NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

Parameter Estimation and Control of Multirotors Using Integral Concurrent Learning

Cheng-Cheng Yang and Teng-Hu Cheng

Speaker : Cheng-Cheng Yang
Date : 2020/12/24

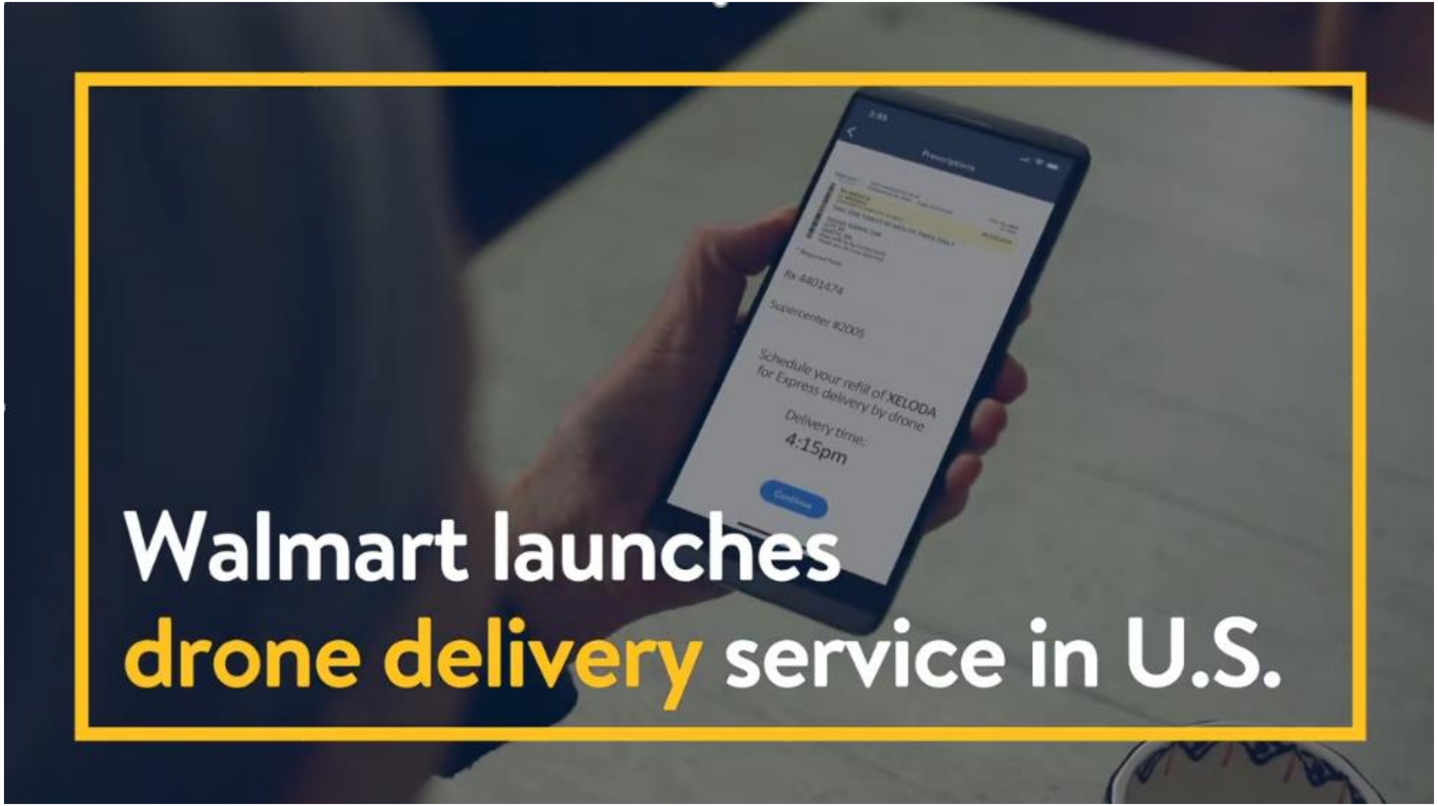
Outlines

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Motivation
- Problem Formulation
- Controller Design
- Stability Analysis
- Simulation and Experiments
- Conclusion

Motivation – from Walmart's drone delivery

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室



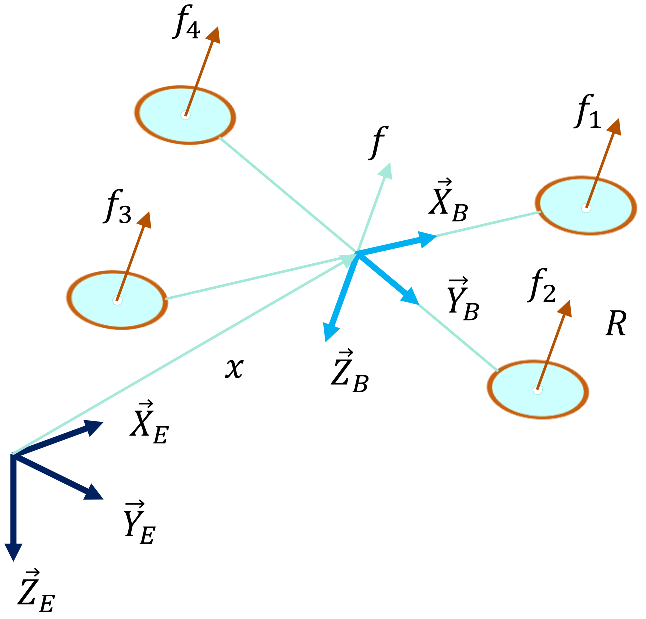
Motivation – from movies Angel Has Fallen

Motivation

- Knowledge of the geometric and inertia parameters is essential to achieving good control performance.
- The payload or sensors attaching to multirotors may change the geometric and inertia parameters.
- Some geometric and inertia parameters like moment of inertia can not be measured through instrument.
- Existing adaptive control method can only guarantee the stability of multirotors system, can not ensure the parameters converge.

Problem Formulation - Definition of Symbols

Symbol	Description
x	Position of the multirotor
v	Velocity of the multirotor
R	Rotation matrix from the body-fixed frame to the inertial frame
Ω	Angular velocity in the body-fixed frame
f	Net thrust control input
M	Moment control input
m	Mass of the multirotor
J	Moment of inertia of the multirotor



Body-fixed frame : $\{\vec{X}_B, \quad \vec{Y}_B, \quad \vec{Z}_B\}$

Inertial frame: $\{\vec{X}_E, \quad \vec{Y}_E, \quad \vec{Z}_E\}$

Problem Formulation - Dynamics of the Multirotor

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- The multirotor is described by both translational and rotational dynamics.
- The translational dynamics considers forces such as the effects of gravity, thrusts, and the external force.
- The rotational dynamics takes the moment of the control input, rotational speed, and moment of inertia into account.

$$\left.\begin{aligned} \dot{x} &= v \\ m\dot{v} &= mge_3 - fRe_3 \end{aligned}\right\} \text{Translational dynamics}$$

$$\left.\begin{aligned} \dot{R} &= R\hat{\Omega} \\ J\dot{\Omega} + \Omega \times J\Omega &= M \end{aligned}\right\} \text{Rotational dynamics}$$

$$J = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}$$

Problem Formulation – Tracking Errors and Estimate Errors

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Position and velocity tracking errors

  $e_x \triangleq x - x_d$

  $e_v \triangleq v - v_d$
- Attitude error function on SO(3) based on Geometric Tracking Control

  $\Psi(R, R_d) \triangleq \frac{1}{2} tr\left[I - R_d^T R\right]$
- Attitude tracking error and the angular velocity tracking error

  $e_R \triangleq \frac{1}{2}\left(R_d^T R - R^T R_d\right)^{\vee}$

  $e_\Omega \triangleq \Omega - R^T R_d \Omega_d$
- Estimate error of mass

  $\tilde{\theta}_m \triangleq \theta_m - \hat{\theta}_m$, $\theta_m = m$ (mass of the multirotor)
- Estimate error of moment of inertia

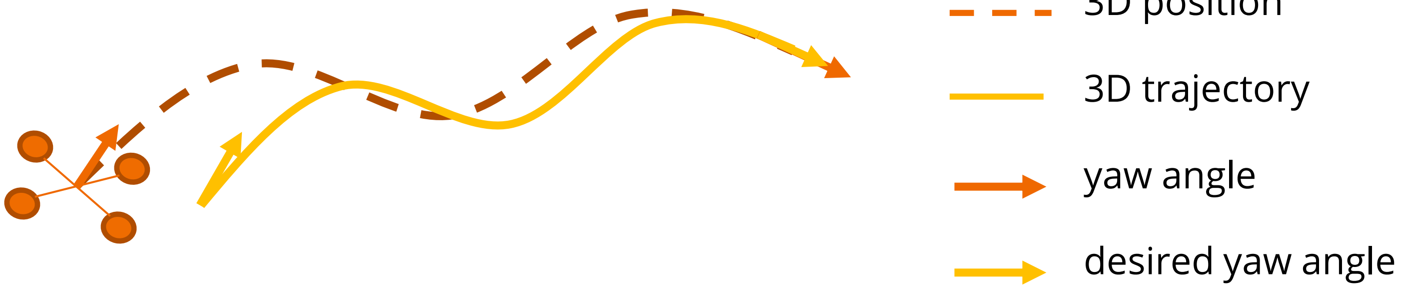
  $\tilde{\theta}_{diag} \triangleq \theta_{diag} - \hat{\theta}_{diag}$, $\theta_{diag} = [J_{xx} \quad J_{yy} \quad J_{zz}]^T$ (moment of inertia of the multirotor)

Problem Formulation – Control Objectives

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Track a desired 3D trajectory
- Track a desired yaw angle
- Estimate the mass of the multirotor
- Estimate the moment of inertia of the multirotor

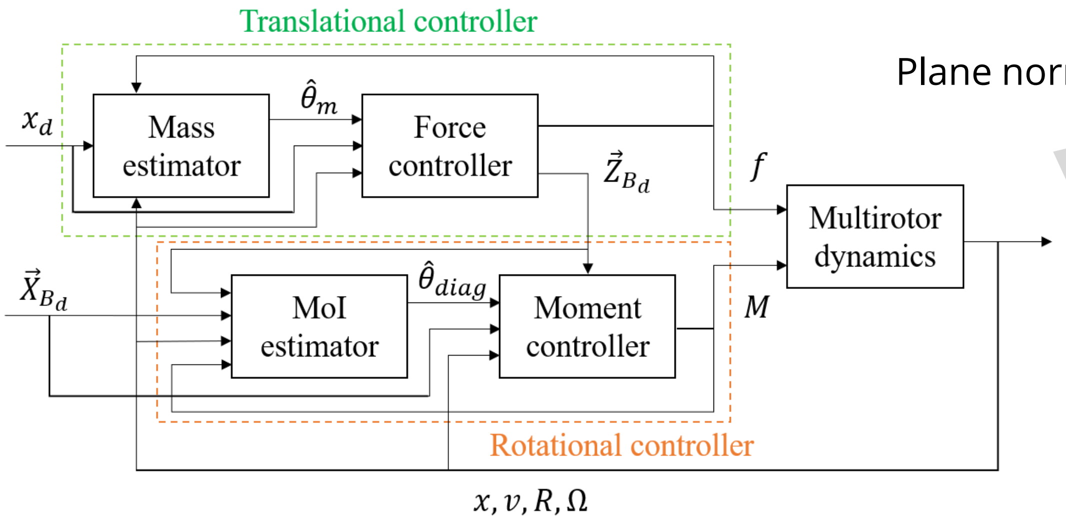
$$\begin{cases} e_x & \to 0 \\ e_v & \to 0 \\ e_R & \to 0 \\ e_\Omega & \to 0 \\ \tilde{\theta}_m & \to 0 \\ \tilde{\theta}_{diag} & \to 0 \end{cases} \quad as\ t \to \infty$$

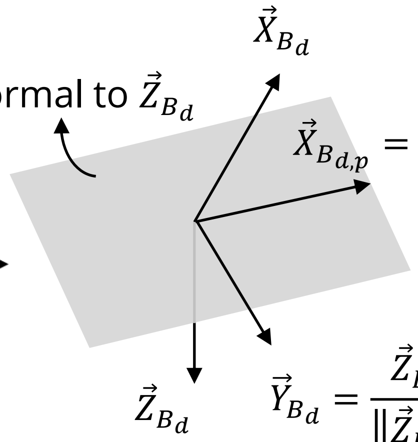


Motivation
Problem Formulation
Controller Design
Stability Analysis
Simulation and Experiments
Conclusion

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

Controller Design – Control Architecture





$$\vec{X}_{B_{d,p}} = \vec{Y}_{B_d} \times \vec{Z}_{B_d}$$

$$\vec{Y}_{B_d} = \frac{\vec{Z}_{B_d} \times \vec{X}_{B_d}}{\left\|\vec{Z}_{B_d} \times \vec{X}_{B_d}\right\|}$$

$$R_d = \left[\vec{X}_{B_{d,p}};\quad \vec{Y}_{B_d};\quad \vec{Z}_{B_d}\right] \in R^{3\times 3}$$

Controller Design – Translational Controller

- Translational controller

$$f = \underbrace{(k_x e_x + k_v e_v}_{} + \underbrace{Y_m \hat{\theta}_m}_{}) \cdot Re_3 \qquad , Y_m = \begin{bmatrix} -\ddot{x}_{d_1} \\ -\ddot{x}_{d_2} \\ g - \ddot{x}_{d_3} \end{bmatrix}$$ is a regression matrix

$$= \boldsymbol{feedback\ term} + \boldsymbol{adaptive\ term}$$

- Integral CL-based adaptive control update law $\dot{\hat{\theta}}_m$

$$\dot{\hat{\theta}}_m = \underbrace{\Gamma_m Y_m^T(e_v + C_1 e_x)}_{} + \underbrace{k_m^{cl}\Gamma_m \sum_{j=1}^{N_m} \left(y_m^{cl}(t_j)\right)^T \left(F(t_j) - y_m^{cl}(t_j)\,\hat{\theta}_m\right)}_{}$$

$$= \boldsymbol{adaptive\ term} + \boldsymbol{ICL - based\ term}$$

$$y_m^{cl}(t_j) \triangleq \begin{cases} 0_{n\times 1} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} Y_m^{cl}(\tau)d\tau & t > \Delta t \end{cases}, \quad F(t_j) \triangleq \begin{cases} 0_{n\times 1} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} fRe_3(\tau)d\tau & t > \Delta t \end{cases}$$

Controller Design – Translational Controller

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Y_m^{cl} defined as follows contains acceleration terms which is not implementable

$$fRe_3 = mge_3 - m\dot{v} = Y_m^{cl}\theta_m, \quad Y_m^{cl} = \begin{bmatrix} -\ddot{x}_1 \\ -\ddot{x}_2 \\ g - \ddot{x}_3 \end{bmatrix}$$

- By integrating Y_m^{cl} to be y_m^{cl} as defined in last page, y_m^{cl} becomes implementable
- Integrating both sides of the translational dynamics $fRe_3 = Y_m^{cl}\theta_m$ yields

$$\int_{t-\Delta t}^{t} fRe_3(\tau)d\tau = \int_{t-\Delta t}^{t} Y_m^{cl}(\tau)\theta_m d\tau \Rightarrow \int \boldsymbol{fRe_3(\tau)}\Big|_{\tau=t} - \int \boldsymbol{fRe_3(\tau)}\Big|_{\tau=t-\Delta t} = y_m^{cl}\theta_m$$

$$\dot{\hat{\theta}}_m = \Gamma_m Y_m^T(e_v + C_1 e_x) + k_m^{cl}\Gamma_m \sum_{j=1}^{N_m} \left(y_m^{cl}(t_j)\right)^T \left(\boldsymbol{F(t_j)} - y_m^{cl}(t_j)\,\hat{\theta}_m\right)$$

$$= \Gamma_m Y_m^T(e_v + C_1 e_x) + k_m^{cl}\Gamma_m \sum_{j=1}^{N_m} \left(y_m^{cl}(t_j)\right)^T y_m^{cl}(t_j)\tilde{\theta}_m$$

Motivation

Problem Formulation

Controller Design

Stability Analysis

Simulation and Experiments

Conclusion

Controller Design – Rotational Controller

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Rotational controller

$$M = \underbrace{-k_R e_R - k_\Omega e_\Omega}_{} \underbrace{- Y_{diag}\hat{\theta}_{diag}}_{} \quad , Y_{diag} = \begin{bmatrix} \bar{\Omega}_1 & \Omega_2 \cdot \Omega_3 & -\Omega_2 \cdot \Omega_3 \\ -\Omega_1 \cdot \Omega_3 & \bar{\Omega}_2 & \Omega_1 \cdot \Omega_3 \\ \Omega_1 \cdot \Omega_2 & -\Omega_1 \cdot \Omega_2 & \bar{\Omega}_3 \end{bmatrix}$$

$$= \boldsymbol{feedback\ term} + \boldsymbol{adaptive\ term}$$

- Integral CL-based adaptive control update law $\dot{\hat{\theta}}_{diag}$

$$\dot{\hat{\theta}}_{diag} = \underbrace{\Gamma_{diag} Y_{diag}^T (e_\Omega + C_2 e_R)}_{} + \underbrace{k_{diag}^{cl} \Gamma_{diag} \sum_{j=1}^{N} \left(y_{diag}^{cl}(t_j)\right)^T \left(\bar{M}(t_j) - y_{diag}^{cl}(t_j)\,\hat{\theta}_{diag}\right)}_{}$$

$$= \boldsymbol{adaptive\ term} + \boldsymbol{ICL - based\ term}$$

$$y_{diag}^{cl}(t_j) \triangleq \begin{cases} 0_{n\times m} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} Y_{diag}^{cl}(\tau)d\tau & t > \Delta t \end{cases} , \quad \bar{M}(t_j) \triangleq \begin{cases} 0_{n\times m} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} M(\tau)d\tau & t > \Delta t \end{cases}$$

Controller Design – Rotational Controller

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Motivation
- Problem Formulation
- Controller Design
- Stability Analysis
- Simulation and Experiments
- Conclusion

- Y_{diag}^{cl} defined as follows contains angular acceleration which is not implementable

$$M = J\dot{\Omega} + \Omega \times J\Omega = Y_{diag}^{cl}\theta_{diag}, \quad Y_{diag}^{cl} = \begin{bmatrix} \dot{\Omega}_1 & -\Omega_2 \cdot \Omega_3 & \Omega_2 \cdot \Omega_3 \\ \Omega_1 \cdot \Omega_3 & \dot{\Omega}_2 & -\Omega_1 \cdot \Omega_3 \\ -\Omega_1 \cdot \Omega_2 & \Omega_1 \cdot \Omega_2 & \dot{\Omega}_3 \end{bmatrix}$$

- By integrating Y_{diag}^{cl} to be y_{diag}^{cl} as defined in last page, y_m^{cl} becomes implementable
- Integrating both sides of the translational dynamics $M = Y_{diag}^{cl}\theta_{diag}$ yields

$$\int_{t-\Delt t}^{t} M(\tau)d\tau = \int_{t-\Delta t}^{t} Y_{diag}^{cl}(\tau)\theta_{diag}d\tau \Rightarrow \int \boldsymbol{M(\tau)}\Big|_{\tau=t} - \int \boldsymbol{M(\tau)}\Big|_{\tau=t-\Delta t} = y_{diag}^{cl}\theta_{diag}$$

$$\dot{\hat{\theta}}_{diag} = \Gamma_{diag}Y_{diag}^{T}(e_{\Omega} + C_2 e_R) + k_{diag}^{cl}\Gamma_{diag}\sum_{j=1}^{N}\left(y_{diag}^{cl}(t_j)\right)^T\left(\boldsymbol{\bar{M}(t_j)} - y_{diag}^{cl}(t_j)\,\hat{\theta}_{diag}\right)$$

$$= \Gamma_{diag}Y_{diag}^{T}(e_{\Omega} + C_2 e_R) + k_{diag}^{cl}\Gamma_{diag}\sum_{j=1}^{N}\left(y_{diag}^{cl}(t_j)\right)^T y_{diag}^{cl}(t_j)\tilde{\theta}_{diag}$$

Stability Analysis – Closed-Loop Error Systems

- Taking the time derivative of error dynamics e_x, e_v defined in Problem Formulation

$$
\begin{aligned}
\dot{e}_x &= \dot{e}_v \\
m\dot{e}_v &= mge_3 - fRe_3 - m\ddot{x}_d \\
&= Y_m\theta_m - \boldsymbol{f}Re_3 \\
&= -k_xe_x - k_ve_v + Y_m\tilde{\theta}_m - X \quad , X = \frac{f}{e_3^TR_d^TRe_3}\left((e_3^TR_d^TRe_3)Re_3 - R_de_3\right)
\end{aligned}
$$

- Taking the time derivative of error dynamics e_R, e_Ω defined in Problem Formulation

$$
\begin{aligned}
\dot{e}_R &= \frac{1}{2}\left(R_d^TR\hat{e}_\Omega + \hat{e}_\Omega R^TR_d\right)^\vee \\
&= \frac{1}{2}\left(tr[R^TR_d]I - R^TR_d\right) \equiv C(R_d^TR)e_\Omega \\
J\dot{e}_\Omega &= J\dot{\Omega} + J\left(\widehat{\Omega}R^TR_d\Omega_d - R^TR_d\dot{\Omega}_d\right) \\
&= J\dot{\Omega} + J\bar{\Omega} = \boldsymbol{M} + Y_{diag}\theta_{diag} = -k_Re_R - k_\Omega e_\Omega + Y_{diag}\tilde{\theta}_{diag}
\end{aligned}
$$

Stability Analysis – Translational Dynamics

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Let Lyapunov function V_1 defined as

$$V_1 = \frac{1}{2} k_x e_x^T e_x + \frac{1}{2} m e_v^T e_v + C_1 m e_x \cdot e_v + \frac{1}{2} \tilde{\theta}_m^T \Gamma_m^{-1} \tilde{\theta}_m$$

- V_1 is P.D. and it can be lower and upper bounded by

$$z_1^T M_{11} z_1 + \frac{1}{2} \tilde{\theta}_m^T \Gamma_m^{-1} \tilde{\theta}_m \leq V_1 \leq z_1^T M_{12} z_1 + \frac{1}{2} \tilde{\theta}_m^T \Gamma_m^{-1} \tilde{\theta}_m$$

$$z_1 \triangleq [\|e_x\|, \quad \|e_v\|]^T$$

$$M_{11} = \frac{1}{2} \begin{bmatrix} k_x & -C_1 m \\ -C_1 m & m \end{bmatrix}$$

$$M_{12} = \frac{1}{2} \begin{bmatrix} k_x & C_1 m \\ C_1 m & m \end{bmatrix}$$

Stability Analysis – Translational Dynamics

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Taking the time derivative of V_1 yields

$$\dot{V}_1 = k_x e_x \cdot \dot{\boldsymbol{e}}_{\boldsymbol{x}} + e_v \cdot \boldsymbol{m}\dot{\boldsymbol{e}}_{\boldsymbol{v}} + C_1 m \dot{\boldsymbol{e}}_{\boldsymbol{x}} \cdot e_v + C_1 e_x \cdot \boldsymbol{m}\dot{\boldsymbol{e}}_{\boldsymbol{v}} - \tilde{\theta}_m^T \Gamma_m^{-1} \dot{\hat{\boldsymbol{\theta}}}_{\boldsymbol{m}}$$

- Substitute $\dot{e}_x$ and $m\dot{e}_v$ defined in the previous page into $\dot{V}_1$

$$\dot{V}_1 \leq -z_1^T W_1 z_1 + z_1^T W_{12} z_2 - k_m^{cl} \tilde{\theta}_m^T \left(\sum_{j=1}^{N} \left(y_m^{cl}(t_j) \right)^T y_m^{cl}(t_j) \right) \tilde{\theta}_m$$

$$W_1 = \begin{bmatrix} k_x C_1 (1-\alpha) & -\frac{1}{2} C_1 k_v (1+\alpha) \\ -\frac{1}{2} C_1 k_v (1+\alpha) & k_v (1-\alpha) - C_1 m \end{bmatrix}, \quad W_{12} = \begin{bmatrix} k_x e_{v,max} + C_1 B & 0 \\ B & 0 \end{bmatrix}$$

- M_{11}, M_{12}, W_1 in V_1 and $\dot{V}_1$ are positive-definite matrices if C_1 satisfies

$$C_1 < min \left\{ \sqrt{\frac{k_x}{m}}, \frac{k_v(1-\alpha)}{m}, \frac{4 k_x k_v (1-\alpha)^2}{k_v^2 (1+\alpha)^2 + 4 m k_x (1-\alpha)} \right\}$$

Stability Analysis – Rotational Dynamics

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Let Lyapunov function V_2 defined as

$$V_2 = \frac{1}{2} e_\Omega \cdot J e_\Omega + k_R \Psi(R, R_d) + J C_2 e_R \cdot e_\Omega + \frac{1}{2} \tilde{\theta}_{diag}^T \Gamma_{diag}^{-1} \tilde{\theta}_{diag}$$

- V_2 is P.D. and it can be lower and upper bounded by

$$z_2^T M_{21} z_2 + \frac{1}{2} \tilde{\theta}_{diag}^T \Gamma_{diag}^{-1} \tilde{\theta}_{diag} \le V_2 \le z_2^T M_{22} z_2 + \frac{1}{2} \tilde{\theta}_{diag}^T \Gamma_{diag}^{-1} \tilde{\theta}_{diag}$$

$$z_2 \triangleq [\|e_R\|, \quad \|e_\Omega\|]^T$$

$$M_{21} = \frac{1}{2} \begin{bmatrix} k_R & -C_2 \lambda_{max}(J) \\ -C_2 \lambda_{max}(J) & \lambda_{min}(J) \end{bmatrix}$$

$$M_{22} = \frac{1}{2} \begin{bmatrix} \dfrac{2k_R}{2 - \psi_2} & -C_2 \lambda_{max}(J) \\ -C_2 \lambda_{max}(J) & \lambda_{min}(J) \end{bmatrix}$$

Stability Analysis – Rotational Dynamics

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Taking the time derivative of V_2 yields

$$\dot{V}_2 = (e_\Omega + C_2 e_R) \cdot (J\dot{\boldsymbol{e}}_{\boldsymbol{\Omega}}) + k_R e_\Omega \cdot e_R + J C_2 \dot{\boldsymbol{e}}_{\boldsymbol{R}} \cdot e_\Omega - \tilde{\theta}_{diag}^T \Gamma_{diag}^{-1} \dot{\hat{\boldsymbol{\theta}}}_{\boldsymbol{diag}}$$

- Substitute $\dot{e}_R$ and $J\dot{e}_\Omega$ defined in the previous page into $\dot{V}_2$

$$\dot{V}_2 \le -z_2^T W_2 z_2 - k_{diag}^{cl} \tilde{\theta}_{diag}^T \left(\sum_{j=1}^{N} \left(y_{diag}^{cl}(t_j) \right)^T y_{diag}^{cl}(t_j) \right) \tilde{\theta}_{diag}$$

$$W_2 = \begin{bmatrix} C_2 k_R & -\dfrac{C_2 k_\Omega}{2} \\ -\dfrac{C_2 k_\Omega}{2} & k_\Omega - C_2 \lambda_{max}(J) \end{bmatrix}$$

- M_{21}, M_{22}, W_2 in V_2 and $\dot{V}_2$ are positive-definite matrices if C_2 satisfies

$$C_2 < min\left\{ \frac{k_\Omega}{\lambda_{max}(J)}, \frac{4k_\Omega k_R}{k_\Omega^2 + 4k_R \lambda_{max}(J)}, \sqrt{\frac{k_R \lambda_{min}(J)}{\lambda_{max}(J)^2}} \right\}$$

Stability Analysis – Overall System

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

- Let $V = V_1 + V_2$ be a Lyapunov function for the system containing rotational and translational dynamics

$$V = V_1 + V_2$$
$$= \frac{1}{2}k_x e_x^T e_x + \frac{1}{2}m e_v^T e_v + C_1 m e_x \cdot e_v + \frac{1}{2}\tilde{\theta}_m^T \Gamma_m^{-1} \tilde{\theta}_m$$
$$+ \frac{1}{2}e_\Omega \cdot J e_\Omega + k_R \Psi(R, R_d) + J C_2 e_R \cdot e_\Omega + \frac{1}{2}\tilde{\theta}_{diag}^T \Gamma_{diag}^{-1} \tilde{\theta}_{diag} \quad \text{... P.D.}$$

- Taking the time derivative of V and substituting $\dot{V}_1$ and $\dot{V}_2$ yields

$$\dot{V} = \dot{V}_1 + \dot{V}_2 \le -z_1^T W_1 z_1 + z_1^T W_{12} z_2 - k_m^{cl} \tilde{\theta}_m^T \left(\sum_{j=1}^{N} \left(y_m^{cl}(t_j) \right)^T y_m^{cl}(t_j) \right) \tilde{\theta}_m$$
$$-z_2^T W_2 z_2 - k_{diag}^{cl} \tilde{\theta}_{diag}^T \left(\sum_{j=1}^{N} \left(y_{diag}^{cl}(t_j) \right)^T y_{diag}^{cl}(t_j) \right) \tilde{\theta}_{diag} \quad \text{... N.D.}$$

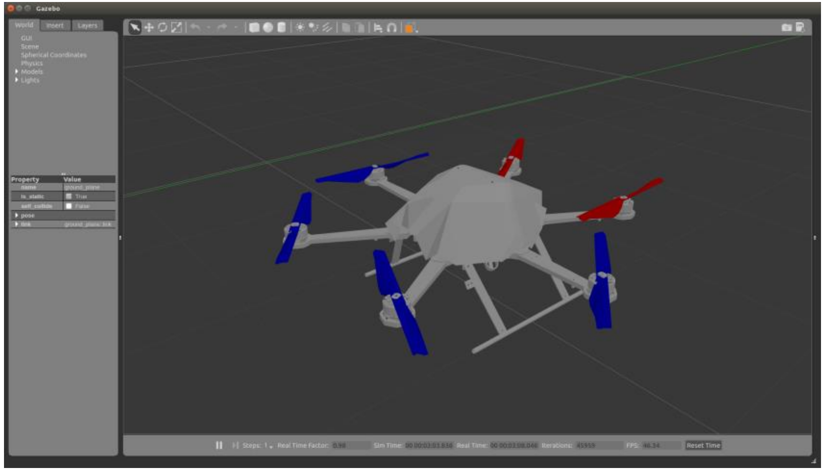
, where $\lambda_{min}(W_2) > {}^{4\|W_{12}\|^2}/_{\lambda_{min}(W_1)}$

- Motivation
- Problem Formulation
- Controller Design
- **Stability Analysis**
- Simulation and Experiments
- Conclusion

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

Simulation – Setup and Ground Truth

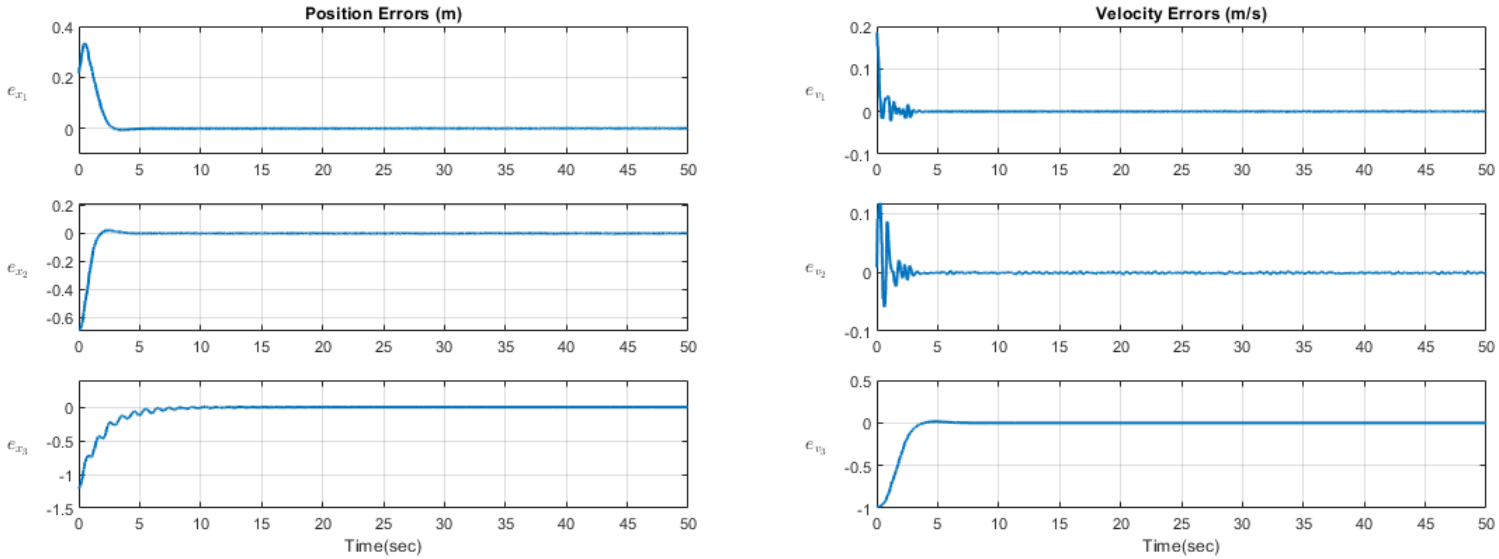
- A six-rotor multirotor was used as our model in ROS Gazebo
- The ground truth of moment of inertia J and m were unknown parameters to be estimated in the simulations, and were used for evaluating the estimate error but not for implementing the controller



Parameter	Value
m	1.568(kg)
J	$\begin{bmatrix} 0.035 & 0 & 0 \\ 0 & 0.046 & 0 \\ 0 & 0 & 0.0977 \end{bmatrix} (kg \cdot m^2)$
d	0.215(m)

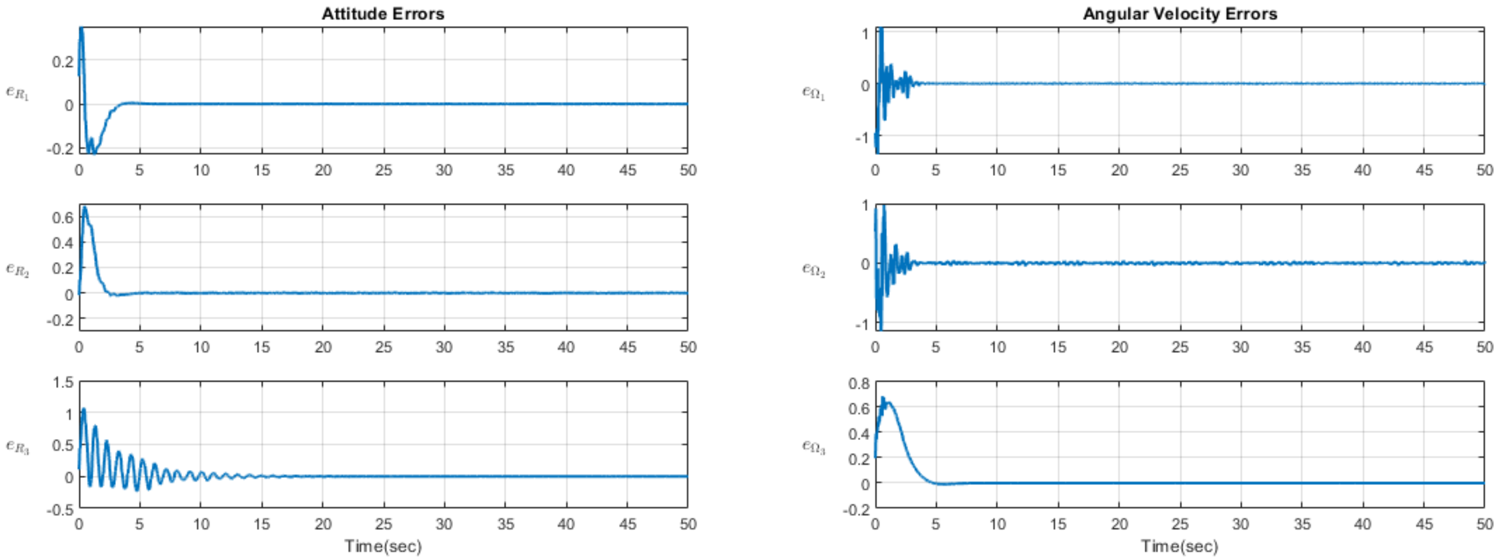
Motivation
Problem Formulation
Controller Design
Stability Analysis
Simulation and Experiments
Conclusion

Simulation – Translational Error Tracking



- The tracking errors of the position and velocity converged asymptotically to zero
- The multirotor can track a desired 3D trajectory without the information of mass and moment of inertia

Simulation – Rotational Error Tracking



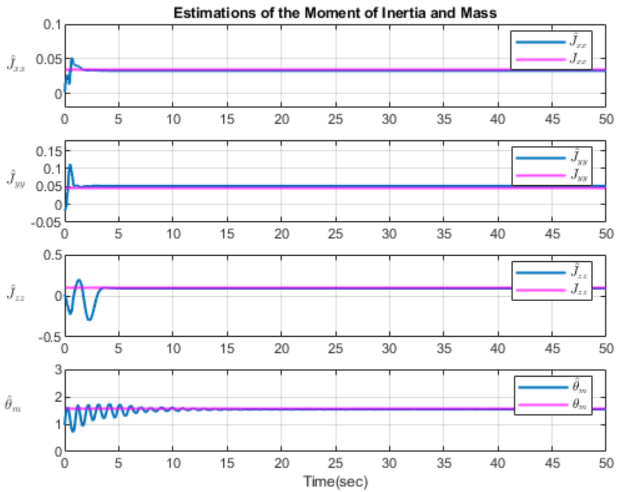
- The tracking errors of the attitude and angular velocity converged asymptotically to zero

Simulation – Estimate Mass and Moment of Inertia

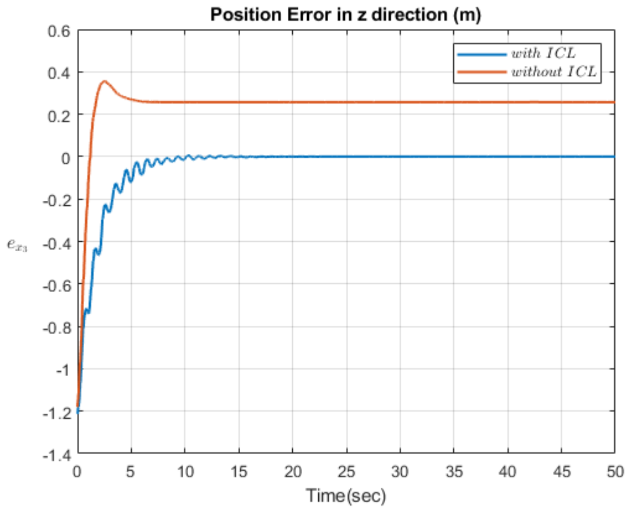
NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

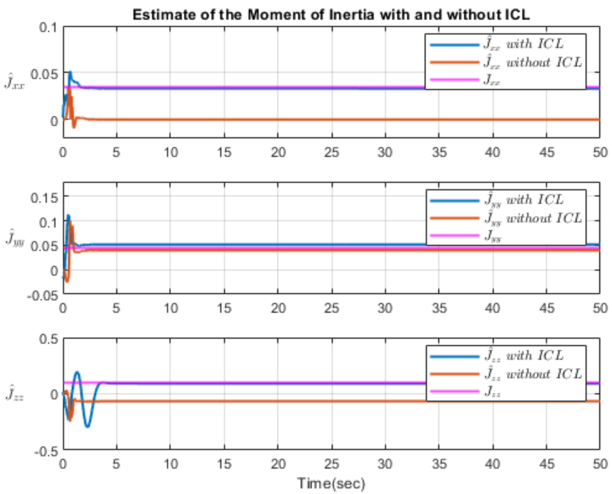
	Estimation value	Ground truth
Mass	1.534	1.568
Moment of inertia	$\begin{bmatrix} 0.033 & 0 & 0 \\ 0 & 0.051 & 0 \\ 0 & 0 & 0.091 \end{bmatrix}$	$\begin{bmatrix} 0.035 & 0 & 0 \\ 0 & 0.046 & 0 \\ 0 & 0 & 0.0977 \end{bmatrix}$

- The estimates of the moment of inertia and mass during flight
- The normalized estimate errors of the moment of inertia and mass were 7.9% and 1.5%, respectively



Simulation – Adaptive Control v.s. Adaptive ICL Control

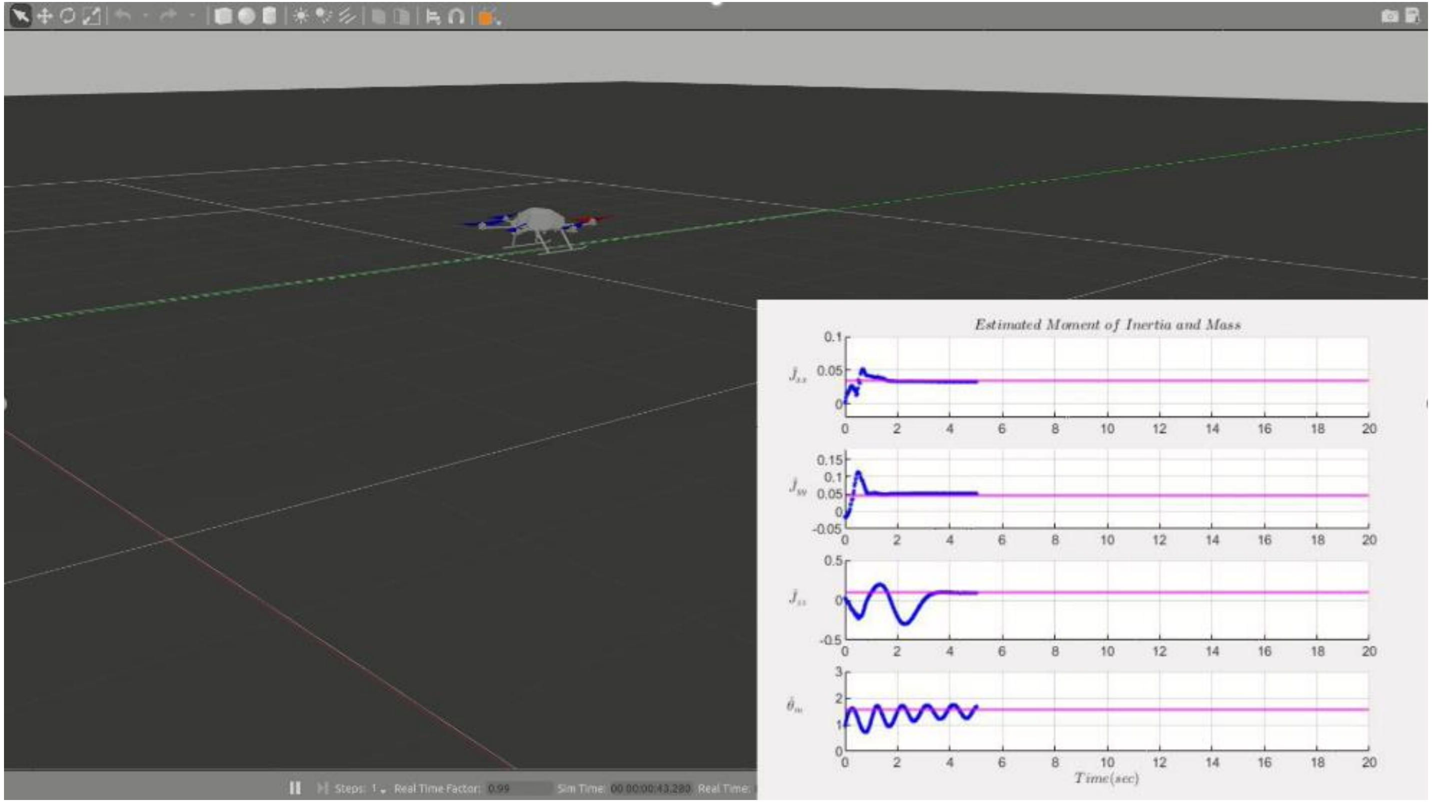




- The figure above compares the controller performance without and with the ICL controller
- This demonstrates the importance and robustness of our developed controller

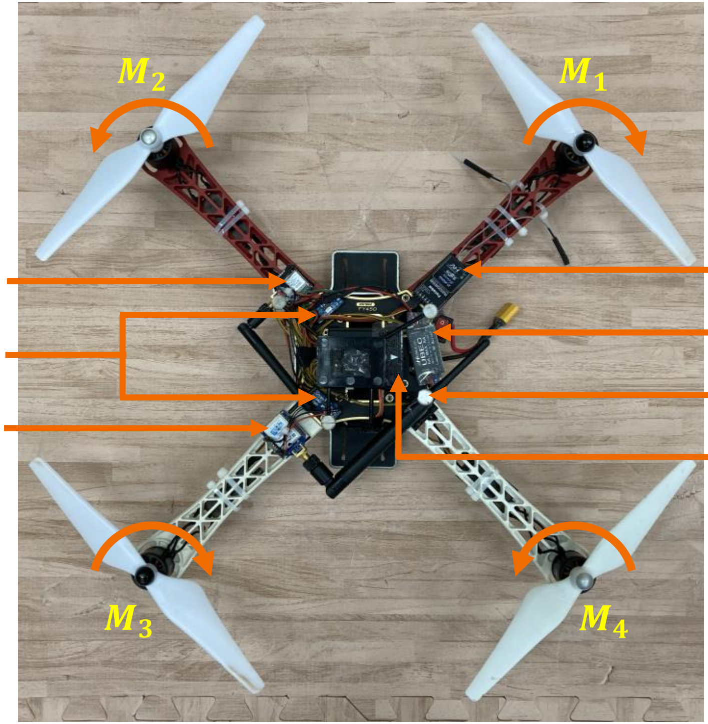
- The figure above compares the estimated parameters when using the adaptive controller and the developed ICL controller
- Asymptotic convergence of the estimate errors converged asymptotically when using the ICL controller

Simulation - Video



Experiments – Hardware Architecture

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室



2.4 GHz Xbee Module

5V to 3V3 switching regulator

900 MHz Xbee Module

RC receiver

UBEC

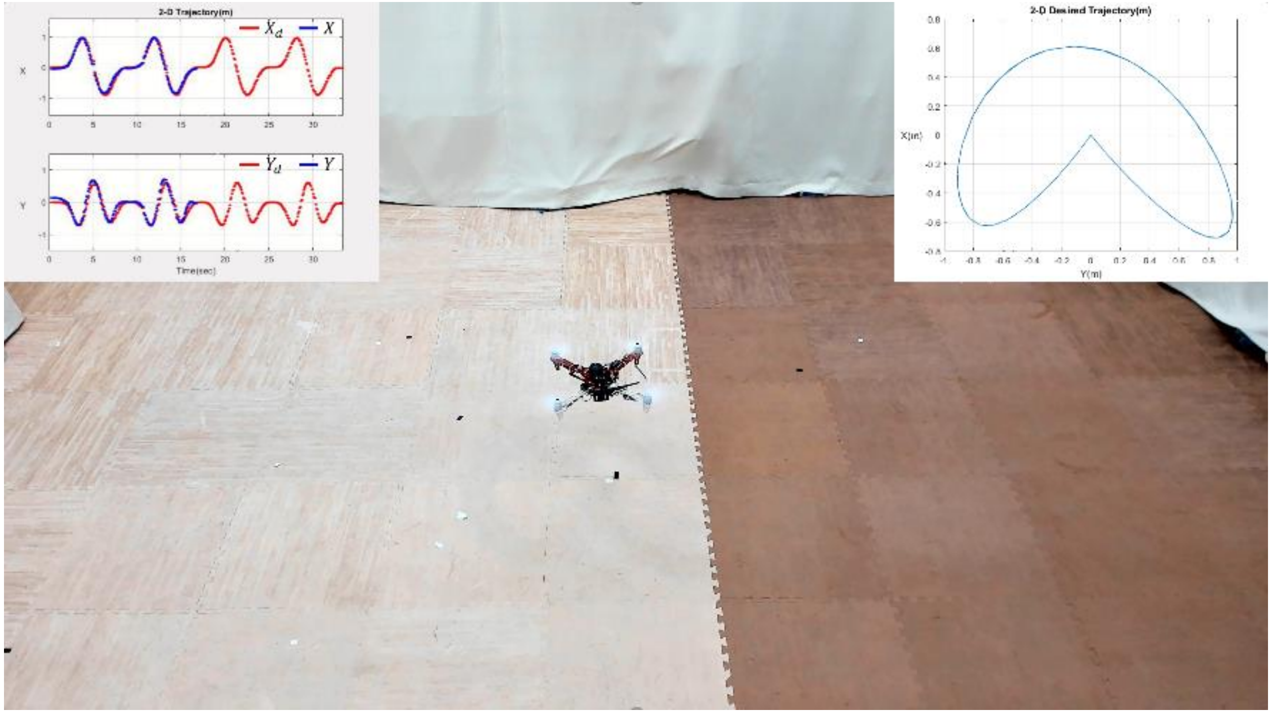
OptiTrack marker

Flight control board

NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

Experiments - Video

- The experiments is based on ncrl-flight-control mainly contributed by Shengwen



Motivation
Problem Formulation
Controller Design
Stability Analysis
Simulation and Experiments
Conclusion

Conclusion

- An ICL controller has been developed for controlling a multirotor with an unknown mass and moment of inertia
- The control architecture can be applied to many types of multirotors of unknown mass, including hexacopters and octocopters
- The ICL controller ensures the steady-state errors resulted from the wrong parameters be eliminated
- The ICL controller can guarantee asymptotic convergence of the system parameters, while the adaptive controller cannot
- Future work can be estimate other parameters of the multirotor, such as off-diagonal elements in the inertia matrix and the center of mass.

Thanks for listening!