CSE471: Advanced Robotics & Control **Spring 2018**

Lecture 7: March 29

Lecturer: *Scribes: Avijit Ashe, Roll:20172067*

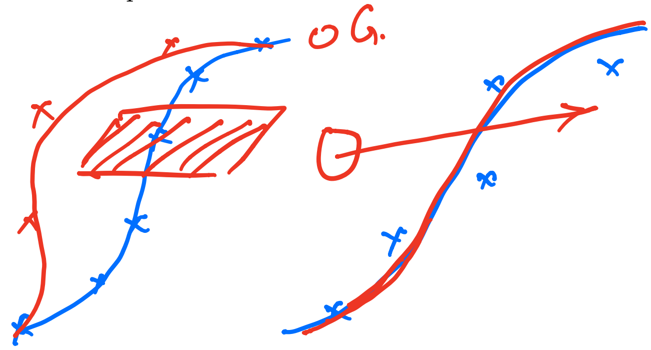
Note: *LaTeX template courtesy of UC Berkeley EECS dept.*
Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

7.1 Derivations for the trajectory optimization problem

We consider a **holonomic system**, and it's motion model. That is, a system whose constraints can be expressed in integrable form that equates to zero across all the positional constraints.

7.1.1 Objectives

- 1. Goal reaching
- 2. Obstacle avoidance
- 3. Velocity and acceleration bounds
- 4. Smoothness



We now build on the objectives above and formulate the entire trajectory optimization problem

7.2 Objective: Goal reaching

We need to minimize this :

$$(x_{robo}(t) - x_g)^2 + (y_{robo}(t) - y_g)^2 \tag{7.1}$$

where, the

$$(x_{robo}, y_{robo})$$

is a function of 't' or discrete time steps, and

$$(x_g, y_g)$$

is the goal in world coordinates

Now, if a holonomic motion model is considered i.e. f(n1, n2...; t) = 0, we get the following form

$$x_{robo}(t) = x_{robo}(t-1) + x_{robo}\dot{(}t-1)x\delta t$$

or

$$x_{robo}(n) = x_{robo}(n-1) + x_{robo}\dot{(}n-1)x\delta t$$

where, the robo is at *nth* position at time *t* given by the physical coordinates *(x(n), y(n))*. The starting point is assumed to be at zero or origin.

The steps are: *(n-1)th* position at time *t-1*, *nth* position at time *t*, it takes a small time-step to reach from (n-1) to n, and $(\dot{x_n}, \dot{y_n})$ are the velocities at *nth* position

So, the objective is to find **the set** of

$$\dot{x}$$

the velocity component along x and

$$\dot{y}$$

the velocity component along y, that would push the robo from any position in the world towards the goal,

$$(x_g, y_g)$$

.

Only that set minimizes the above equation 7.1

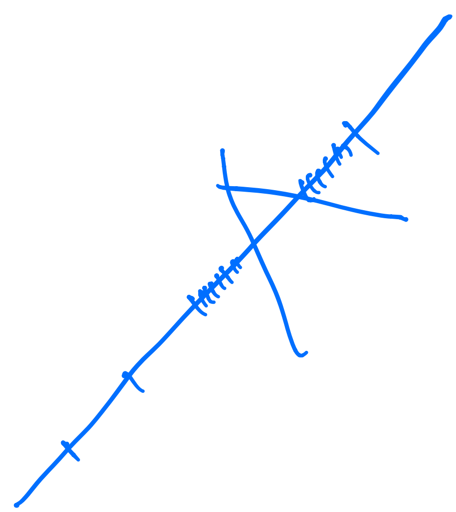
More specifically, we try to solve for a set of $\dot{x_0}, \dot{x_1}, \dot{x_2}, \dot{x_3}, ...\dot{x_{n-1}}$ and $\dot{y_0}, \dot{y_1}, \dot{y_2}, \dot{y_3}, ...\dot{y_{n-1}}$ that would push the robo from any position in the world towards the goal, (x_g, y_g).

Now onwards, (x_{robo}, y_{robo}) is written as (x_r, y_r)

Now, the equation for holonomic model is used below

$$
\begin{aligned}
x_n &= x_{n-1} + \dot{x_{n-1}}\delta t \\
x_{n-1} &= x_{n-2} + \dot{x_{n-2}}\delta t \\
x_{n-2} &= x_{n-3} + \dot{x_{n-3}}\delta t
\end{aligned}
$$

$$\cdots$$
$$\cdots$$
$$\cdots$$



$$x_1 = x_0 + \dot{x_0}\delta t$$

That is,

$$
\begin{aligned}
x_n &= x_0 + (\dot{x_0}, \dot{x_1}, \dot{x_2}, \dot{x_3}, ...\dot{x_{n-1}})\delta t \\
&= x_0 + \sum_{i=0}^{n-1} \dot{x_i}\delta t
\end{aligned}
$$

and similarly for the y-component like

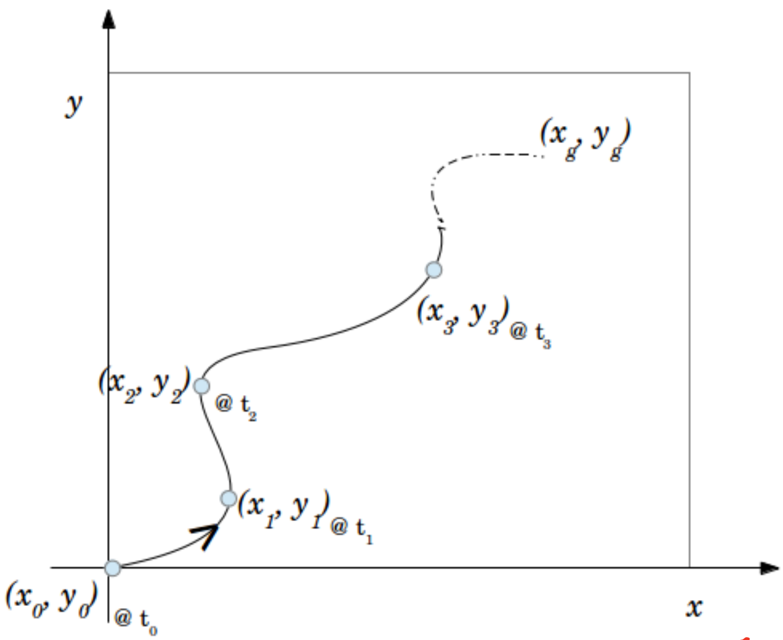
$$
\begin{aligned}
y_n &= y_0 + (\dot{y_0}, \dot{y_1}, \dot{y_2}, \dot{y_3}, ...\dot{y_{n-1}})\delta t \\
&= y_0 + \sum_{i=0}^{n-1} \dot{y_i}\delta t
\end{aligned}
$$

7.2.1 Deriving goal reaching as a quadriatic equation

Let us assume that we have just 3 control points (or way points through which the robo must pass through as close as possible) for simplicity

So, we have $\dot{x_0}, \dot{x_1}, \dot{x_2}$ and $\dot{y_0}, \dot{y_1}, \dot{y_2}$

Here is a small illustration of the same



· Min $X^T P X + q^T X$
sut to:
$GX \leq h$.
$AX = b$.

Figure 7.1: Positional constraints on robo trajectory

The cost function from equation 7.1 can be rewritten as

$$(x_3 - x_g)^2 + (y_3 - y_g)^2$$

Upon expanding it using equation 7.2, we get

$$(x_0 + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t - x_g)^2 + (y_0 + (\dot{y_0} + \dot{y_1} + \dot{y_2})\delta t - y_g)^2$$

Writing the x-term in quadriatic form, we have

$$(\quad (x_0 - x_g) \quad + \quad (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t) \quad)^2 \tag{7.2}$$

following $(a+b)^2 = a^2 + b^2 + 2ab$, where $a = (x_0 - x_g)$ and $b = (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t$. So, we have

$$(x_o - x_g)^2 \quad + \quad (\dot{x_0} + \dot{x_1} + \dot{x_2})^2 \delta t^2 \quad + \quad 2(x_0 - x_g)(\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t$$

7.2.2 Rewriting 1st term from the quadriatic form

$(x_0 - x_g)^2$ is just a constant C1 as they're both known $(x_0, y_0) = (0, 0)$ or the origin and (x_g, y_g) can be some arbitrary values e.g. (10, 10) in units

7.2.3 Rewriting 2nd term from the quadriatic form

Let

$$X = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x}_0 & \dot{x}_1 & \dot{x}_2 \end{bmatrix} I_{3\times 3} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}$$

That is, $\dot{x_0} + \dot{x_1} + \dot{x_2}$ can be written in matrix-vector form like

$$\begin{bmatrix} \dot{x_0} & \dot{x_1} & \dot{x_2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

and let's call it K^T

So, $(\dot{x_0} + \dot{x_1} + \dot{x_2})^2 \delta t^2 = K^T K$ from linear algebra, like so

$$K^T K = \begin{bmatrix} \dot{x_0} & \dot{x_1} & \dot{x_2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{x_0} \\ \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} \dot{x_0} & \dot{x_1} & \dot{x_2} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{x_0} \\ \dot{x_1} \\ \dot{x_2} \end{bmatrix} \tag{7.3}$$

7.2.4 Rewriting last term from the quadriatic form

$$2(x_0 - x_g)(\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t = 2(x_0 - x_g)\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{x_0} \\ \dot{x_1} \\ \dot{x_2} \end{bmatrix} = q_1^T X$$

The final matrix vector form can be rewritten as

$$\boxed{min_{X,Y} \quad X^T A X + q_1^T X + C1 \quad + \quad Y^T B Y + q_2^T Y + C2} \tag{7.4}$$

where,

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, B = A$$

$$q_1 = 2(x_0 - x_g)\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \quad q_2 = 2(y_0 - y_g)\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

If done for n time instance.

$$q_1 = 2(x_0 - x_g)\delta \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}_{1\times n} \quad q_2 = 2(y_0 - y_g)\delta t \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}_{1\times n}.$$

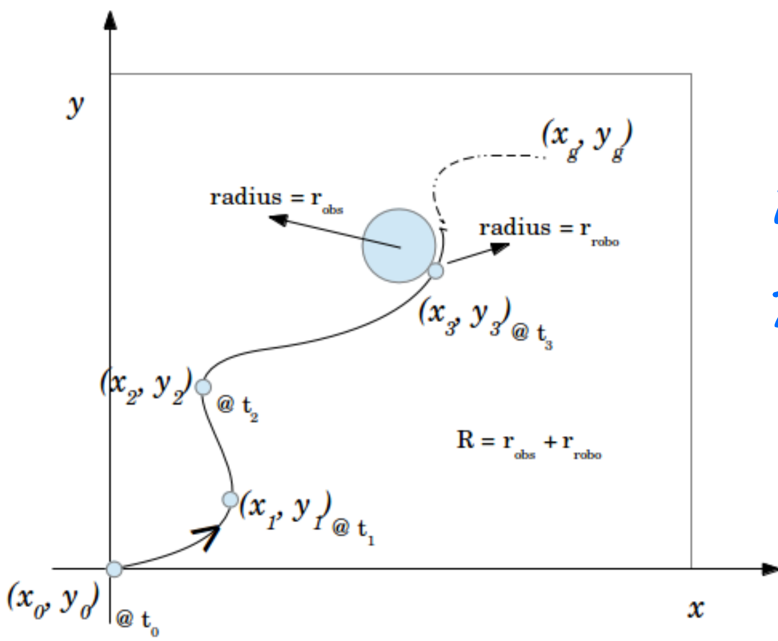
Then $\underset{1\times 2n}{q^T} \underset{2n\times 1}{X} = [\underset{1\times n}{q_1} \quad \underset{1\times n}{q_2}] \begin{bmatrix} \dot{x}_0 \\ \vdots \\ \dot{x}_n \\ \dot{y}_0 \\ \vdots \\ \dot{y}_n \end{bmatrix}_{2n\times 1}$

$$X = \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix}, Y = \begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \dot{y}_2 \\ \vdots \\ \dot{y}_{n-1} \end{bmatrix}$$

7.3 Objective: Obstacle avoidance

If we have an obstacle at (x_ob, y_ob) then an appropriate way to avoid it could be written as

$$(x_n - x_{ob})^2 + (y_n - y_{ob})^2 \cancel{<}\, > R^2$$
$$-\left[(x_n - x_{ob})^2 + (y_n - y_{ob})^2 - R^2\right] < 0$$



$\dot{x}_0 \leq \dot{x}_{max}$

$\dot{x}_i \leq \dot{x}_{max}$

$\dot{y}_i \leq \dot{y}_{max}$

To get it in the matrix-vector form $AX = B$ we linearize the objective function 7.2 using Taylor's expansion up to 1st order.

We use only 3 control/way points for ease thus framing the equation as

$$-\left[(x_3 - x_{ob})^2 + (y_3 - y_{ob})^2 - R^2\right] < 0$$

expanding the terms using expansion of 7.2 from above quadriatic form

$$-\left[(x_0 + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t - x_{ob})^2 + (y_0 + (\dot{y_0} + \dot{y_1} + \dot{y_2})\delta t - y_{ob})^2 - R^2\right] < 0$$
$$-\left[(x_0 - x_{ob} + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t)^2 + (y_0 - y_{ob} + (\dot{y_0} + \dot{y_1} + \dot{y_2})\delta t)^2 - R^2\right] < 0$$

Thus, ignoring the -ve sign, R^2 and using just x-component at a time we apply MTS (multivariate taylor series) $(x_0 - x_{ob} + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t)^2$

7.3.1 Multivariate Taylor Series

We linearize 7.2 for x-term now as

$$f(\dot{x_0} + \dot{x_1} + \dot{x_2}) = f(\dot{x_0^*} + \dot{x_1^*} + \dot{x_2^*}) + \frac{\delta f}{\delta \dot{x_0}}|_{\dot{x_0},\dot{x_1},\dot{x_2},}(\dot{x_0} - \dot{x_0^*}) + \frac{\delta f}{\delta \dot{x_1}}|_{\dot{x_0},\dot{x_1},\dot{x_2},}(\dot{x_1} - \dot{x_1^*}) + \frac{\delta f}{\delta \dot{x_2}}|_{\dot{x_0},\dot{x_1},\dot{x_2},}(\dot{x_2} - \dot{x_2^*})$$

$$f(\dot{x_0} + \dot{x_1} + \dot{x_2}) = f(\dot{x_0^*} + \dot{x_1^*} + \dot{x_2^*}) + \begin{bmatrix} \frac{\delta f}{\delta \dot{x_0}} & \frac{\delta f}{\delta \dot{x_1}} & \frac{\delta f}{\delta \dot{x_2}} \end{bmatrix}_{\dot{x_0},\dot{x_1},\dot{x_2},} \begin{bmatrix} (\dot{x_0} - \dot{x_0^*}) \\ (\dot{x_1} - \dot{x_1^*}) \\ (\dot{x_2} - \dot{x_2^*}) \end{bmatrix}$$

Now, $f = (x_0 - x_{ob} + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t)^2$, So,

$$\frac{df}{d\dot{x}_0} = 2\left((x_0 - x_{ob}) + (\dot{x}_0 + \dot{x}_1 + \dot{x}_2)\delta t\right)\delta t$$

$$\frac{\delta f}{\delta \dot{x_0}} = 2\Delta t((x_0 - x_{ob}) + (\dot{x_0} + \dot{x_1} + \dot{x_2})\Delta t)$$
$$= 2\Delta t((x_0 - x_{ob}) + 2\Delta t^2 \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{x_0^*} \\ \dot{x_1^*} \\ \dot{x_2^*} \end{bmatrix}$$

and similarly for

$$\frac{\delta f}{\delta \dot{x_1}}, \frac{\delta f}{\delta \dot{x_2}}$$

Now, considering the quadriatic form of 7.2 $(x_0 - x_{ob} + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t)^2 = X^T AX + q_1^T X + C1$, we get

7.3.2 1st term of Taylor

$X_*^T AX_* + q_1^T X_* + C1$ where $X_* = \begin{bmatrix} \dot{x_0^*} \\ \dot{x_1^*} \\ \dot{x_2^*} \end{bmatrix}$

7.3.3 2nd term of Taylor

$$\frac{\delta X_*^T AX_* + q_1^T X_* + C1}{\delta \dot{x_0}} = (2AX_* + 2q_1)^T \begin{bmatrix} (\dot{x_0} - \dot{x_0^*}) \\ (\dot{x_1} - \dot{x_1^*}) \\ (\dot{x_2} - \dot{x_2^*}) \end{bmatrix}$$

Finally, we have

$$-\left[(x_0 - x_{ob} + (\dot{x_0} + \dot{x_1} + \dot{x_2})\delta t)^2 + (y_0 - y_{ob} + (\dot{y_0} + \dot{y_1} + \dot{y_2})\delta t)^2 - R^2\right] < 0$$

in quadriatic form is

$$\boxed{-\left[X^T AX + q_1^T X + C1 \quad + \quad X^T BY + q_2^T Y + C2\right]} \tag{7.5}$$

where,

$$A = \Delta t^2 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, B = A$$

$$q_1 = 2(x_0 - x_{ob})\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \quad q_2 = 2(y_0 - y_{ob})\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$X = \begin{bmatrix} \dot{x_0} \\ \dot{x_1} \\ \dot{x_2} \end{bmatrix} \quad , Y = \begin{bmatrix} \dot{y_0} \\ \dot{y_1} \\ \dot{y_2} \end{bmatrix}$$

such that after linearization it becomes

$-\left[(X^T AX + q_1^T X + C1 + Y^T BY + q_2^T Y + C2)|_{X^*,Y^*} + (2AX + 2q_1)^T (X - X^*)|_{X^*} + (2BY + 2q_2)^T (Y - Y^*)|_{Y^*}\right] - R^2\right] < 0$

7.4 Objective with constraints

Finally, we get the objectives for goal reaching and obstacle avoidance together into a convex optimization equation

$$-\left[X^T AX + q_1^T X + C1 \quad + \quad X^T BY + q_2^T Y + C2\right] \tag{7.6}$$

such that

$$-\left[Constant + A_{matrix}(X - X^*) + B_{matrix}(Y - Y^*) - R^2\right] < 0$$

where,

$$A_{matrix} = (2AX + 2q_1)^T|_{X^*}, \quad A = \Delta t^2 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad q_1 = 2(x_0 - x_{ob})\delta t \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

References

[ML13] Figures 7.1 to 7.4 generated in Matlab

Min $(x_n - x_g)^2 + (y_n - y_g)^2$

sub to $x_n = x_{n-1} + \dot{x}_{n-1}\,\delta t$; $y_n = y_{n-1} + \dot{y}_{n-1}\,\delta t$

$\vdots \qquad\qquad \vdots$

$x_1 = x_0 + \dot{x}_0\,\delta t$; $y_1 = y_0 + \dot{y}_0\,\delta t$

$(x_n - x_g)^2$ can be written as

$$\left((x_0 - x_g) + (\dot{x}_0 + \dot{x}_1 + \dots + \dot{x}_{n-1})\,\delta t\right)^2 +$$

$$\left((y_0 - y_g) + (\dot{y}_0 + \dot{y}_1 + \dots + \dot{y}_{n-1})\,\delta t\right)^2.$$

This will have terms like:

$$(\dot{x}_0 + \dot{x}_1 + \dots + \dot{x}_{n-1})^2\,\delta t^2 \ \& \ (\dot{y}_0 + \dot{y}_1 + \dots + \dot{y}_{n-1})^2\,\delta t^2.$$

$$\downarrow$$

$$\underbrace{\begin{bmatrix} \dot{x}_0 & \cdots & \dot{x}_{n-1} \end{bmatrix}}_{1\times n} \underbrace{\begin{bmatrix} 1 & 1 & \cdots & 1 \\ \vdots & & & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}}_{n\times n} \underbrace{\begin{bmatrix} \dot{x}_0 \\ \vdots \\ \dot{x}_{n-1} \end{bmatrix}}_{n\times 1}$$

$= X^T A X$. & similarly for the $\dot{y}$'s as

$Y^T B Y$

Then $X^T A X + Y^T B Y$ is nothing but.

$$\underbrace{[X^T \quad Y^T]}_{1\times 2n} \begin{bmatrix} A_{n\times n} & 0_{n\times n} \\ 0_{n\times n} & B_{n\times n} \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix}_{2n\times 1}$$

$$= X^T_{(1\times 2n)} P_{2n\times 2n} X_{2n\times 1}$$

and from before $q^T X$ is of the form

$$[q_1{}^T{}_{1\times n} \quad q_2{}^T{}_{1\times n}] X_{2n\times 1}$$

The above is the optimal control formulation for goal reaching cost with holonomic constraints where control variable are velocities.

HW: Derive for control as acceleration or double integrator system. with velocity bounds

$\dot{x}_m \leq \dot{x}_1 \leq \dot{x}_M$

$\dot{y}_m \leq \dot{y}_1 \leq \dot{y}_M$

$\vdots$

$\dot{x}_m \leq \dot{x}_n \leq \dot{x}_M$

$\dot{y}_m \leq \dot{y}_n \leq \dot{y}_M$

The formulation $X^T P X + q^T X$ s.t $GX \leq h$.
is an optimization problem or optimal control problem.

- → solve for X the control variable.
- → forward evolve by this control for dt
- → resolve the optimization problem from the current state obtained after forward evolution.
- → Repeat till goal state is reached.

↓

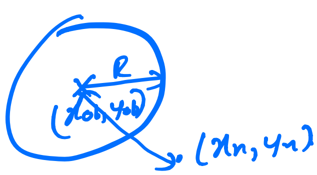
MPC.

Obstacle avoidance constraint:

for time instant n:

$$(x_n - x_{ob})^2 + (y_n - y_{ob})^2 > R^2 \longrightarrow (1).$$

R ↓ robot + obstacle size.
approximated as a circle



$$\Rightarrow -\left[(x_n - x_{ob})^2 + (y_n - y_{ob})^2 - R^2\right] < 0 \longrightarrow (2).$$

Consider $n = 3$

$$x_3 = x_0 + (\dot{x}_0 + \dot{x}_1 + \dot{x}_2)\,\delta t, \quad \text{Ilarly for } y_3 \longrightarrow (3).$$

To get it of the form $ax \leq h$, which is convex, we need to linearize (3) about a certain $\hat{\dot{x}}_0, \hat{\dot{x}}_1, \hat{\dot{x}}_2$

$$(x_3 - x_{ob})^2 = (x_0 + (\dot{x}_1 + \dot{x}_2 + \dot{x}_3)\delta t - x_{ob})^2.$$

Linearize about $\hat{\dot{x}}_1, \hat{\dot{x}}_2, \hat{\dot{x}}_3$ to get.

$$(x_3 - x_{ob})^2 = (x_0 + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t - x_{ob})^2.$$

$$+ 2\left[(x_0 - x_{ob}) + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t\right]\delta t$$

$$\left[(\dot{x}_0 - \hat{\dot{x}}_0) + (\dot{x}_1 - \hat{\dot{x}}_1) + (\dot{y}_1 - \hat{\dot{y}}_1)\right] \longrightarrow (4).$$

$$= c_{1x} - 2\left[(x_0 - x_{ob})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t^2\right].\left[\hat{\dot{x}}_0 + \hat{\dot{x}}_1 + \hat{\dot{x}}_2\right]$$

$$+ 2\left[(x_0 - x_{ob})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t^2\right]\left[\dot{x}_0 + \dot{x}_1 + \dot{x}_2\right]$$

$$= c_{1x} + c_{2x} + \begin{bmatrix} q_{1x} & q_{1x} & q_{1x} \end{bmatrix} \begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} \longrightarrow (5).$$

$$c_{1x} = (x_0 + (\dot{x}_1 + \dot{x}_2 + \dot{x}_3)\delta t - x_{ob})^2 \longrightarrow (6)$$

$$c_{2x} = -2\left[(x_0 - x_{ob})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_2)\delta t^2\right].$$

$$\left[\hat{\dot{x}}_0 + \hat{\dot{x}}_1 + \hat{\dot{x}}_2\right] \longrightarrow (7).$$

$$q_{1x} = 2\left[(x_0 - x_{ob})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t^2\right] \longrightarrow (8)$$

||rly get c_{1y}, c_{2y}, q_{1y}

We need to do this for all $X_1, \dots, X_n$

where $X_i = [x_i \; y_i]^T$

We need to stack together terms to get it of the form $Gx \leq h$.

Generalizing the Obstacle Avoidance Constraint over n time steps into the future:

From (6) we get for $n=3$

$$C_{1x} = C_{31x} \text{(new notation)} = (x_0 - x_{0t} + (\dot{x}_1 + \dot{x}_2 + \dot{x}_3)\delta t)^2 \longrightarrow (9)$$

$$C_{31y} = (y_0 - y_{0t} + (\dot{y}_1 + \dot{y}_2 + \dot{y}_3)\delta t)^2 \longrightarrow (10)$$

$$C_{31} = C_{31x} + C_{31y} \longrightarrow (11)$$

C_{2x} or C_{32x} (new notation)

$$= -2[(\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)[(x_0 - x_{0t})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t^2] \longrightarrow (12)$$

and similarly

$$C_{32y} = -2[(\hat{\dot{y}}_1 + \hat{\dot{y}}_2 + \hat{\dot{y}}_3)[(y_0 - y_{0t})\delta t + (\hat{\dot{y}}_1 + \hat{\dot{y}}_2 + \hat{\dot{y}}_3)\delta t^2] \longrightarrow (13)$$

$$C_{32} = C_{32x} + C_{32y} \longrightarrow (14)$$

$$q_{3x} = 2((x_0 - x_{0t})\delta t + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \hat{\dot{x}}_3)\delta t^2) \longrightarrow (15)$$

$$q_{3y} = 2[(y_0 - y_{0t})\delta t + (\hat{\dot{y}}_1 + \hat{\dot{y}}_2 + \hat{\dot{y}}_3)\delta t^2] \longrightarrow (16)$$

This easily generalizes over a time horizon.

For example $t=n$ we have

$$C_{n1} = [(x_0 - x_{0t}) + (\hat{\dot{x}}_1 + \hat{\dot{x}}_2 + \dots + \hat{\dot{x}}_n)\delta t)^2 + [(y_0 - y_{0t}) + (\hat{\dot{y}}_1 + \hat{\dot{y}}_2 + \dots + \hat{\dot{y}}_n)\delta t]^2 \longrightarrow (17)$$

$$C_{n2} = -2\left[(\hat{\dot{x}}_1+\hat{\dot{x}}_2+\cdots+\hat{\dot{x}}_n)\right]\left[(x_0-x_{0t})\delta t + (\hat{\dot{x}}_1+\hat{\dot{x}}_2+\cdots+\hat{\dot{x}}_n)\delta t^2\right] + \left[(\hat{\dot{y}}_1+\hat{\dot{y}}_2+\cdots+\hat{\dot{y}}_n)\left[(y_0-y_{0t})\delta t + (\hat{\dot{y}}_1+\hat{\dot{y}}_2+\cdots+\hat{\dot{y}}_n)\delta t^2\right]\right] \rightarrow (18)$$

$$q_{nx} = 2\left([x_0-x_{0t}]\delta t + (\hat{\dot{x}}_1+\hat{\dot{x}}_2+\cdots+\hat{\dot{x}}_n)\delta t^2\right) \rightarrow (19).$$

$$q_{ny} = 2\left[(y_0-y_{0t})\delta t + (\hat{\dot{y}}_1+\hat{\dot{y}}_2+\cdots+\hat{\dot{y}}_n)\delta t^2\right) \rightarrow (20)$$

Then the obstacle avoidance constraint at nth time step is

$$-\left[C_{n1}+C_{n2}+\left[Q_{nx\,(1,n)} \quad Q_{ny\,(1,n)}\right]\begin{bmatrix}\dot{x}_1\\ \vdots \\ \dot{x}_n \\ \dot{y}_1 \\ \vdots \\ \dot{y}_n\end{bmatrix}\right] < R^2 \rightarrow (21)$$

where

$$Q_{nx} = [q_{nx} \cdots q_{nx}]_{(1,n)} \rightarrow (22)$$

$$Q_{ny} = [q_{ny} \cdots q_{ny}]_{(1,n)} \rightarrow (23)$$

Then the overall set of inequality constraints for a single obstacle is

$$\underbrace{-\begin{bmatrix} q_{1x\,(1,1)} & \overbrace{0_2 \cdots\cdots 0_n}^{n-1 \text{ times}} & q_{1y} & \overbrace{0 \cdots\cdots 0}^{n-1 \text{ times.}} \\ q_{2x} & q_{2x} \; \overbrace{0 \cdots 0}^{n-2} & q_{2y} & q_{2y} \; \overbrace{0 \cdots 0}^{n-2} \\ q_{3x} & q_{3x} \; q_{3x} \cdots 0 & q_{3y} & q_{3y} \; q_{3y} \cdots 0 \\ \vdots & & & \\ \vdots & & & \\ q_{nx} & q_{nx} \cdots q_{nx} & q_{ny} & \cdots \; q_{ny} \end{bmatrix}}_{G} \underbrace{\begin{bmatrix}\dot{x}_1\\ \vdots \\ \dot{x}_n \\ \dot{y}_1 \\ \vdots \\ \dot{y}_n\end{bmatrix}}_{x} < \begin{bmatrix} C_{11}+C_{12}+R^2 \\ C_{21}+C_{22}+R^2 \\ \vdots \\ C_{n1}+C_{n2}+R^2 \end{bmatrix} \rightarrow (24)$$

$$\boxed{G \qquad x < h}$$

The matrix G is $n \times 2n$ for a time horizon of n as there are n obstacle avoidance constraint at each of these n time steps.

If there are m obstacles then we get a matrix G of size $nm \times 2n$

Incorporating trust region constraint

$$\left.\begin{aligned} \dot{x}_1 &\le \hat{\dot{x}}_1 + \Delta x \longrightarrow (1) \\ \dot{x}_1 &\geq \hat{\dot{x}}_1 - \Delta x \longrightarrow (2) \end{aligned}\right\} \begin{aligned} \dot{x}_1 &\le \hat{\dot{x}}_1 + \Delta x \longrightarrow (3) \\ -\dot{x}_1 &\le \Delta x - \hat{\dot{x}}_1 \longrightarrow (4) \end{aligned}$$

Hence we get a set of constraints of the form

$$\dot{x}_1 \le \hat{\dot{x}}_1 + \Delta x \longrightarrow (1)$$
$$-\dot{x}_1 \le \Delta x - \hat{\dot{x}}_1 \longrightarrow (2)$$
$$\dot{x}_2 \le \hat{\dot{x}}_2 + \Delta x \longrightarrow (3)$$
$$-\dot{x}_2 \le \Delta x - \hat{\dot{x}}_2 \longrightarrow (4)$$
$$\vdots$$
$$\dot{x}_n \le \hat{\dot{x}}_n + \Delta x \longrightarrow (5)$$
$$-\dot{x}_n \le \Delta x - \hat{\dot{x}}_n \longrightarrow (6)$$

and similarly for $\dot{y}_1 \ldots \dot{y}_n$

Then we get the new trust region matrix G_{trust}, where $G_{trust}\, x \le h_{trust} \longrightarrow (7)$

where $G_{trust} =$

$$
\begin{array}{c}
\overbrace{\hphantom{1 \quad 0 \quad 0 \quad \cdots \quad 0}}^{n} \; \overbrace{\hphantom{0 \quad \cdots \quad 0}}^{n} \\
\begin{bmatrix}
1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
-1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
0 & 1 & 0 & \cdots & 0 & 0 & 0\,0 \cdots & 0 \\
0 & -1 & 0 & \cdots & 0 & 0 & 0\,0 \cdots & 0 \\
\vdots & & & & & & & \\
0 & 0 & \cdots & & 1 & 0 & \cdots & 0 \\
0 & 0 & \cdots & & -1 & 0 & \cdots & 0 \\
0 & 0 & \cdots & & 0 & 1 & \cdots & 0 \\
0 & 0 & \cdots & & 0 & -1 & \cdots & 0 \\
\vdots & & & & & & & \\
0 & \cdots & & & 0 & 0 & \cdots & 1 \\
0 & \cdots & & & 0 & 0 & \cdots & -1
\end{bmatrix}_{(4n \times 2n)}
\end{array}
\begin{bmatrix}
\dot{x}_1 \\ \vdots \\ \dot{x}_n \\ \dot{y}_1 \\ \vdots \\ \dot{y}_n
\end{bmatrix}
\leq
\begin{bmatrix}
\dot{x}_1 + \Delta x \\ \Delta x - \dot{x}_1 \\ \dot{x}_2 + \Delta x \\ \Delta x - \dot{x}_2 \\ \vdots \\ \dot{x}_n + \Delta x \\ \Delta x - \dot{x}_n \\ \dot{y}_1 + \Delta y \\ \Delta y - \dot{y}_1 \\ \vdots \\ \dot{y}_n + \Delta y \\ \Delta y - \dot{y}_n
\end{bmatrix}
$$

We can then stack the trust region matrix or constraint with the G matrix above to get one large matrix $G_e x \leq h_e$.