

Robust Neural Control Design for Multi-drone Slung Payload Manipulation with Control Contraction Metrics

Yi Lok Lo, Xinyuan Liang, Longhao Qian, Hugh H. T. Liu

University of Toronto Institute for Aerospace Studies (UTIAS)

Motivation & Approach

MODERN developments in cable-suspended payload transportation using multirotors present various challenges related to system performance, stability, and safety. However, compared to a single-agent slung payload system, a multi-drone design offers a more scalable solution with better range, higher payload capacity, and additional redundancy.

- A robust non-linear control scheme using neural control contraction metric (N-CCM) for a three-drone, point-mass slung-payload system is proposed.
- A UDE to compensate for persistent external disturbances. We show that the UDE compensator provides a bounded and converging disturbance estimation error.
- The control law inherits bounded control output to satisfy control saturation constraints while guaranteeing the stability of the system.

System Geometry

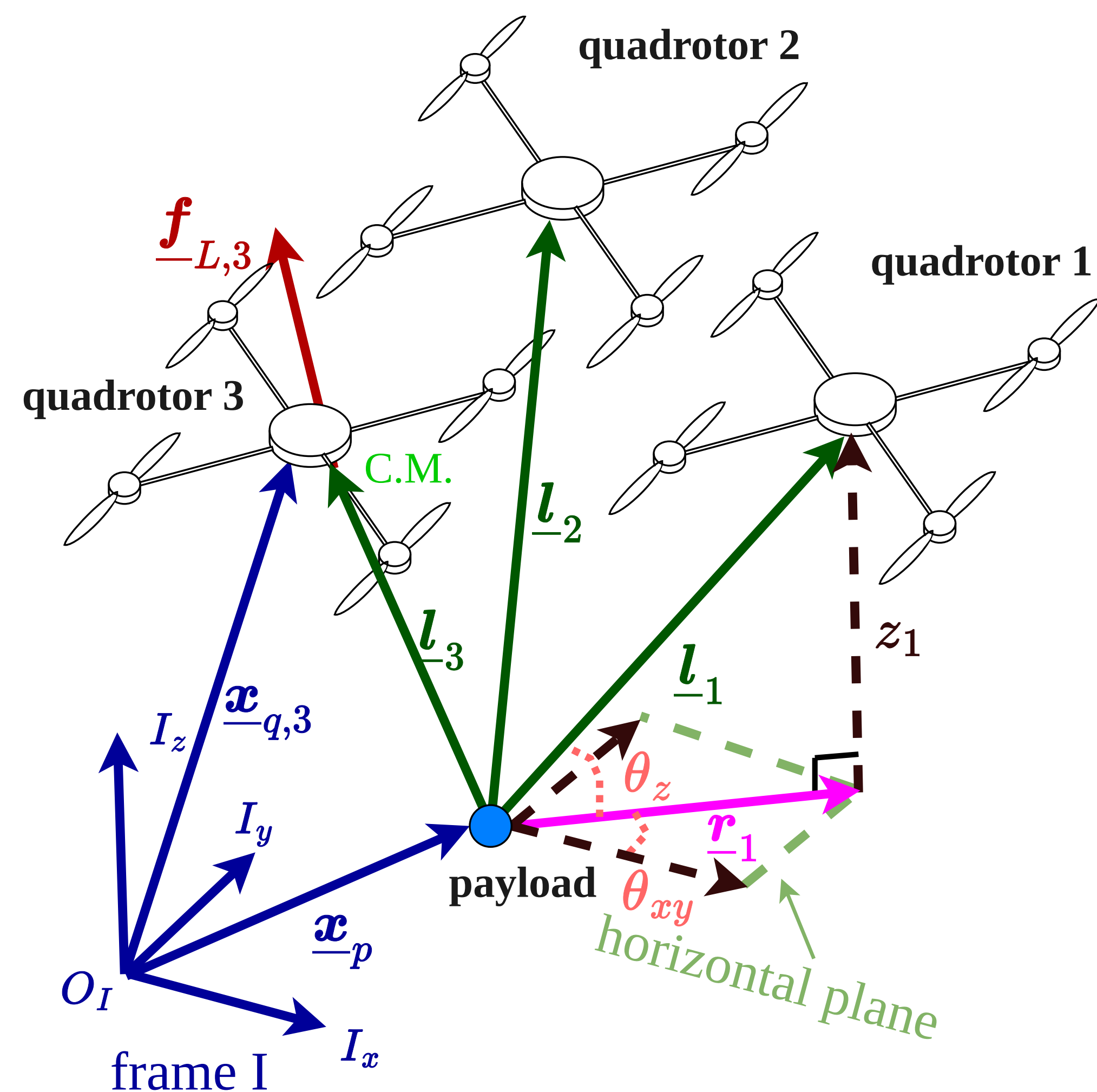


Figure 1: System geometry.

System Model

A point-mass slung payload with mass m_p is carried by three quadrotors with position $\mathbf{x}_p$ in the inertial frame, each producing a three-dimensional (3D) lift force $\mathbf{f}_{L,j}$. The mass of each quadrotor is m_j with position $\mathbf{x}_{q,j}$ in the inertial frame, $j = 1, 2, 3$. The cable vector defined in frame I (inertial frame) is $\mathbf{l}_j \in \mathbb{R}^3$, with equal length $l = \|\mathbf{l}_j\|$. Each cable forms a horizontal projection $\mathbf{r}_j$, the vertical and horizontal angles to this projection are θ_z and θ_{xy} . The total payload system with velocity vector $\mathbf{u} = [\mathbf{v}_p^T \mathbf{v}_1^T \mathbf{v}_2^T \mathbf{v}_3^T]^T$ and the full state $\mathbf{x} = [\mathbf{x}_p^T \mathbf{r}_1^T \mathbf{r}_2^T \mathbf{r}_3^T \mathbf{u}^T]^T$ evolves according to:

$$\Sigma_p : \begin{cases} \mathbf{M}\dot{\mathbf{u}} + \mathbf{C}\mathbf{u} = \mathbf{f}_{g,p} + \mathbf{H}\boldsymbol{\zeta} + \mathbf{H}_\delta\boldsymbol{\delta} \\ \dot{\mathbf{x}}_p = \mathbf{v}_p \\ \dot{\mathbf{r}}_j = \mathbf{v}_j \end{cases} \quad (1)$$

where $\mathbf{v}_p$ is the payload velocity, $\boldsymbol{\zeta}$ is the control input, and $\boldsymbol{\delta}$ is the disturbance vector. Given that the total mass of the system is $m_t = m_p + m_1 + m_2 + m_3$, the system matrices are the following:

$$\begin{aligned} \mathbf{M} &= \begin{bmatrix} m_t \mathbf{1}_{3 \times 3} & \text{hstack}(m_j \mathbf{B}_j) \\ \text{vstack}(m_j \mathbf{B}_j^T) & \text{diag}(m_j \mathbf{B}_j^T \mathbf{B}_j) \end{bmatrix}, \\ \mathbf{C} &= \begin{bmatrix} \mathbf{0}_{3 \times 3} & \text{hstack}(m_j \dot{\mathbf{B}}_j) \\ \mathbf{0}_{6 \times 3} & \text{diag}(m_j \mathbf{B}_j^T \dot{\mathbf{B}}_j) \end{bmatrix}, \quad \mathbf{f}_{g,p} = \begin{bmatrix} m_p \mathbf{g}_I \\ \mathbf{0}_{6 \times 1} \end{bmatrix}, \\ \mathbf{H} &= \begin{bmatrix} \text{hstack}(\mathbf{1}_{3 \times 3}) \\ \text{diag}(\mathbf{B}_j^T) \end{bmatrix}, \quad \boldsymbol{\zeta} = [\text{vstack}(\delta \mathbf{f}_{L,j})], \\ \mathbf{H}_\delta &= \begin{bmatrix} \mathbf{1}_{3 \times 3} & \text{hstack}(\mathbf{1}_{3 \times 3}) \\ \mathbf{0}_{6 \times 3} & \text{diag}(\mathbf{B}_j^T) \end{bmatrix}, \quad \boldsymbol{\delta} = \begin{bmatrix} \delta_p \\ \text{vstack}(\delta_j) \end{bmatrix}. \end{aligned} \quad (2)$$

After this manipulation, we calculated that:

$$\mathbf{G}(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{9 \times 9} \\ \mathbf{M}^{-1} \mathbf{H} \end{bmatrix}, \quad \mathbf{G}_\delta(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{9 \times 12} \\ \mathbf{M}^{-1} \mathbf{H}_\delta \end{bmatrix}, \quad (3)$$

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{u} \\ \mathbf{M}^{-1}(\mathbf{f}_{g,p} - \mathbf{C}\mathbf{u}) \end{bmatrix}.$$

The final control-affine system is given below:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\boldsymbol{\zeta} + \mathbf{G}_\delta(\mathbf{x})\boldsymbol{\delta} \quad (4)$$

Neural Robust Control with CCM

The CCM-based control law is realized by training neural networks representing $\boldsymbol{\zeta}_{nn}$ and $\mathbf{P}(\mathbf{x}, t)$ to satisfy differential stability conditions simultaneously.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\boldsymbol{\zeta}_{nn}, \quad (5)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state and $\boldsymbol{\zeta}_{nn} \in \mathbb{R}^m$ is the control input. A smooth control law can be found as

$$\boldsymbol{\zeta}_{nn} = \mathbf{k}(\mathbf{x}, \mathbf{x}^*, \mathbf{k}^*; \theta_{K_1}, \theta_{K_2}), \quad (6)$$

$$\dot{\mathbf{P}} + \text{sym}(\mathbf{P}(\mathbf{A} + \mathbf{G}(\mathbf{x})\mathbf{K})) + 2\lambda\mathbf{P} \prec \mathbf{0}, \quad (7)$$

$$\mathbf{W} - w \cdot \mathbf{1} \prec \mathbf{0}, \quad (8)$$

where $\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} + \sum_{i=1}^m \frac{\partial \mathbf{g}_i}{\partial \mathbf{x}} \boldsymbol{\zeta}_{nn,i}$, $\mathbf{K} = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}$, and $\mathbf{g}_i$ is the i^{th} column of $\mathbf{G}$, $\boldsymbol{\zeta}_{nn,i}$ is the i^{th} element of $\boldsymbol{\zeta}_{nn}$. w is a positive constant that represents the largest eigenvalue of $\mathbf{W}$. The dual conditions for training are:

$$\mathbf{G}_{ann}^T \left(-\frac{\partial \mathbf{W}}{\partial \mathbf{x}} \mathbf{f} + \text{sym} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \mathbf{W} \right) + 2\lambda \mathbf{W} \right) \mathbf{G}_{ann} \prec \mathbf{0}, \quad (9)$$

$$\mathbf{G}_{ann}^T \left(\frac{\partial \mathbf{W}}{\partial \mathbf{x}} \mathbf{g}_i - \text{sym} \left(\frac{\partial \mathbf{g}_i}{\partial \mathbf{x}} \mathbf{W} \right) \right) \mathbf{G}_{ann} = \mathbf{0}, i = 1, \dots, m. \quad (10)$$

The output control signal after the hard control constraints is:

$$\boldsymbol{\zeta}_{nn,sat} = \tanh(a \cdot \mathbf{k}(\mathbf{x}, \mathbf{x}^*, \mathbf{0}; \theta_{K_1}, \theta_{K_2})) \cdot \mathbf{f}_b + \mathbf{k}^*. \quad (11)$$

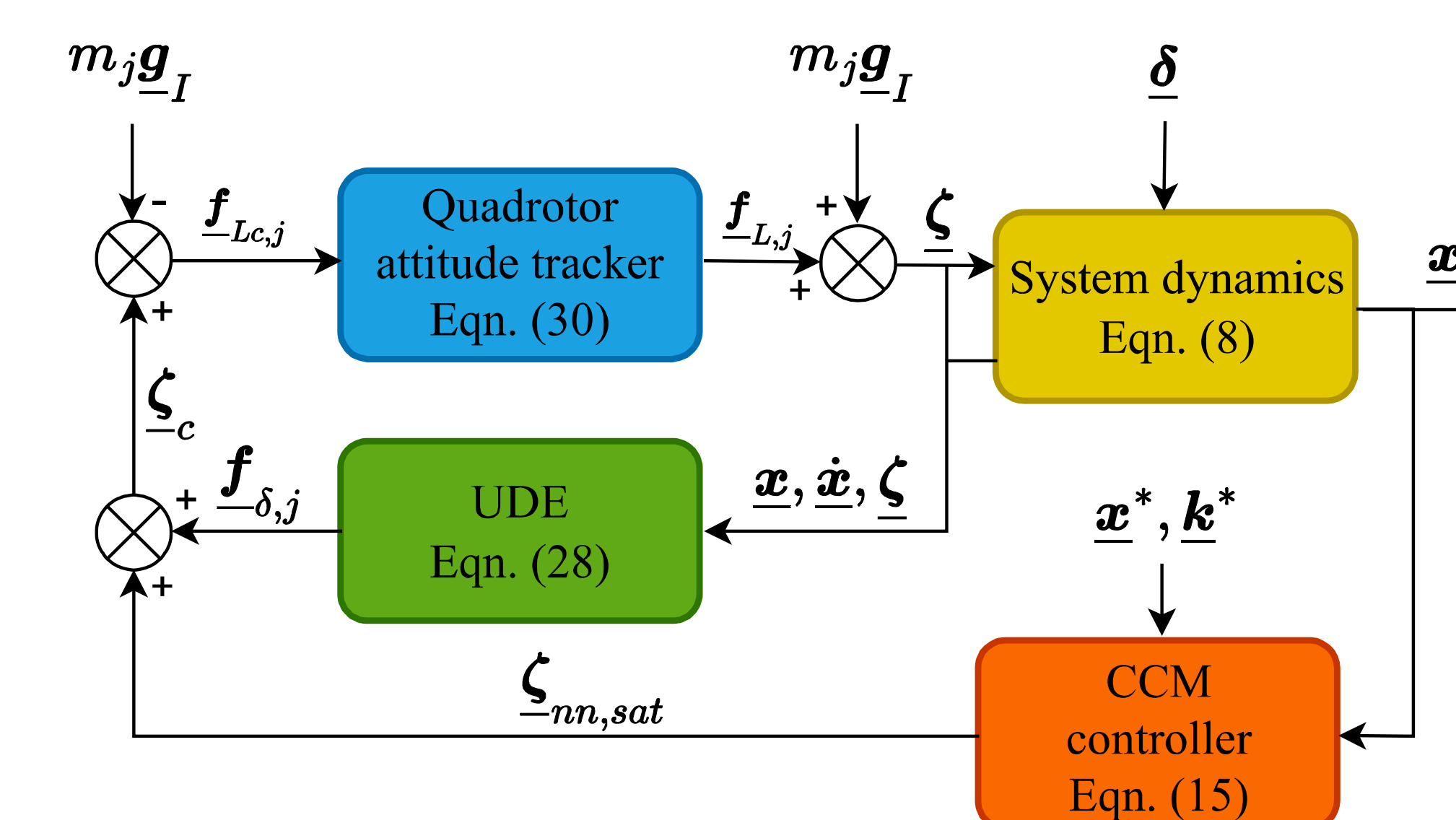


Figure 2: The complete closed-loop system.

Results

The performance of the figure-8 trajectory tracking is demonstrated. The disturbance force is a summation of constant and stochastic noise $\boldsymbol{\delta} = \boldsymbol{\delta}_c + \boldsymbol{\delta}_s$, where $\boldsymbol{\delta}_c = [0.3, -0.2, 0.5, 0.3, \dots, 0.3]^T \in \mathbb{R}^{12 \times 1}$ and $\boldsymbol{\delta}_s \sim 0.3 \cdot \mathcal{U}(0, 1)$ is uniformly distributed.

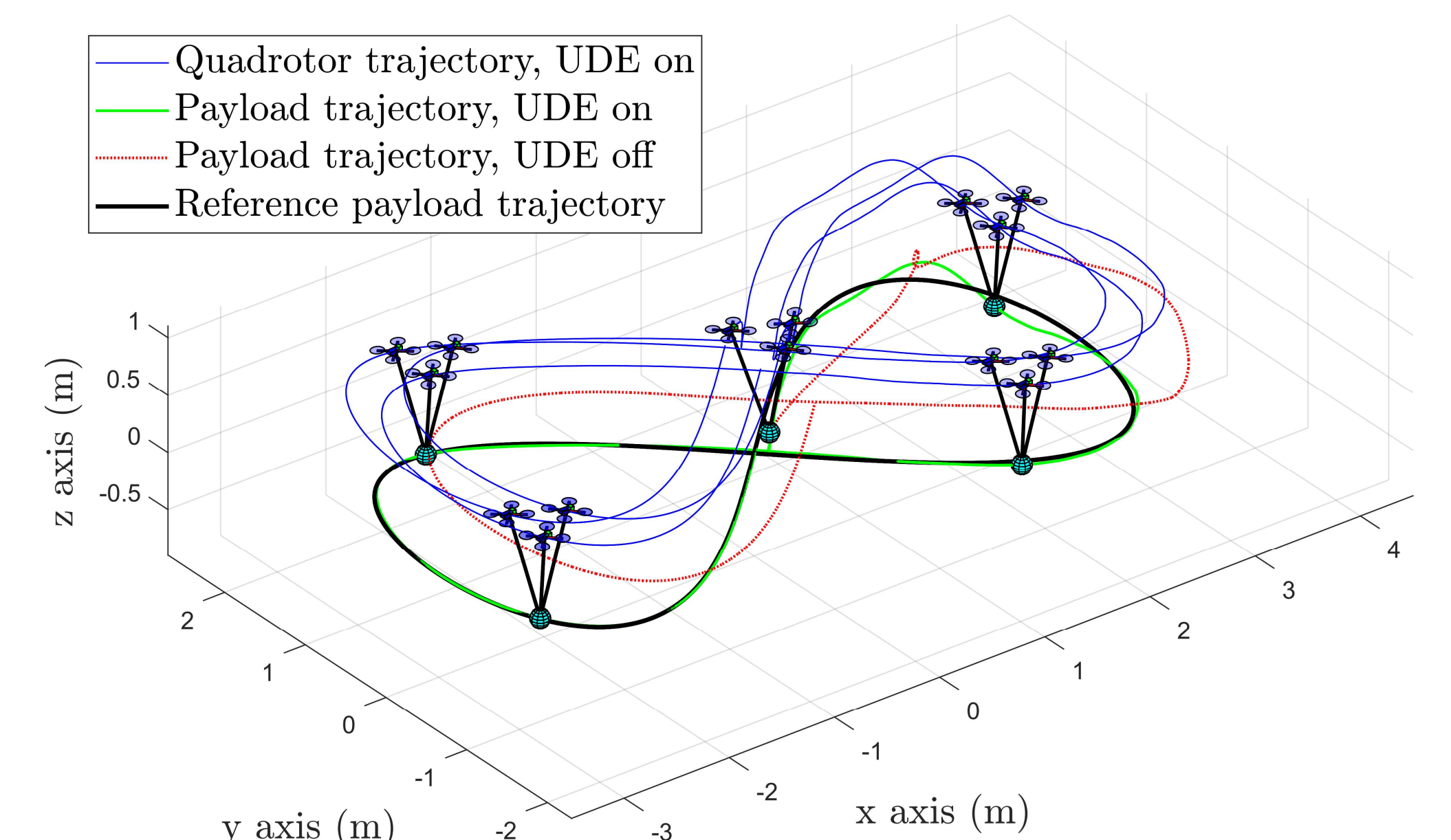


Figure 3: Trajectory tracking results.

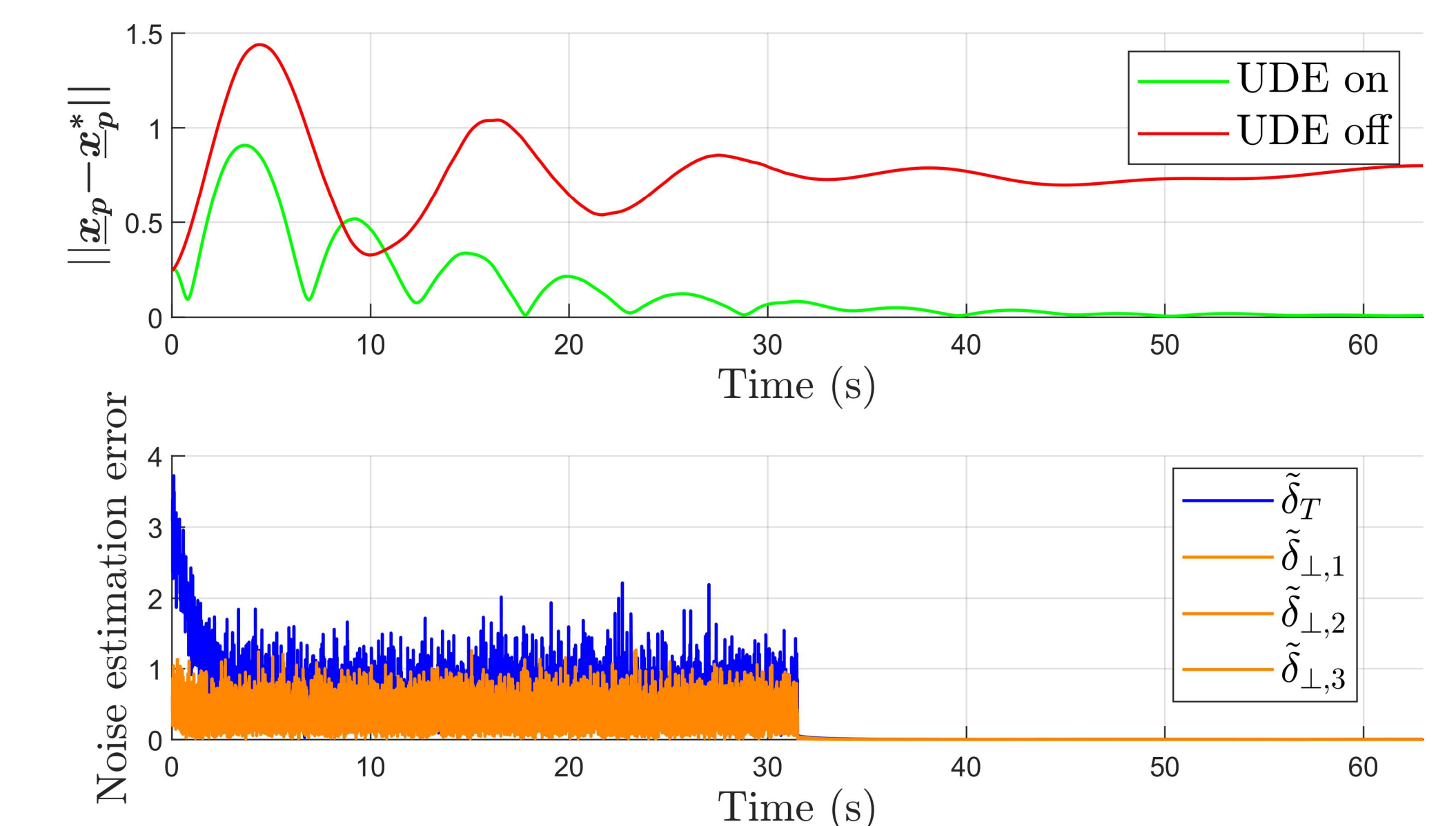


Figure 4: Payload trajectory tracking errors and UDE estimation errors.

Conclusions & Future Work

An extensive derivation of the dynamics, contraction metric, and disturbance estimation is provided. Stability and robustness are proved, with results illustrated by numerical simulations. Future work will focus on physical experiments and state constraints of the contraction metric.