保障安全的Control Barrier Function MPC

MPC 为了实现最佳性能，通常使状态和输入维持在约束的**极限附近**，这无法保证安全问题（可能会产生碰撞）。

距离约束 MPC

MPC-DC:

$$J_t^*(\mathbf{x}_t) = \min_{\mathbf{u}_{t:t+N-1|t}} p(\mathbf{x}_{t+N|t}) + \sum_{k=0}^{N-1} q(\mathbf{x}_{t+k|t}, \mathbf{u}_{t+k|t}) \quad (2a)$$

$$\text{s.t.} \quad \mathbf{x}_{t+k+1|t} = f(\mathbf{x}_{t+k|t}, \mathbf{u}_{t+k|t}),\ k = 0, ..., N-1 \quad (2b)$$

$$\mathbf{x}_{t+k|t} \in \mathcal{X}, \mathbf{u}_{t+k|t} \in \mathcal{U},\ k = 0, ..., N-1 \quad (2c)$$

$$\mathbf{x}_{t|t} = \mathbf{x}_t, \quad (2d)$$

$$\mathbf{x}_{t+N|t} \in \mathcal{X}_f, \quad (2e)$$

$$g(\mathbf{x}_{t+k|t}) \geq 0,\ k = 0, ..., N-1. \quad (2f)$$

其中 g 代表安全距离约束。

控制障碍函数 CBF

一个控制障碍函数 $h(x)$ 满足以下条件:

- $h(x)$ 在状态空间内是一个连续可微函数。
- 安全集合定义为: $C = \{x \in R^n \mid h(x) \geq 0\}$

为了确保系统状态始终处于安全集合C 内，需要控制输入 u 满足以下条件:

$\dot{h}(x) \geq -\alpha(h(x))$ 其中， α 通常选择为线性函数，例如 $\alpha(h) = \lambda h$，其中 $\lambda > 0$ 。

设系统状态方程为: $\dot{x} = f(x) + g(x)u$ ，那么使用链式法则，可以表示:

$$\dot{h}(x) = \frac{\partial h}{\partial f}(f(x) + g(x)u) \geq -\alpha(h(x))$$

以上就是控制障碍函数作为约束条件时，能够保证安全（保持在安全距离）。

对 $\dot{h}(x) \geq -\alpha(h(x))$ 的解释:

- 当 $h(x) = 0$ 时，不等式变为 $\dot{h}(x) \geq 0$ ，这意味着 $h(x)$ 不会变负，因此 x 不会离开安全集合 C 。
- 当 $h(x) > 0$ 时，不等式 $\dot{h}(x) \geq -\alpha(h(x))$ 确保了 $h(x)$ 的减少率受限，即 $h(x)$ 不会快速减少到负值。这确保了状态在有限时间内不会离开安全集合 C 。

CBF-MPC

MPC-CBF:

$$J_t^*(\mathbf{x}_t) = \min_{\mathbf{u}_{t:t+N-1|t}} p(\mathbf{x}_{t+N|t}) + \sum_{k=0}^{N-1} q(\mathbf{x}_{t+k|t}, \mathbf{u}_{t+k|t}) \tag{10a}$$

$$\text{s.t.} \quad \mathbf{x}_{t+k+1|t} = f(\mathbf{x}_{t+k|t}, \mathbf{u}_{t+k|t}),\ k = 0, ..., N-1 \tag{10b}$$

$$\mathbf{x}_{t+k|t} \in \mathcal{X}, \mathbf{u}_{t+k|t} \in \mathcal{U},\ k = 0, ..., N-1 \tag{10c}$$

$$\mathbf{x}_{t|t} = \mathbf{x}_t, \tag{10d}$$

$$\mathbf{x}_{t+N|t} \in \mathcal{X}_f, \tag{10e}$$

$$\Delta h(\mathbf{x}_{t+k|t}, \mathbf{u}_{t+k|t}) \geq -\gamma h(\mathbf{x}_{t+k|t}),\ k = 0, ..., N-1 \tag{10f}$$

具体的 CBF 例子

$$h_k = (\mathbf{x}_k(1) - x_{obs})^2 + (\mathbf{x}_k(2) - y_{obs})^2 - r_{obs}^2, \tag{17}$$

where x_{obs}, y_{obs}, and r_{obs} describe x/y-coordinate and radius of the obstacle with $x_{obs} = -2m$, $y_{obs} = -2.25m$ and $r_{obs} = 1.5m$, shown as a red circle in Fig. 4. The start and target positions are $(-5, -5)$ and $(0, 0)$, which are labelled as blue and red diamonds in Fig. 4, respectively.

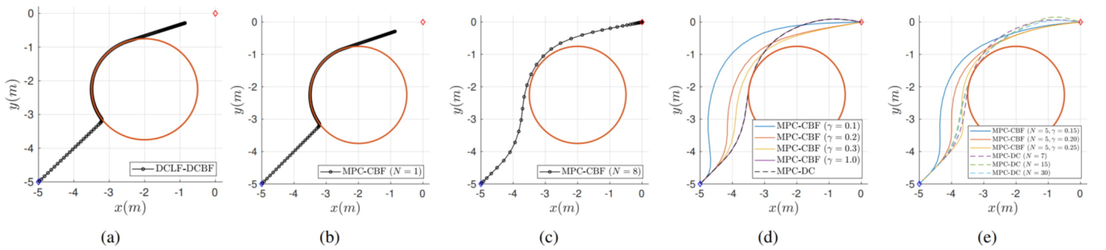

Fig. 4: A 2D double integrator avoids an obstacle using different control designs. The obstacle is represented by a red circle and the start and target positions are located at $(-5, -5)$ and $(0, 0)$, labelled as blue and red diamonds, respectively. (a) a DCLF-DCBF controller; (b) a MPC-CBF controller with $N = 1$; (c) a MPC-CBF controller with $N = 8$ and $\gamma = 0.5$; (d) a MPC-DC controller with $N = 8$ and four MPC-CBF controllers with $N = 8$ and different choices of γ; (e) three MPC-CBF controller with $N = 5$ and different values of γ and three MPC-DC controllers with different values of horizon N. Notice that for $N = 5$, MPC-DC becomes infeasible when the state is close to the boundary of the obstacle, whose trajectory is therefore excluded from (e).

结合 RG-MPC-AUV

增加一个线性的 CBF 作为约束即可。