009). From flatness-based trajectory tracking to path following. In *2009 IEEE Intelligent Vehicles Symposium*, 1271–1275. IEEE.
- Xu, X., Waters, T., Pickem, D., Glotfelter, P., Egerstedt, M., Tabuada, P., Grizzle, J.W., and Ames, A.D. (2017). Realizing simultaneous lane keeping and adaptive speed regulation on accessible mobile robot testbeds. In *Conference on Control Technology and Applications (CCTA)*, 1769–1775. IEEE.
- Zhang, X., Liniger, A., and Borrelli, F. (2020). Optimization-based collision avoidance. *IEEE Transactions on Control Systems Technology*, 29(3), 972–983.