Assignment 2- Report

Avinash Prabhu- 2018102027
Dipanwita Guhathakurta- 2018112004

Link to Videos-

https://drive.google.com/drive/folders/1zW2nCIS5h4kHIVlmrm0YTOgIkI1yPo6q?usp=sharing

Github Repo-

https://github.com/susiejojo/Holonomic_MPC

MPC overall structure (with obstacles case)

```
initialise initial guess velocities Vx, Vy as random (n×1) vector
while (!isnear(goal)){
        get P,q,G,h wrt Vx, Vy guess
        loop over obstacle_array {
                if obstacle in sensor-range{
                        append obstacle constraint
                        append trust region constraint
                }
        }
        sol = cvxopt.solvers.qp(P, q, G, h)
        Vx = sol['x'][:n]
        Vy = sol['y'][n:]
        for i in range(control_horizon){
                execute motion using Vx[i], Vy[i]
                i.e  x += Vx[i] Δt
                     y += Vy[i] Δt
                append x, y to path
        }
        Update Vx and Vy guesses
        Vx_guess = [Vx[c:n]  Vx[n-1]*ones(c,1)]
        Vy_guess = [Vy[c:n]  Vy[n-1]*ones(c,1)]
```

Handling obstacle avoidance

Planned velocities for planning horizon

Executing controls for control horizon

Updating our guesses

Without Obstacles

Planning horizon = ~~p~~ n
Control horizon = c .

Holonomic MPC without obstacles:

Holonomic kinematics: Initial robot position = (x_0, y_0)

$$x_1 = x_0 + v_0 \Delta t$$
$$x_2 = x_1 + v_1 \Delta t$$
$$\vdots$$
$$\therefore x_n = x_{n-1} + v_{n-1} \Delta t$$

i.e. $x_n = x_0 + \sum_{i=0}^{n-1} \dot{x}_i \Delta t$

Here our controls are $\dot{x}_0, \dot{x}_1, \ldots \dot{x}_{n-1}$.

Similarly $y_n = y_0 + \sum_{i=0}^{n-1} \dot{y}_i \Delta t$, controls being $\dot{y}_0, \dot{y}_1 \ldots \dot{y}_{n-1}$.

Cost fn (goal-reaching):

$$J = (x_n - x_g)^2 + (y_n - y_g)^2$$ Goal position = (x_g, y_g)

Now consider $x_n - x_g = (x_0 - x_g) + \sum_{i=0}^{n-1} \dot{x}_i \Delta t$

$$= (x_0 - x_g) + \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix} \Delta t$$

Similarly $y_n - y_g = (y_0 - y_g) + \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix} \Delta t$

$$\therefore (x_n - x_g)^2 = (x_0 - x_g)^2 + 2(x_0 - x_g) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix} \Delta t$$

$$+ \begin{bmatrix} \dot{x}_0 & \dot{x}_1 & \cdots & \dot{x}_{n-1} \end{bmatrix} (\Delta t)^2 \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{n\times 1} \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix}$$

and $$(y_n - y_g)^2 = (y_0 - y_g)^2 + 2(y_0 - y_g) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix} \Delta t$$

$$+ \begin{bmatrix} \dot{y}_0 & \dot{y}_1 & \cdots & \dot{y}_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{n\times 1} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix} (\Delta t)^2$$

We construct $\mathcal{X} = \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \\ \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix}_{2n\times 1}$

Now
$$(x_n - x_g)^2 + (y_n - y_g)^2 = (x_0 - x_g)^2 + (y_0 - y_g)^2 + 2\Delta t (x_0 - x_g) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix}$$
$$+ 2\Delta t (y_0 - y_g) \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times n} \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix} + \begin{bmatrix} \dot{x}_0 & \dot{x}_1 & \cdots & \dot{x}_{n-1} \end{bmatrix} (\Delta t)^2 \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & & & \vdots \\ 1 & \cdots & \cdots & 1 \end{bmatrix}_{n\times n} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix}$$
$$+ \begin{bmatrix} \dot{y}_0 & \dot{y}_1 & \cdots & \dot{y}_{n-1} \end{bmatrix} (\Delta t)^2 \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & & & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{n\times n} \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix}$$

w.r.t $\mathcal{X}$,
$$J = (x_0 - x_g)^2 + (y_0 - y_g)^2 + 2\Delta t \Big[\overbrace{(x_0 - x_g)\ (x_0 - x_g) \cdots (x_0 - x_g)}^{n \text{ terms}}\ \overbrace{(y_0 - y_g) \cdots (y_0 - y_g)}^{n \text{ terms}} \Big]_{1\times 2n} \cdot \mathcal{X}$$
$$+ \mathcal{X}^T (\Delta t)^2 \left[\begin{array}{cccc|cccc} 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ \vdots & & & & \vdots & & & \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ \hline 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\ \vdots & & & & 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \end{array}\right]_{2n\times 2n} \mathcal{X}$$

We can now write J in QP form as
$$J = \frac{1}{2} \mathcal{X}^T P \mathcal{X} + q^T \mathcal{X} + c$$
where
$$P = 2(\Delta t)^2 \left[\begin{array}{cccc|cccc} 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ \hline 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\ \vdots & & & & 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \end{array}\right]_{2n\times 2n} = \mathrm{kron}(\mathrm{eye}(2), A)$$

A being $\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & & & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{n\times n}$

$$q = 2\Delta t \Big[(x_0 - x_g)\ (x_0 - x_g) \cdots (x_0 - x_g)\ (y_0 - y_g) \cdots (y_0 - y_g) \Big]^T$$

Constraints:

(1)
$\dot{x}_0 \le v_{max}$
$\dot{x}_1 \le v_{max}$
$\vdots$
$\dot{x}_{n-1} \le v_{max}$

$$\left.\begin{array}{l} \dot{x}_0 \ge 0 \\ \vdots \\ \dot{x}_{n-1} \ge 0 \end{array}\right\} = \begin{array}{l} -\dot{x}_0 \le 0 \\ \vdots \\ -\dot{x}_{n-1} \le 0 \end{array}$$

Similarly for y

$$G = \left[\begin{array}{ccccc} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & & \cdots & 1 \\ \hline -1 & 0 & 0 & & 0 \\ 0 & -1 & 0 & & 0 \\ & & & & -1 \\ 0 & 0 & & & \end{array}\right] \begin{array}{l} \updownarrow 2n \text{ constraints} \\ \\ \updownarrow 2n \text{ constraints} \end{array} \qquad h = \begin{bmatrix} v_{max} \\ v_{max} \\ \vdots \\ v_{max} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{array}{l} \updownarrow 2n \\ \\ \updownarrow 2n \end{array}$$

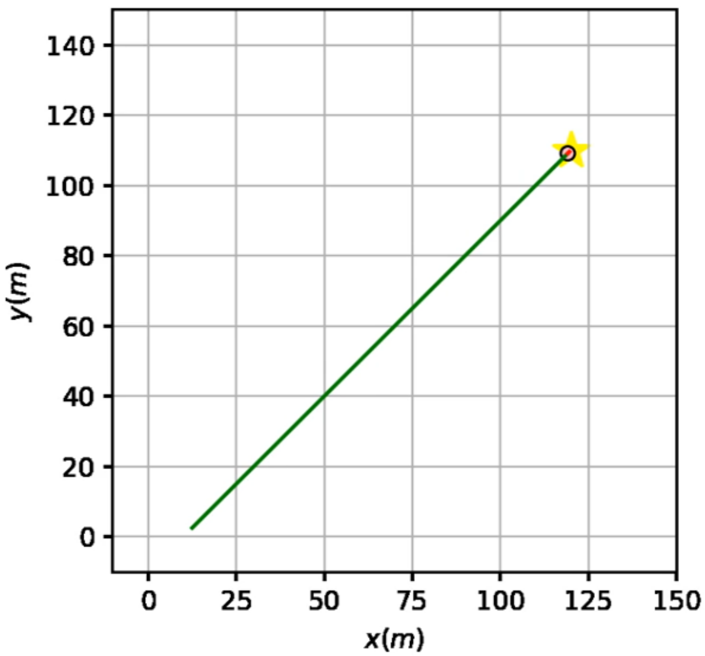
So we finally have

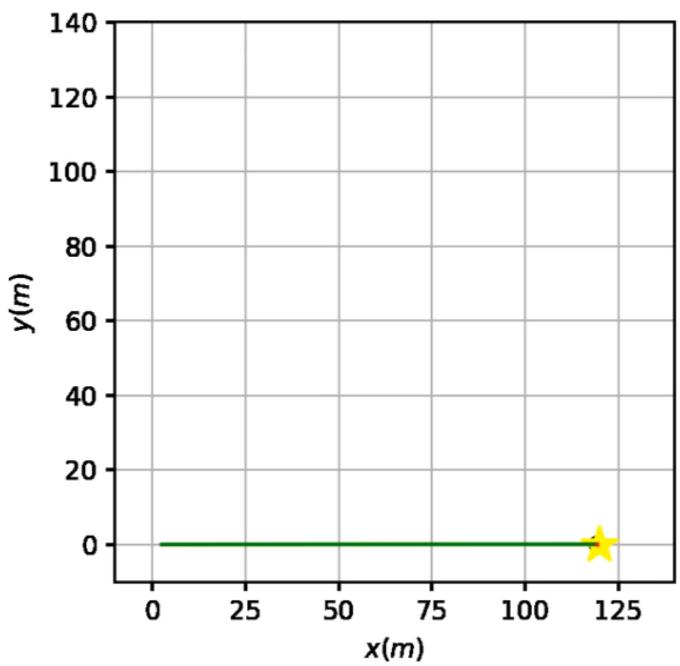
$$\min_{x} J = \min_{x} \left(\frac{1}{2} x^T P x + q^T x \right) \quad \text{subject to}$$

$$\cancel{G^T x \leq h}$$

$$G x \leq h .$$

Outputs





With Obstacles

Holonomic MPC with obstacles

The cost function remains the same as in the case without obstacles. However we now introduce the collision avoiding constraints

Let the obstacle position be (x_{ob}, y_{ob})

$$(x_1 - x_{ob})^2 + (y_1 - y_{ob})^2 \geq R^2 \qquad (R = r + r_{ob})$$

$$(x_2 - x_{ob})^2 + (y_2 - y_{ob})^2 \geq R^2$$ where r is radius of robot, r_{ob} is radius of obstacle.

$\vdots$

$$(x_n - x_{ob})^2 + (y_n - y_{ob})^2 \geq R^2$$

These constraints are concave w.r.t $\dot{x}_0, \dot{x}_1 \ldots \dot{x}_{n-1}, \dot{y}_0, \dot{y}_1, \ldots \dot{y}_{n-1}$. So we need to linearise them.

Consider $(x_n - x_{ob})^2 + (y_n - y_{ob})^2 \geq R^2$

i.e $$\left(x_0 + \sum_{i=0}^{n-1} \dot{x}_i \Delta t - x_{ob}\right)^2 + \left(y_0 + \sum_{i=0}^{n-1} \dot{y}_i \Delta t - y_{ob}\right)^2 \geq R^2$$

We will linearise wrt $\dot{x}_0, \dot{x}_1, \ldots \dot{x}_{n-1}, \dot{y}_0, \dot{y}_1 \ldots \dot{y}_{n-1}$.

$$f(\dot{x}_0, \dot{x}_1, \dot{x}_2 \ldots \dot{x}_{n-1}, \dot{y}_0, \dot{y}_1, \ldots \dot{y}_{n-1})$$
$$= f(\dot{x}_0^*, \dot{x}_1^*, \dot{x}_2^*, \dot{x}_3^*, \ldots \dot{x}_{n-1}^*, \dot{y}_0^*, \ldots \dot{y}_{n-1}^*)$$
$$+ \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} (\dot{x}_0 - \dot{x}_0^*) + \left.\frac{\partial f}{\partial \dot{x}_1}\right|_{\dot{x}_1 = \dot{x}_1^*} (\dot{x}_1 - \dot{x}_1^*) \ldots$$
$$+ \ldots \left.\frac{\partial f}{\partial \dot{y}_1}\right|_{\dot{y}_1 = \dot{y}_1^*} (\dot{y}_1 - \dot{y}_1^*) \ldots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} (\dot{y}_{n-1} - \dot{y}_{n-1}^*)$$

$$\left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} = \cancel{2\Delta t (\dot{x}_0 - x} \quad 2\Delta t \left(x_0 - x_{ob} + (\dot{x}_0^* + \dot{x}_1^* + \ldots \dot{x}_{n-1}^*)\Delta t\right)\Delta t$$
$$= 2\Delta t (x_0 - x_{ob}) + 2(\Delta t)^2 [1 \; 1 \; \ldots \; 1]_{1 \times n} \begin{bmatrix} \dot{x}_0^* \\ \dot{x}_1^* \\ \vdots \\ \dot{x}_{n-1}^* \end{bmatrix}$$

$$f(\dot{x}_0^*, \dot{x}_1^*, \dots \dot{y}_{n-1}^*) = \left(x_0 + \sum_{i=0}^{n-1} \dot{x}_i^* \Delta t - x_{ob}\right)^2 + \left(y_0 + \sum_{i=0}^{n-1} \dot{y}_i^* \Delta t - y_{ob}\right)^2$$

$\rightarrow$ constant $= c_1$

$$f(\dot{x}_0, \dot{x}_1, \dots \dot{y}_{n-1}) = c_1 + \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0 - \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0^* + \dots$$

$$\dots - \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1}^*$$

Let $\left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0^* + \left.\frac{\partial f}{\partial \dot{x}_1}\right|_{\dot{x}_1 = \dot{x}_1^*} \dot{x}_1^* + \dots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1}^* = c_2.$

$$\therefore f(\dot{x}_0, \dot{x}_1, \dots \dot{y}_{n-1}) = c_1 - c_2 + \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0 + \left.\frac{\partial f}{\partial \dot{x}_1}\right|_{\dot{x}_1 = \dot{x}_1^*} \dot{x}_1 + \dots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1}$$

This is linear w.r.t $\dot{x}_0, \dot{x}_1, \dots \dot{y}_{n-1}$.

Linearised obstacle avoiding constraint:

$$c_1 - c_2 + \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0 + \dots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1} \geq R^2$$

i.e $-\left(c_1 - c_2 + \left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0 + \dots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1}\right) \leq -R^2$

$$-\left(\left.\frac{\partial f}{\partial \dot{x}_0}\right|_{\dot{x}_0 = \dot{x}_0^*} \dot{x}_0 + \dots \left.\frac{\partial f}{\partial \dot{y}_{n-1}}\right|_{\dot{y}_{n-1} = \dot{y}_{n-1}^*} \dot{y}_{n-1}\right) \leq -R^2 + (c_1 - c_2)$$

Now this linearisation is valid only within a trust region around the linearisation/operating point. So we add further constraints:

$\dot{x}_0 \leq \dot{x}_0^* + \Delta\dot{x}$ and $\dot{x}_0 \geq \dot{x}_0^* - \Delta\dot{x}$

$\dot{x}_1 \leq \dot{x}_1^* + \Delta\dot{x}$ i.e $-\dot{x}_0 \leq -(\dot{x}_0^* - \Delta\dot{x})$

$-\dot{x}_1 \leq -(\dot{x}_1^* - \Delta\dot{x})$

$\vdots$

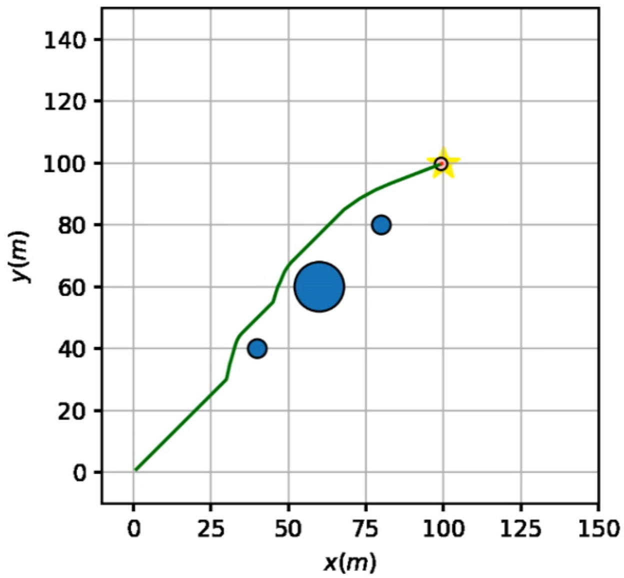
$\dot{x}_{n-1} \leq \dot{x}_{n-1}^* + \Delta\dot{x}$ $\quad -\dot{x}_{n-1} \leq -(\dot{x}_{n-1}^* - \Delta\dot{x})$

All of these constraints get stacked, with the velocity bounds in matrices G and h as shown in the without obstacles case.

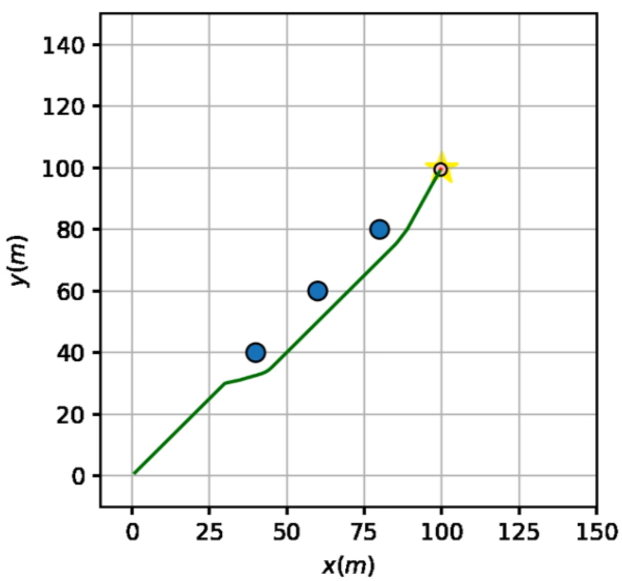
For multiple obstacles we can perform the similar linearisation for each of the constraints:

$(x_n - x_{obs}[i])^2 + (y_n - y_{obs}[i])^2 \geq R_i^2$ where $R = r + r_{obs}[i]$

Outputs



140
120
100
80
60
40
20
0
y(m)
0 25 50 75 100 125 150
x(m)



140
120
100
80
60
40
20
0
y(m)
0 25 50 75 100 125 150
x(m)