Continuous-Time Line-of-Sight Constrained Trajectory Planning for 6-Degree of Freedom Systems

Christopher R. Hayner[1], John M. Carson III[2], Behçet Açıkmeşe[1], Karen Leung[1]

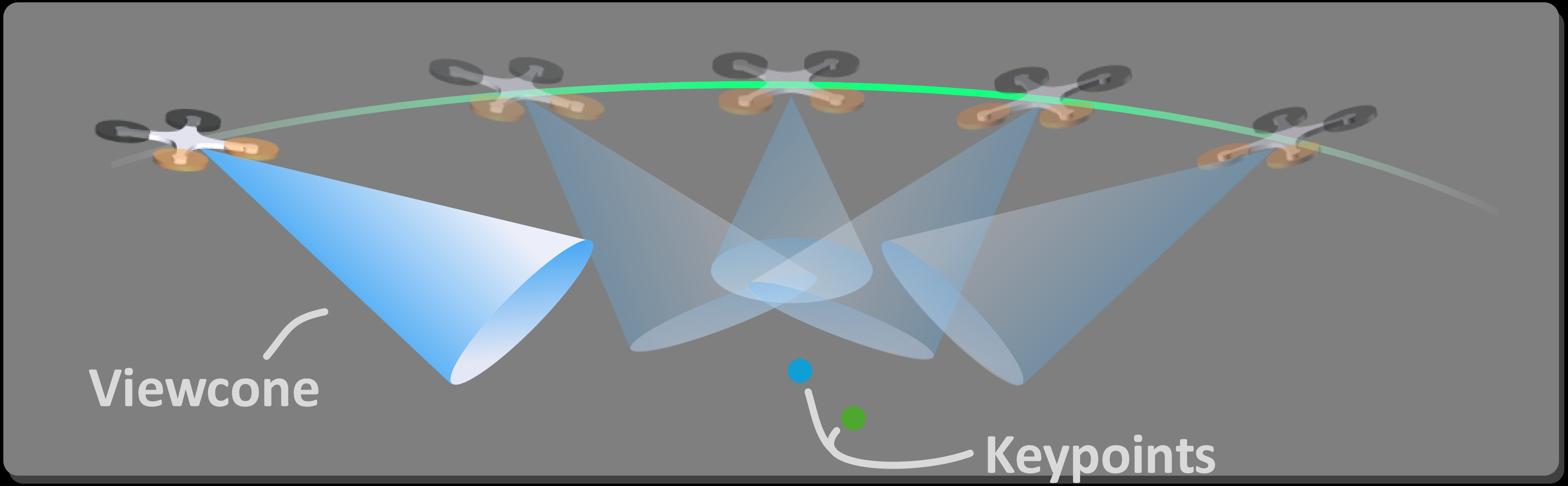
[1]William E. Boeing Department of Aeronautics & Astronautics, University of Washington, Seattle WA

[2]NASA Johnson Space Center, Houston TX

IEEE Robotics and Automation Letters (RA-L) 2025 (Accepted)

We wish to keep a set of keypoints within the Line-of-Sight (LoS) of the robot **throughout** its movement

We consider this problem as ***LoS Guidance***



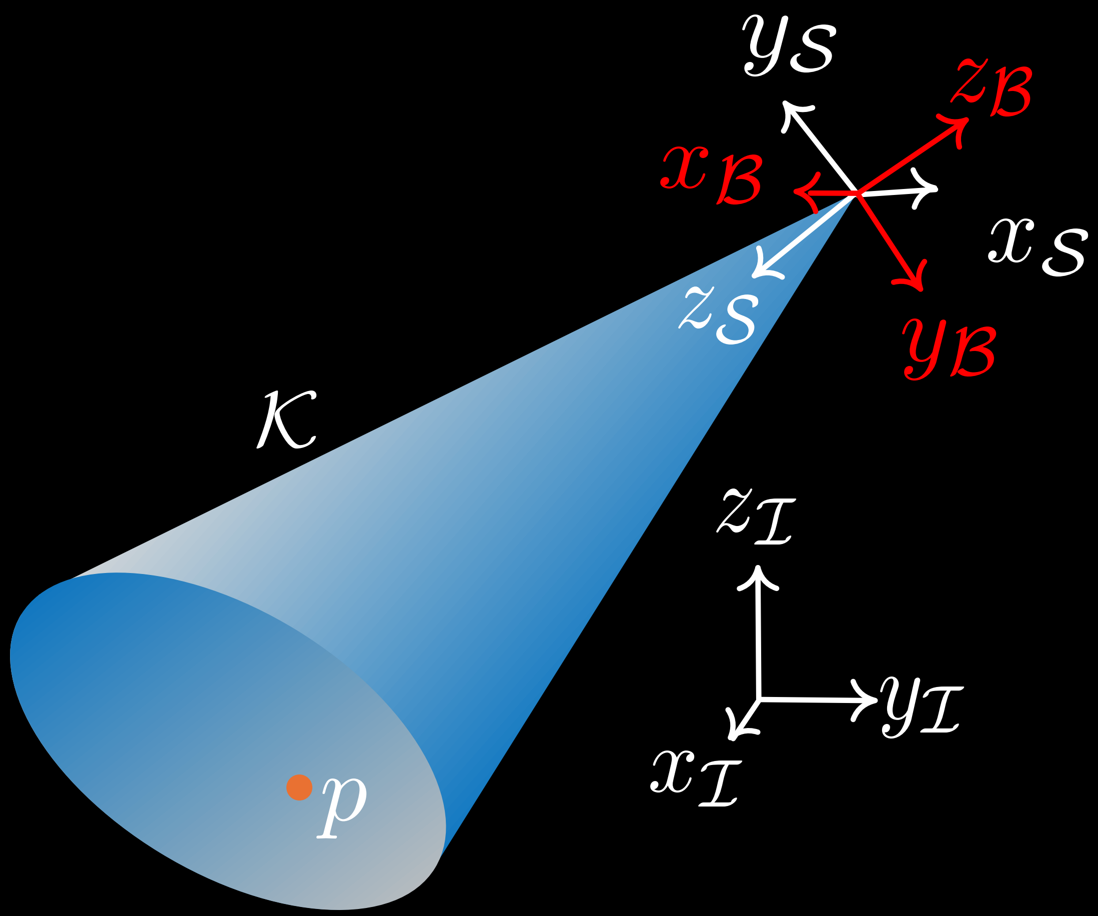
Nonlinear Program

$$
\begin{aligned}
\underset{x,u,t_f}{\text{minimize}} \quad & L_f(t_f, x(t_f), u(t_f)), && \text{Terminal Cost} \\
\text{subject to} \quad & \dot{x}(t) = f(t, x(t), u(t)), && \text{Dynamics} \\
& p(t) \in \mathcal{K}(b(x(t)), c(x(t)), A^{\mathrm{C}}(t)), && \text{Line-of-Sight Constraint} \\
& g(t, x(t), u(t)) \leq 0_{n_g}, && \text{General Inequality Constraints} \\
& h(t, x(t), u(t)) = 0_{n_h}, && \text{General Equality Constraints} \\
& x(t_i) = x_i, x(t_f) = x_f, && \text{Boundary Constraints}
\end{aligned}
$$

Line-of-Sight Constraint

Notation Introduction

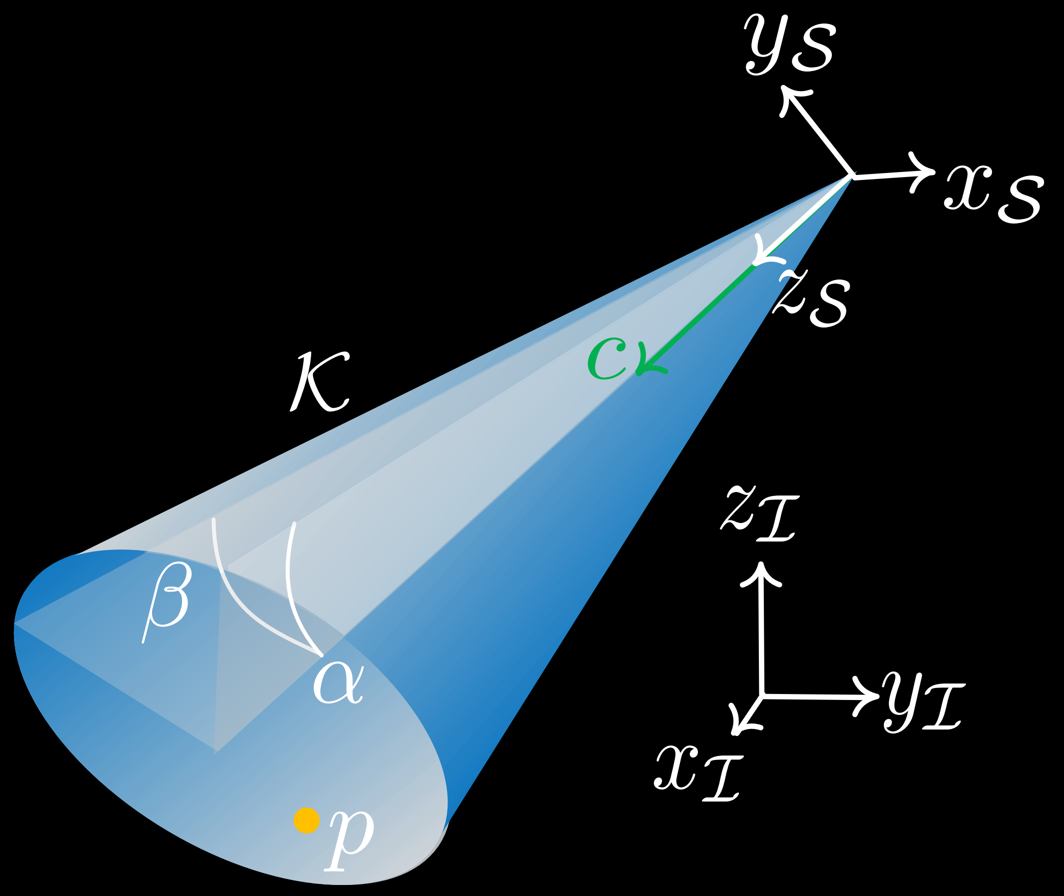
$\square_{\mathcal{I}}$ $\square_{\mathcal{B}}$ $\square_{\mathcal{S}}$



Line-of-Sight Constraint

Norm Cone Component

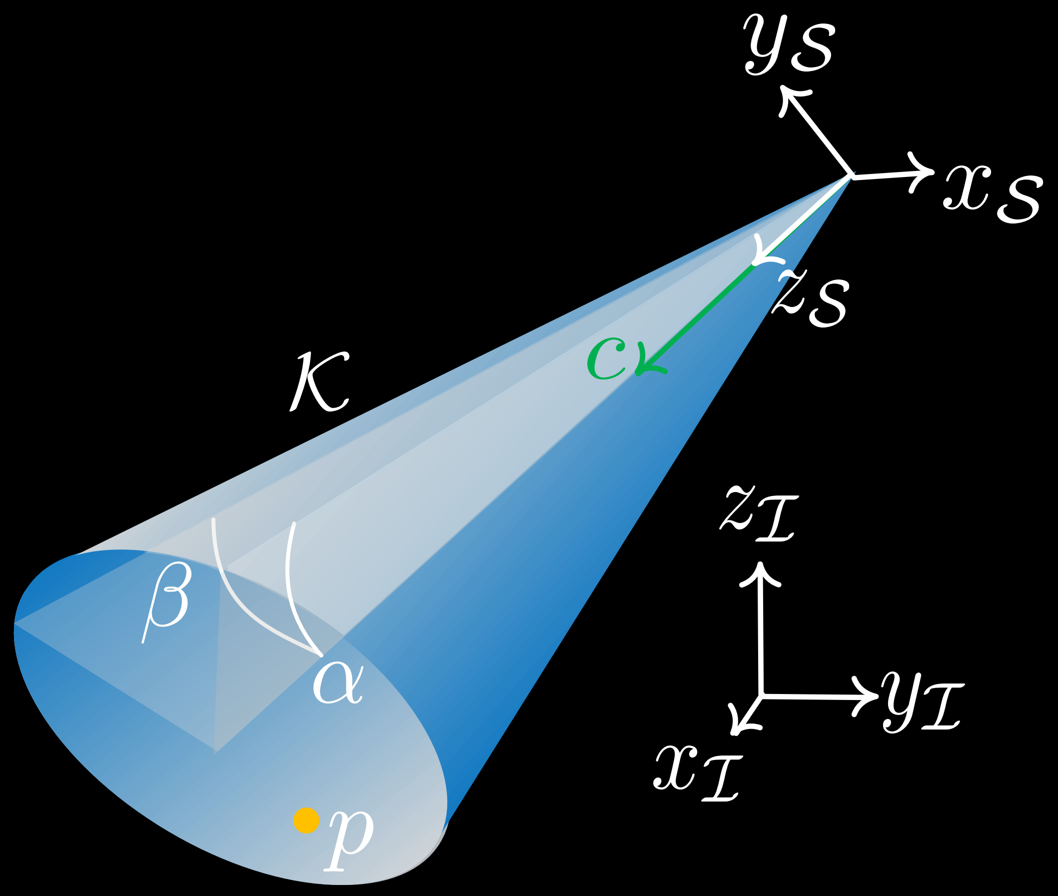
$$\|A^{\mathrm{C}} p_{\mathcal{S}}(t)\|_\rho \leq c^\top p_{\mathcal{S}}(t)$$



Line-of-Sight Constraint

Norm Cone Component

$$\underbrace{\|A^{\mathrm{C}} p_{\mathcal{S}}(t)\|_{\rho} \leq c^{\top} p_{\mathcal{S}}(t)}_{\text{Convex in } p_{\mathcal{S}}(t)}$$



Line-of-Sight Constraint

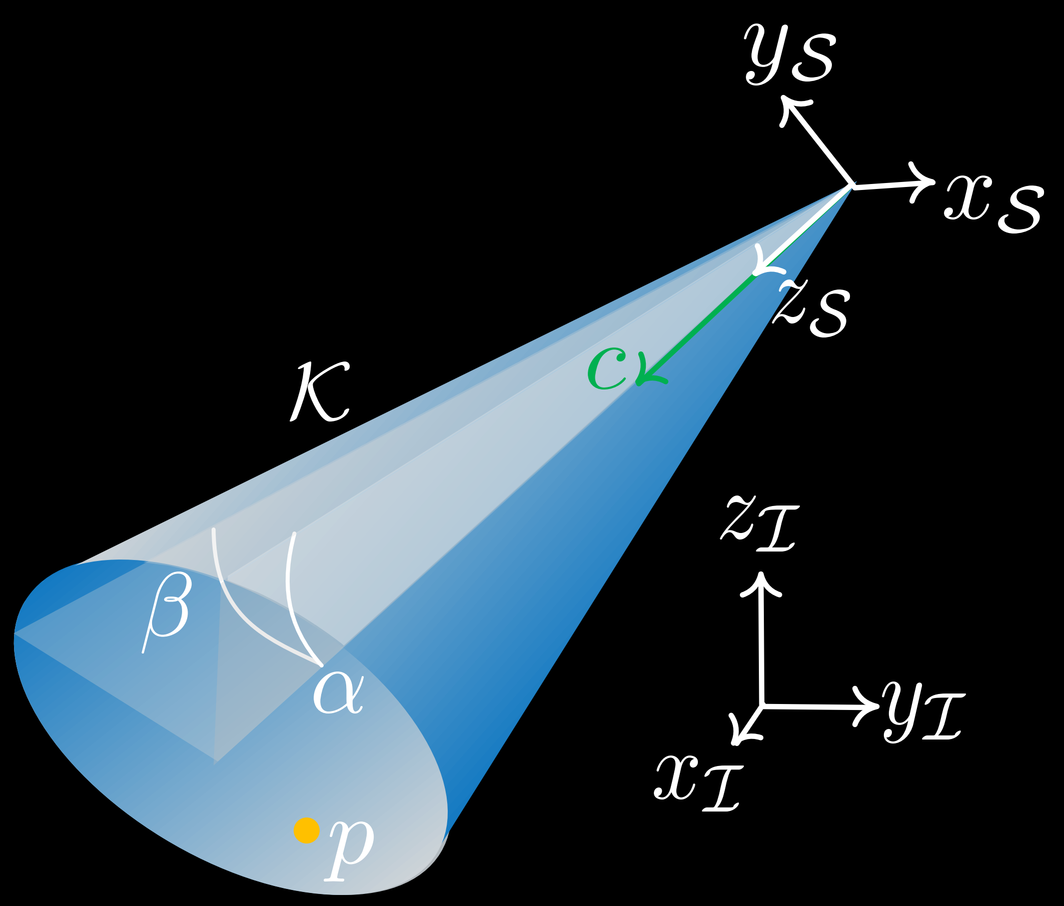
Norm Cone Component

$$\|A^{\mathrm{C}} p_{\mathcal{S}}(t)\|_\rho \leq c^\top p_{\mathcal{S}}(t)$$

Notation Introduction

$$A^{\mathrm{C}} \triangleq \begin{bmatrix} \frac{1}{\tan(\alpha)} & 0 & 0 \\ 0 & \frac{1}{\tan(\beta)} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$c = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top$$

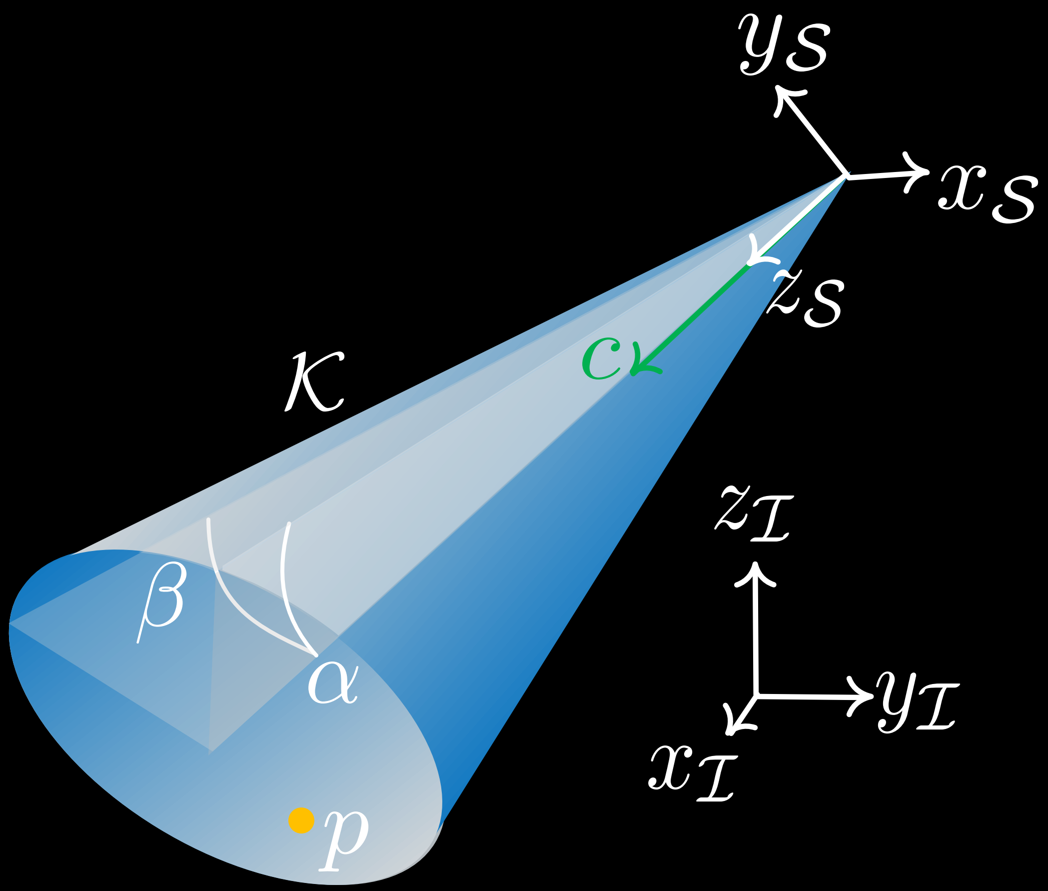


Line-of-Sight Constraint

Norm Cone Component

$$\|A^{\mathrm{C}} p_{\mathcal{S}}(t)\|_{\rho} \leq c^{\top} p_{\mathcal{S}}(t)$$

Note: By choosing different values for ρ, α, β the LoS constraint can be tailored specific sensor types.

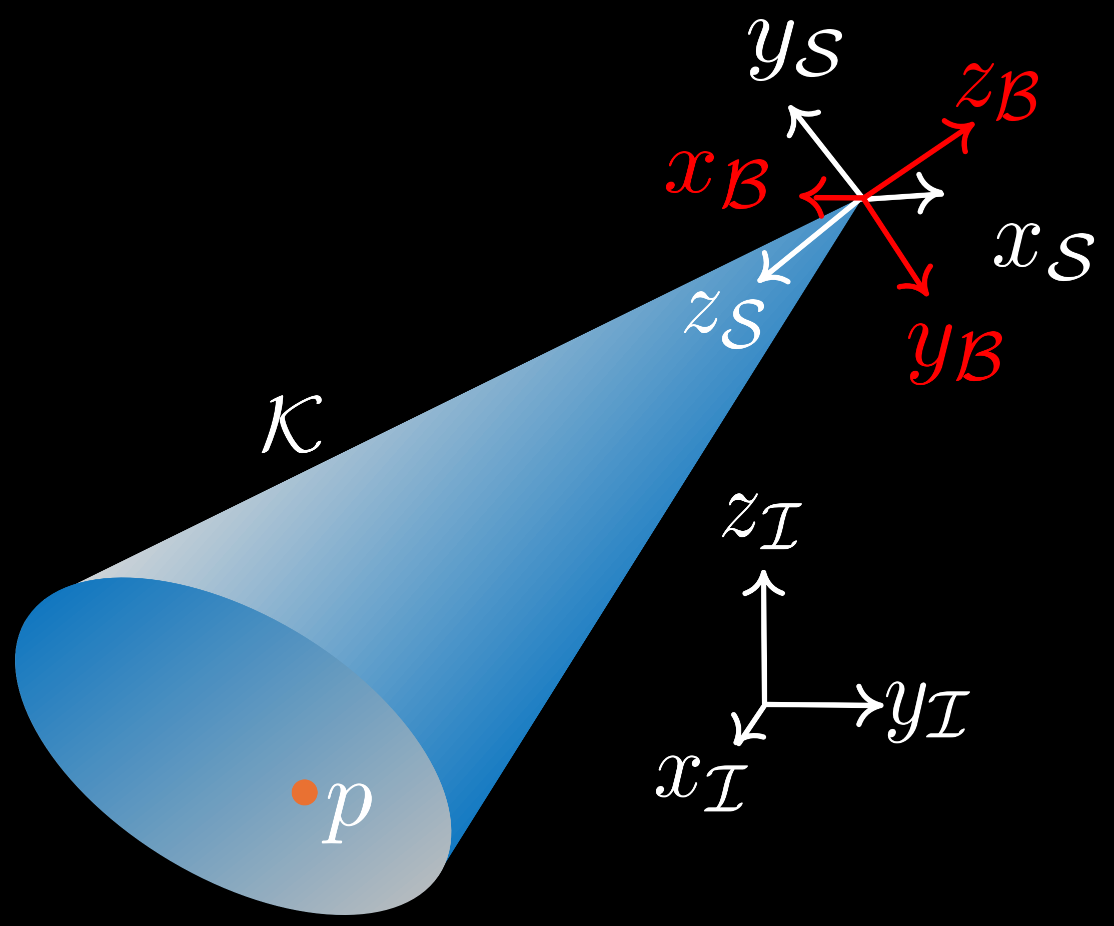


Line-of-Sight Constraint

Notation Introduction

$$q_{\mathcal{S}\to\mathcal{B}}$$

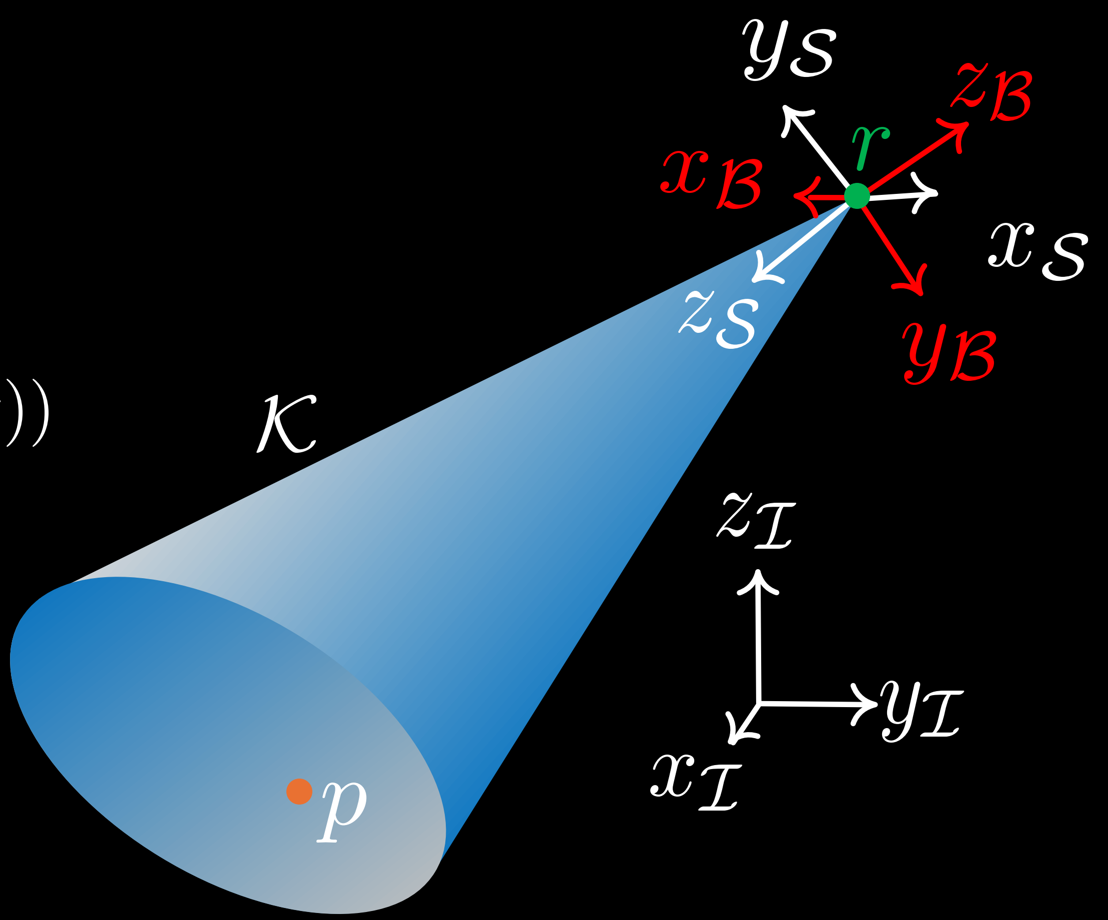
$$C : \mathcal{S}^3 \mapsto SO(3)$$



Line-of-Sight Constraint

Transformation Component

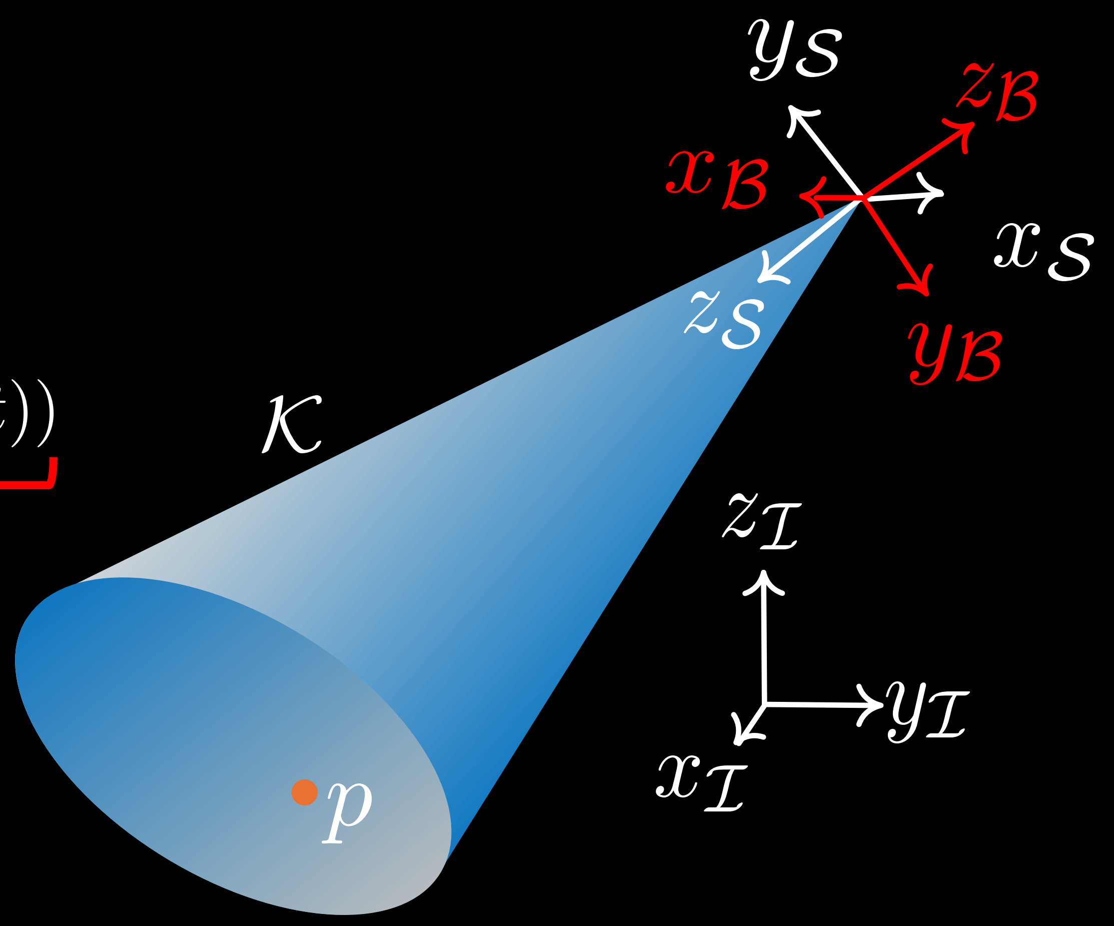
$$p_{\mathcal{S}}(t) = C(q_{\mathcal{S}\to\mathcal{B}})C(q_{\mathcal{B}\to\mathcal{I}}(t))(p_{\mathcal{I}}(t) - r_{\mathcal{I}}(t))$$



Line-of-Sight Constraint

Transformation Component

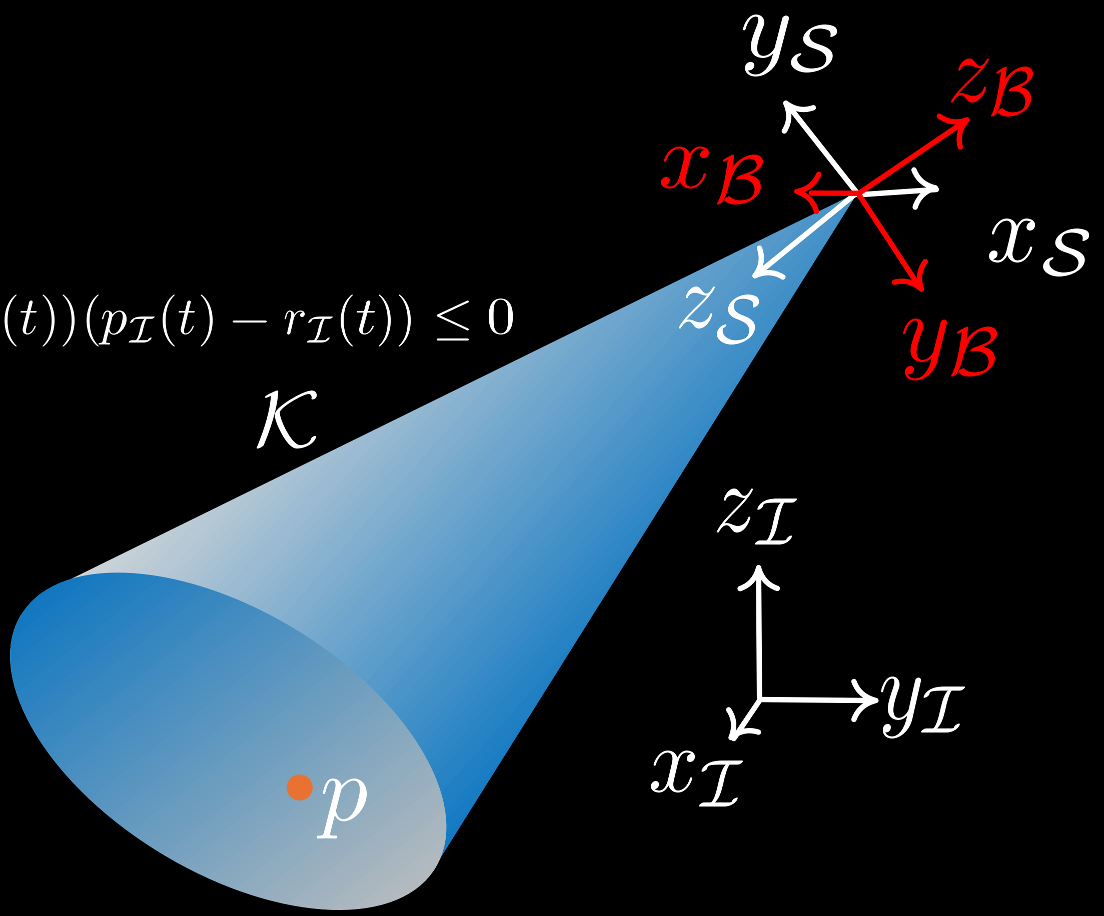
$$p_{\mathcal{S}}(t) = C(q_{\mathcal{S}\to\mathcal{B}})\underbrace{C(q_{\mathcal{B}\to\mathcal{I}}(t))(p_{\mathcal{I}}(t) - r_{\mathcal{I}}(t))}_{\text{Nonconvex in } q_{\mathcal{B}\to\mathcal{I}},\, r_{\mathcal{I}}}$$



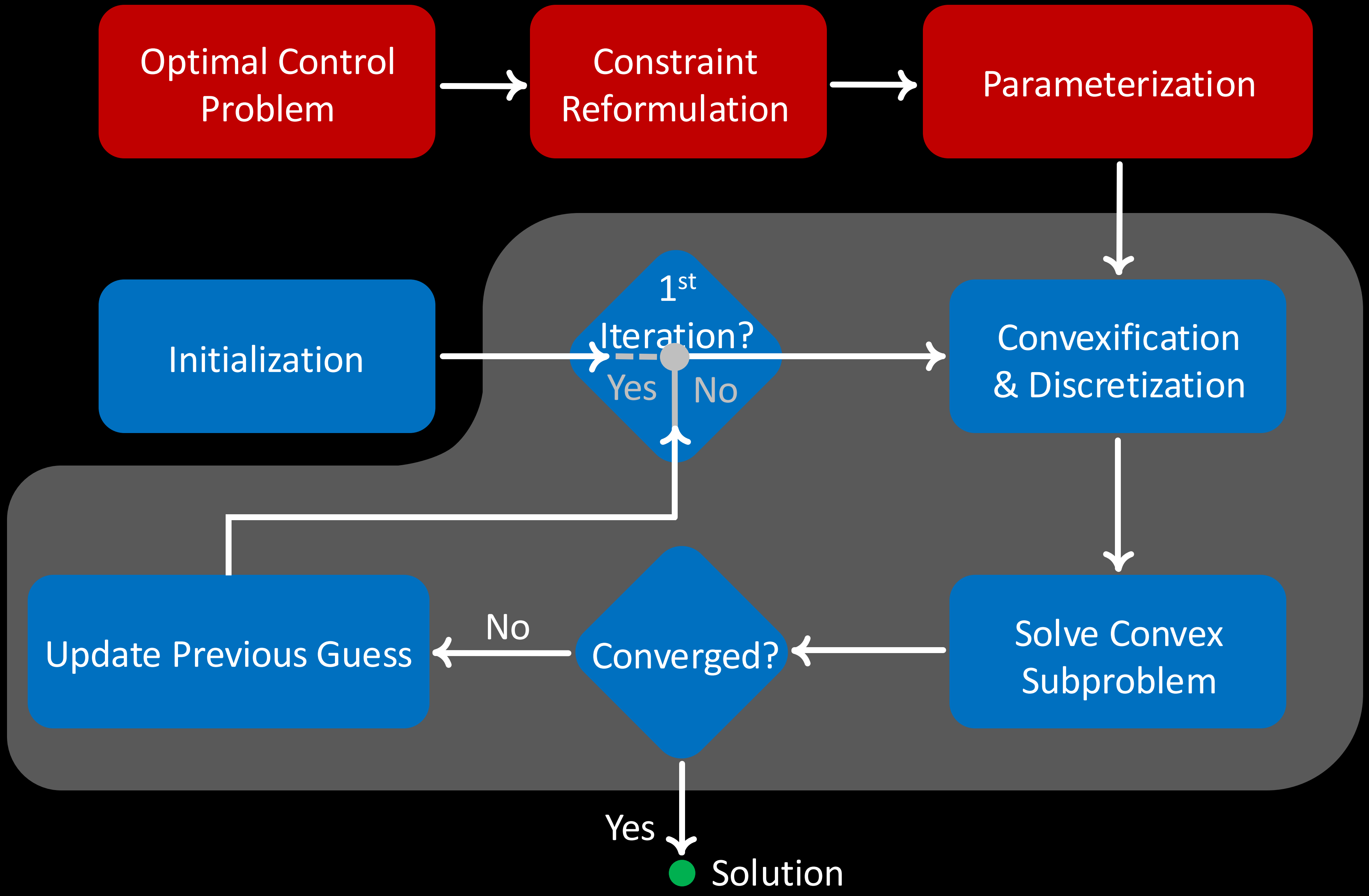
Line-of-Sight Constraint

Full Constraint

$$g_{\mathrm{LoS}} \triangleq \|A^C C(q_{\mathcal{S}\to\mathcal{B}})C(q_{\mathcal{B}\to\mathcal{I}}(t))(p_{\mathcal{I}}(t) - r_{\mathcal{I}}(t))\| \\ - c^\top C(q_{\mathcal{S}\to\mathcal{B}})C(q_{\mathcal{B}\to\mathcal{I}}(t))(p_{\mathcal{I}}(t) - r_{\mathcal{I}}(t)) \leq 0$$

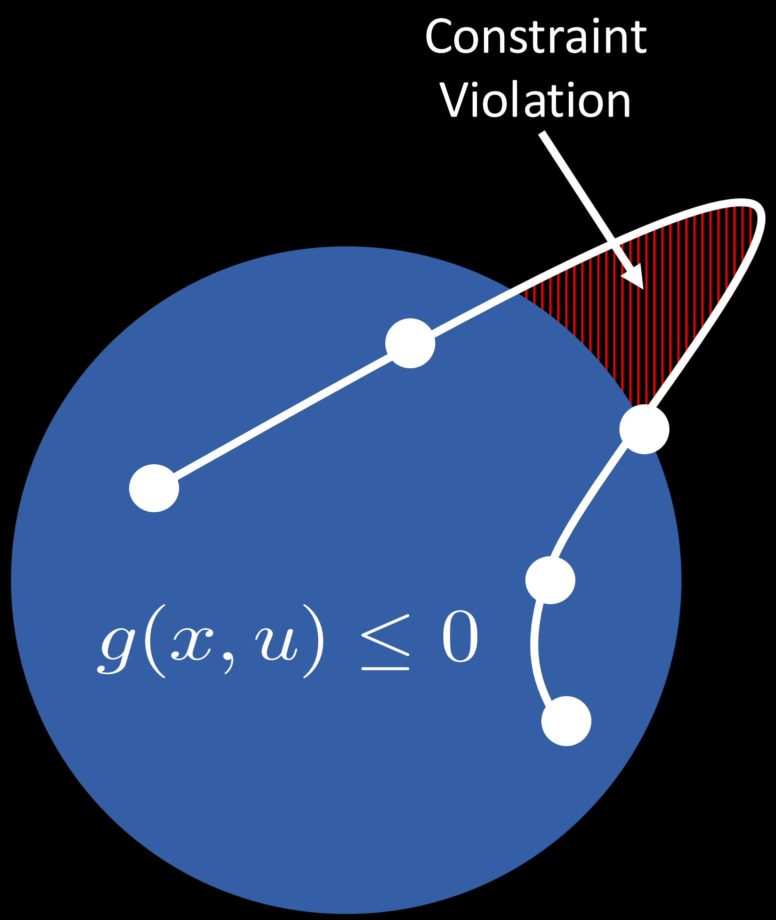


Continuous-Time Successive Convexification CT-SCvx



Constraint Reformulation

$$\left.\begin{aligned} g(t, x(t), u(t)) &\leq 0_{n_g} \\ h(t, x(t), u(t)) &= 0_{n_h} \end{aligned}\right. \forall t \in [0, t_f]$$



Elango et al. 2024

Constraint Reformulation

$$\begin{aligned} g(t, x(t), u(t)) &\leq 0_{n_g} \\ h(t, x(t), u(t)) &= 0_{n_h} \end{aligned} \quad \forall t \in [0, t_f] \iff \int_0^{t_f} \max\{0, g(x(t), u(t))\}^2 + h(x(t), u(t))^2 dt = 0$$

Constraint Reformulation

$$\dot{y}(t) = \max\{0, g(x(t), u(t))\}^2 + h(x(t), u(t))^2$$

$$y(0) = y(t_f)$$

Constraint Reformulation

$$\tilde{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\frac{d\tilde{x}}{dt} = \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = \begin{bmatrix} f(x,u) \\ \max\{0, g(x,u)\}^2 + h(x,u)^2 \end{bmatrix}$$

System Dynamics

We use a 6-Degree of Freedom rigid body model

Position.	$\dot{r}_{\mathcal{I}}(t) = v_{\mathcal{I}}(t),$
Velocity.	$\dot{v}_{\mathcal{I}}(t) = \dfrac{1}{m}\left(C_{\mathcal{B}\to\mathcal{I}}(q_{\mathcal{B}\to\mathcal{I}}(t))T_{\mathcal{B}}(t)\right) + g_{\mathcal{I}},$
Attitude.	$\dot{q}_{\mathcal{I}\to\mathcal{B}} = \dfrac{1}{2}\Omega(\omega_{\mathcal{B}}(t))q_{\mathcal{I}\to\mathcal{B}}(t),$
Angular Velocity.	$\dot{\omega}_{\mathcal{B}}(t) = J_{\mathcal{B}}^{-1}\left(M_{\mathcal{B}}(t) - \left[\omega_{\mathcal{B}}(t)\times\right]J_{\mathcal{B}}\omega_{\mathcal{B}}(t)\right),$

As a baseline approach we solved the same problem but only applying constraints at individual nodes

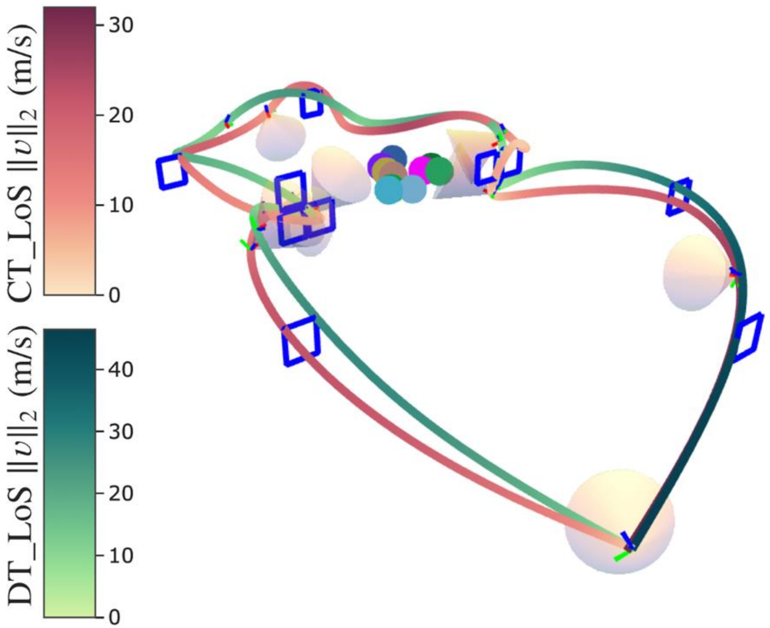
CT-LoS (Proposed)

$$\begin{aligned}
\underset{\hat{\tilde{x}},\hat{u}}{\text{minimize}} \quad & \lambda_{\text{obj}} L_f(\tilde{x}_N, u_N) + \\
& \sum_{k=0}^{N} \lambda_{\text{tr}} \left\| \begin{bmatrix} \hat{\tilde{x}}_k \\ \hat{u}_k \end{bmatrix} - \begin{bmatrix} \bar{\hat{\tilde{x}}}_k \\ \bar{\hat{u}}_k \end{bmatrix} \right\|_2^2 + \lambda_{\text{vc}} \|\nu_k\|_1 \\
\text{subject to} \quad & \Delta\tilde{x}_k = \bar{A}_{k-1}\Delta\tilde{x}_{k-1} + \bar{B}^-_{k-1}\Delta u_{k-1} \\
& \quad + \bar{B}^+_{k-1}\Delta u_k + \nu_{k-1}, \\
& y_k - y_{k-1} \leq \varepsilon_{\text{LICQ}}, \\
& \tilde{x}_0 = \tilde{x}_i, \tilde{x}_N = \tilde{x}_f,
\end{aligned}$$

DT-LoS (Baseline)

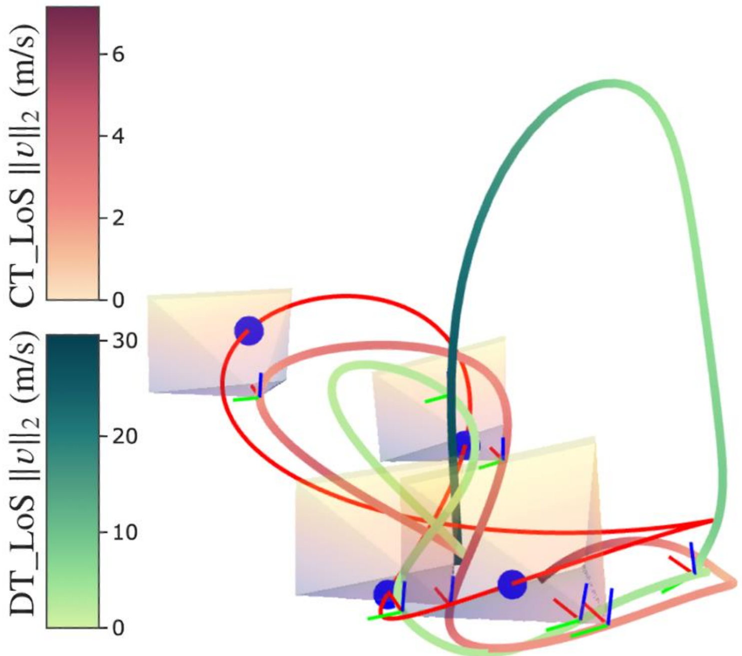
$$\begin{aligned}
\underset{\hat{x},\hat{u}}{\text{minimize}} \quad & \lambda_{\text{obj}} L_f(x_N, u_N) + \lambda_{\text{vc}} \|\nu_k\|_1 + \\
& + \lambda_{\text{vb}} \max\{0, \nu_{g_{\text{LoS}}}\} \\
& + \sum_{k=0}^{N} \lambda_{\text{tr}} \left\| \begin{bmatrix} \hat{x}_k \\ \hat{u}_k \end{bmatrix} - \begin{bmatrix} \bar{\hat{x}}_k \\ \bar{\hat{u}}_k \end{bmatrix} \right\|_2^2 \\
\text{subject to} \quad & \Delta x_k = \bar{A}_{k-1}\Delta x_{k-1} + \bar{B}^-_{k-1}\Delta u_{k-1} \\
& \quad + \bar{B}^+_{k-1}\Delta u_k + \nu_{k-1}, \\
& L^{\ell}_{g_{\text{LoS}}}(x_k, u_k) = \nu^{\ell}_{g_{\text{LoS}}}, \\
& x^i_{\text{min}} \leq x^i_k \leq x^i_{\text{max}}, \\
& u^j_{\text{min}} \leq u^j_{k-1} \leq u^j_{\text{max}}, \\
& x_0 = x_i, \ x_N = x_f,
\end{aligned}$$

We demonstrated both the proposed and baseline in two challenging and representative scenarios.



Relative Navigation Scenario

- 10 gates in a predefined sequence
- 10 static keypoints to keep within LoS
- Minimal time
- $\rho = 2, \alpha = \beta = \pi/4$



Tracking Scenario

- 1 dynamic keypoint to keep within LoS
- Minimal Fuel
- $\rho = \infty, \alpha = \pi/6, \beta = \pi/8$

We sought to address the following questions in our experiments

Q1. How well does CT-LoS satisfy the LoS constraint throughout the entire trajectory compared to the baseline?

Q2. What tradeoffs are made to achieve better LoS violation performance?

Q3. How does CT-LoS scale as the problem size increase?

We sought to address the following questions in our experiments

Q1. How well does CT-LoS satisfy the LoS constraint throughout the entire trajectory compared to the baseline?

Q2. What tradeoffs are made to achieve better LoS violation performance?

Q3. How does CT-LoS scale as the problem size increase?

We used the following metric to address the above questions

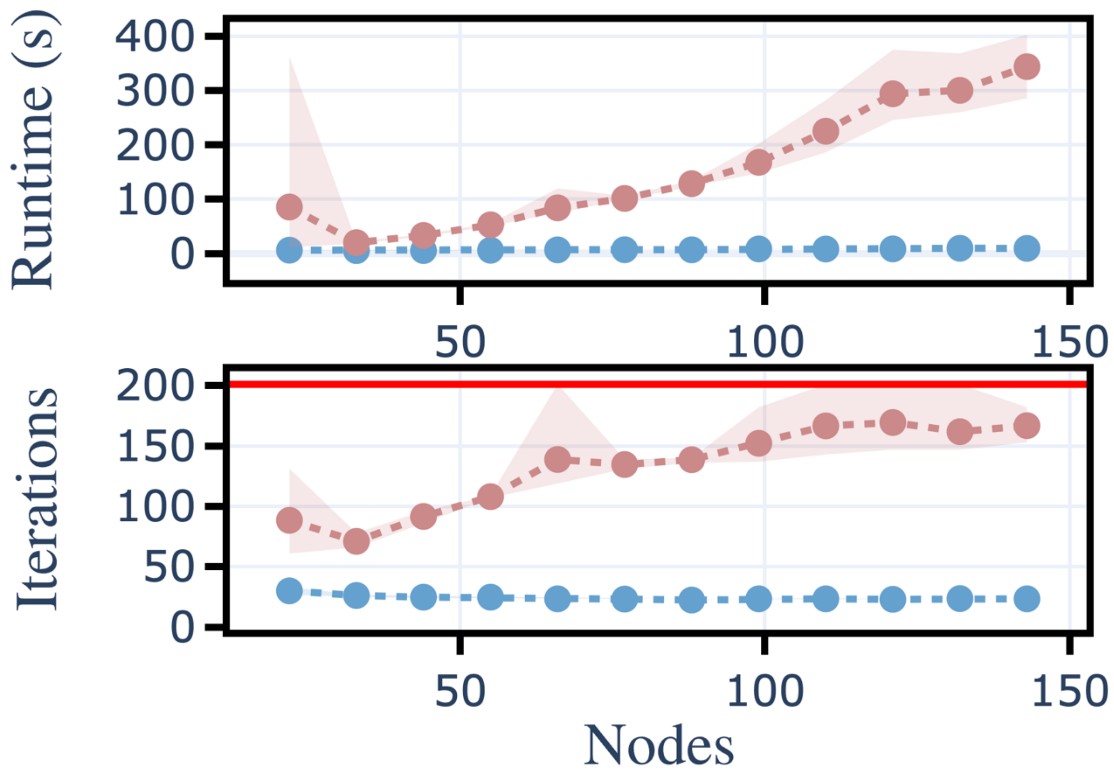
1. LoS Constraint Violation over the full trajectory

2. The total runtime

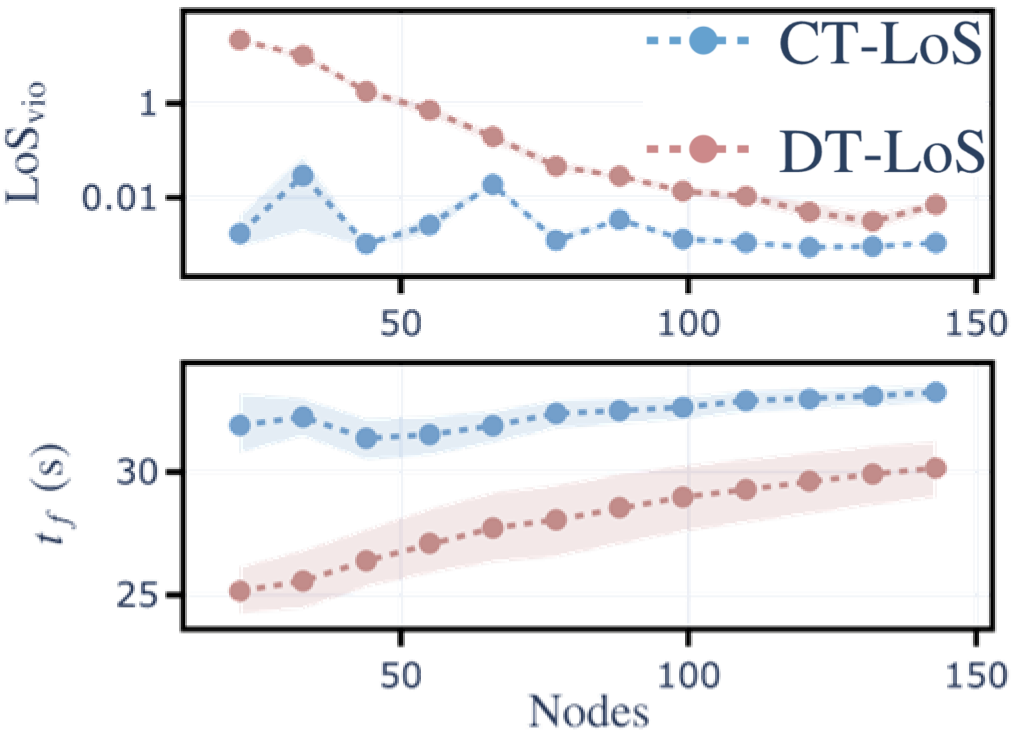
3. Original Object Cost (Minimal Time or Minimal Fuel)

4. The number of iterations

Relative Navigation Scenario

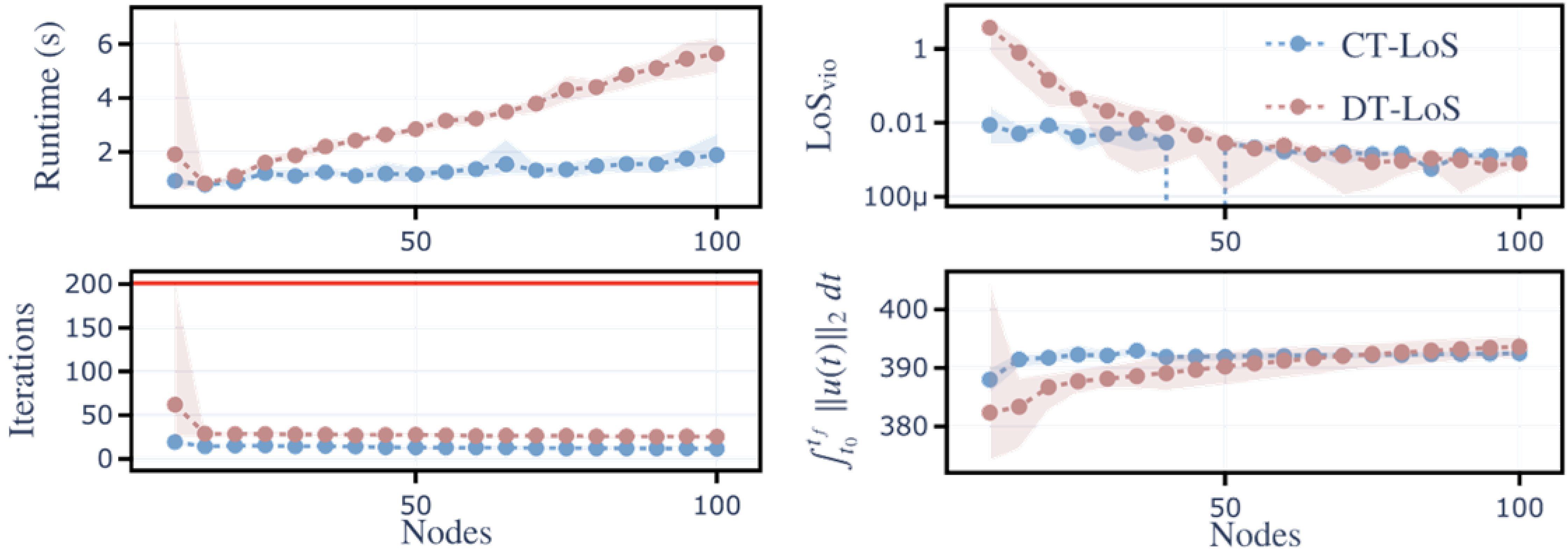


CT-LoS Scales much better with respect to discretization grid size.



CT-LoS is significantly more performant than DT-LoS for LoS violation. However, it sacrifices some objective performance

Cinematography Scenario



Takeaways

Q1. How well does CT-LoS satisfy the LoS constraint throughout the entire trajectory compared to the baseline?

> ***A1.*** *Across both scenarios, the proposed method consistently shows either lower or equivalent LoS Violation to the baseline.*

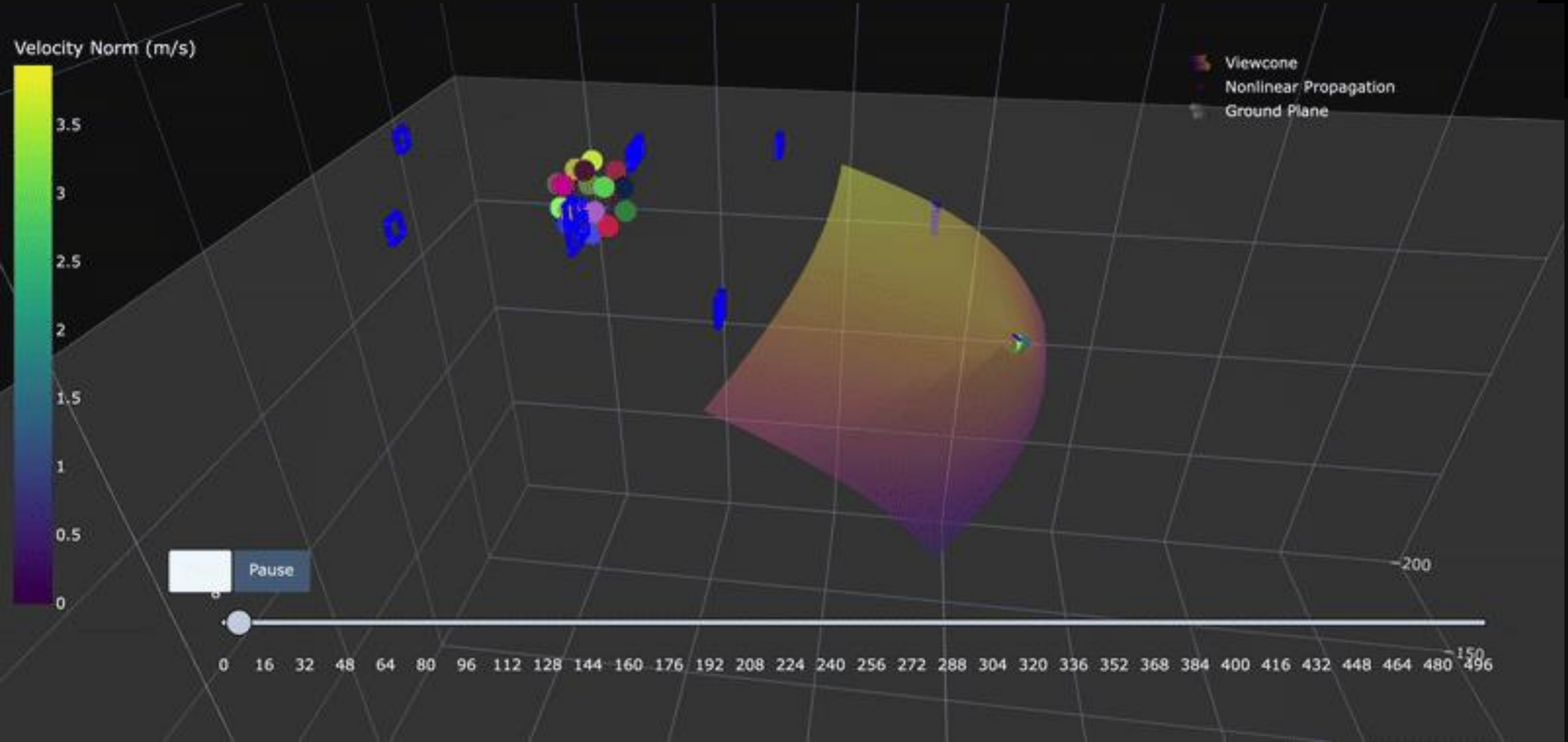
Q2. What tradeoffs are made to achieve better LoS violation performance?

> ***A2.*** *DT-LoS better objective performance as it inherently solves a less constrained approximation of the original nonconvex problem*

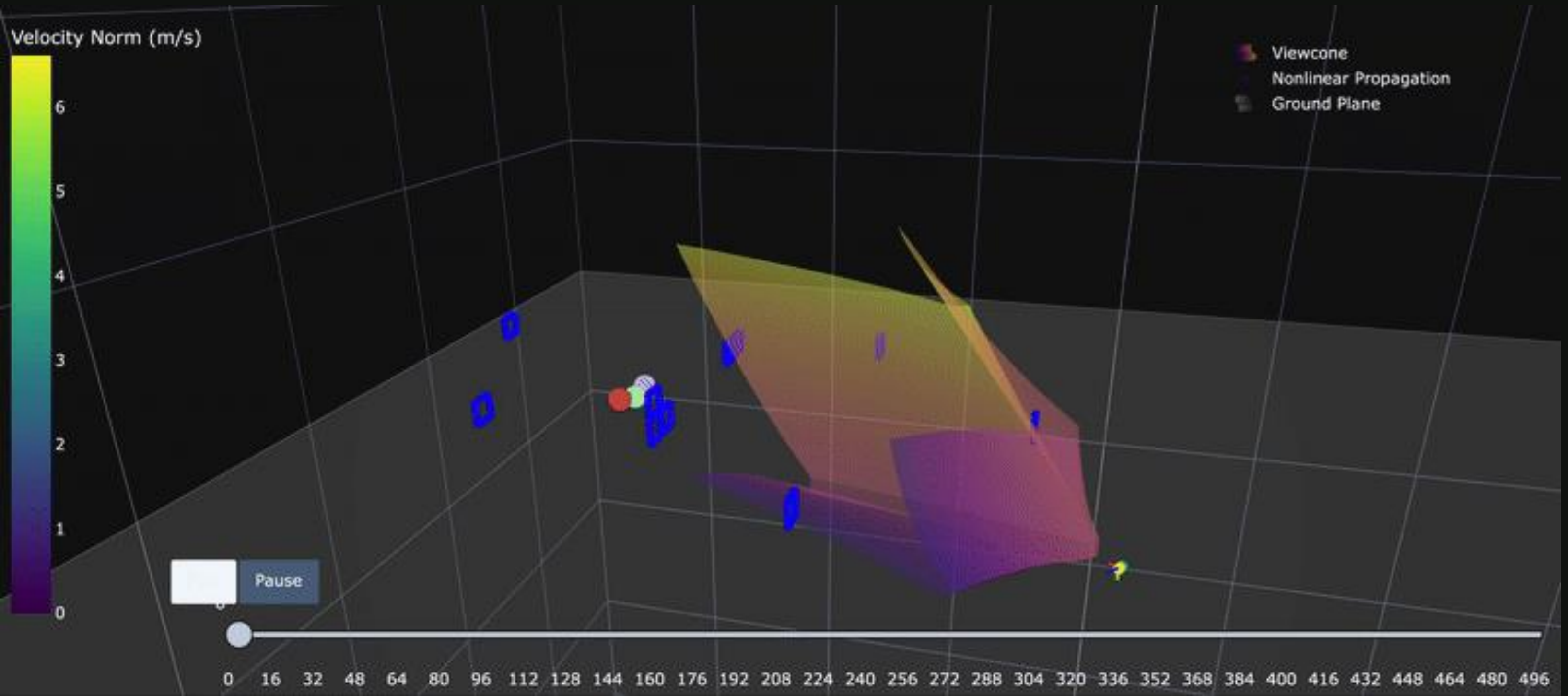
Q3. How does CT-LoS scale as the problem size increase?

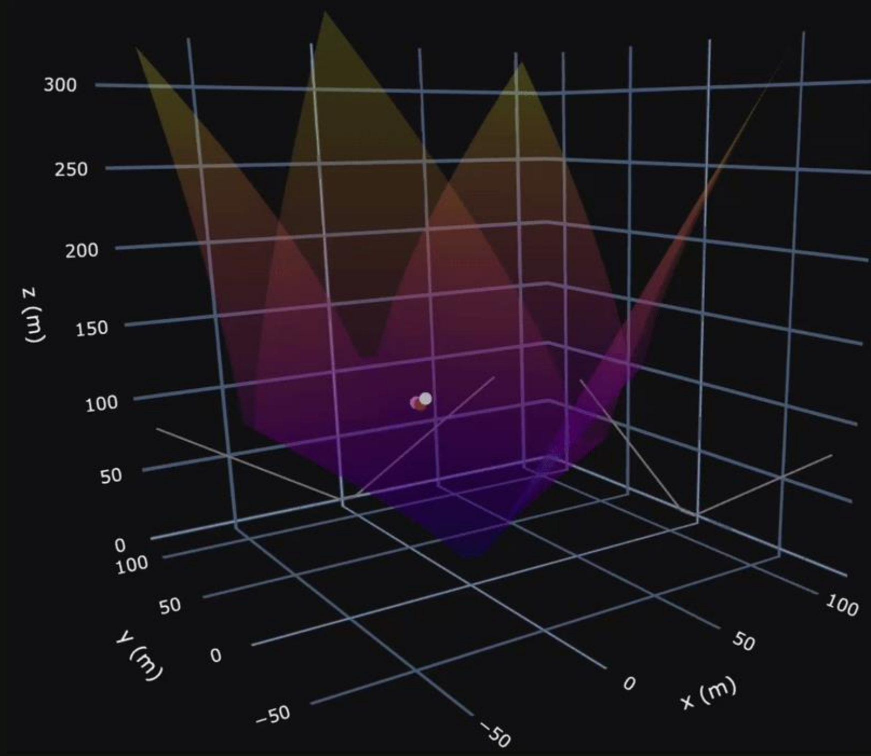
> ***A3.*** *As the discretization grid size increases, CT-LoS is significantly less affected than DT-LoS as the convex subproblem has significantly fewer nodal constraints*

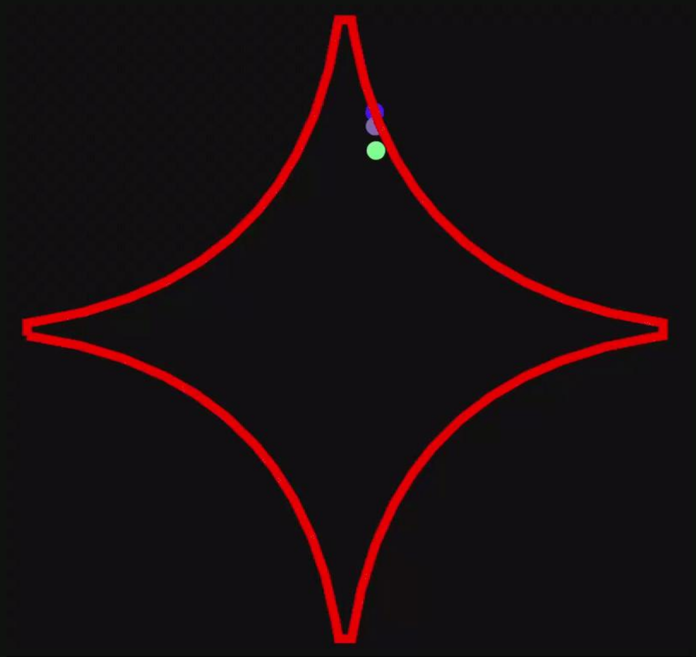
Challenging Cases: Polytopic Containment



Challenging Cases: Weird Norms







Challenging Cases: Corkscrew Maneuver