Obstacle Avoidance using Potential Fields

Bjarke Engsig Larsen

Emil Vincent Ancker

Mathias Emil Slettemark-Nielsen

Mikkel Larsen

Overview

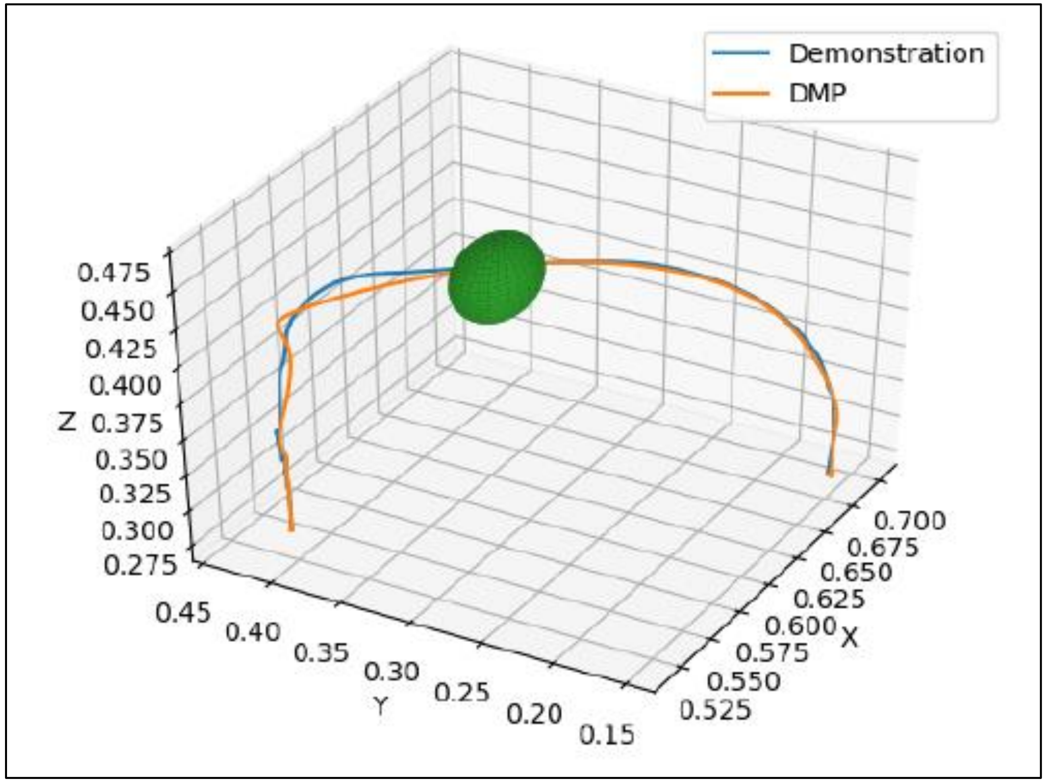
- Obstacle Avoidance using Potential Fields
- Link Collision Avoidance
- Pose Estimation

End-effector Collision Avoidance using Potential Fields

- Spherical obstacles
- DMP

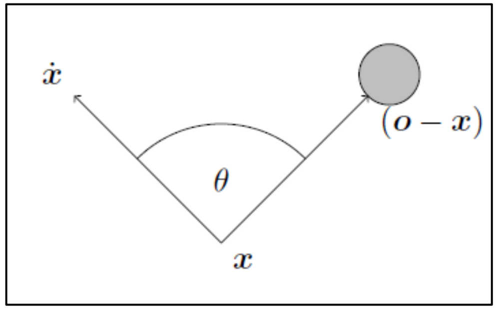
$$\tau \dot{z} = \alpha_z(\beta_z(g - x) - z) + f + C_t,$$

$$\tau \dot{x} = z,$$

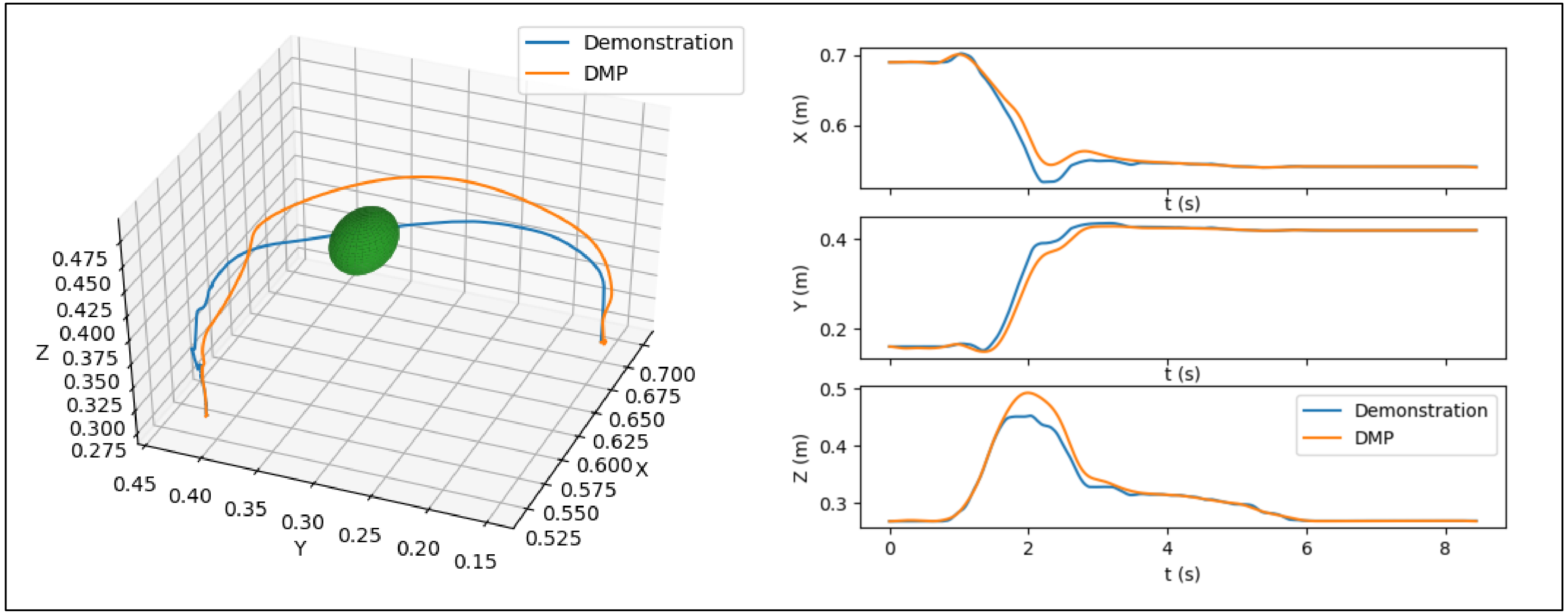


[5] Akshara Rai et al. “Learning coupling terms for obstacle avoidance”.

Potential Field as a Function of Steering Angle



$$C_t = \gamma_0 R \dot{x} \theta e^{-\beta_0 |\theta|}$$

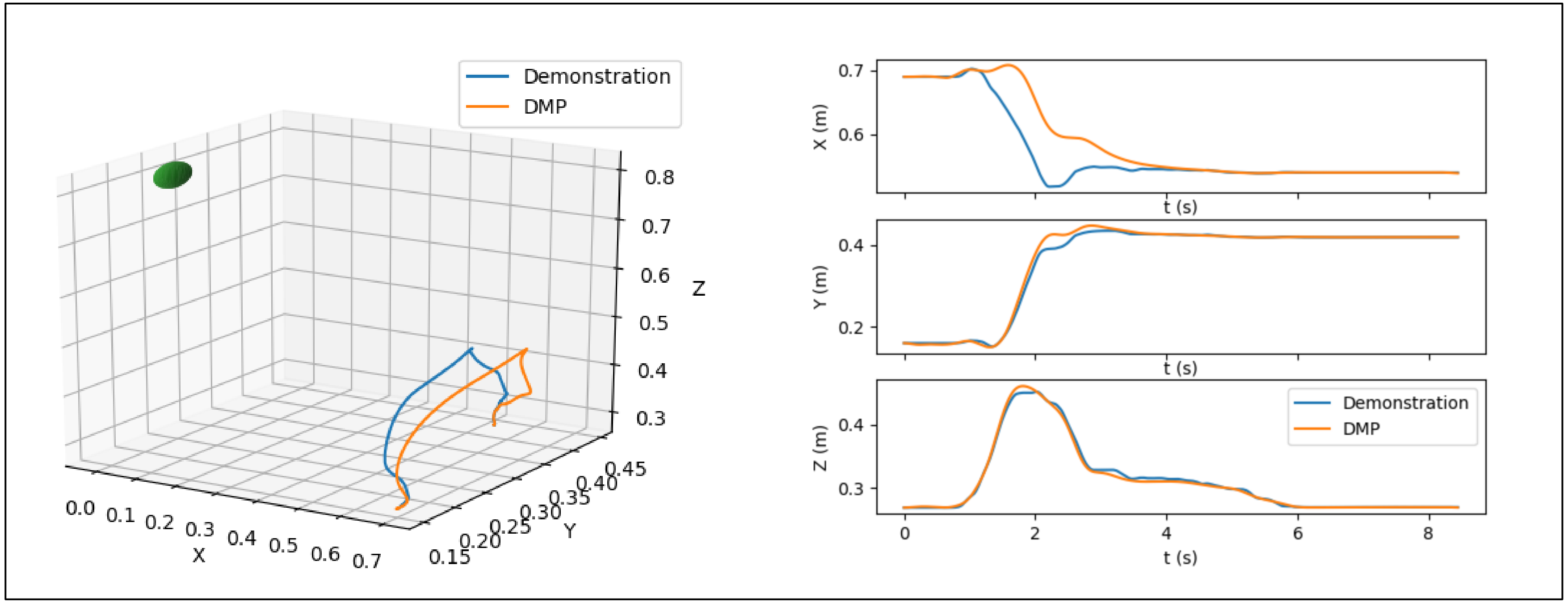


[4] W. Warren, B. Fajen, and D. Belcher. “Behavioral dynamics of steering, obstacle avoidance, and route selection”.
[5] Akshara Rai et al. “Learning coupling terms for obstacle avoidance”.

Potential Field as a Function of Steering Angle

Problems

- Obstacle far away
- $\theta = 0$



$$C_t = \gamma_0 R \dot{x} \theta e^{-\beta_0 |\theta|}$$

[4] W. Warren, B. Fajen, and D. Belcher. "Behavioral dynamics of steering, obstacle avoidance, and route selection".
[5] Akshara Rai et al. "Learning coupling terms for obstacle avoidance".

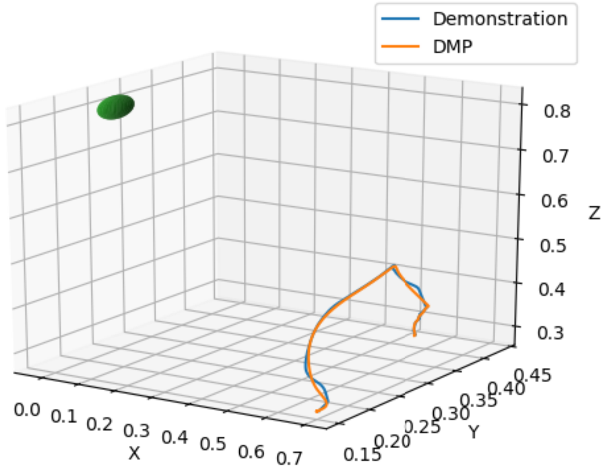
Coupling terms DMP

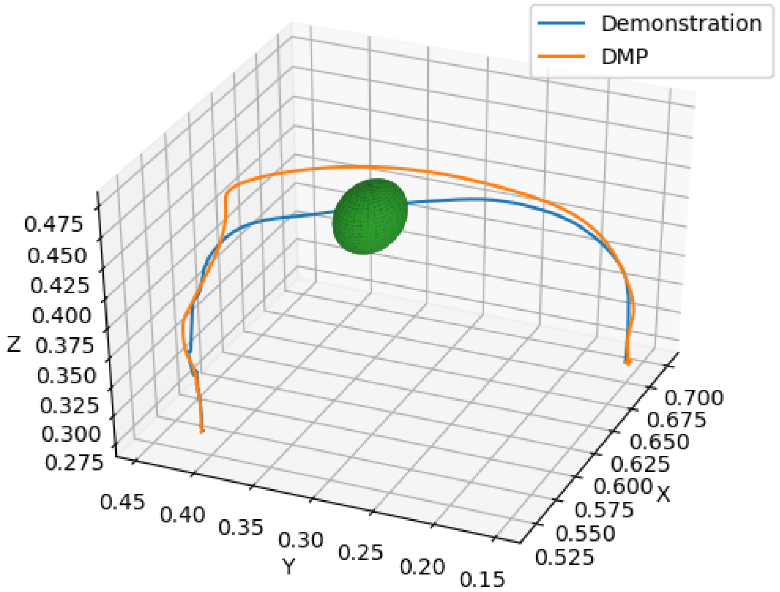
$$C_t = \gamma_1 R\dot{\boldsymbol{x}}\phi_1 + \gamma_2 R\dot{\boldsymbol{x}}\phi_2 + \gamma_3 R\dot{\boldsymbol{x}}\phi_3$$

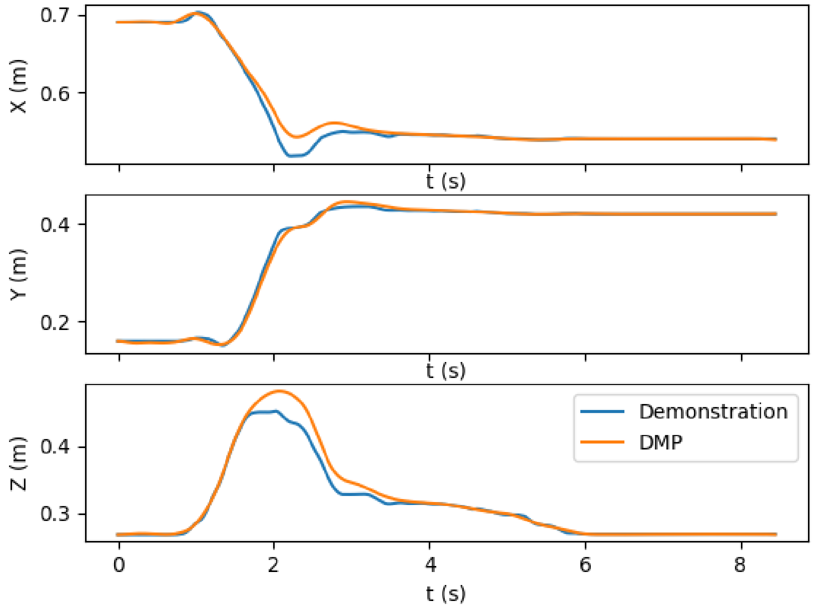
$$\phi_1 = \theta e^{-\beta_0|\theta|} \cdot e^{-kd}$$

$$\phi_2 = \theta e^{-\beta_0|\theta_p|} \cdot e^{-kd_p}$$

$$\phi_3 = e^{-kd}$$



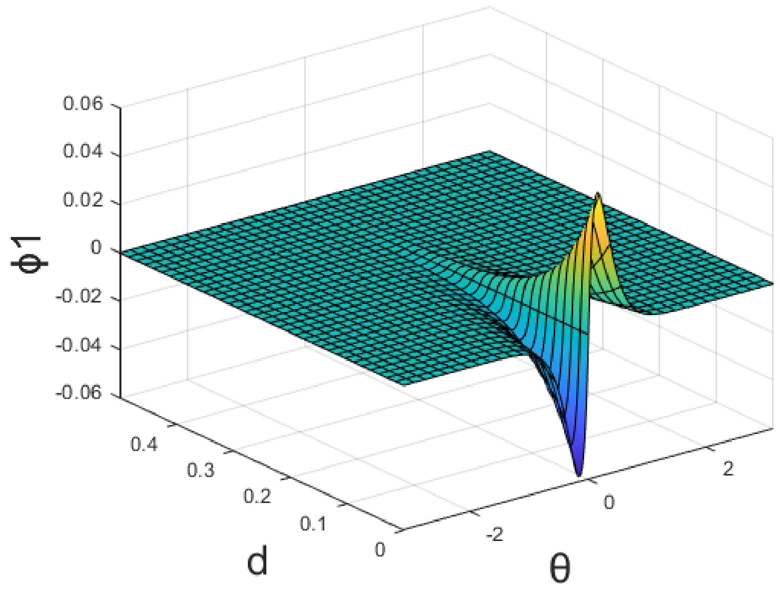


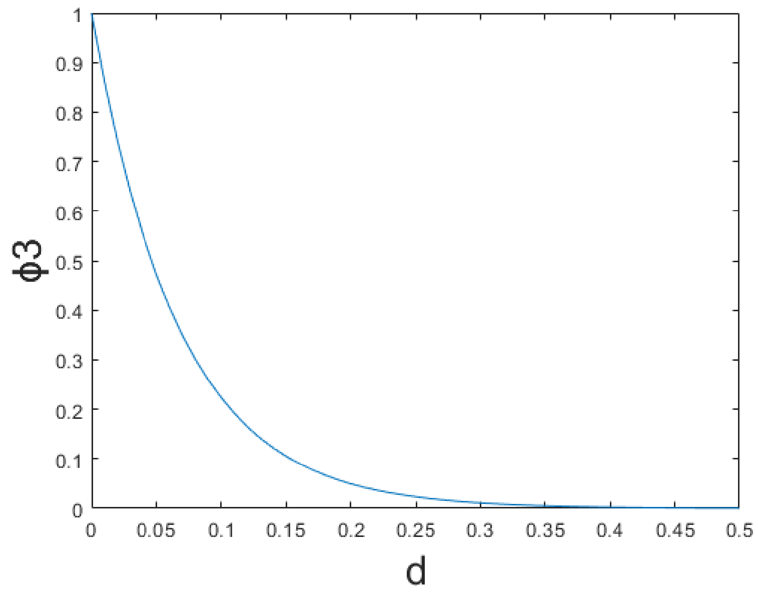


[5] Akshara Rai et al. “Learning coupling terms for obstacle avoidance”.

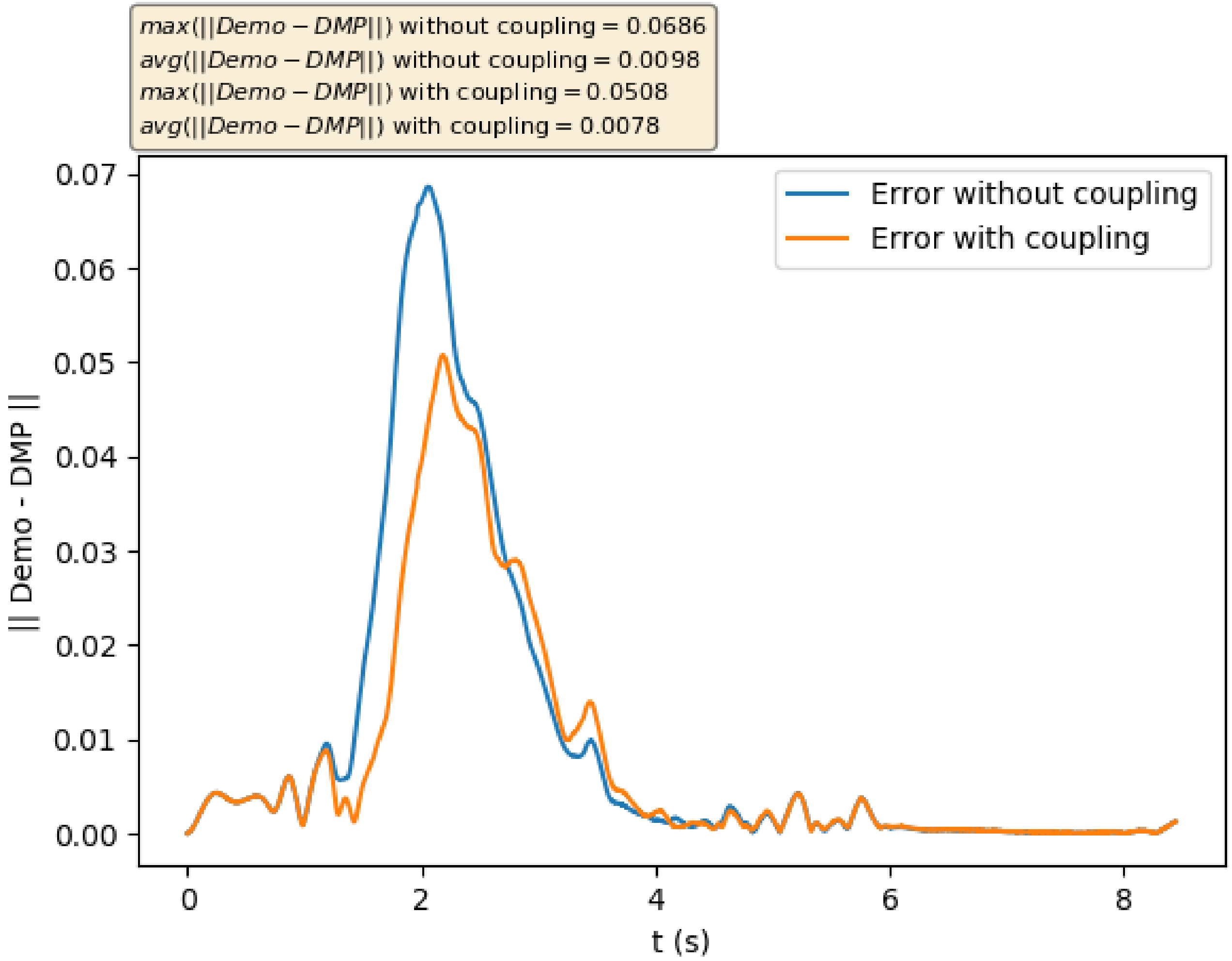
Coupling terms DMP

- Choosing distance scaling constant k
  - $\uparrow k \rightarrow$ less impact of obstacles
  - $\downarrow k \rightarrow$ high impact of obstacles
  - $k = 15$
- Correct weights of three terms
  - γ_1, γ_2 and γ_3



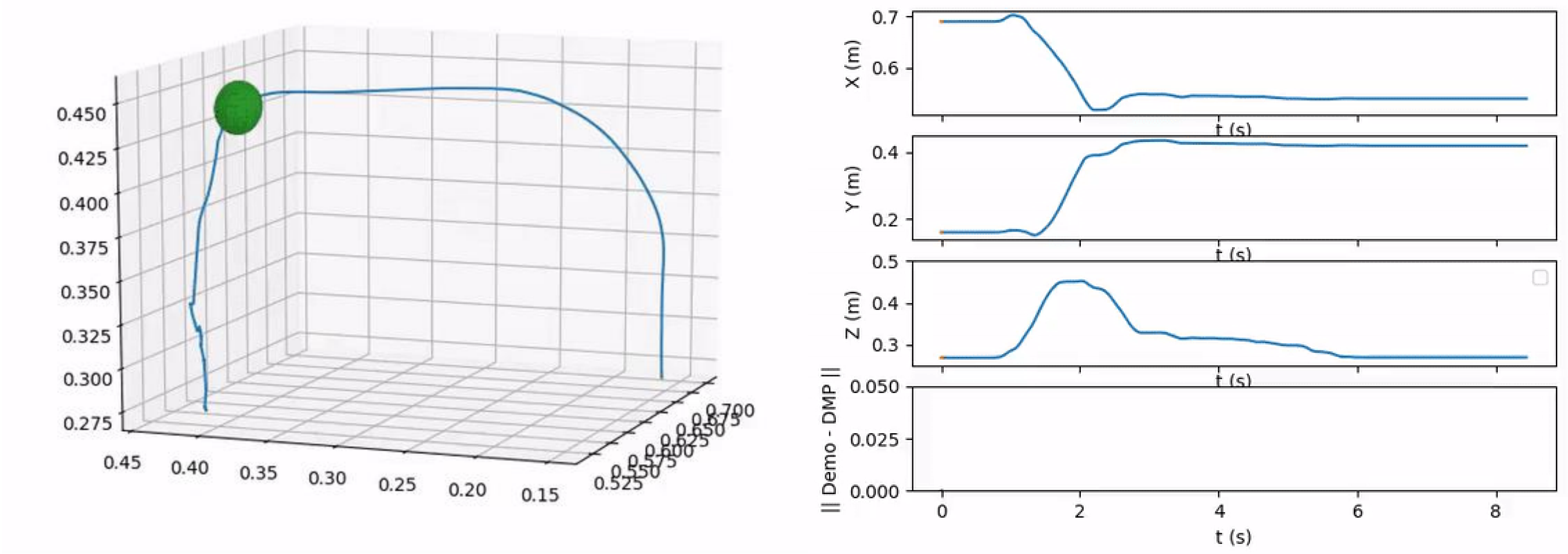


Coupling terms DMP



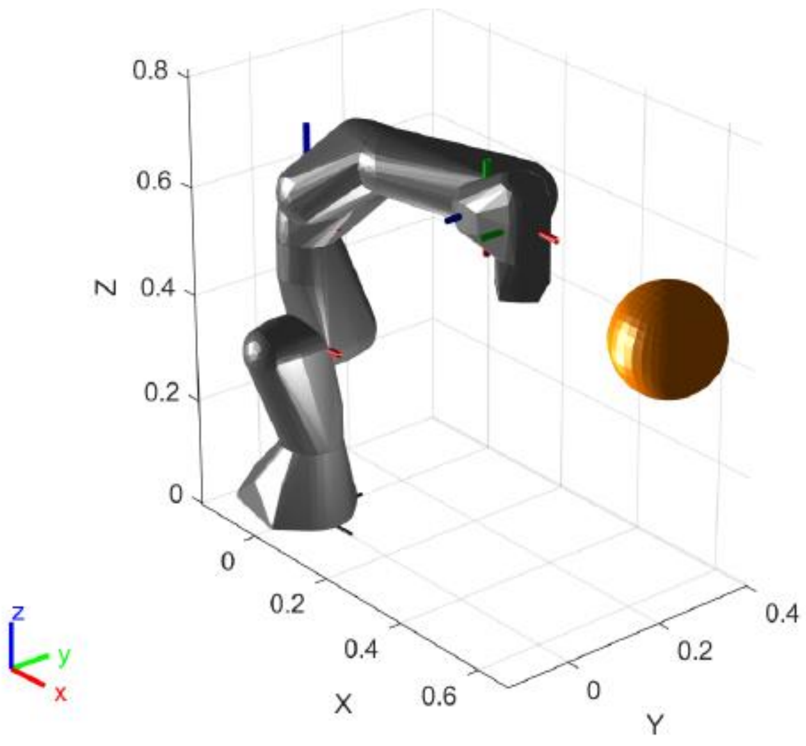
Online obstacle avoidance

- Evaluating position of obstacle

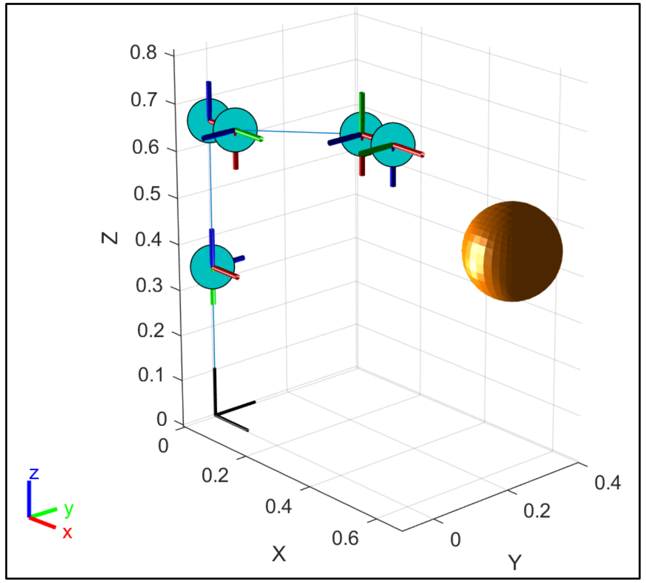


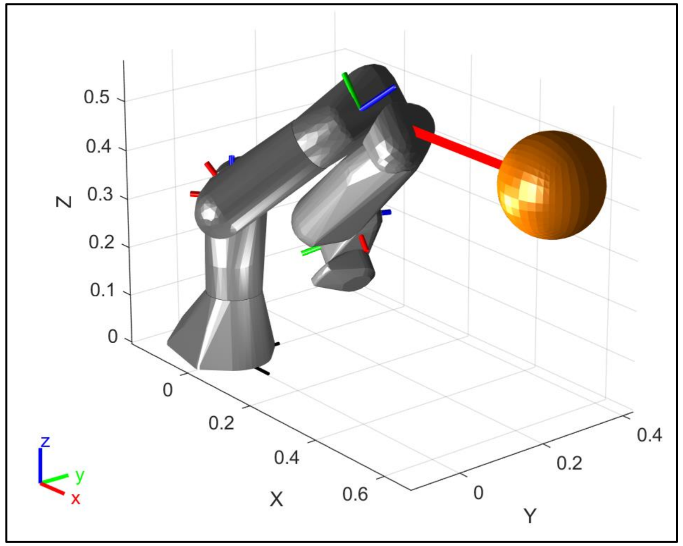
Link Collision Avoidance

- 7-DOF Franka Emika Panda
- Spherical obstacles



Pseudoinverse based Link Collision Avoidance



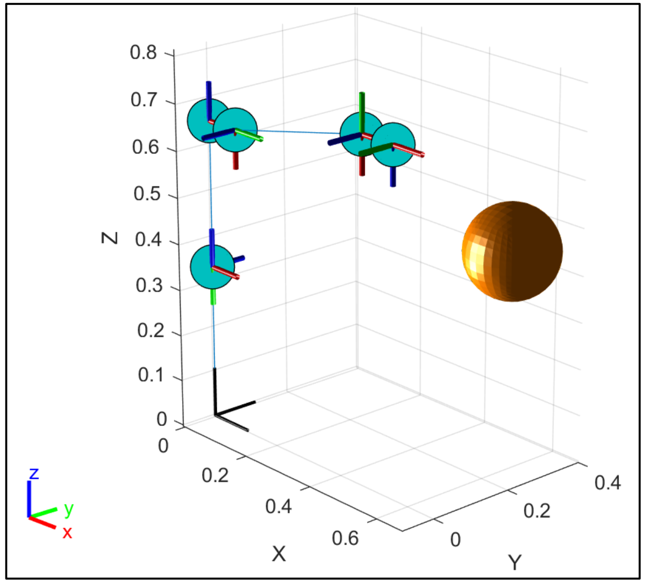


$$\dot{\boldsymbol{x}}_0 = -\gamma_L \nabla U(\boldsymbol{x}_0),$$

$$\nabla_{\boldsymbol{x}} U(\boldsymbol{x}) = \eta \left(\frac{1}{p(\boldsymbol{x})} - \frac{1}{p_0} \right) \left(\frac{-1}{p(\boldsymbol{x})^2} \right) \nabla_{\boldsymbol{x}} p(\boldsymbol{x})$$

[7] Anthony A. Maciejewski and Charles A. Klein. “Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments”.
[8] Dae-Hyung Park et al. “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields”.
[9] Samuel R Buss. “Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods”

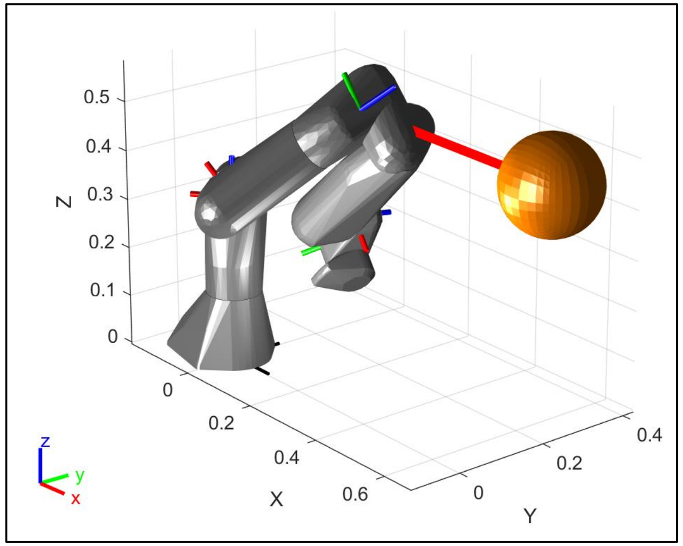
Pseudoinverse based Link Collision Avoidance



$$\dot{\boldsymbol{x}}_0 = -\gamma_L \nabla U(\boldsymbol{x}_0),$$

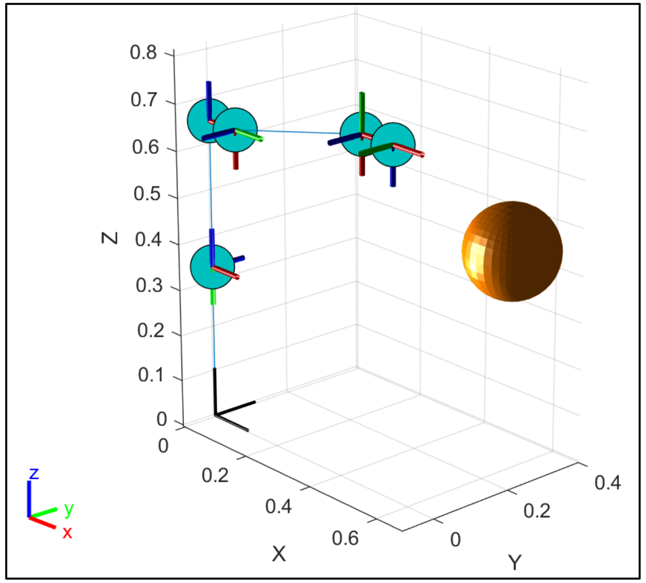
$$\nabla_{\boldsymbol{x}} U(\boldsymbol{x}) = \eta \left(\frac{1}{p(\boldsymbol{x})} - \frac{1}{p_0} \right) \left(\frac{-1}{p(\boldsymbol{x})^2} \right) \nabla_{\boldsymbol{x}} p(\boldsymbol{x})$$

$$\boxed{\dot{q} = \boldsymbol{J}_e^{+} \dot{\boldsymbol{x}}_e + \left(\boldsymbol{I} - \boldsymbol{J}_e^{+} \boldsymbol{J}_e \right) \boldsymbol{\mathcal{E}}}$$



[7] Anthony A. Maciejewski and Charles A. Klein. “Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments”.
[8] Dae-Hyung Park et al. “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields”.
[9] Samuel R Buss. “Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods”

Pseudoinverse based Link Collision Avoidance



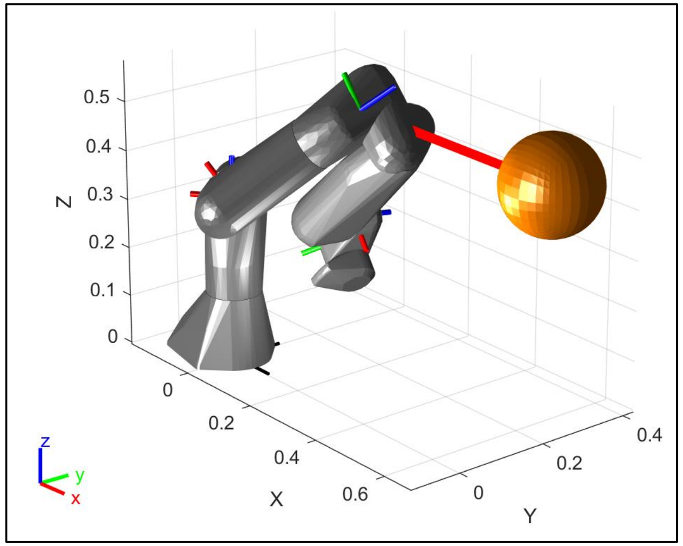
$$\dot{\boldsymbol{x}}_0 = -\gamma_L \nabla U(\boldsymbol{x}_0),$$

$$\nabla_{\boldsymbol{x}} U(\boldsymbol{x}) = \eta \left(\frac{1}{p(\boldsymbol{x})} - \frac{1}{p_0} \right) \left(\frac{-1}{p(\boldsymbol{x})^2} \right) \nabla_{\boldsymbol{x}} p(\boldsymbol{x})$$

$$\dot{q} = J_e^+ \dot{x}_e + \left(I - J_e^+ J_e \right) \mathcal{E}$$

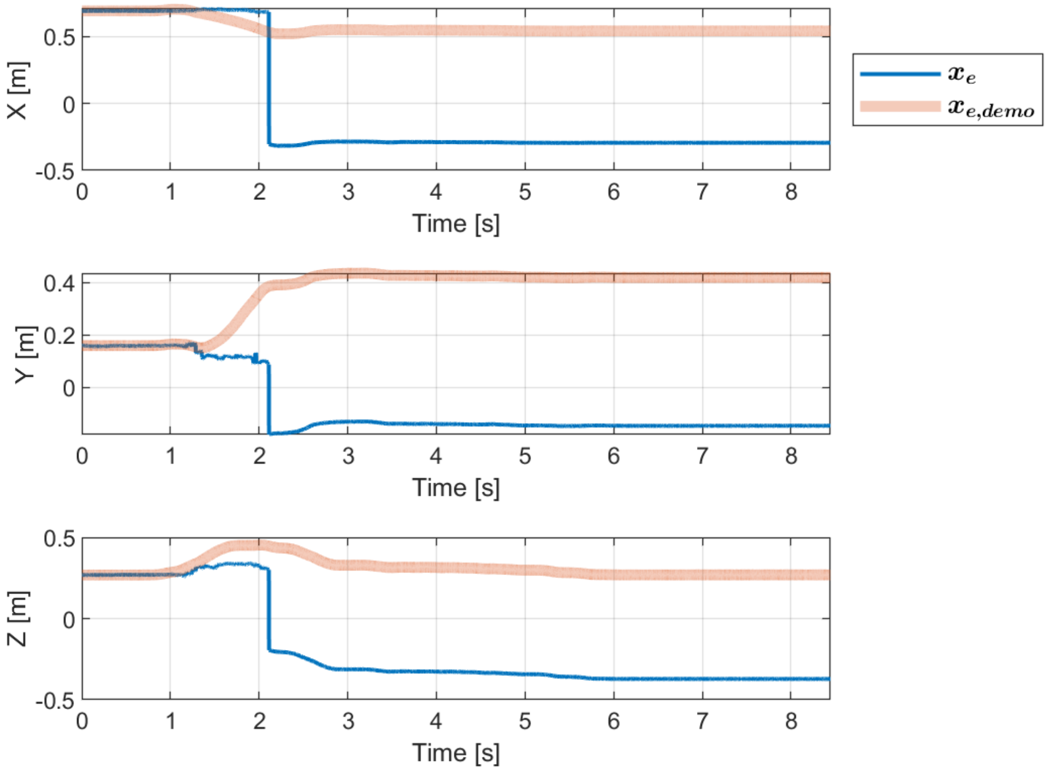
$$\boldsymbol{\mathcal{E}} = \left(J_0 (I - J_e^+ J_e) \right)^+ \left(\dot{x}_0 - J_0 J_e^+ \dot{x}_e \right)$$

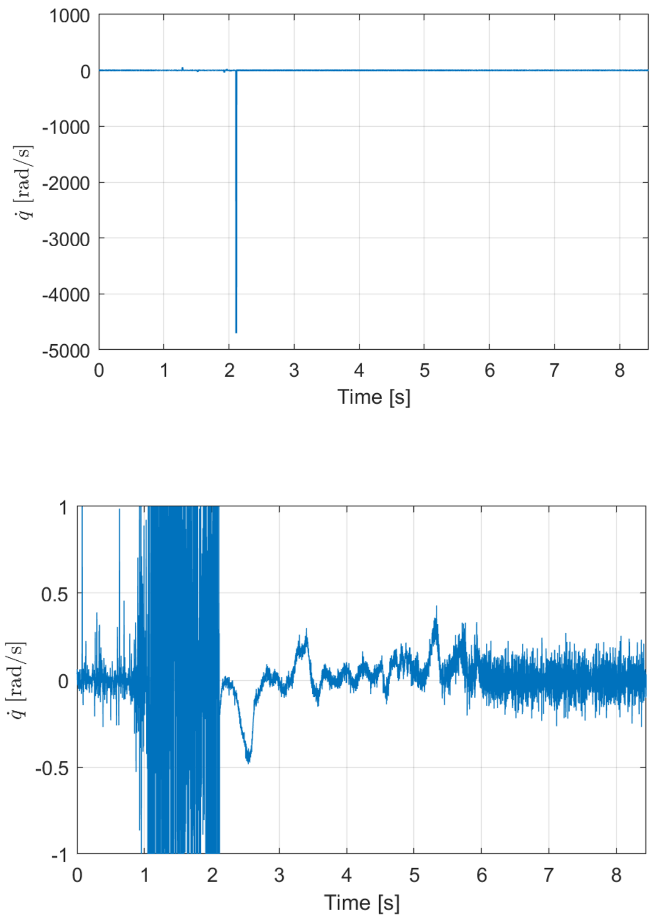
$$\dot{q} = J_e^+ \dot{x}_e + \left(J_0 (I - J_e^+ J_e) \right)^+ \left(\dot{x}_0 - J_0 J_e^+ \dot{x}_e \right)$$



[7] Anthony A. Maciejewski and Charles A. Klein. “Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments”.
[8] Dae-Hyung Park et al. “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields”.
[9] Samuel R Buss. “Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods”

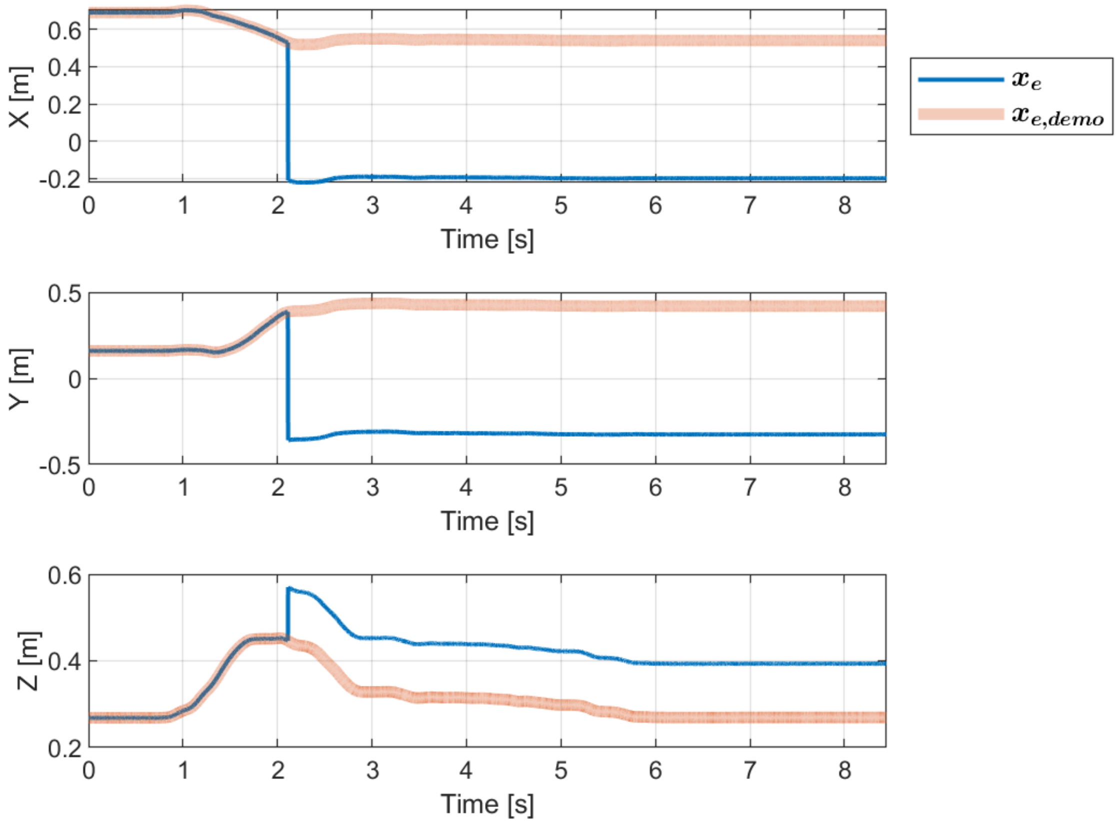
Pseudoinverse based Link Collision Avoidance

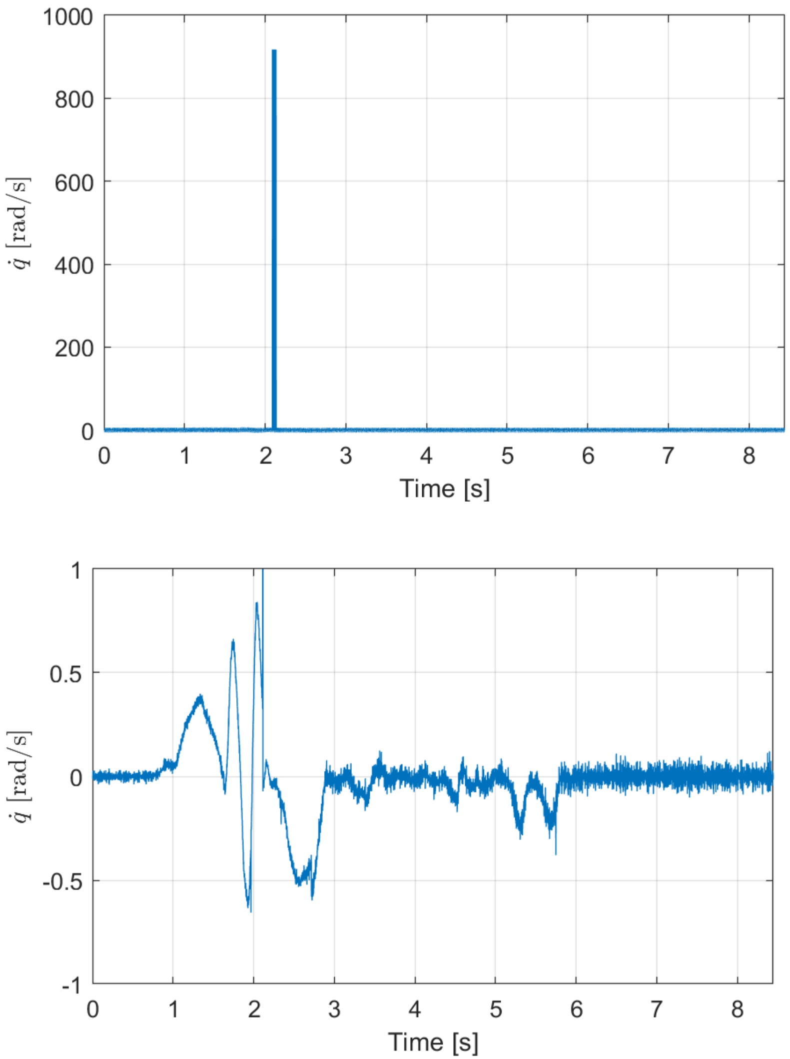




Pseudoinverse based Link Collision Avoidance

Threshold on pseudoinverse





Damped Least Squares based Link Collision Avoidance

$$\min \ f(\dot{\boldsymbol{q}}) = ||\boldsymbol{J}\dot{\boldsymbol{q}} - \dot{\boldsymbol{x}}_e||^2 + \lambda^2||\dot{\boldsymbol{q}}||^2$$

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_e^T \boldsymbol{f}$$

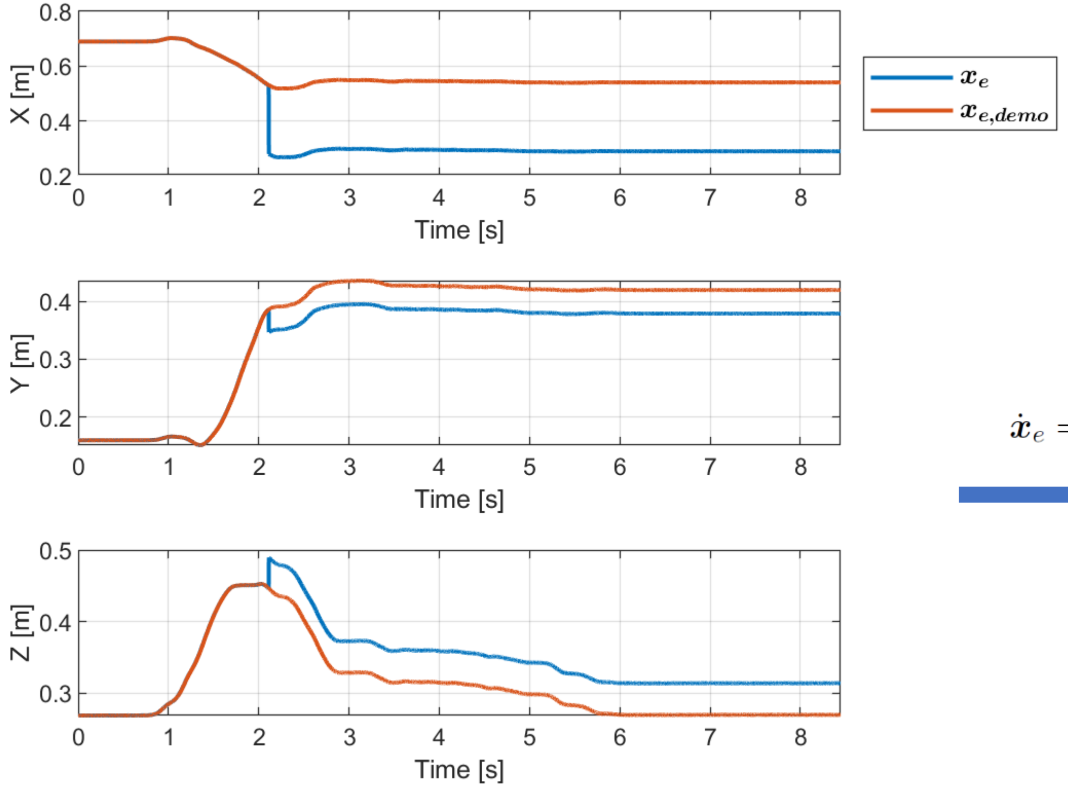
$$(\boldsymbol{J}_e \boldsymbol{J}_e^T + \lambda^2 \boldsymbol{I})\boldsymbol{f} = \dot{\boldsymbol{x}}_e$$

- Combination of the two methods

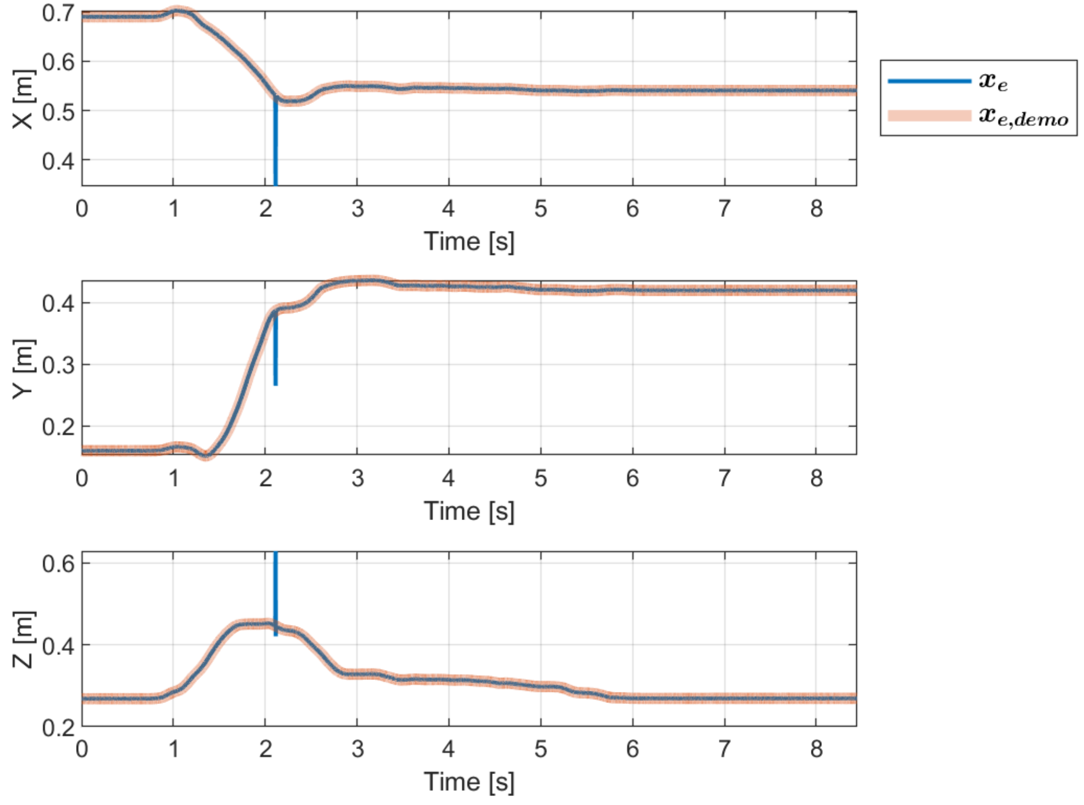
$$\dot{q} = \boldsymbol{J}_e^T \boldsymbol{f} + \left(\boldsymbol{J}_0(I - \boldsymbol{J}_e^+ \boldsymbol{J}_e)\right)^+ (\dot{x}_0 - \boldsymbol{J}_0 \boldsymbol{J}_e^+ \dot{x}_e)$$

[9] Samuel R Buss. "Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods"

Damped Least Squares based Link Collision Avoidance



$$\dot{x}_e = \frac{x_{e,demo} - x_e}{dt}$$



Damped Least Squares based Link Collision Avoidance

Algorithm 2: Link collision avoidance using IK solved with damped least squares

Requires initialization of: $\boldsymbol{T}$ (trajectory), Δt (time step), $\boldsymbol{q}$ (initial configuration).

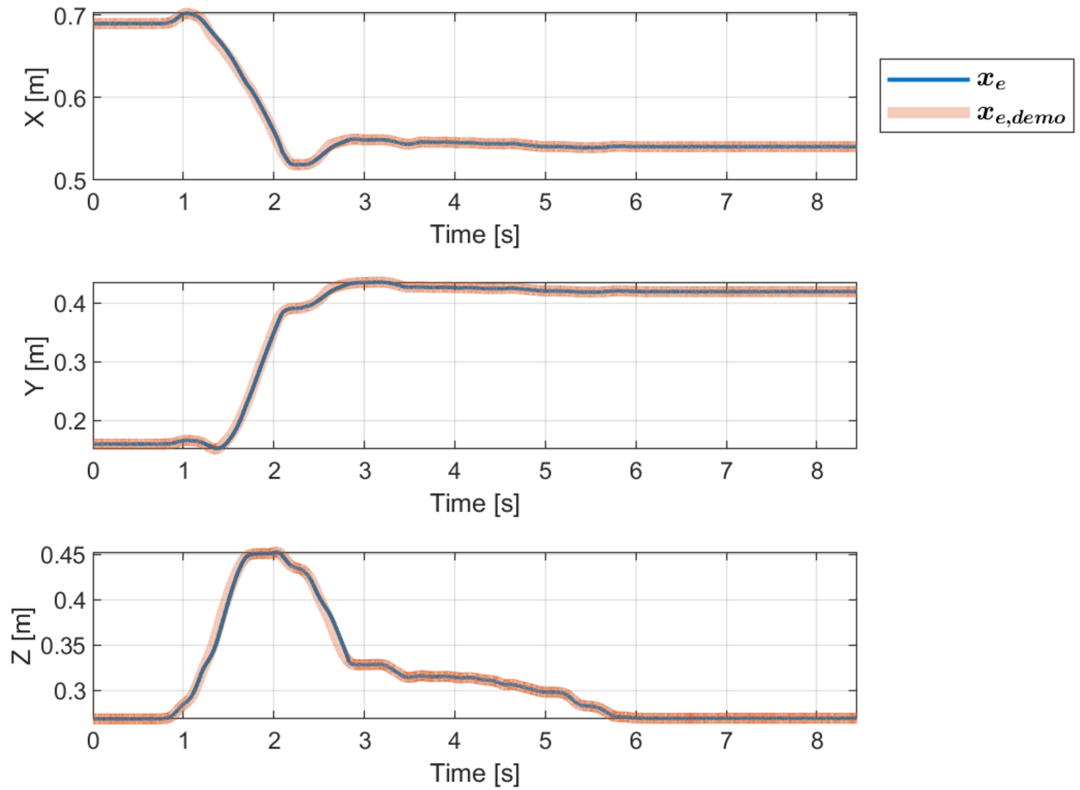
while *points left in* $\boldsymbol{T}$ **do**

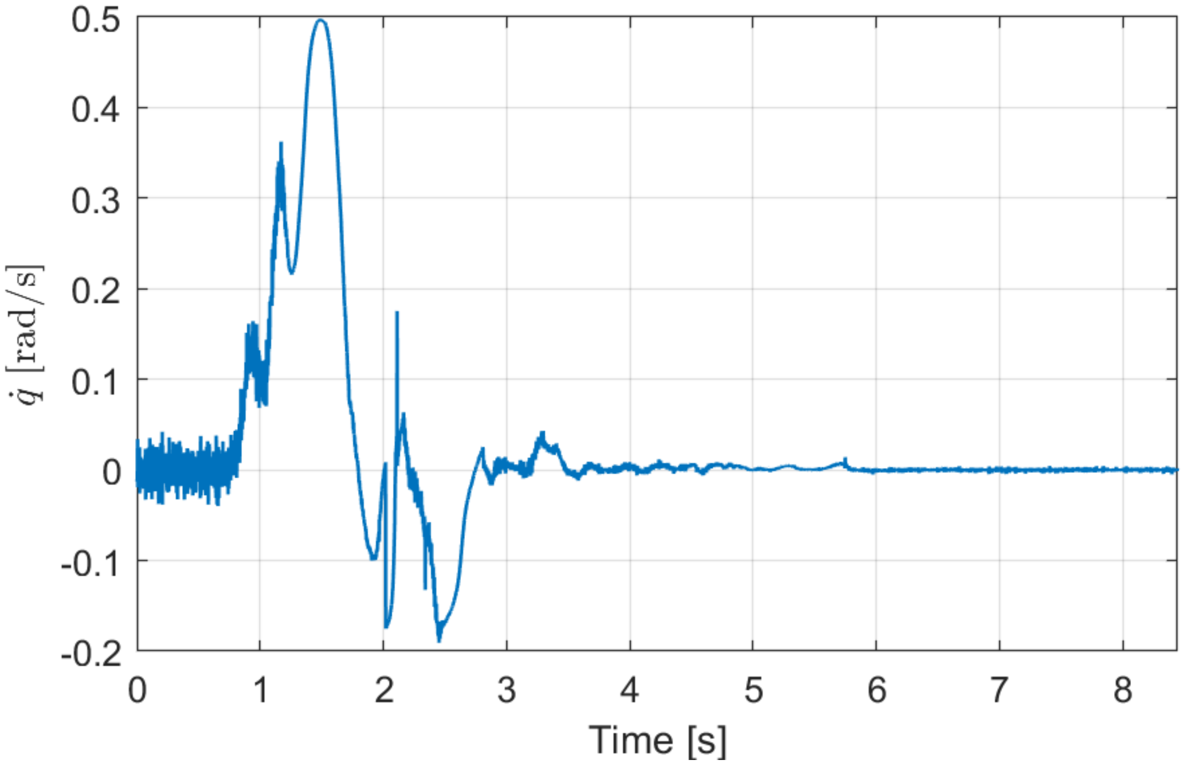
- $\dot{\boldsymbol{x}}_e = \frac{\boldsymbol{T}_{i+1} - \boldsymbol{x}_e}{\Delta t}$
- $\boldsymbol{x}_0 = \arg\min_{\boldsymbol{x}} ||(\boldsymbol{o} - \boldsymbol{x}_i)||^2$ ▷ find point on robot closest to obstacle
- $\dot{\boldsymbol{x}}_0 = -\gamma_L \nabla U(\boldsymbol{x}_0)$
- $\boldsymbol{J}_0 = \text{ComputeJacobian}(\boldsymbol{x}_0)$
- $\boldsymbol{J}_e = \text{ComputeJacobian}(\boldsymbol{x}_e)$
- $\boldsymbol{f} = \text{LinearSolve}\big((\boldsymbol{J}_e \boldsymbol{J}_e^T + \lambda^2 \boldsymbol{I})\boldsymbol{f} = \dot{\boldsymbol{x}}_e\big)$
- $\dot{\boldsymbol{q}} = \boldsymbol{J}_e^T \boldsymbol{f} + \big(\boldsymbol{J}_0(I - \boldsymbol{J}_e^+ \boldsymbol{J}_e)\big)^+ (\dot{\boldsymbol{x}}_0 - \boldsymbol{J}_0 \boldsymbol{J}_e^+ \dot{\boldsymbol{x}}_e)$
- $\dot{\boldsymbol{q}} = \text{ClampAbsMax}(\dot{\boldsymbol{q}},\ 1)$
- $\boldsymbol{q} = \boldsymbol{q} + \dot{\boldsymbol{q}} \Delta t$ ▷ configuration is being updated

end

$$\text{ClampAbsMax}(\dot{\boldsymbol{q}}, d) = \begin{cases} \dot{\boldsymbol{q}}, & ||\dot{\boldsymbol{q}}|| < d \\ d \frac{\dot{\boldsymbol{q}}}{||\dot{\boldsymbol{q}}||}, & \text{otherwise} \end{cases}$$

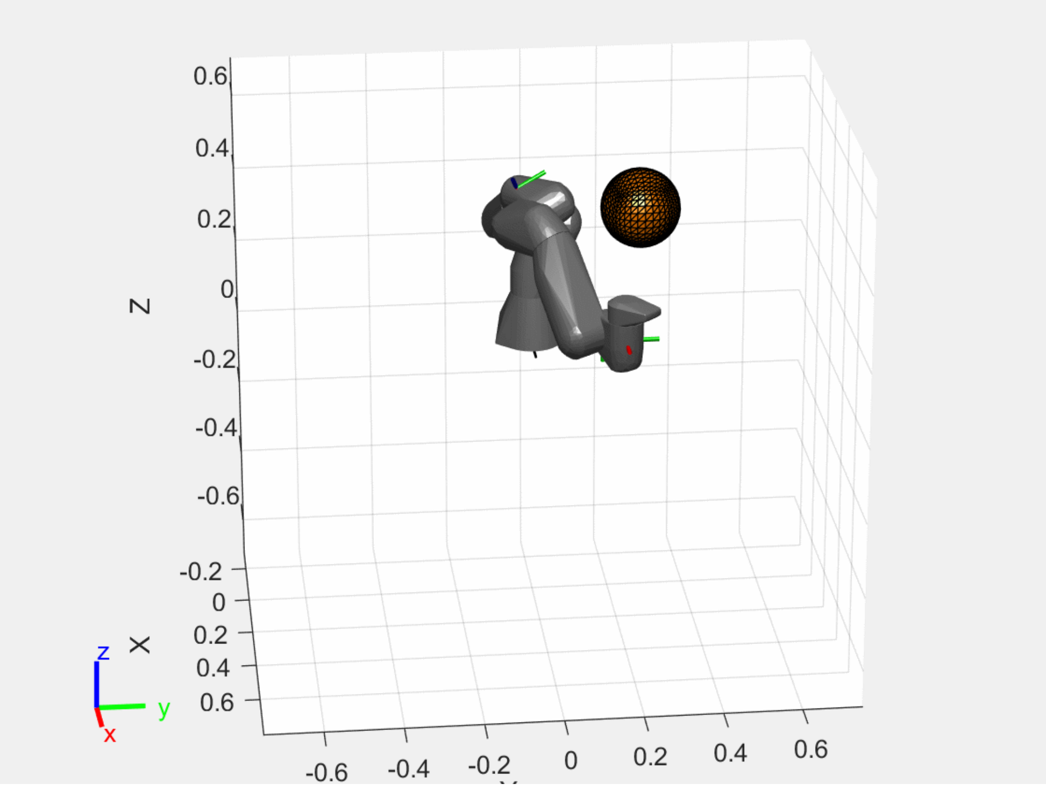
Damped Least Squares based Link Collision Avoidance



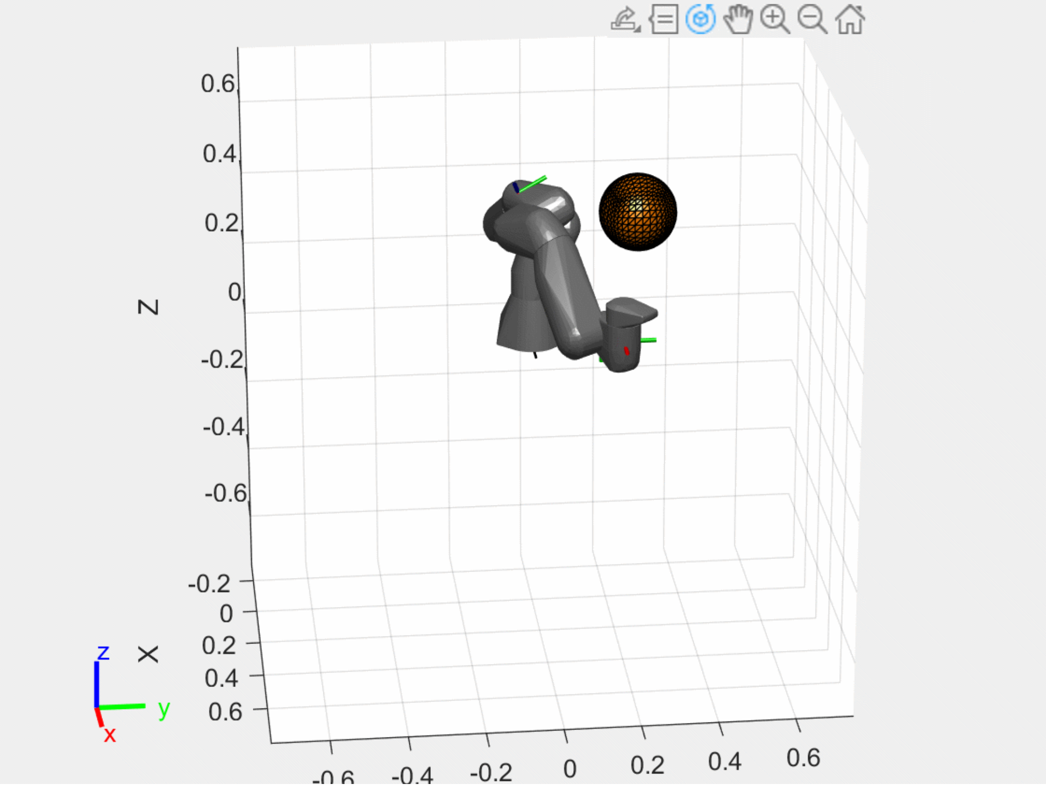


Damped Least Squares based Link Collision Avoidance

$\gamma_L = 0$

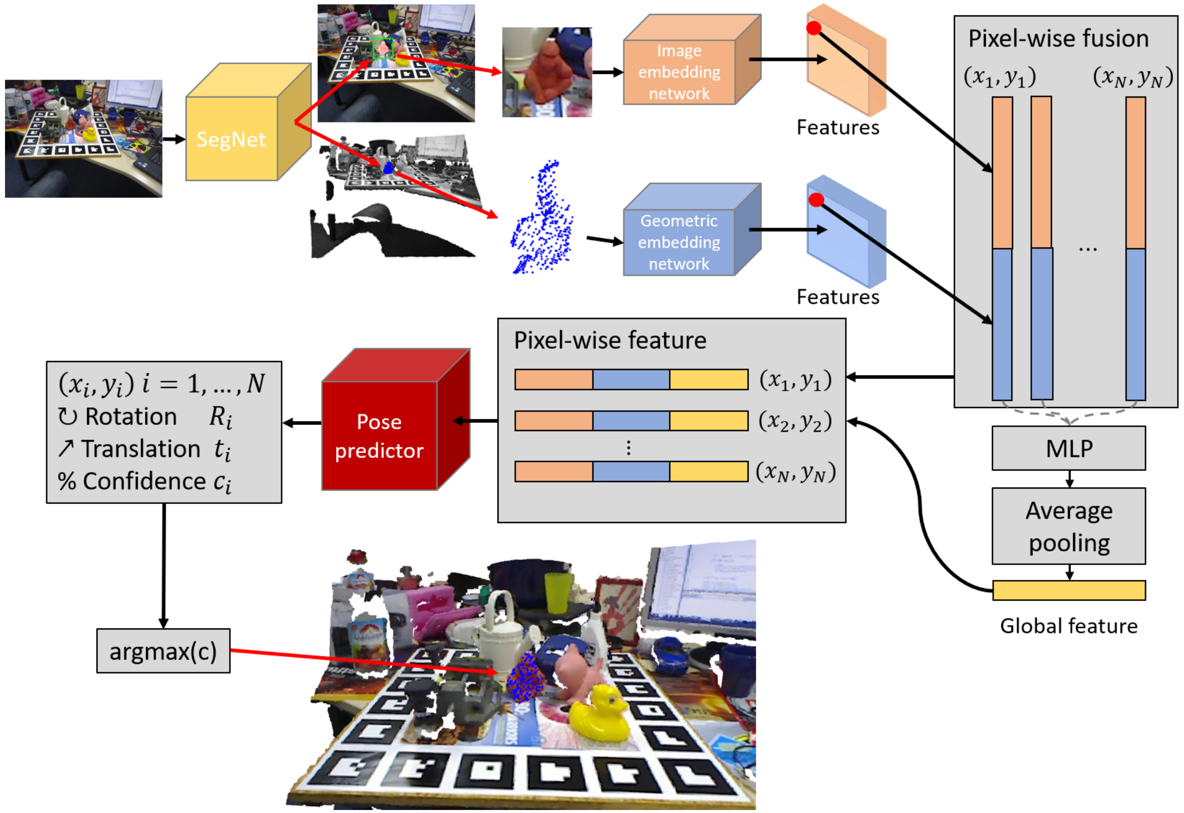


$\gamma_L = 100$



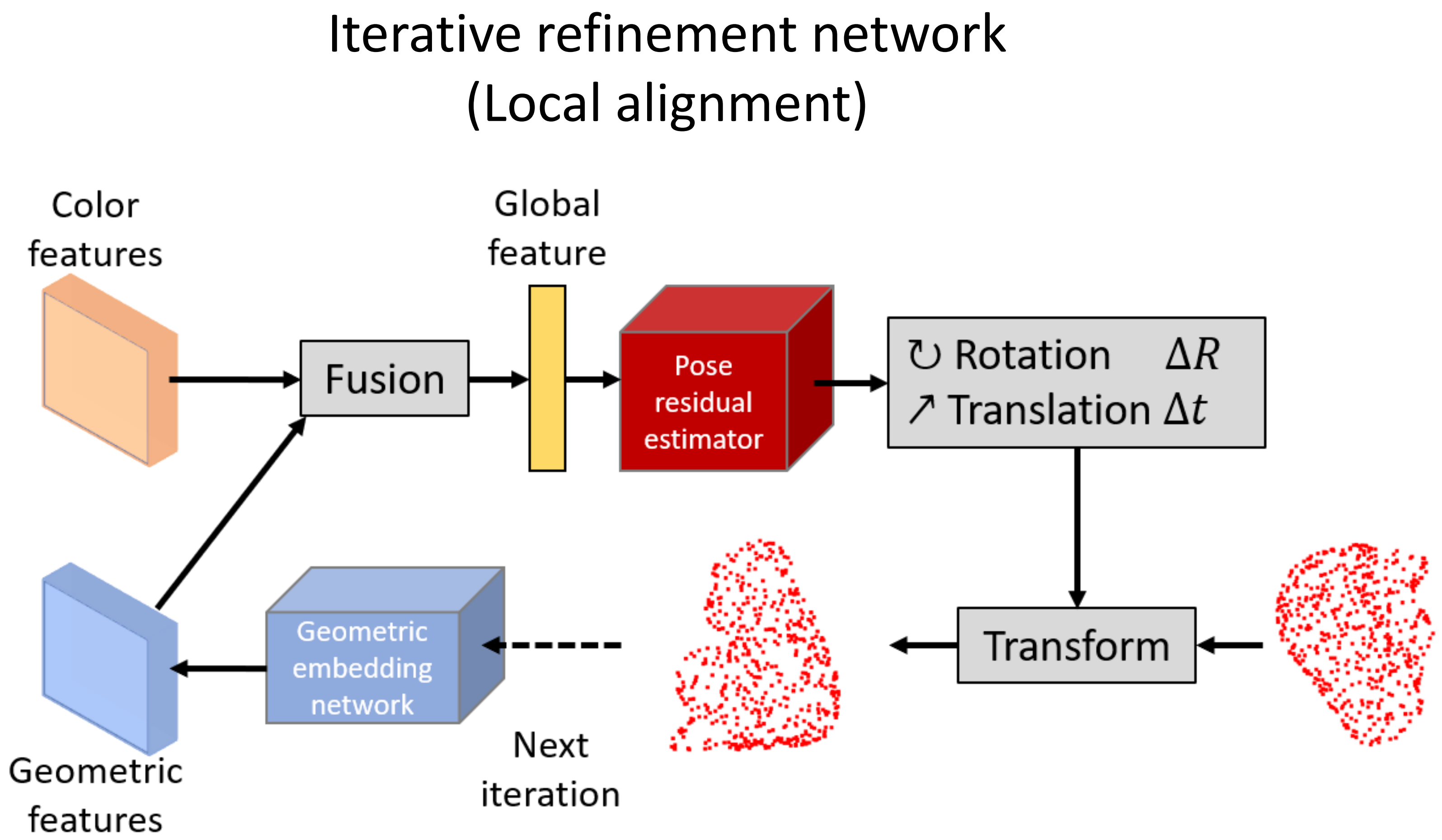
Pose Estimation - Iterative Dense Fusion

Pose predictor network (Global alignment)



[13] Chen Wang et al. “DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion”.

Pose Estimation - Iterative Dense Fusion

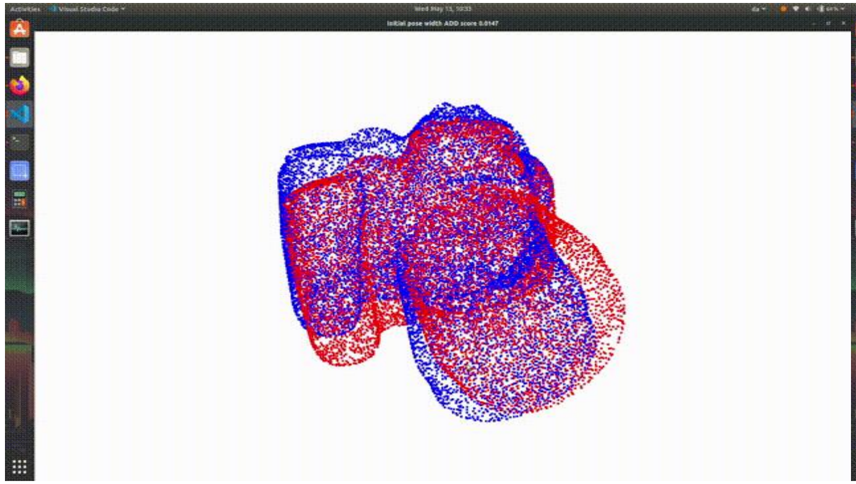


[13] Chen Wang et al. “DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion”.

Iterative Refinement Network

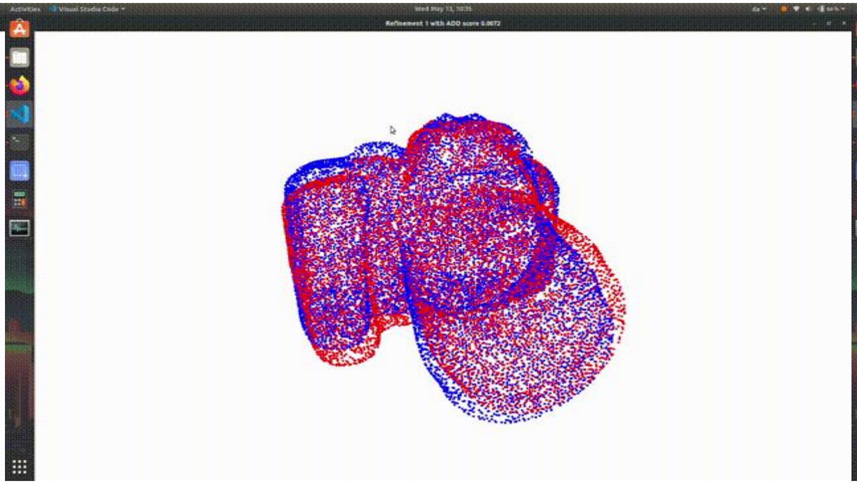
Pose predictor net

ADD score: 0.0147



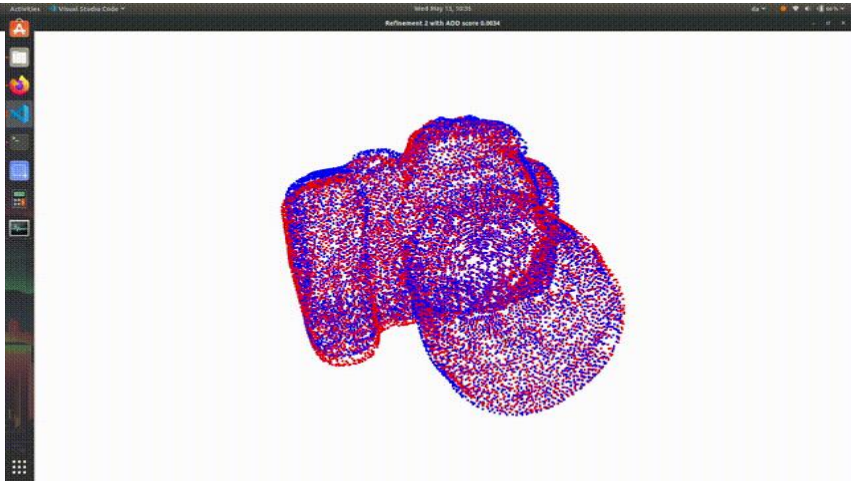
Pose refine net

ADD score: 0.0074



Pose refine net

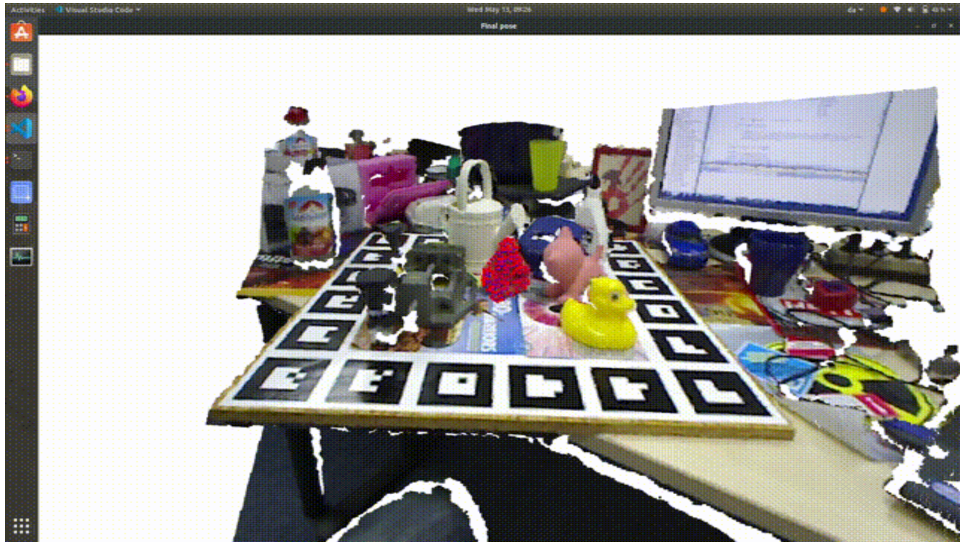
ADD score: 0.0034

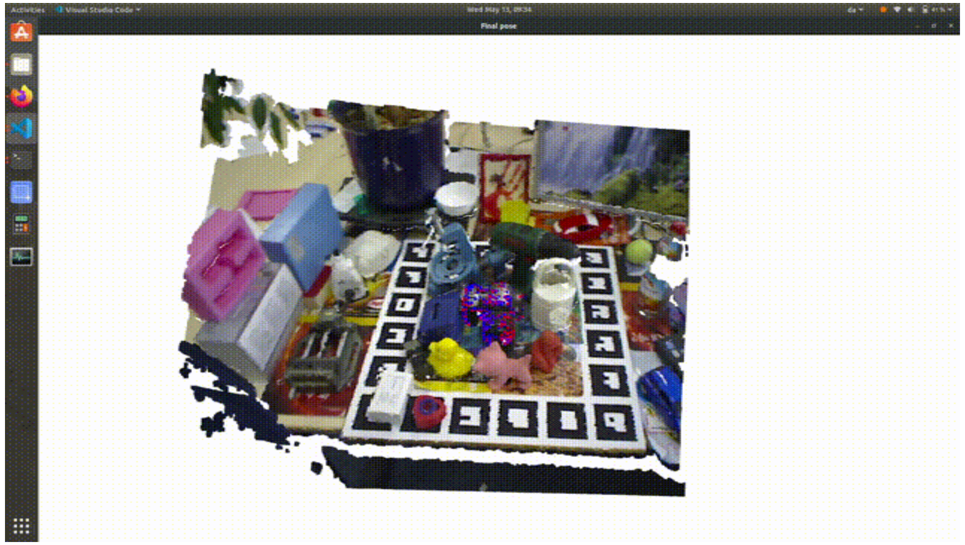


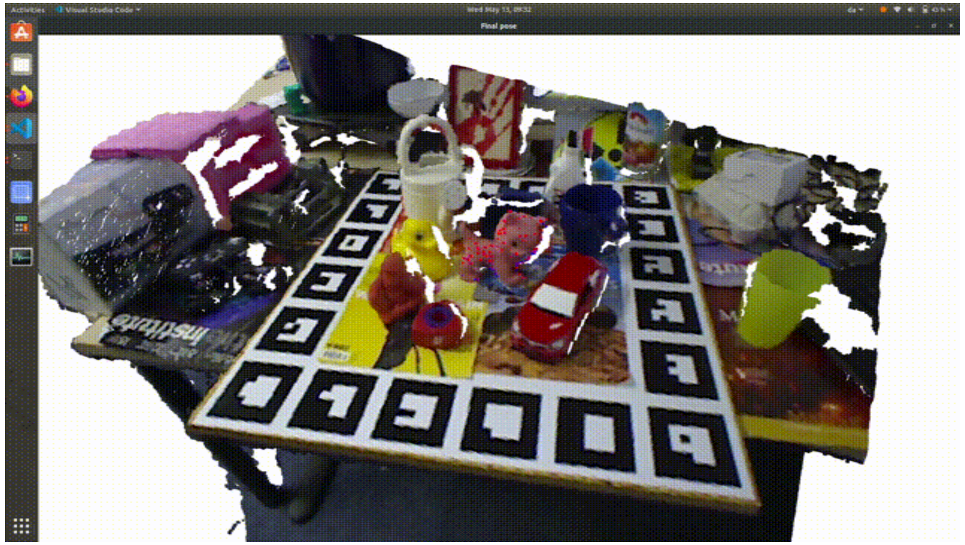
$$ADD\ score = \underset{x \in \mathcal{M}}{\text{average}} \left\| (Rx + T) - (\tilde{R}x + \tilde{T}) \right\|$$

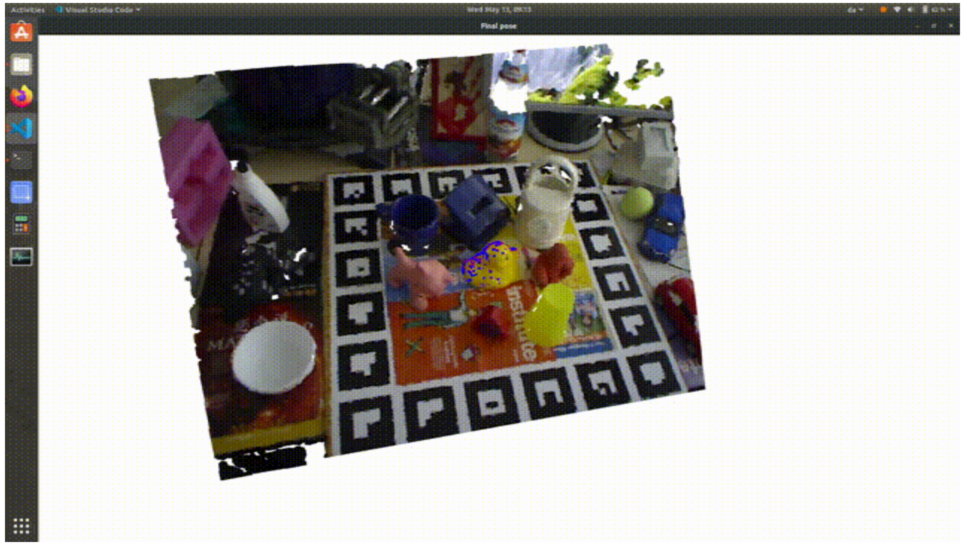
[12] Stefan Hinterstoisser et al. “Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes”

Visualization of Pose Estimation









Results of DenseFusion vs. 3D-3D

	DenseFusion		3D - 3D	
	Accuracy [%]	ADD/ADD-S [cm]	Accuracy [%]	ADD/ADD-S [cm]
Ape	93.71	0.005(±0.00603)	51.38	0.02457(±0.02217)
Bench vise	94.67	0.00767(±0.0146)	59.94	0.03895(±0.04337)
Can	97.45	0.00638(±0.00928)	44.61	0.04761(±0.04139)
Camera	94.69	0.00675(±0.0143)	52.17	0.04602(±0.04364)
Cat	96.71	0.00468(±0.00852)	57.78	0.02940(±0.03124)
Drill	88.31	0.0143(±0.0263)	65.01	0.03546(±0.04351)
Duck	93.90	0.00629(±0.0191)	46.49	0.02989(±0.03083)
Eggbox	99.81	0.00763(±0.0011)	99.81	0.00744(±0.002588)
Glue	99.81	0.00481(±0.00192)	99.90	0.00611(±0.002708)
Hole pouncher	94.01	0.00683(±0.0102)	37.68	0.04104(±0.03168)
Iron	98.47	0.00651(±0.0111)	41.47	0.07298(±0.05758)
Lamp	97.02	0.00638(±0.0143)	66.41	0.03337(±0.04200)
Phone	96.35	0.00693(±0.0121)	73.97	0.02137(±0.02907)
All	95.76	0.00693(±0.0133)	61.28	0.03510(±0.03240)

References

[4] W. Warren, B. Fajen, and D. Belcher. “Behavioral dynamics of steering, obstacle avoidance, and route selection”.
[5] Akshara Rai et al. “Learning coupling terms for obstacle avoidance”.
[7] Anthony A. Maciejewski and Charles A. Klein. “Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments”.
[8] Dae-Hyung Park et al. “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields”.
[9] Samuel R Buss. “Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods”
[12] Stefan Hinterstoisser et al. “Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes”
[13] Chen Wang et al. “DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion”.