We take spheroid agents of dimension $\frac{l_{xy}}{2}, \frac{l_z}{2}$ for smoothness of trajectory.

$$\min_{x_i(t),\, y_i(t),\, z_i(t)} \sum_i \sum_T \ddot{x}_i^2(t) + \ddot{y}_i^2(t) + \ddot{z}_i^2(t)$$

→ we are minimising norm of the accln of each agent

such that $x_i(t), y_i(t), z_i(t) \in \mathcal{C}_{boundary}$ → (1)

i.e. $x_i(0), x_i(t_f)$ are known (initial & final positions of each agent)

Collision avoidance constraints:

$$f(x_i(t), y_i(t), z_i(t)) = -\frac{(x_i(t) - x_j(t))^2}{l_{xy}^2} - \frac{(y_i(t) - y_j(t))^2}{l_{xy}^2} - \frac{(z_i(t) - z_j(t))^2}{l_z^2} + 1$$

(This comes from linearising the constraint

$$(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 \geq \left(2\left(\frac{l_{xy}}{2}\right)\right)^2 \;\&\; \left(\frac{l_z}{2}\right)\big)^2$$

$$\cancel{= \arg\min_{\varepsilon} \frac{1}{2} \varepsilon^T (Q + \rho F^T F)\varepsilon + \rho F^T \left(g + s + \frac{\lambda}{\rho}\right)\varepsilon}$$

Converting the quadratic collision avoidance constraints to polar representation, we can write

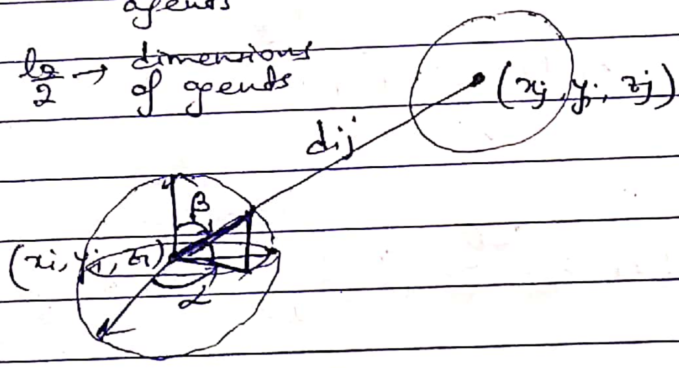
$f = 0$

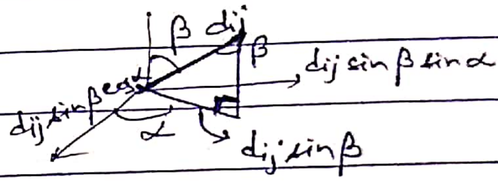
where

$$f = \begin{Bmatrix} x_i(t) - x_j(t) - l_{xy}\, d_{ij}(t) \sin\beta_{ij}(t) \cos\alpha_{ij}(t) \\ y_i(t) - y_j(t) - l_{xy}\, d_{ij}(t) \sin\beta_{ij}(t) \sin\alpha_{ij}(t) \\ z_i(t) - z_j(t) - l_z\, d_{ij}(t) \cos\beta_{ij}(t) \end{Bmatrix} \longrightarrow (2)$$

$d_{ij} \rightarrow$ ~~dist b/w~~ centres of 2 agents

$\frac{l_{xy}}{2}, \frac{l_z}{2} \rightarrow$ dimensions of agents





Using ADMM, we can write a given minimisation problem $\min_{\varepsilon} \frac{1}{2}\varepsilon^T Q \varepsilon$ subject to $f(\varepsilon) + s = 0 \quad s \geq 0$

as $\min_{\varepsilon} \frac{1}{2}\varepsilon^T Q \varepsilon + \frac{\rho}{2}\left\| f(\varepsilon) + s + \frac{\lambda}{\rho} \right\|_2^2$

from (1) & (2), we write augmented cost function.

$$\mathcal{L} = \sum_i \ddot{x}_i^2(t) + \ddot{y}_i^2(t) + \ddot{z}_i^2(t) + \sum_{i,j} \frac{\rho}{2}\Big(x_i(t) - x_j(t)$$

$$- l_{xy}\, d_{ij}(t) \sin\beta_{ij}(t) \cos\alpha_{ij}(t) + \frac{\lambda_{x_{ij}}}{\rho}\Big)^2 + \cdots \frac{\lambda_{z_{ij}}}{\rho}\Big)^2$$

Using Bernstein Polynomial for the trajectories of the agents, we can write

If n_samples be the # time samples we consider.

$$\begin{bmatrix} x_i(t_1) \\ x_i(t_2) \\ \vdots \\ x_i(t_n) \end{bmatrix} = P c_x \qquad \begin{bmatrix} \dot{x}_i(t_1) \\ \dot{x}_i(t_2) \\ \vdots \\ \dot{x}_i(t_n) \end{bmatrix} = \dot{P} c_x \qquad \begin{bmatrix} \ddot{x}_i(t_1) \\ \ddot{x}_i(t_2) \\ \vdots \\ \ddot{x}_i(t_n) \end{bmatrix} = \ddot{P} c_x$$

$$P = \begin{bmatrix} B_0(t) & B_1(t) & \cdots & B_n(t) \end{bmatrix}_{n\_samples \times n} \qquad c_x = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix}_{n \times 1}$$

where $B_i(t) = \begin{bmatrix} B_i(0) \\ \vdots \\ B_i(t_f) \end{bmatrix}$

$$\dot{P} = \begin{bmatrix} \dot{B}_0(t) & \dot{B}_1(t) & \cdots & \dot{B}_n(t) \end{bmatrix}$$

$$\ddot{P} = \begin{bmatrix} \ddot{B}_0(t) & \ddot{B}_1(t) & \cdots & \ddot{B}_n(t) \end{bmatrix}_{n\_samples \times n}$$

Geometric structure

Now $\sum_i \ddot{x}_i^2(t) = \frac{1}{2} c_x^T \varphi c_x$ $\qquad \varphi = \begin{bmatrix} \ddot{P}^T\ddot{P} & \\ & \ddots \\ & & \ddot{P}^T\ddot{P} \end{bmatrix}$

Similarly for $y_i(t)$, $z_i(t)$ - initial & final constraints

Now $x_i(t) \in$ Boundary $\Rightarrow A_{eq} c_x = b_{eq}^x$

$\cancel{b_{eq}^x = \begin{bmatrix} x_1(0) \\ x_1(t_f) \\ x_2(0) \\ x_2(t_f) \\ \vdots \end{bmatrix}}$

$$b_{eq}^x = \begin{bmatrix} x_1(0) \\ x_2(0) \\ \vdots \\ x_k(0) \\ v_1(0) \\ v_2(0) \\ \vdots \\ v_k(0) \\ a_1(0) \\ \vdots \\ a_k(0) \\ x_1(t_f) \\ \vdots \\ x_n(t_f) \\ v_1(t_f) \\ \vdots \\ v_n(t_f) \\ a_1(t_f) \\ \vdots \\ a_n(t_f) \end{bmatrix}_{6k \times 1} \begin{matrix} x(0) \\ v(0) \\ a(0) \\ x(t_f) \\ v(t_f) \\ a(t_f) \end{matrix}$$

k = # robots

initial & final position, velocity & accel- constraints

$$A_{eq} = \begin{bmatrix} A & & \\ & \ddots & \\ & & A \end{bmatrix} = \mathrm{kron}(eye, A)$$

$$A = \begin{bmatrix} P[0] \\ \dot{P}[0] \\ \ddot{P}[0] \\ P[n] \\ \dot{P}[n] \\ \ddot{P}[n] \end{bmatrix}_{6k \times n}$$

$$\sum_{i,j} \frac{\rho}{2}\left(x_i(t) - x_j(t) - l_{xy}\, d_{ij}^k(t) \sin\beta_{ij}^k(t) \cos\alpha_{ij}^k(t) + \frac{\lambda_{x_{ij}}^k}{\rho}\right)^2$$

$$= \frac{\rho}{2} \left\| A_{fc}\, c_x - b_{fc}^{k^x} \right\|_2^2 \quad \text{where}$$

$$A_{fc} = \begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_r \end{bmatrix}$$

$$A_i = \left[\begin{pmatrix} P \\ P \\ P \\ \vdots \\ P \end{pmatrix}_{n-i} \begin{bmatrix} -P & & \\ & \ddots & \\ & & -P \end{bmatrix}_{n-i} \right]$$

$$b_{fc}^{k^x} = l_{xy}\, d_{ij}^k \sin\beta^k \cos\alpha^k - \frac{\lambda_{x_{ij}}^k}{\rho}$$

this depend on k where ~~k is iteration~~ index

$\boxed{\text{Dimensions}}$: $P, \dot{P}, \ddot{P} \rightarrow$ n samples $\times$ n

n = order of Bernstein poly

$c_x \rightarrow n \times 1$

p = # robots

$\underbrace{A_{eq}}_{6p \times n}\, c_x = \underbrace{b_{ep}^x}_{6p \times 1}$

We have finally

$$\min_{c_x} \underbrace{\frac{1}{2} c_x^T \left(Q_x + \rho A_{fc}^T A_{fc}\right) c_x - \left(\rho A_{fc}^T b_{fc}^x\right)^T c_x}_{f(c_x)}$$

subject to. $A_{eq}\, c_x = b_{ep}^x$. i.e $\underbrace{A_{ep}\, c_x - b_{ep}^x}_{h(c_x)} = 0$

From stationarity condition of KKT constraints

$$\cancel{f(\mathcal{Q}) + \mu(A_{eq} c_x - b_{eq}^x)}$$

$$\cancel{f'(c_x) + \mu h'(c_x) = 0}$$

i.e. $\cancel{(\mathcal{Q}_x + \rho A_{fe}^T A_{fe}) c_x + \mu A_{eq}^T}$

or $\cancel{(\mathcal{Q}_x + \rho A_{fe}^T A_{fe}) \quad \mathcal{Q}_x c_x + A_{eq}^T \mu}$

From stationarity condition of KKT constraints,

μ = KKT multiplier vector

$$f'(c_x) + \mu h'(c_x) = 0$$

i.e. $(\mathcal{Q}_x + \rho A_{fe}^T A_{fe}) c_x + A_{eq}^T \mu = \rho A_{fe}^T b_{fe}^x \longrightarrow (3)$

$\cancel{\text{i.e. } \mathcal{Q}_x c_x + A_{eq}^T \mu =}$

Also, $A_{eq} c_x = b_{eq}$ (from primal feasibility condition) $\longrightarrow (4)$

From (3) & (4) we can write,

$$\underbrace{\begin{bmatrix} (\mathcal{Q}_x + \rho A_{fe}^T A_{fe}) & A_{eq}^T \\ A_{eq} & 0 \end{bmatrix}}_{\mathcal{Q}} \begin{bmatrix} c_x \\ \mu \end{bmatrix} = \begin{bmatrix} \rho A_{fe}^T b_{fe}^{k\,x} \\ b_{eq}^x \end{bmatrix}$$

As only b_{fe}^x depends on k LHS is independent of k and can be precomputed & cached.

$$S = \begin{bmatrix} c_x \\ \mu \end{bmatrix}_x = \mathcal{Q}^{-1} \begin{bmatrix} \rho A_{fe}^T b_{fe}^{k\,x} \\ b_{eq}^x \end{bmatrix}$$

$$c_x = S\,[0 : n\_robot \times n]$$

n = ~~degree~~ order of Bernstein poly.

Similarly for c_y, c_z.

for each iter, updates to $d_{ij}, \alpha_{ij}, \beta_{ij}, \lambda_{ij}^x, \lambda_{ij}^y, \lambda_{ij}^z$ as:

$${}^{k+1}\alpha_{ij} = \tan^{-1} \frac{({}^{k+1}y_i - {}^{k+1}y_j)}{({}^{k+1}x_i - {}^{k+1}x_j)}$$

(approximation from polar coordinate diagram)

$${}^{k+1}\beta_{ij} = \tan^{-1} \left(\frac{{}^{k+1}x_i - {}^{k+1}x_j}{l_{xy} \cos {}^{k+1}\alpha_{ij}} \times \frac{l_z}{{}^{k+1}z_i - {}^{k+1}z_j} \right)$$

$${}^{k+1}\lambda_{ij}^x = {}^{k}\lambda_{ij}^x + \rho (res_x)$$

$$res_x = {}^{k+1}x_i - {}^{k+1}x_j - l_{xy}\, {}^{k+1}d_{ij} \sin {}^{k+1}\beta_{ij} \cos {}^{k+1}\alpha_{ij}$$

similarly for ${}^{k+1}\lambda_{ij}^y$ & ${}^{k+1}\lambda_{ij}^z$ (with respective residuals)

$$\cancel{{}^{k+1}d_{ij} = \arg\min_{d_{ij}} \sum_{i,j} \frac{\rho}{2}\, {}^{k+1}x_i - {}^{k+1}x_j - l_{xy}\, d_{ij} \sin {}^{k+1}\beta_{ij} \cos {}^{k+1}\alpha_{ij}}$$

$${}^{k+1}d_{ij} = \cancel{\arg\min_{d_{ij}} \sum_{i,j} \frac{\rho}{2} (res_x^2 + res_y^2 + res_z^2)}$$

$${}^{k+1}d_{ij} = \arg\min_{d_{ij}} \sum_{i,j} \frac{\rho}{2} \left[\left(res_x + \frac{{}^{k}\lambda_{x_{ij}}}{\rho}\right)^2 + \left(res_y + \frac{{}^{k}\lambda_{y_{ij}}}{\rho}\right)^2 + \left(res_z + \frac{{}^{k}\lambda_{z_{ij}}}{\rho}\right)^2 \right]$$