Outlines

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

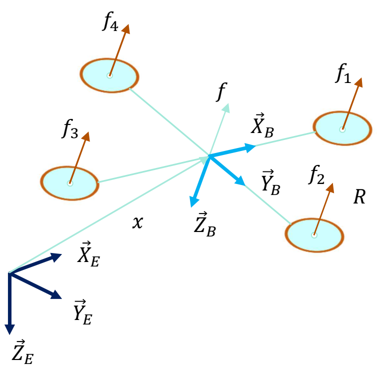
Motivation

- 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.

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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 : $\{\vec{X}_B,\ \vec{Y}_B,\ \vec{Z}_B\}$

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

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Problem Formulation - Dynamics of the Multirotor

- 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 control input, rotational speed, and moment of inertia into account.

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

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

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

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Problem Formulation – Tracking Errors and Estimate Errors

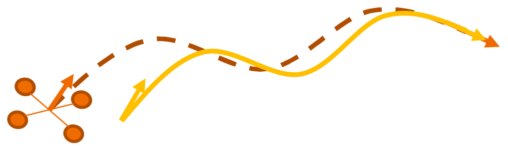
- Position and velocity tracking errors
  $e_x \triangleq x - x_d$
  $e_v \triangleq v - v_d$
- Attitude error function on SO(3) based on Geometric Tracking Control
  $\Psi(R, R_d) \triangleq \frac{1}{2} tr[I - R_d^T R]$
- Attitude tracking error and the angular velocity tracking error
  $e_R \triangleq \frac{1}{2}(R_d^T R - R^T R_d)^{\vee}$
  $e_\Omega \triangleq \Omega - R^T R_d \Omega_d$
- 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
  $\tilde{\theta}_J \triangleq \theta_J - \hat{\theta}_J, \theta_J = [J_{xx} \quad J_{yy} \quad J_{zz}]^T$ (moment of inertia of the multirotor)

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Problem Formulation – Control Objectives

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

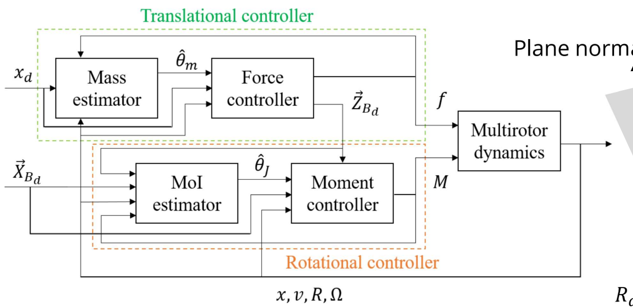
$$\begin{cases} e_x \to 0 \\ e_v \to 0 \\ e_R \to 0 \\ e_\Omega \to 0 \\ \tilde{\theta}_m \to 0 \\ \tilde{\theta}_J \to 0 \end{cases} \quad as\ t \to \infty$$

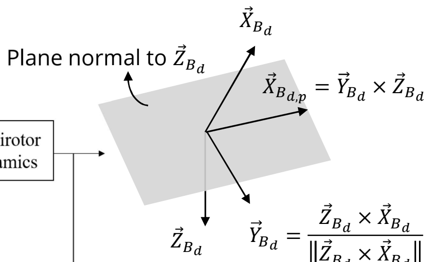


3D position
3D trajectory
yaw angle
desired yaw angle

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Controller Design – Control Architecture





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

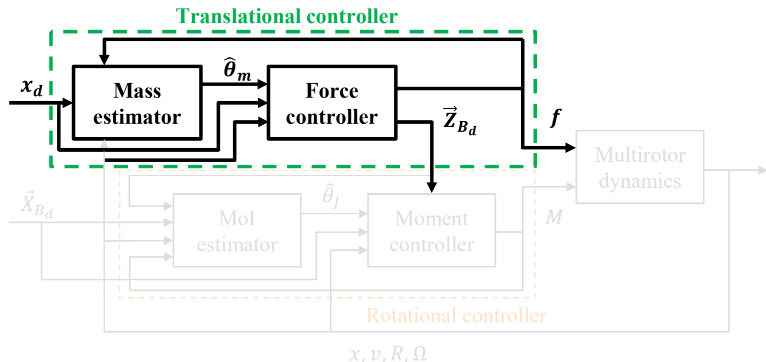
$\vec{Y}_{B_d} = \frac{\vec{Z}_{B_d} \times \vec{X}_{B_d}}{\|\vec{Z}_{B_d} \times \vec{X}_{B_d}\|}$

$R_d = [\vec{X}_{B_{d,p}}; \quad \vec{Y}_{B_d}; \quad \vec{Z}_{B_d}] \in R^{3\times3}$

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Controller Design – Translational Controller

- Translational controller



Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Controller Design – Translational Controller

- Translational controller

$$f = (k_x e_x + k_v e_v + Y_m \hat{\theta}_m) \cdot R e_3 \quad , Y_m = \begin{bmatrix} -\ddot{x}_{d_1} \\ -\ddot{x}_{d_2} \\ g - \ddot{x}_{d_3} \end{bmatrix} \text{ is a regression matrix}$$

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

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

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

$$= \textit{adaptive term} + \textit{ICL} - \textit{based te}$$

$$y_m^{cl}(t_j) \triangleq \begin{cases} 0_{n\times 1} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} Y_m^{cl}(\tau) d\tau & t > \Delta t \end{cases} , \quad F(t_j) \triangleq \begin{cases} 0_{n\times 1} & t \in [0, \Delta t] \\ \int_{t-\Delta t}^{t} f R e_3(\tau) d\tau & t > \Delta t \end{cases}$$

Controller Design – Translational Controller

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

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

$$f R e_3 = m g e_3 - m \dot{v} = Y_m^{cl} \theta_m , \quad Y_m^{cl} = \begin{bmatrix} -\ddot{x}_1 \\ -\ddot{x}_2 \\ g - \ddot{x}_3 \end{bmatrix}$$

- 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 $f R e_3 = Y_m^{cl} \theta_m$ yields

$$\int_{t-\Delta t}^{t} f R e_3(\tau) d\tau = \int_{t-\Delta t}^{t} Y_m^{cl}(\tau) \theta_m d\tau \Rightarrow \int \boldsymbol{fRe_3(\tau)} \Big|_{\tau=t} - \int \boldsymbol{fR\ {}_3(\tau)} \Big|_{\tau=t-\Delta t} = y_m^{cl} \theta_m$$

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

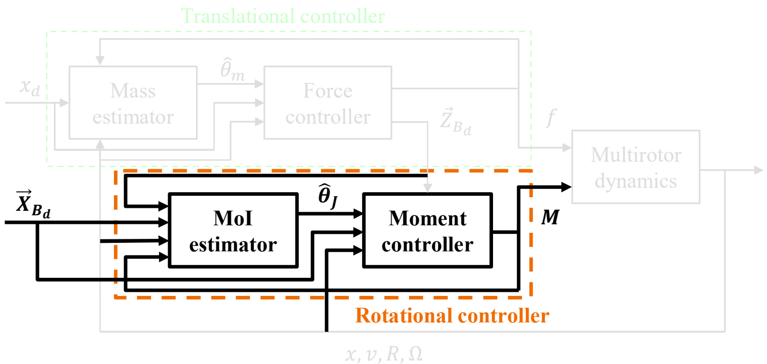
$$= \Gamma_m Y_m^T (e_v + C_1 e_x) + k_m^{cl} \Gamma_m \sum_{j=1}^{N_m} \left(y_m^{cl}(t_j) \right)^T y_m^{cl}(t_j) \tilde{\theta}_m$$

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
Experiments
Conclusion

- Rotational controller



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
Experiments
Conclusion

- Rotational controller

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

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

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

$$\dot{\hat{\theta}}_J = \Gamma_J Y_J^T (e_\Omega + C_2 e_R) + k_J^{cl} \Gamma_J \sum_{j=1}^{N_J} \left(y_J^{cl}(t_j) \right)^T \left(\bar{M}(t_j) - y_J^{cl}(t_j)\, \hat{\theta}_J \right)$$

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

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

Controller Design – Rotational Controller

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

$$M = J\dot{\Omega} + \Omega \times J\Omega = Y_J^{cl}\theta_J, \qquad Y_J^{cl} = \begin{bmatrix} \dot{\Omega}_1 & -\Omega_2 \cdot \Omega_3 & \Omega_2 \cdot \Omega_3 \\ \Omega_1 \cdot \Omega_3 & \dot{\Omega}_2 & -\Omega_1 \cdot \Omega_3 \\ -\Omega_1 \cdot \Omega_2 & \Omega_1 \cdot \Omega_2 & \dot{\Omega}_3 \end{bmatrix}$$

- By integrating Y_J^{cl} to be y_J^{cl} as defined in last page, y_J^{cl} becomes implementable
- Integrating both sides of the translational dynamics $M = Y_J^{cl}\theta_J$ yields

$$\int_{t-\Delta t}^{t} M(\tau)d\tau = \int_{t-\Delta t}^{t} Y_J^{cl}(\tau)\theta_J d\tau \Rightarrow \int \boldsymbol{M(\tau)}\Big|_{\tau=t} - \int \boldsymbol{M(\tau)}\Big|_{\tau=t-\Delta t} = y_J^{cl}\theta_J$$

$$\dot{\hat{\theta}}_J = \Gamma_J Y_J^T(e_\Omega + C_2 e_R) + k_J^{cl}\Gamma_J \sum_{j=1}^{N_J} \left(y_J^{cl}(t_j)\right)^T \left(\boldsymbol{\bar{M}(t_j)} - y_J^{cl}(t_j)\,\hat{\theta}_J\right)$$

$$= \Gamma_J Y_J^T(e_\Omega + C_2 e_R) + k_J^{cl}\Gamma_J \sum_{j=1}^{N_J} \left(y_J^{cl}(t_j)\right)^T y_J^{cl}(t_j)\tilde{\theta}_J$$

Stability Analysis – Closed-Loop Error Systems

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

$$\dot{e}_x = \dot{e}_v$$

$$m\dot{e}_v = mge_3 - fRe_3 - m\ddot{x}_d$$

$$= Y_m\theta_m - \boldsymbol{f}Re_3$$

$$= -k_x e_x - k_v e_v + Y_m\tilde{\theta}_m - X \quad , X = \frac{f}{e_3^T R_d^T R e_3}\left((e_3^T R_d^T R e_3)Re_3 - R_d e_3\right)$$

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

$$\dot{e}_R = \frac{1}{2}\left(R_d^T R\hat{e}_\Omega + \hat{e}_\Omega R^T R_d\right)^\vee$$

$$= \frac{1}{2}\left(tr[R^T R_d]I - R^T R_d\right) \equiv C(R_d^T R)e_\Omega$$

$$J\dot{e}_\Omega = J\dot{\Omega} + J(\hat{\Omega}R^T R_d\Omega_d - R^T R_d\dot{\Omega}_d)$$

$$= J\dot{\Omega} + J\bar{\Omega} = \boldsymbol{M} + Y_J\theta_J = -k_R e_R - k_\Omega e_\Omega + Y_J\tilde{\theta}_J$$

Stability Analysis – Translational Dynamics

- 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

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

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

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

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

Stability Analysis – Translational Dynamics

- Taking the time derivative of V_1 yields

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

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

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

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

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

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

Stability Analysis – Rotational Dynamics

- Let Lyapunov function V_2 defined as

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

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

$$z_2^T M_{21} z_2 + \frac{1}{2} \tilde{\theta}_J^T \Gamma_J^{-1} \tilde{\theta}_J \le V_2 \le z_2^T M_{22} z_2 + \frac{1}{2} \tilde{\theta}_J^T \Gamma_J^{-1} \tilde{\theta}_J$$

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

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

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

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

Stability Analysis – Rotational Dynamics

- Taking the time derivative of V_2 yields

$$\dot{V}_2 = (e_\Omega + C_2 e_R) \cdot (J \dot{e}_\Omega) + k_R e_\Omega \cdot e_R + J C_2 \dot{e}_R \cdot e_\Omega - \tilde{\theta}_J^T \Gamma_J^{-1} \dot{\hat{\theta}}_J$$

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

$$\dot{V}_2 \le -z_2^T W_2 z_2 - k_J^{cl} \tilde{\theta}_J^T \left(\sum_{j=1}^{N_J} \left(y_J^{cl}(t_j) \right)^T y_J^{cl}(t_j) \right) \tilde{\theta}_J$$

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

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

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

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

Stability Analysis – Overall System

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

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

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

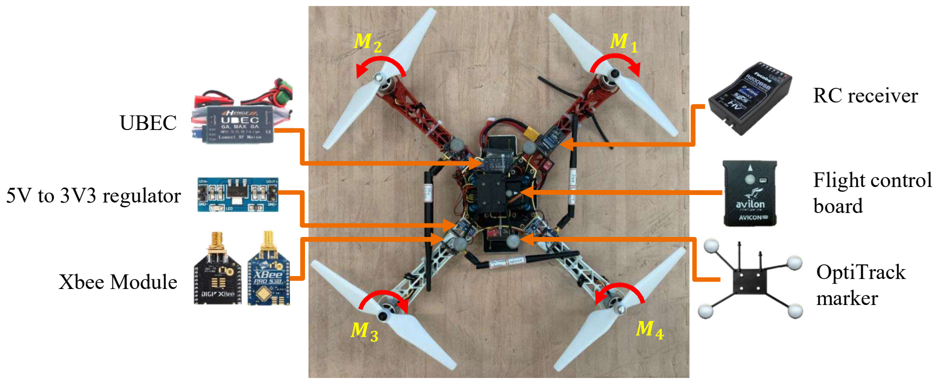
$$\dot{V} = \dot{V}_1 + \dot{V}_2 \le -z_1^T W_1 z_1 + z_1^T W_{12} z_2 - k_m^{cl} \tilde{\theta}_m^T \left(\sum_{j=1}^{N_m} \left(y_m^{cl}(t_j) \right)^T y_m^{cl}(t_j) \right) \tilde{\theta}_m$$
$$-z_2^T W_2 z_2 - k_J^{cl} \tilde{\theta}_J^T \left(\sum_{j=1}^{N_J} \left(y_J^{cl}(t_j) \right)^T y_J^{cl}(t_j)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

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

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

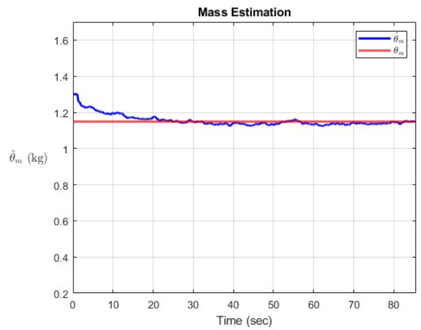
Experiments – Hardware Architecture



Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Experiments – Mass Estimation

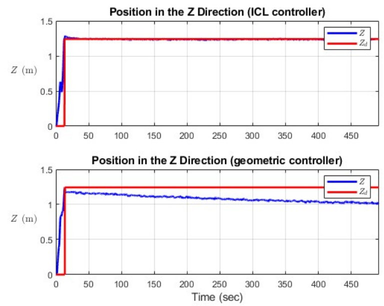
- The mass estimation of the multirotor with ICL controller converged to 1.15kg and has 1% error with ground truth



Mass estimation	Mass ground truth
1.15 (kg)	1.16 (kg)

Experiments – Mass Estimation

- The accurate mass estimation also make the multirotor have better tracking performance in the z direction



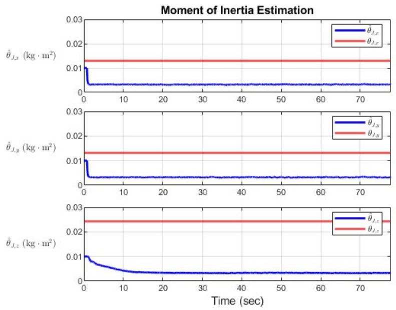
Steady-state error in the z direction	
ICL controller	Geometric controller
<0.5 (cm)	>10 (cm)

ICL controller : $f = \left(k_x e_x + k_v e_v + \boldsymbol{Y_m \hat{\theta}_m}\right) \cdot Re_3$

Geometric controller : $f = (k_x e_x + k_v e_v + \boldsymbol{mg\ _3 - m\ddot{x}_d}) \cdot Re_3$

Experiments – Moment of Inertia Estimation

- The moment of inertia estimation converge to $[0.0033, 0.0032, 0.0033]$ $(kg \cdot m^2)$ which is smaller than the ground truth



Moment of inertia estimation $(kg \cdot m^2)$	Moment of inertia ground truth $(kg \cdot m^2)$
$\begin{bmatrix} 0.0033 & 0 & 0 \\ 0 & 0.0032 & 0 \\ 0 & 0 & 0.0033 \end{bmatrix}$	$\begin{bmatrix} 0.013 & 0 & 0 \\ 0 & 0.013 & 0 \\ 0 & 0 & 0.024 \end{bmatrix}$

Experiments – Moment of Inertia Estimation

- We suppose the incorrect convergence is resulted from the measurement noise from gyroscope
- To present the experimental scene more realistically, white Gaussian noise is applied to the simulation
- Moreover, we designed an estimator to eliminate the influence of noise on the system

$$M = -k_R e_R - k_\Omega e_\Omega - Y_J \hat{\theta}_J \ , Y_J = \begin{bmatrix} \bar{\Omega}_1 & \Omega_2 \cdot \Omega_3 & -\Omega_2 \cdot \Omega_3 \\ -\Omega_1 \cdot \Omega_3 & \bar{\Omega}_2 & \Omega_1 \cdot \Omega_3 \\ \Omega_1 \cdot \Omega_2 & -\Omega_1 \cdot \Omega_2 & \bar{\Omega}_3 \end{bmatrix}$$

$$\dot{\hat{\theta}}_J = \Gamma_J Y_J^T (e_\Omega + C_2 e_R) + k_J^{cl} \Gamma_J \sum_{j=1}^{N_J} \left(y_J^{cl}(t_j)\right)^T \left(\bar{M}(t_j) - y_J^{cl}(t_j)\, \hat{\theta}_J\right) \ , Y_J^{cl} = \begin{bmatrix} \dot{\Omega}_1 & -\Omega_2 \cdot \Omega_3 & \Omega_2 \cdot \Omega_3 \\ \Omega_1 \cdot \Omega_3 & \dot{\Omega}_2 & -\Omega_1 \cdot \Omega_3 \\ -\Omega_1 \cdot \Omega_2 & \Omega_1 \cdot \Omega_2 & \dot{\Omega}_3 \end{bmatrix}$$

Experiments – Estimator of the Angular Velocity

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

- The rotational dynamics of the multirotor ca be written as

$$\dot{\Omega} = J^{-1}M - J^{-1}\Omega \times J\Omega = \begin{bmatrix} \frac{M_1}{\hat{J}_{xx}} + \frac{\hat{J}_{yy}}{\hat{J}_{xx}}\Omega_2\Omega_3 - \frac{\hat{J}_{zz}}{\hat{J}_{xx}}\Omega_2\Omega_3 \\ \frac{M_2}{\hat{J}_{yy}} + \frac{\hat{J}_{xx}}{\hat{J}_{yy}}\Omega_1\Omega_3 - \frac{\hat{J}_{zz}}{\hat{J}_{yy}}\Omega_1\Omega_3 \\ \frac{M_3}{\hat{J}_{zz}} + \frac{\hat{J}_{xx}}{\hat{J}_{zz}}\Omega_1\Omega_2 - \frac{\hat{J}_{yy}}{\hat{J}_{zz}}\Omega_1\Omega_2 \end{bmatrix}$$

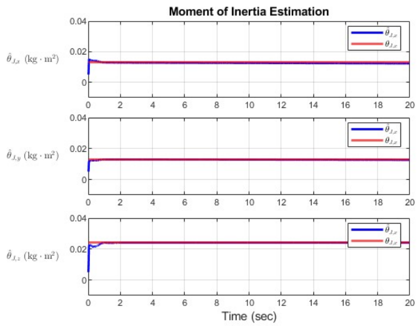
$$\hat{\Omega}_k^- = \hat{\Omega}_{k-1} + \dot{\Omega}_{k-1}\Delta t = \begin{bmatrix} \hat{\Omega}_{k-1,1} + \left(\frac{M_1}{\hat{J}_{xx}} + \frac{\hat{J}_{yy}}{\hat{J}_{xx}}\Omega_2\Omega_3 - \frac{\hat{J}_{zz}}{\hat{J}_{xx}}\Omega_2\Omega_3\right)\cdot\Delta t \\ \hat{\Omega}_{k-1,2} + \left(\frac{M_2}{\hat{J}_{yy}} + \frac{\hat{J}_{xx}}{\hat{J}_{yy}}\Omega_1\Omega_3 - \frac{\hat{J}_{zz}}{\hat{J}_{yy}}\Omega_1\Omega_3\right)\cdot\Delta t \\ \hat{\Omega}_{k-1,3} + \left(\frac{M_3}{\hat{J}_{zz}} + \frac{\hat{J}_{xx}}{\hat{J}_{zz}}\Omega_1\Omega_2 - \frac{\hat{J}_{yy}}{\hat{J}_{zz}}\Omega_1\Omega_2\right)\cdot\Delta t \end{bmatrix}$$

- The estimated angular velocity can be generated as

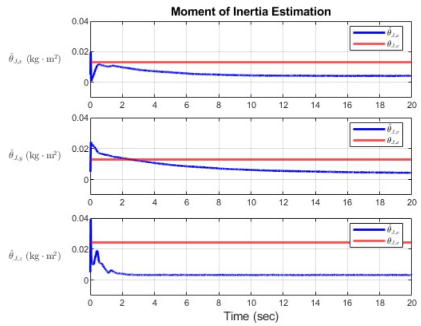
$\hat{\Omega}_k = \hat{\Omega}_k^- + K_\Omega(\Omega - \hat{\Omega}_k^-)$, K_Ω is a estimator gain

Experiments – Moment of Inertia Estimation

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



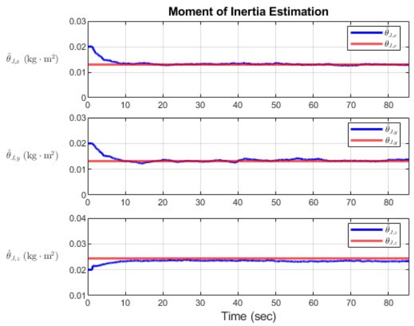
Moment of inertia estimation **with noise and estimator** in the simulation



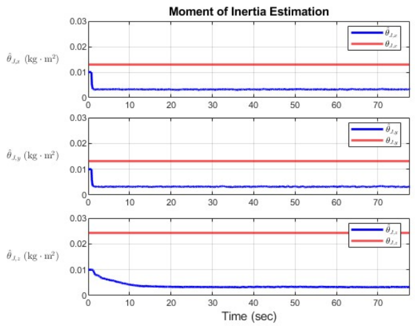
Moment of inertia estimation **with noise, without estimator** in the simulation

Experiments – Moment of Inertia Estimation

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



Moment of inertia estimation **with noise and estimator** in the experiments

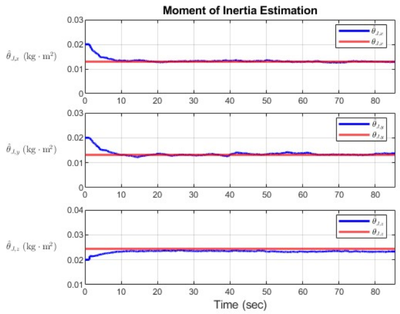


Moment of inertia estimation **with noise, without estimator** in the experiments

Experiments – Moment of Inertia Estimation

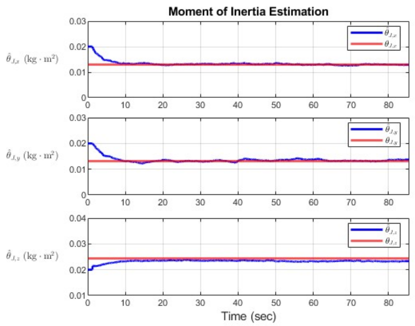
NETWORKED CONTROL ROBOTICS LAB 網路控制機器人實驗室

- The moment of inertia estimation converge to $[0.013, 0.014, 0.022]\ (kg \cdot m^2)$ and has 8% error with ground truth

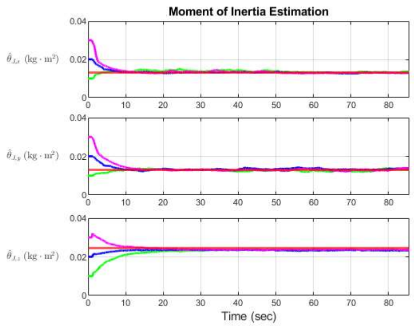


Moment of inertia estimation $(kg \cdot m^2)$	Moment of inertia ground truth $(kg \cdot m^2)$
$\begin{bmatrix} 0.013 & 0 & 0 \\ 0 & 0.014 & 0 \\ 0 & 0 & 0.022 \end{bmatrix}$	$\begin{bmatrix} 0.013 & 0 & 0 \\ 0 & 0.013 & 0 \\ 0 & 0 & 0.024 \end{bmatrix}$

Experiments – Moment of Inertia Estimation



Moment of inertia estimation **with noise and estimator** in the experiments



Moment of inertia estimation **with different initial values** in the experiments

Experiments – Trajectory Generation

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

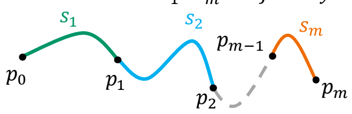
- Formulate the trajectory generation problem as a quadratic programming (QP) problem
- Write the trajectory passing through given waypoints as piecewise polynomial function of order n as

$$s_i(t) = \sum_{j=0}^{n} \sigma_{ij} t^j, t_{i-1} \le t < t_i, i \in \{1,2,\cdots,m\},$$

with cost function and constraints defined as

$$\min \int_{t_0}^{t_m} \left\| \frac{d^4 s_i}{dt^4} \right\|^2 dt,$$
$$s.t. A\sigma = b$$

$p_0 \sim p_m$: *waypoints*
$s_1 \sim s_m$: *trajectory*



Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

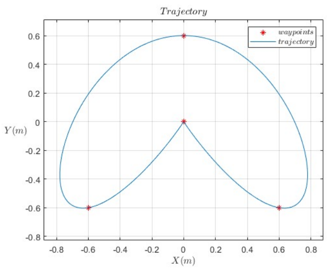
Experiments – Trajectory Generation

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

- The waypoints are given as

$$p_0 = \begin{bmatrix}0\\0\end{bmatrix} \quad p_1 = \begin{bmatrix}0.6\\-0.6\end{bmatrix} \quad p_2 = \begin{bmatrix}0\\0.6\end{bmatrix} \quad p_3 = \begin{bmatrix}-0.6\\-0.6\end{bmatrix} \quad p_4 = \begin{bmatrix}0\\0\end{bmatrix}$$

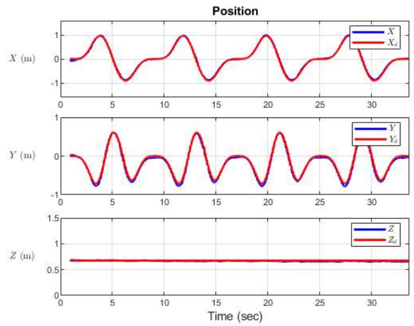
- The desired trajectory is generation as



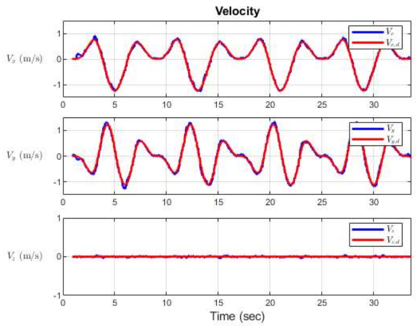
Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

Experiments – Tracking Performance

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



Position tracking performance of the multirotor using ICL controller



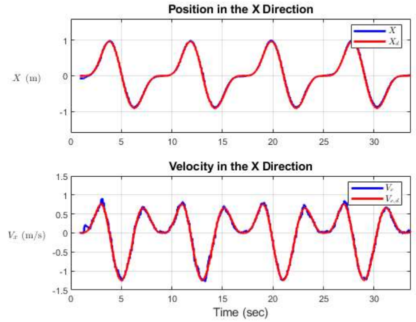
Velocity tracking performance of the multirotor using ICL controller

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

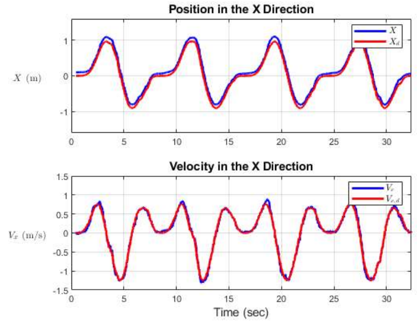
Experiments – Comparison

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

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



Tracking performance of the multirotor using **ICL controller**

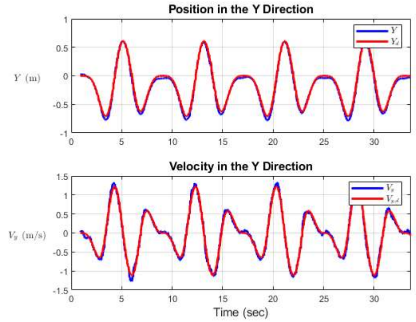


Tracking performance of the multirotor using **geometric controller**

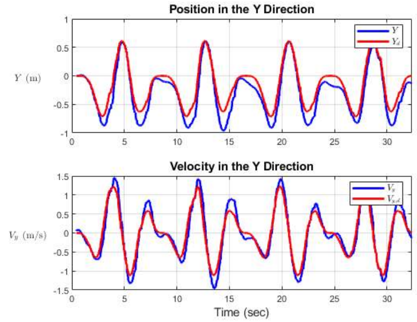
Experiments – Comparison

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

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



Tracking performance of the multirotor using **ICL controller**

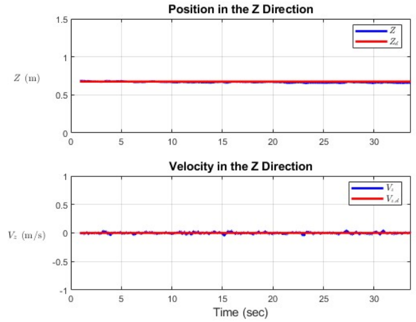


Tracking performance of the multirotor using **geometric controller**

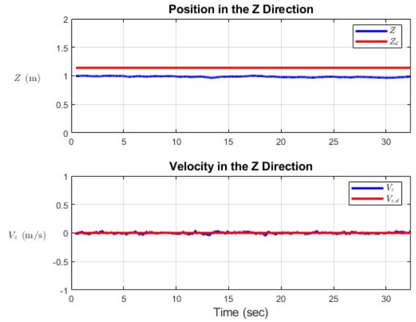
Experiments – Comparison

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

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



Tracking performance of the multirotor using **ICL controller**

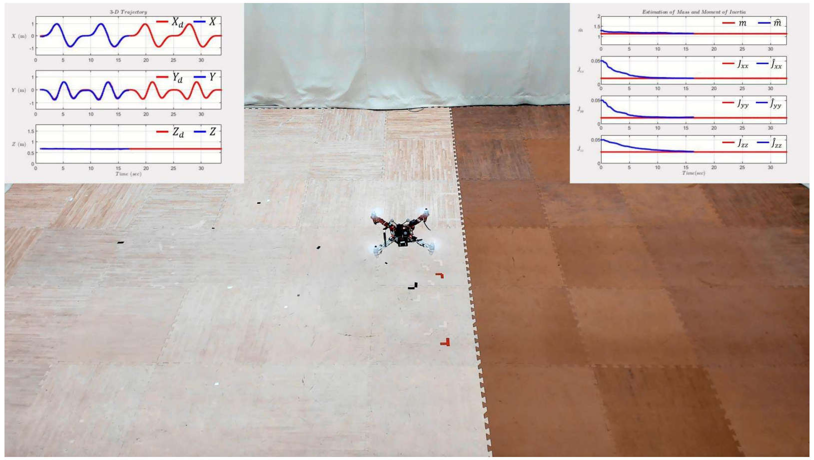


Tracking performance of the multirotor using **geometric controller**

Experiments – Video (ICL controller)

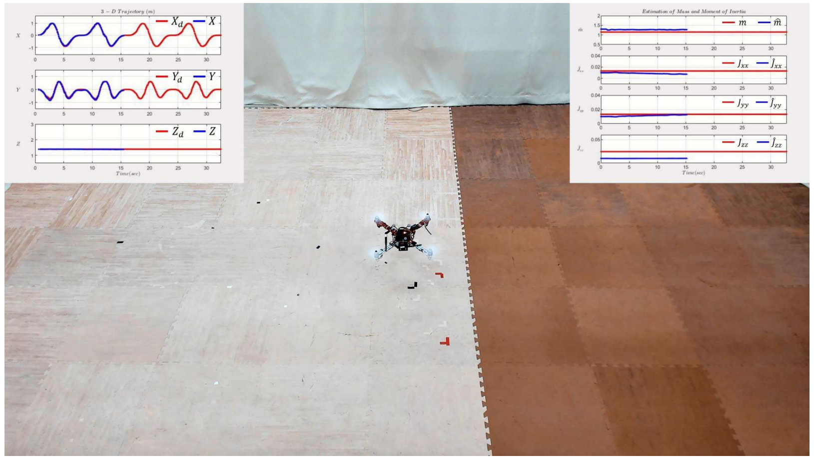
NETWORKED CONTROL ROBOTICS LAB
網路控制機器人實驗室

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



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

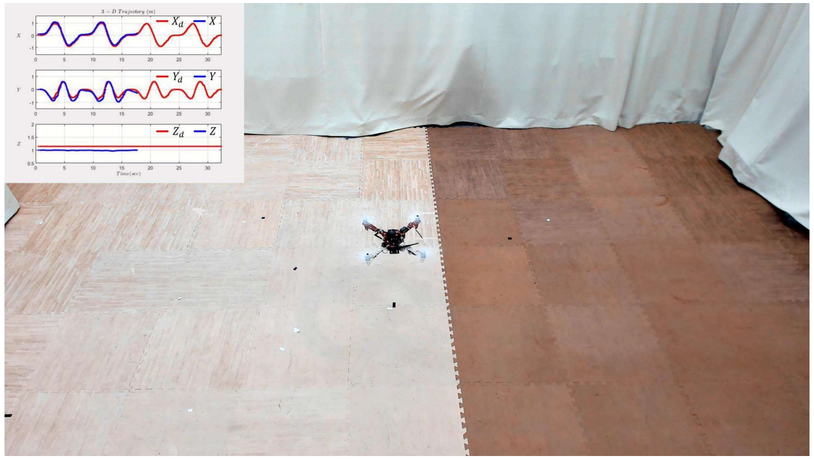
Experiments – Video (adaptive controller)



Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

Experiments – Video (geometric controller)



Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

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
- 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.

Motivation
Problem Formulation
Controller Design
Stability Analysis
Experiments
Conclusion

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

Thanks for listening!