Imperial College London

Time-Optimal Control of a Pseudo-Helicopter

Author: James Gross

CID: 01305321

Supervisor: Dr. Eric Kerrigan

Imperial College London

Aims

- Design model predictive controllers with a minimum time objective for the Quanser 3 DOF helicopter
- Compare the performance of the controllers

Objectives

- Improve and validate mathematical model of the helicopter
- Determine optimal path
- Design tracking and non-tracking controllers
- Investigate performance of the controllers in the absence and presence of disturbances

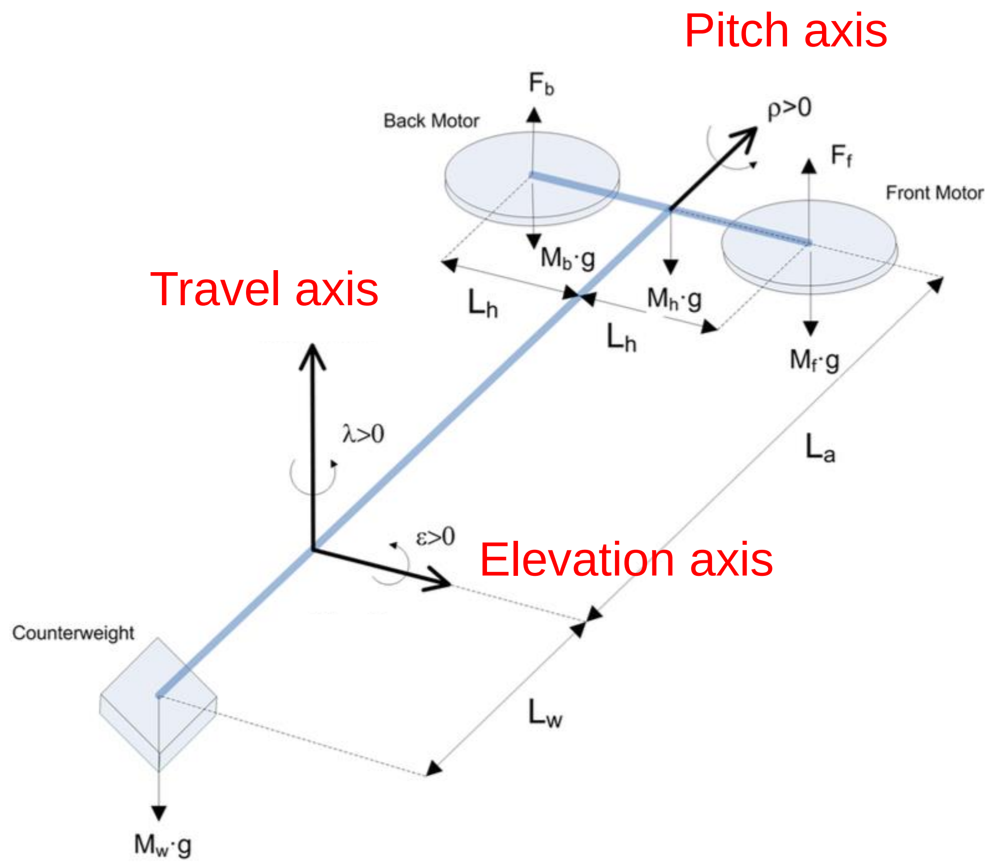
Imperial College London

Contents

Imperial College London

Introduction to the Pseudo-Helicopter

- 3 degrees of freedom: elevation, pitch, travel
- Highly coupled, nonlinear system
- Constraints on motor voltage, elevation, and pitch



Imperial College London

Introduction to the Pseudo-Helicopter

- Inputs:
  - Front motor voltage V_f
  - Back motor voltage V_b
- Outputs:
  - Elevation ϵ
  - Pitch ρ
  - Travel λ

Imperial College London

Mathematical Model and Model Validation

Euler-Lagrange equations:

$$\frac{\partial}{\partial q_i} L(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) - \frac{d}{dt}\frac{\partial}{\partial \dot{q}_i} L(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) = Q_i \text{ for } i = 1,2, \dots, n$$

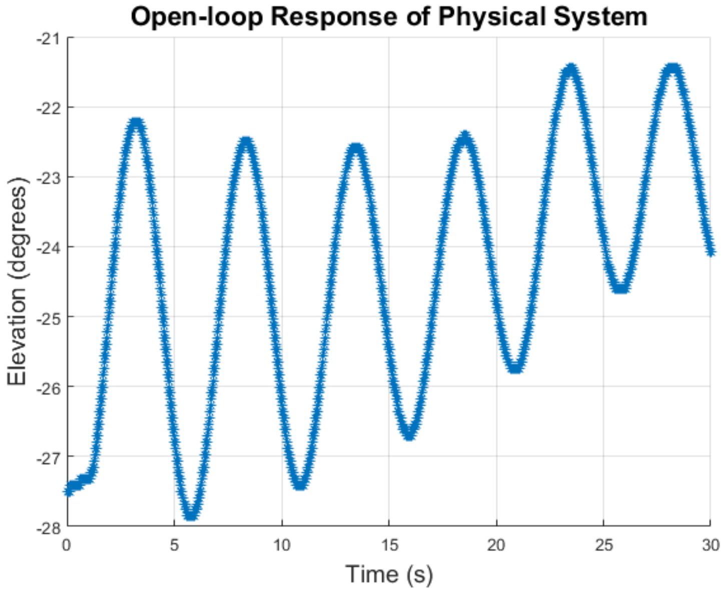
where $\boldsymbol{q} = [\epsilon \quad \rho \quad \lambda]^T$ and $L(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) = T - V$

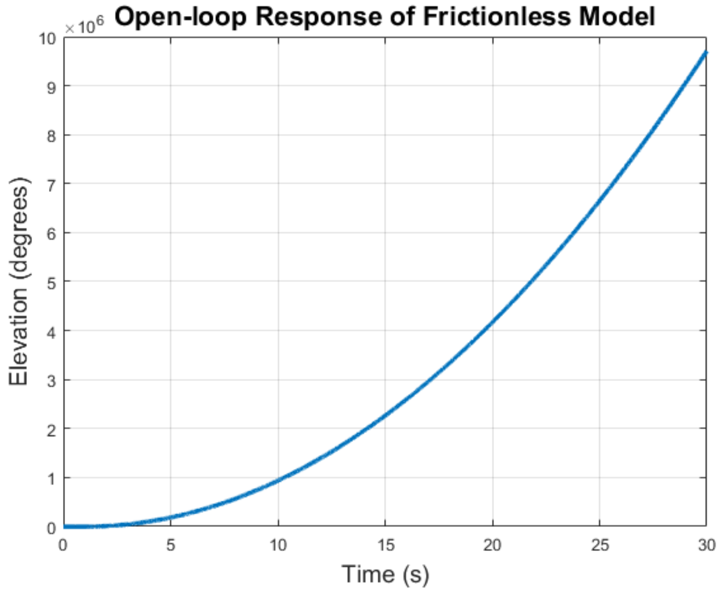
$$\rightarrow \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$$

where $\boldsymbol{x} = [\epsilon \quad \rho \quad \lambda \quad \dot{\epsilon} \quad \dot{\rho} \quad \dot{\lambda}]^T$ and $\boldsymbol{u} = [V_f \quad V_b]^T$

Imperial College London

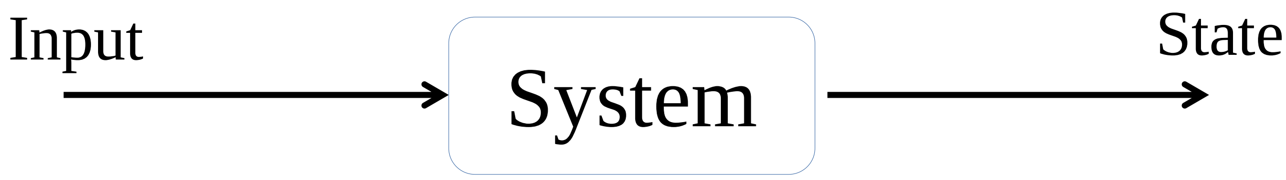
Mathematical Model and Model Validation



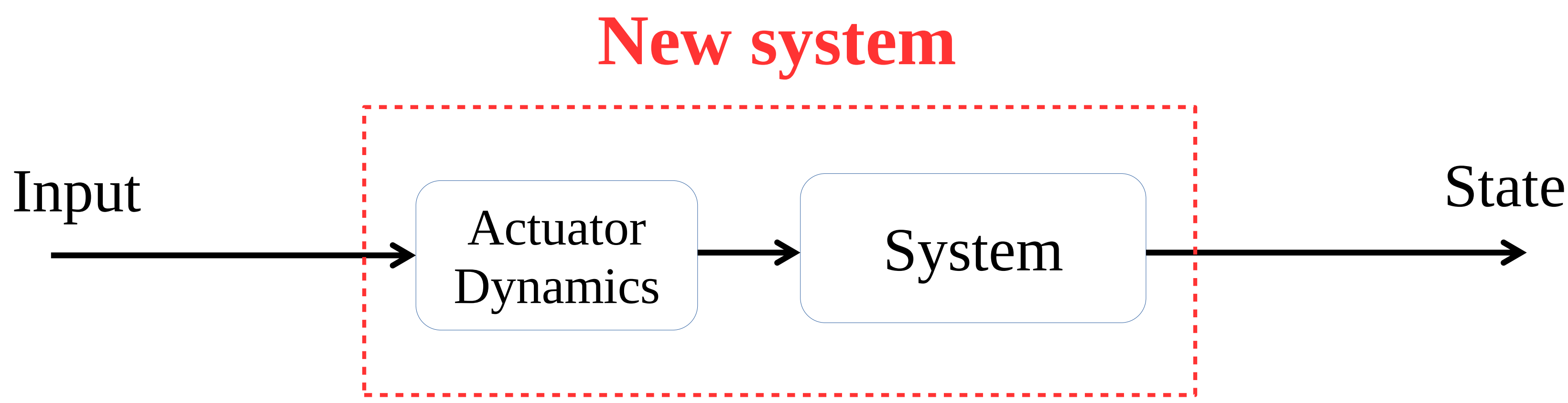


Imperial College London

Mathematical Model and Model Validation



No Actuator Lag



First Order Actuator Lag

Mathematical Model and Model Validation

$$\tilde{Q}_i = Q_i + F_{v_i}\dot{q}_i + F_{c_i}sign(q_i)$$

Mechanical damping

Kinetic friction

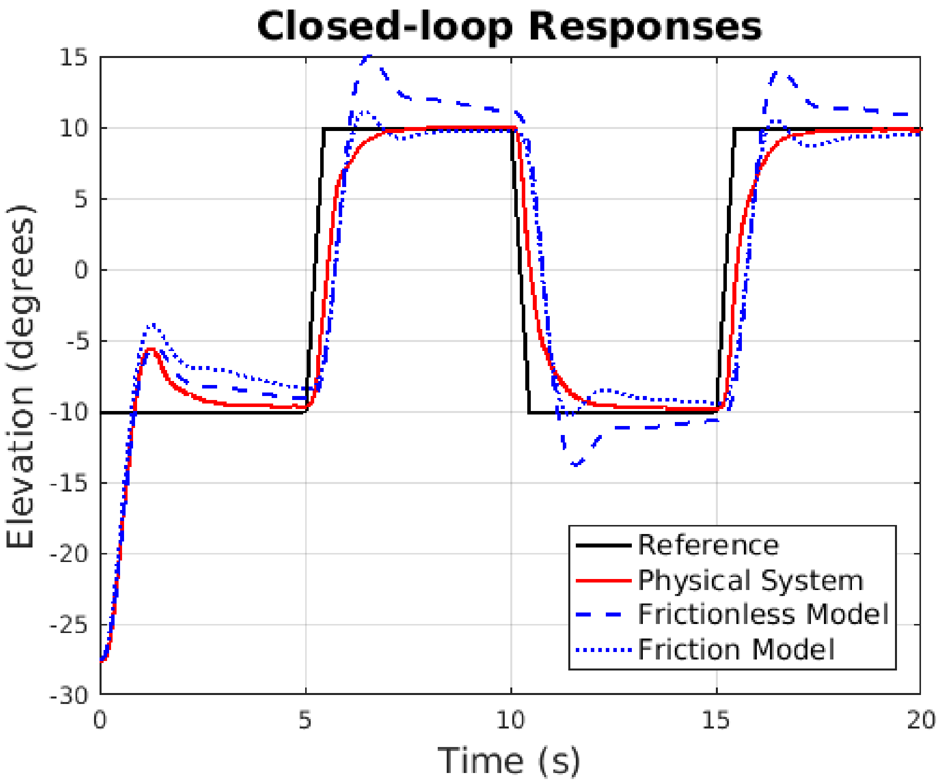
- Friction in pitch and travel directions negligible
- Simple friction model preferred
- $sign$ function replaced with smooth approximation

Imperial College London

Mathematical Model and Model Validation

- Open-loop:
  - RMS deviation of each model from experimental data showed a 533% improvement for the friction model

- Closed-loop:
  - Compared closed-loop responses of the models and the physical system using same LQR gains matrix



Optimal Path Planning

- Objective: get from start point to end point in minimum time

- Start point:

$$x_0 = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$$

- End point:

$$x_f \in \begin{bmatrix} 0 & 0 & \pi & 0 & 0 & 0 & -\infty & -\infty \\ 0 & 0 & \pi & 0 & 0 & 0 & \infty & \infty \end{bmatrix}$$

- Cost function:

$$V(x, u, t) = t_f$$

- Hard constraints:

$$\epsilon \in [-27.5°, 27.5°]$$

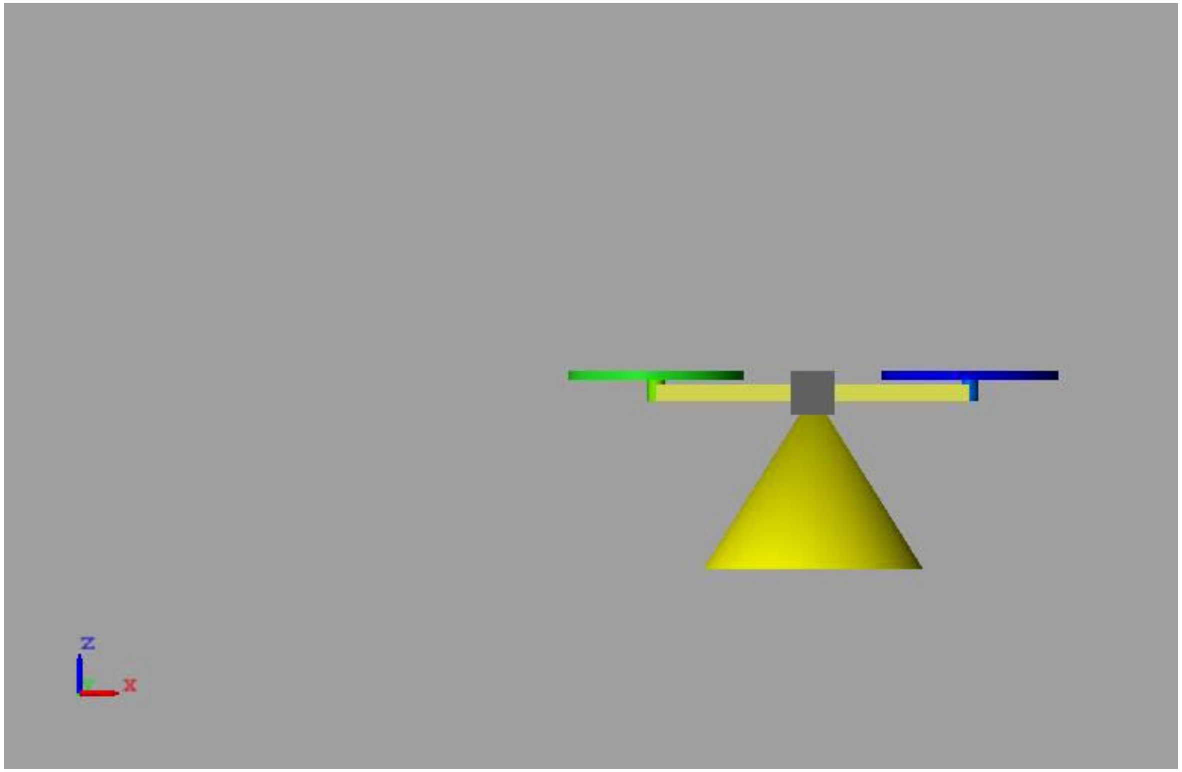
$$\rho \in [-90°, 90°]$$

$$\boldsymbol{u} \in \begin{bmatrix} -24 & 24 \\ -24 & 24 \end{bmatrix}$$

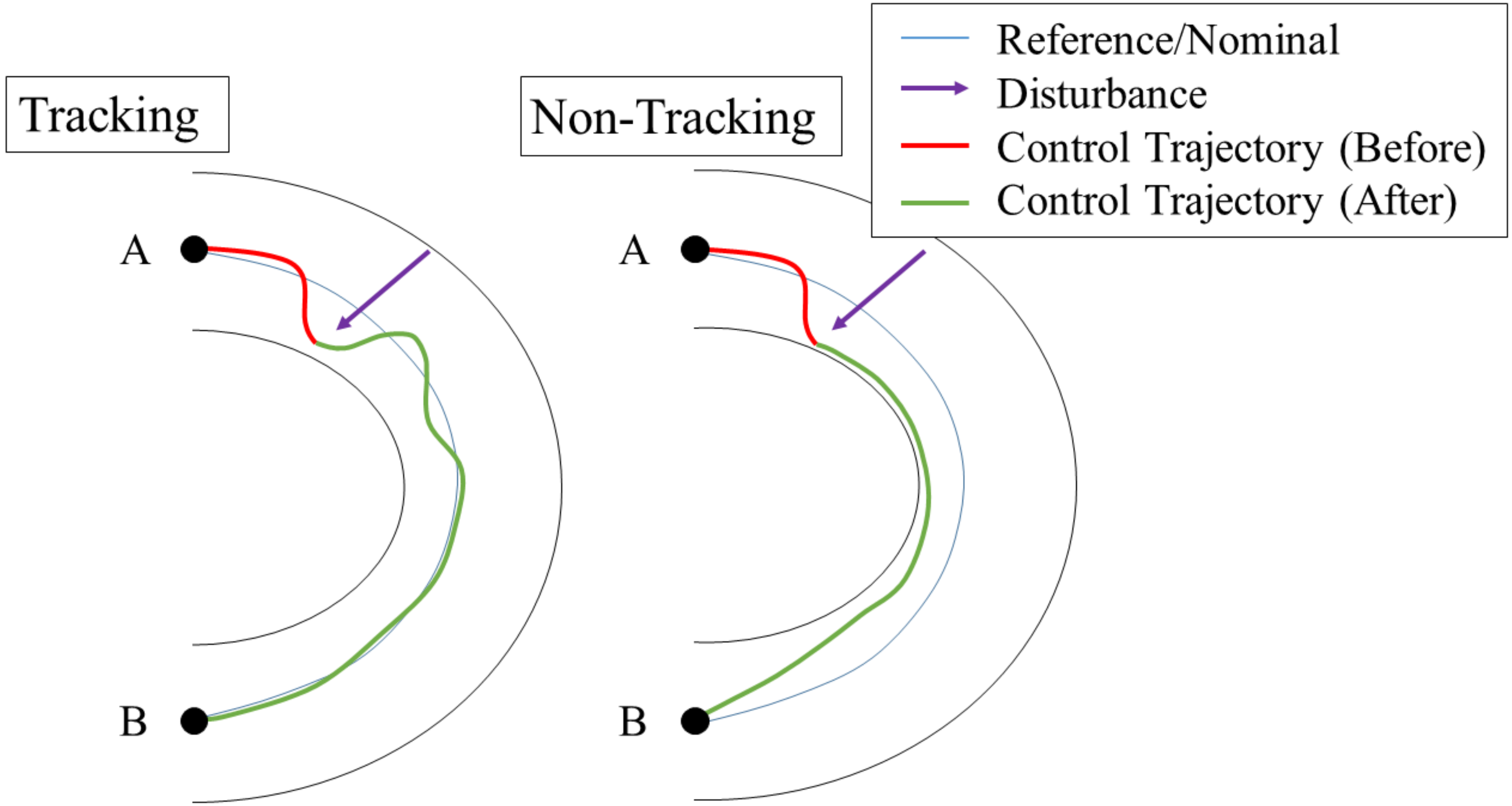
Imperial College
London

Optimal Path Planning

$t_f = 2.1$ **seconds**



Tracking vs Non-Tracking



Imperial College London

Tracking vs Non-Tracking

Linear tracking receding horizon controller:

- Objective: track the optimal path using a linear state space model as the predictive controller

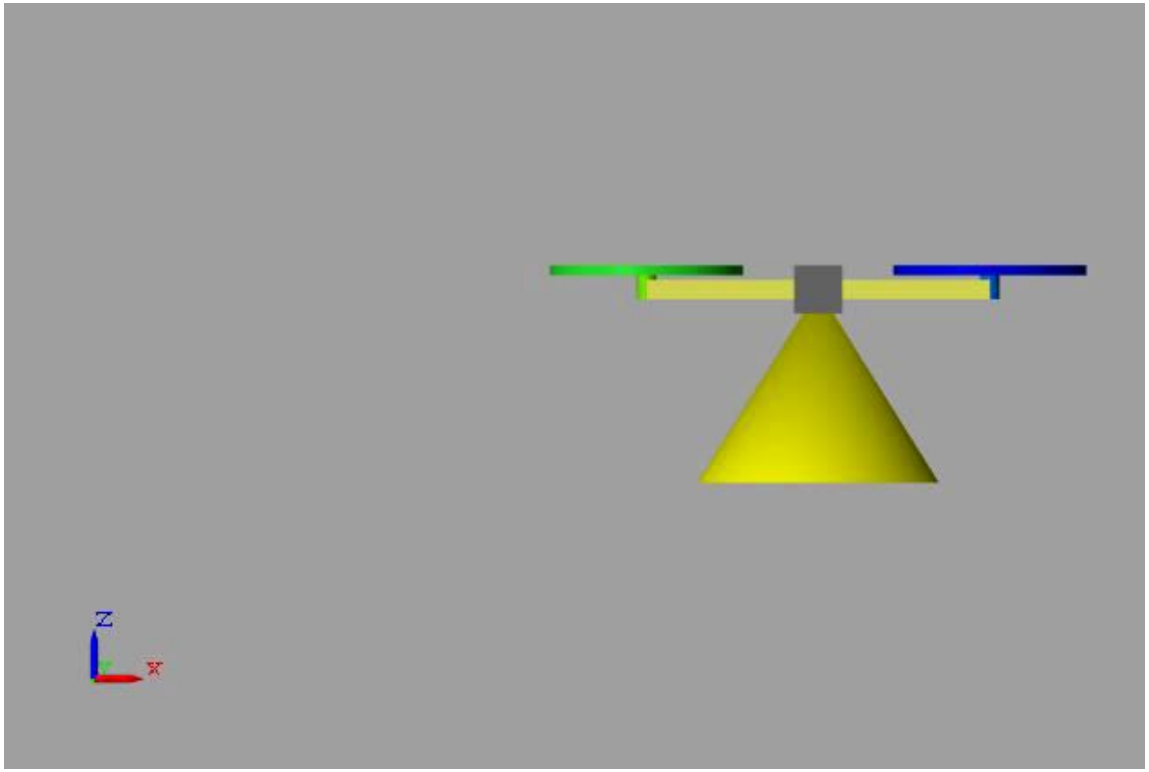
- Cost function:

$$V(\boldsymbol{x}, \boldsymbol{u}, t) = \int_{t_{k_0}}^{t_{k_f}} \|\boldsymbol{y} - \boldsymbol{y}_r\|^2 dt$$

Imperial College London

Tracking vs Non-Tracking

$$\left\| \boldsymbol{y_f - y_{des}} \right\|_2 = \mathbf{1.124}$$



Imperial College London

Tracking vs Non-Tracking

Nonlinear non-tracking variable horizon controller:

- Objective: get from start point to end point in the minimum time possible using nonlinear frictionless model as predictive plant

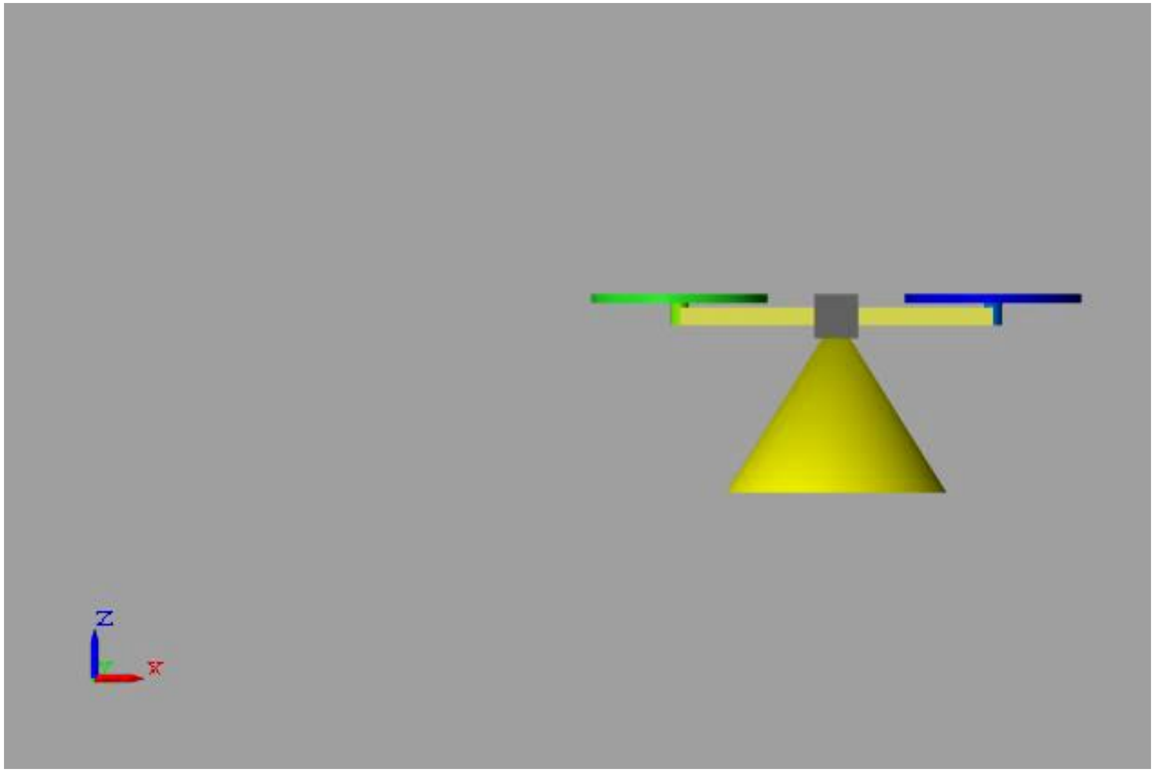
- Cost function:

$$V(\boldsymbol{x}, \boldsymbol{u}, t) = t_f$$

Imperial College London

Tracking vs Non-Tracking

$$\left\| y_f - y_{des} \right\|_2 = 0.646$$



Disturbance Handling

- White noise disturbance:
  - Added Gaussian white noise with signal to noise ratio 2
  - $\dot{x} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\xi}(t)$
- Bias disturbance:
  - Added disturbance in the travel direction
  - $\ddot{\lambda} = f_\lambda(x, u) + \xi(t)$

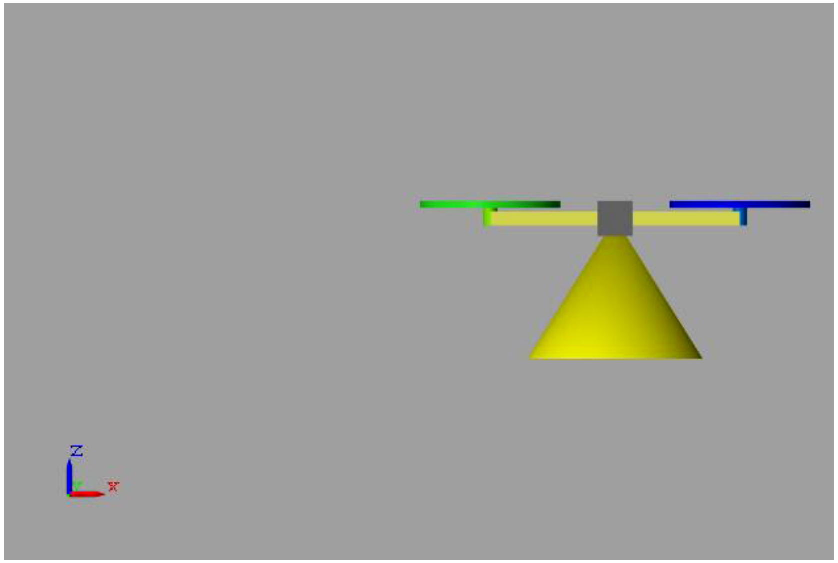
$$\xi(t) = \begin{cases} 2 & if\ t < 0.2 \\ 0 & otherwise \end{cases}$$

Imperial College London

White noise disturbance

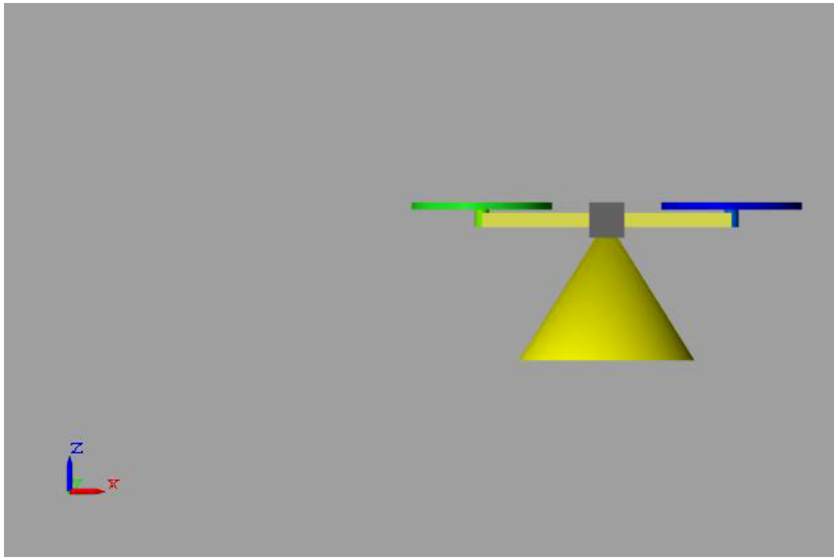
Disturbance Handling

Tracking



$\|y_f - y_{des}\|_2 = 2.539$

Non-Tracking



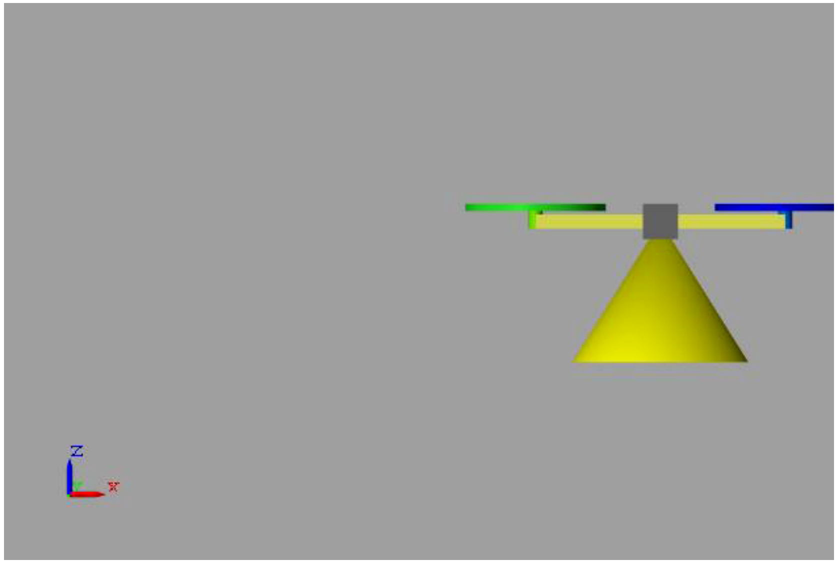
$\|y_f - y_{des}\|_2 = 0.7718$

Imperial College London

Bias disturbance

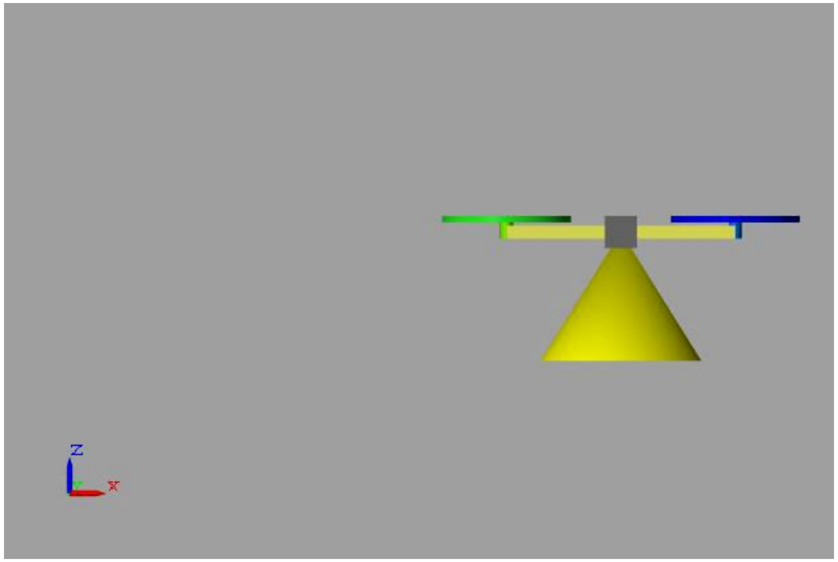
Disturbance Handling

Tracking



$$\left\|y_f - y_{des}\right\|_2 = \mathbf{0.868}$$

Non-Tracking



$$\left\|y_f - y_{des}\right\|_2 = \mathbf{0.336}$$

Future Work

- Reduce computational time, possibly by parallelising the code or calculating multiple inputs each iteration
- Reduce the effects of plant-model mismatch by either introducing an input disturbance model or by using integral action
- Improve non-tracking controller performance by switching to a minimum energy cost function when in the vicinity of the desired end point

Imperial College London

Conclusions

- Modified the nonlinear model by adding previously ignored dynamics, improving open-loop performance by 533%
- Determined the optimal path for a minimum time control objective
- Designed tracking and non-tracking controllers for the control objective
- Compared the performance of the controllers without disturbance, with non-tracking offering 74% better performance
- Compared the performance of the controllers with disturbance:

	Tracking	Non-tracking
White Noise	126% reduction	20% reduction
Bias	23% improvement	48% improvement