MICHIGAN ENGINEERING
UNIVERSITY OF MICHIGAN

MECHENG 599 Final Project Presentation

Robot Motion Planning Using Model Predictive Control and Control Barrier Function

Lihan Lian

Methodology (CBF)

- Control Barrier Function (CBF)
  - Define safe set C using the control barrier function h(x).
  - Link control input to safety constraints by using the Lie derivatives.
  - Works for control affine systems.

$$\dot{x} = f(x) + g(x)u$$

where α is an extended class K infinity function.

$$\sup_{u \in U} \left[L_f h(x) + L_g h(x) u \right] \geq -\alpha(h(x)).$$

For safety-critical control, we consider a set $\mathcal{C}$ defined as the superlevel set of a continuously differentiable function $h : \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}$:

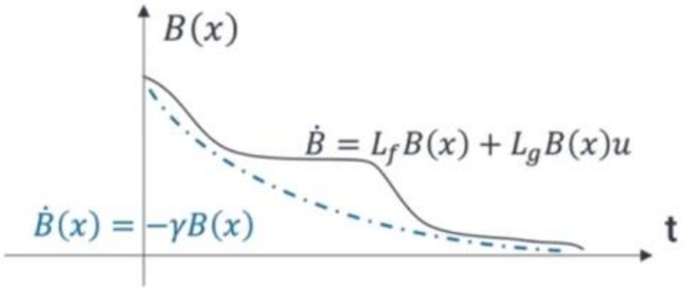
$$\mathcal{C} = \{\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n : h(\mathbf{x}) \geq 0\}. \tag{4}$$

Throughout this paper, we refer to $\mathcal{C}$ as a safe set. The function h is a control barrier function (CBF) [1] if $\frac{\partial h}{\partial \mathbf{x}} \neq 0$ for all $\mathbf{x} \in \partial\mathcal{C}$ and there exists an extended class $\mathcal{K}_\infty$ function γ such that for the control system (1), h satisfies

$$\exists\, \mathbf{u} \text{ s.t. } \dot{h}(\mathbf{x}, \mathbf{u}) \geq -\gamma(h(\mathbf{x})), \;\; \gamma \in \mathcal{K}_\infty. \tag{5}$$

This condition can be extended to the discrete-time domain which is shown as follows

$$\Delta h(\mathbf{x}_k, \mathbf{u}_k) \geq -\gamma h(\mathbf{x}_k), \;\; 0 < \gamma \leq 1, \tag{6}$$



Source: Jason Choi - "Introduction to Control Lyapunov Function and Control Barrier Function" (YouTube)

Problem Formulation

Dynamics and Cost Function

- Nonlinear dynamics

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

- Quadratic stage cost

$$\begin{aligned} J^*(x_k) = \min \quad & \sum_{t=k}^{k+N-1} \left[(x_{t|k} - x_{goal})^T Q (x_{t|k} - x_{goal}) \right. \\ & \left. + \ u_{t|k}^T R u_{t|k} \right] \\ & + \ (x_{k+N|k} - x_{goal})^T P (x_{k+N|k} - x_{goal}) \end{aligned}$$

Constraints

- Nonlinear state constraint must be satisfied

$$\sqrt{(x_{t|k}[0] - x_{obs})^2 + (x_{t|k}[1] - x_{obs})^2} - r_{obs} - r_{rob} \geq 0$$

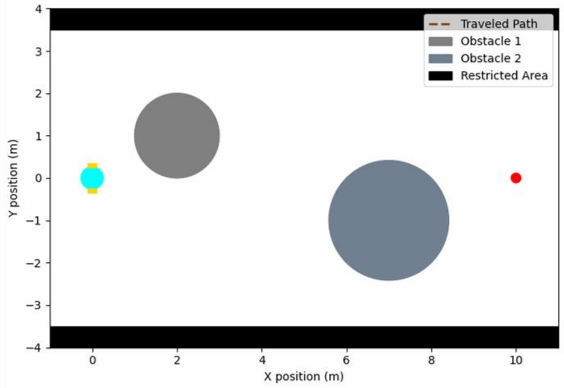
- Adjustable tolerance of error achieved by MPC-CBF

$$h(x_{t|k}) = \sqrt{(x_{t|k}[0] - x_{obs})^2 + (x_{t|k}[1] - x_{obs})^2} - r_{obs} - r_{rob}$$

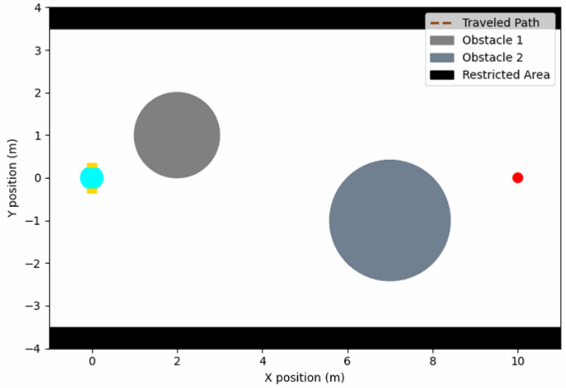
$$\Delta h(x_{t|k}) \geq -\gamma h(x_{t|k})$$

Implementation for Two Obstacles

- Validation of Both Algorithms
  - In a feedback control loop manner.
  - Based on shooting methods (ipopt). (N = 25, dt = 0.02sec, γ = 0.8)
  - Same Euclidean distance function (control barrier function) and safe distance are used for MPC-CBF.
- Implementation Details
  - Two static obstacles, x0 = [0,0,0], x_goal = [10,0,0].
  - Implemented in Python 3.10, use CasADi as optimization solver. Tested on Ubuntu OS.
  - Code is available at github (motion-planning-mpc).



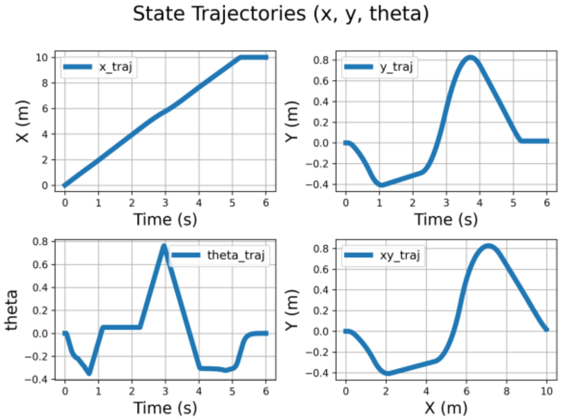
MPC - DC

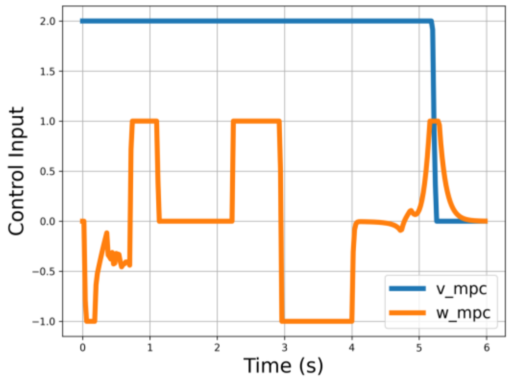


MPC - CBF

Example (MPC-DC)

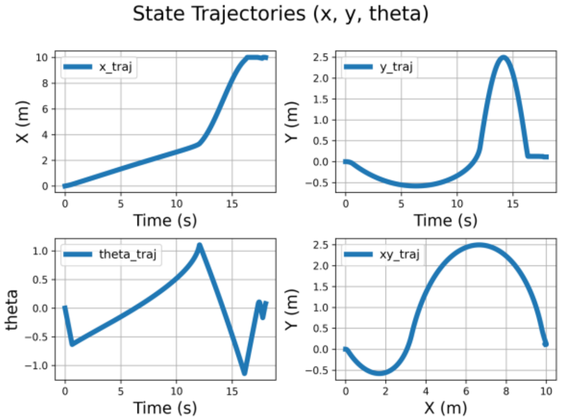
- MPC-DC
  - Q = np.diag([100,100,10]), R = np.diag([0.1,0.1]), H = 30*Q
  - state_min = [0, -3.5, -3.14], state_max = [10,3.5,3.14]
  - v_min = -2, v_max = 2, w_min = -1, w_max = 1
  - Euclidean distance is used to directly enforce the safety constraints.
  - Take around 6 seconds to reach the goal.
  - Tends to be more aggressive and resulting trajectory is relatively closer to the obstacles.

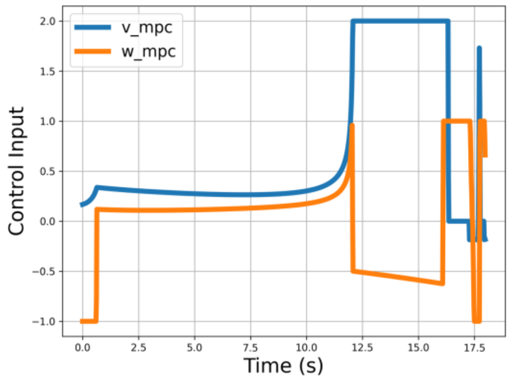




Example (MPC-CBF)

- MPC-CBF
  - Q = np.diag([100,100,10]), R = np.diag([0.1,0.1]), H = 30*Q
  - state_min = [0, -3.5, -3.14], state_max = [10,3.5,3.14]
  - v_min = -2, v_max = 2, w_min = -1, w_max = 1
  - Euclidean distance is used as the control barrier function to enforce the safety constraints.
  - Take around 18 seconds to reach the goal (γ = 0.8).
  - Tends to be more conservative. Resulting smaller control input at the beginning and relatively further from the obstacles.





MICHIGAN ENGINEERING
UNIVERSITY OF MICHIGAN

Thank you!

Any questions ?