Almost Global Tracking Control of a Quadrotor UAV on SE(3)

Erjen Lefeber, Sebastiaan van den Eijnden,
and Henk Nijmeijer

56th IEEE Conference on Decision and Control

Technische Universiteit
Eindhoven
University of Technology

December 12, 2017

Tracking control of drones

Three approaches for modeling dynamics and deriving controllers:

- Euler angles
- (Unit) quaternions
- SE(3)

Introduction

Tracking control of drones

Three approaches for modeling dynamics and deriving controllers:

- Euler angles
- (Unit) quaternions
- SE(3)

Euler angles

Singularities in representation (gimbal lock)

Introduction

Tracking control of drones

Three approaches for modeling dynamics and deriving controllers:

- Euler angles
- (Unit) quaternions
- SE(3)

Euler angles

Singularities in representation (gimbal lock)

(Unit) quaternions

Let both q and $\bar{q}$ represent same attitude.
Need same control actions: $u(q) = u(\bar{q})$, otherwise: ambiguity.

Technische Universiteit Eindhoven University of Technology

Introduction

Remaining option: SE(3)

Both problems are overcome by considering dynamics on SE(3)

Introduction

Remaining option: SE(3)

Both problems are overcome by considering dynamics on SE(3)

Shortcoming of most papers on SE(3)

Almost global result under assumption of *non-zero thrust follower*.

Remaining option: SE(3)

Both problems are overcome by considering dynamics on SE(3)

Shortcoming of most papers on SE(3)

Almost global result under assumption of *non-zero thrust follower*.
Consequence: only local result.

Introduction

Remaining option: SE(3)

Both problems are overcome by considering dynamics on SE(3)

Shortcoming of most papers on SE(3)

Almost global result under assumption of *non-zero thrust follower*.
Consequence: only local result.

Contribution

Almost global result under assumption of *non-zero thrust reference*.

Introduction

Comparable result

M.-D. Hua, T. Hamel, P. Morin, and C. Samson, "A control approach for thrust-propelled underactuated vehicles and its application to VTOL drones," IEEE Transactions on Automatic Control, vol. 54, no. 8, pp. 1837–1853, 2009.

Technische Universiteit Eindhoven University of Technology

Introduction

Comparable result

M.-D. Hua, T. Hamel, P. Morin, and C. Samson, "A control approach for thrust-propelled underactuated vehicles and its application to VTOL drones," IEEE Transactions on Automatic Control, vol. 54, no. 8, pp. 1837–1853, 2009.

Major differences

In this paper

- torques as input (vs. velocities)
- uniform almost global asymptotic stability

Problem

Drone dynamics

$$
\begin{aligned}
\dot{\rho} &= R\nu \\
\dot{\nu} &= -S(\omega)\nu + gR^T e_3 - \frac{f}{m} e_3 \\
\dot{R} &= RS(\omega) \\
J\dot{\omega} &= S(J\omega)\omega + \tau,
\end{aligned}
$$

Problem

Drone dynamics

$$
\begin{aligned}
\dot{\rho} &= R\nu \\
\dot{\nu} &= -S(\omega)\nu + gR^T e_3 - \frac{f}{m} e_3 \\
\dot{R} &= RS(\omega) \\
J\dot{\omega} &= S(J\omega)\omega + \tau,
\end{aligned}
$$

Reference dynamics

$$
\begin{aligned}
\dot{\rho}_r &= R_r \nu_r \\
\dot{\nu}_r &= -S(\omega_r)\nu_r + gR_r^T e_3 - \frac{f_r}{m} e_3 \\
\dot{R}_r &= R_r S(\omega_r) \\
J\dot{\omega}_r &= S(J\omega_r)\omega_r + \tau_r
\end{aligned}
$$

Problem

Drone dynamics

$$\dot{\rho} = R\nu$$
$$\dot{\nu} = -S(\omega)\nu + gR^T e_3 - \frac{f}{m} e_3$$
$$\dot{R} = RS(\omega)$$
$$J\dot{\omega} = S(J\omega)\omega + \tau,$$

Reference dynamics

$$\dot{\rho}_r = R_r \nu_r$$
$$\dot{\nu}_r = -S(\omega_r)\nu_r + gR_r^T e_3 - \frac{f_r}{m} e_3$$
$$\dot{R}_r = R_r S(\omega_r)$$
$$J\dot{\omega}_r = S(J\omega_r)\omega_r + \tau_r$$

Feasible reference trajectory

Trajectory $(\rho_r, R_r, \nu_r, \omega_r, f_r, \tau_r)$ satisfying reference dynamics, with $0 < f_r^{\min} \leq f_r(t)$ and $\omega_r(t)$ bounded.

Problem

Error coordinates

$$\tilde{\rho} = R_r^T(\rho - \rho_r) \qquad \tilde{R} = R_r^T R$$

$$\tilde{\nu} = -\tilde{R}^T S(\omega_r)\tilde{\rho} + \nu - \tilde{R}^T \nu_r \qquad \tilde{\omega} = \omega - \tilde{R}^T \omega_r$$

Technische Universiteit Eindhoven University of Technology

Error coordinates

$$\tilde{\rho} = R_r^T(\rho - \rho_r) \qquad \tilde{R} = R_r^T R$$

$$\tilde{\nu} = -\tilde{R}^T S(\omega_r)\tilde{\rho} + \nu - \tilde{R}^T \nu_r \qquad \tilde{\omega} = \omega - \tilde{R}^T \omega_r$$

Error measure

$$\varepsilon(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega}) = \|\tilde{\rho}\| + \|\log \tilde{R}\| + \|\tilde{\nu}\| + \|\tilde{\omega}\|$$

Problem

Error coordinates

$$\tilde{\rho} = R_r^T(\rho - \rho_r) \qquad \tilde{R} = R_r^T R$$

$$\tilde{\nu} = -\tilde{R}^T S(\omega_r)\tilde{\rho} + \nu - \tilde{R}^T \nu_r \qquad \tilde{\omega} = \omega - \tilde{R}^T \omega_r$$

Error measure

$$\varepsilon(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega}) = \|\tilde{\rho}\| + \|\log \tilde{R}\| + \|\tilde{\nu}\| + \|\tilde{\omega}\|$$

Problem

For $(\rho_r, R_r, \nu_r, \omega_r, f_r, \tau_r)$ being a given feasible reference trajectory, find appropriate control laws

$$f = f(\rho, R, \nu, \omega, \rho_r, R_r, \nu_r, \omega_r) > 0, \qquad \tau = \tau(\rho, R, \nu, \omega, \rho_r, R_r, \nu_r, \omega_r)$$

such that for the resulting closed-loop system

$$\lim_{t\to\infty} \varepsilon\big(\tilde{\rho}(t), \tilde{R}(t), \tilde{\nu}(t), \tilde{\omega}(t)\big) = 0.$$

Technische Universiteit
Eindhoven
University of Technology

Approach

Controller design

Two steps:

- Position tracking (body-fixed accelerations as *virtual input*)
- Attitude control (using actual inputs)

Show stability result using cascade analysis

Position tracking

Error definition

Express tracking error in body fixed frame of the reference:

$$\begin{bmatrix}\rho_e \\ \nu_e\end{bmatrix} = \begin{bmatrix}R_r^T(\rho_r - \rho) \\ \nu_r - R_r^T R \nu\end{bmatrix}$$

TU/e Technische Universiteit Eindhoven University of Technology

Position tracking

Error definition

Express tracking error in body fixed frame of the reference:

$$\begin{bmatrix} \rho_e \\ \nu_e \end{bmatrix} = \begin{bmatrix} R_r^T(\rho_r - \rho) \\ \nu_r - R_r^T R \nu \end{bmatrix}$$

Tracking error dynamics

$$\begin{aligned} \dot{\rho}_e &= -S(\omega_r)\rho_e + \nu_e \\ \dot{\nu}_e &= -S(\omega_r)\nu_e + \underbrace{\tfrac{f}{m} R_r^T R e_3 - \tfrac{f_r}{m} e_3}_{u} \end{aligned}$$

Technische Universiteit Eindhoven University of Technology

Position tracking

Error definition

Express tracking error in body fixed frame of the reference:

$$\begin{bmatrix} \rho_e \\ \nu_e \end{bmatrix} = \begin{bmatrix} R_r^T(\rho_r - \rho) \\ \nu_r - R_r^T R \nu \end{bmatrix}$$

Tracking error dynamics

$$\begin{aligned} \dot{\rho}_e &= -S(\omega_r)\rho_e + \nu_e \\ \dot{\nu}_e &= -S(\omega_r)\nu_e + \underbrace{\frac{f}{m} R_r^T R e_3 - \frac{f_r}{m} e_3}_{u} \end{aligned}$$

Failing alternative

Express tracking error in body fixed frame of the drone:

$$\begin{bmatrix} \rho_e \\ \nu_e \end{bmatrix} = \begin{bmatrix} R^T(\rho - \rho_r) \\ \nu - R^T R_r \nu_r \end{bmatrix}$$

Tracking error dynamics

$$\begin{aligned} \dot{\rho}_e &= -S(\omega)\rho_e + \nu_e \\ \dot{\nu}_e &= -S(\omega)\nu_e + \underbrace{\frac{f_r}{m} R^T R_r e_3 - \frac{f}{m} e_3}_{\text{no full control}} \end{aligned}$$

Technische Universiteit Eindhoven University of Technology

Result

Consider the dynamics

$$\dot{\rho}_e = -S(\omega_r)\rho_e + \nu_e$$
$$\dot{\nu}_e = -S(\omega_r)\nu_e + u$$

in closed loop with the dynamic state feedback

$$\begin{aligned} u &= R_r^T(K_P P_e + K_p p_e) - R_r^T K_P R_r \sigma_1(\rho_e + R_r^T P_e) \\ &\quad - k_\rho \sigma_2(\rho_e + R_r^T P_e) - K_\nu \sigma_3(\nu_e + R_r^T p_e) \\ \dot{P}_e &= p_e \\ \dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\rho_e + R_r^T P_e) \end{aligned}$$

where $K_P = K_P^T > 0$, $K_p = K_p^T > 0$, $K_\nu = K_\nu^T > 0$, and $k_\rho > 0$.

The origin of the closed-loop system is UGAS.

Proof

Define $\bar{\rho}_e = \rho_e + R_r^T P_e$, $\bar{\nu}_e = \nu_e + R_r^T p_e$. Then

$$\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e)
\end{aligned}$$

Technische Universiteit Eindhoven University of Technology

Proof

Define $\bar{\rho}_e = \rho_e + R_r^T P_e$, $\bar{\nu}_e = \nu_e + R_r^T p_e$. Then

$$\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e)
\end{aligned}$$

Differentiating $V_1(\bar{\rho}_e, \bar{\nu}_e) = k_\rho V_{\sigma_2}(\bar{\rho}_e) + \frac{1}{2}\bar{\nu}_e^T \bar{\nu}_e$ yields

$$\dot{V}_1(\bar{\rho}_e, \bar{\nu}_e) = -\bar{\nu}_e^T K_\nu \sigma_3(\bar{\nu}_e) = Y_1(\bar{\nu}_e) \leq 0$$

Proof

Define $\bar{\rho}_e = \rho_e + R_r^T P_e$, $\bar{\nu}_e = \nu_e + R_r^T p_e$. Then

$$\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e)
\end{aligned}$$

Differentiating $V_1(\bar{\rho}_e, \bar{\nu}_e) = k_\rho V_{\sigma_2}(\bar{\rho}_e) + \frac{1}{2}\bar{\nu}_e^T \bar{\nu}_e$ yields

$$\dot{V}_1(\bar{\rho}_e, \bar{\nu}_e) = -\bar{\nu}_e^T K_\nu \sigma_3(\bar{\nu}_e) = Y_1(\bar{\nu}_e) \leq 0$$

Differentiating $V_2(\bar{\rho}_e, \bar{\nu}_e) = \bar{\nu}_e^T \bar{\rho}_e$ yields

$$\dot{V}_2(\bar{\rho}_e, \bar{\nu}_e) = \bar{\nu}_e^T \bar{\nu}_e - \bar{\rho}_e^T k_\nu \sigma_3(\bar{\nu}_e) - \bar{\rho}_e^T k_\rho \sigma_2(\bar{\rho}_e) = Y_2(\bar{\rho}_e, \bar{\nu}_e).$$

Position tracking

Proof

Define $\bar{\rho}_e = \rho_e + R_r^T P_e$, $\bar{\nu}_e = \nu_e + R_r^T p_e$. Then

$$\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e)
\end{aligned}$$

Differentiating $V_1(\bar{\rho}_e, \bar{\nu}_e) = k_\rho V_{\sigma_2}(\bar{\rho}_e) + \frac{1}{2}\bar{\nu}_e^T \bar{\nu}_e$ yields

$$\dot{V}_1(\bar{\rho}_e, \bar{\nu}_e) = -\bar{\nu}_e^T K_\nu \sigma_3(\bar{\nu}_e) = Y_1(\bar{\nu}_e) \leq 0$$

Differentiating $V_2(\bar{\rho}_e, \bar{\nu}_e) = \bar{\nu}_e^T \bar{\rho}_e$ yields

$$\dot{V}_2(\bar{\rho}_e, \bar{\nu}_e) = \bar{\nu}_e^T \bar{\nu}_e - \bar{\rho}_e^T k_\nu \sigma_3(\bar{\nu}_e) - \bar{\rho}_e^T k_\rho \sigma_2(\bar{\rho}_e) = Y_2(\bar{\rho}_e, \bar{\nu}_e).$$

Use a nested Matrosov result and a cascaded result to show UGAS

Desired thrust and orientation

From position tracking we have desired virtual input $u = \frac{f}{m} R_r^T R e_3 - \frac{f_r}{m} e_3$.

Then $f = \|mu + f_r e_3\|$ and $R_r^T R e_3 = \frac{mu+f_r e_3}{\|mu+f_r e_3\|} = f_d = \begin{bmatrix} f_{d1} & f_{d2} & f_{d3} \end{bmatrix}^T$.

Technische Universiteit Eindhoven University of Technology

Attitude control

Desired thrust and orientation

From position tracking we have desired virtual input $u = \frac{f}{m} R_r^T R e_3 - \frac{f_r}{m} e_3$.

Then $f = \|mu + f_r e_3\|$ and $R_r^T R e_3 = \frac{mu + f_r e_3}{\|mu + f_r e_3\|} = f_d = \begin{bmatrix} f_{d1} & f_{d2} & f_{d3} \end{bmatrix}^T$.

Since $0 < f_r^{\min} \leq f_r(t)$, we can make $\|u\| \leq \frac{f_r^{\min} - \epsilon}{m}$, so $f_{d3} > 0$.

Desired thrust and orientation

From position tracking we have desired virtual input $u = \frac{f}{m} R_r^T R e_3 - \frac{f_r}{m} e_3$.

Then $f = \|mu + f_r e_3\|$ and $R_r^T R e_3 = \frac{mu + f_r e_3}{\|mu + f_r e_3\|} = f_d = \begin{bmatrix} f_{d1} & f_{d2} & f_{d3} \end{bmatrix}^T$.

Since $0 < f_r^{\min} \leq f_r(t)$, we can make $\|u\| \leq \frac{f_r^{\min} - \epsilon}{m}$, so $f_{d3} > 0$. Define

$$R_d = \begin{bmatrix} 1 - \frac{f_{d1}^2}{1+f_{d3}} & -\frac{f_{d1} f_{d2}}{1+f_{d3}} & f_{d1} \\ -\frac{f_{d1} f_{d2}}{1+f_{d3}} & 1 - \frac{f_{d2}^2}{1+f_{d3}} & f_{d2} \\ -f_{d1} & -f_{d2} & f_{d3} \end{bmatrix} \in SO(3) \qquad \omega_d = \begin{bmatrix} -\dot{f}_{d2} + \frac{f_{d2} \dot{f}_{d3}}{1+f_{d3}} \\ \dot{f}_{d1} - \frac{f_{d1} \dot{f}_{d3}}{1+f_{d3}} \\ \frac{f_{d2} \dot{f}_{d1} - f_{d1} \dot{f}_{d2}}{1+f_{d3}} \end{bmatrix}$$

Then $R_d e_3 = f_d$. Note: R_d rotates from e_3 to f_d in spanned plane.

Attitude control

Desired thrust and orientation

From position tracking we have desired virtual input $u = \frac{f}{m} R_r^T R e_3 - \frac{f_r}{m} e_3$.

Then $f = \|mu + f_r e_3\|$ and $R_r^T R e_3 = \frac{mu + f_r e_3}{\|mu + f_r e_3\|} = f_d = \begin{bmatrix} f_{d1} & f_{d2} & f_{d3} \end{bmatrix}^T$.

Since $0 < f_r^{\min} \leq f_r(t)$, we can make $\|u\| \leq \frac{f_r^{\min} - \epsilon}{m}$, so $f_{d3} > 0$. Define

$$R_d = \begin{bmatrix} 1 - \frac{f_{d1}^2}{1+f_{d3}} & -\frac{f_{d1} f_{d2}}{1+f_{d3}} & f_{d1} \\ -\frac{f_{d1} f_{d2}}{1+f_{d3}} & 1 - \frac{f_{d2}^2}{1+f_{d3}} & f_{d2} \\ -f_{d1} & -f_{d2} & f_{d3} \end{bmatrix} \in SO(3) \qquad \omega_d = \begin{bmatrix} -\dot{f}_{d2} + \frac{f_{d2} \dot{f}_{d3}}{1+f_{d3}} \\ \dot{f}_{d1} - \frac{f_{d1} \dot{f}_{d3}}{1+f_{d3}} \\ \frac{f_{d2} \dot{f}_{d1} - f_{d1} \dot{f}_{d2}}{1+f_{d3}} \end{bmatrix}$$

Then $R_d e_3 = f_d$. Note: R_d rotates from e_3 to f_d in spanned plane.

Define attitude errors:

$$R_e = R_d^T (R_r^T R) \qquad \omega_e = \omega - R^T R_r \omega_r - R_e^T \omega_d$$

Attitude error dynamics

$$\begin{aligned}\dot{R}_e &= R_e S(\omega_e) \\ J\dot{\omega}_e &= \tau - JR^T R_r J^{-1}[S(J\omega_r)\omega_r + \tau_r] + S(J\omega)\omega \\ &\quad + JS(\omega_e)[\omega - \omega_e] + JR_e^T[S(\omega_d)R_d^T\omega_r - \dot{\omega}_d].\end{aligned}$$

TU/e Technische Universiteit Eindhoven University of Technology

Attitude control

Attitude error dynamics

$$\dot{R}_e = R_e S(\omega_e)$$

$$J\dot{\omega}_e = \tau - JR^T R_r J^{-1}[S(J\omega_r)\omega_r + \tau_r] + S(J\omega)\omega$$
$$+ JS(\omega_e)[\omega - \omega_e] + JR_e^T[S(\omega_d)R_d^T\omega_r - \dot{\omega}_d].$$

Controller (standard)

$$\tau = -K_\omega\omega_e + K_R\sum_{i=1}^{3} k_i(e_i \times R_e^T e_i) + JR^T R_r J^{-1}[S(J\omega_r)\omega_r + \tau_r]$$
$$- S(J\omega)\omega - JS(\omega_e)[\omega - \omega_e] - JR_e^T[S(\omega_d)R_d^T\omega_r - \dot{\omega}_d]$$

with distinct $k_i > 0$ and $K_\omega = K_\omega^T > 0$, $K_R = K_R^T > 0$.

Result: $(R_e, \omega_e) = (I, 0)$ ULES and UaGAS

Combined result

Cascaded system

$$
\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e) + \frac{f}{m} R_r^T R(I - R_e^T) e_3 \\
\dot{R}_e &= R_e S(\omega_e) \\
J\dot{\omega}_e &= -K_\omega \omega_e + K_R \sum_{i=1}^{3} k_i (e_i \times R_e^T e_i)
\end{aligned}
$$

Combined result

Cascaded system

$$
\begin{aligned}
\dot{P}_e &= p_e \\
\dot{p}_e &= -K_P P_e - K_p p_e + K_P R_r \sigma_1(\bar{\rho}_e) \\
\dot{\bar{\rho}}_e &= -S(\omega_r)\bar{\rho}_e + \bar{\nu}_e \\
\dot{\bar{\nu}}_e &= -S(\omega_r)\bar{\nu}_e - k_\rho \sigma_2(\bar{\rho}_e) - K_\nu \sigma_3(\bar{\nu}_e) + \frac{f}{m} R_r^T R(I - R_e^T) e_3 \\
\dot{R}_e &= R_e S(\omega_e) \\
J\dot{\omega}_e &= -K_\omega \omega_e + K_R \sum_{i=1}^{3} k_i (e_i \times R_e^T e_i)
\end{aligned}
$$

Result

If the functions σ_1, σ_2, σ_3, and $P_e(t_0)$ and $p_e(t_0)$ are properly chosen guaranteeing that $\|u\| \le \frac{f_r^{\min} - \epsilon}{m}$ for some $0 < \epsilon < f_r^{\min}$, then the origin $(P_e, p_e, \bar{\rho}, \bar{\nu}, R_e, \omega_e) = (0, 0, 0, 0, I, 0)$ is ULES and UaGAS.

Final result

Problem: we have solved a different problem

We have shown convergence of $(P_e, p_e, \bar{\rho}, \bar{\nu}, R_e, \omega_e)$ to $(0, 0, 0, 0, I, 0)$.
However, we need to show that $\varepsilon(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega})$ converges to 0.

Final result

Problem: we have solved a different problem

We have shown convergence of $(P_e, p_e, \bar{\rho}, \bar{\nu}, R_e, \omega_e)$ to $(0, 0, 0, 0, I, 0)$.
However, we need to show that $\varepsilon(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega})$ converges to 0.

Corollary (final result)

The derived controller also makes $\varepsilon(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega})$ converge to 0, i.e., $(\tilde{\rho}, \tilde{R}, \tilde{\nu}, \tilde{\omega})$ converges to $(0, I, 0, 0)$.

Technische Universiteit Eindhoven University of Technology

Simulations

Data

AR-Drone: $m=0.456$ [kg]
$J=\mathrm{diag}(0.0022, 0.0025, 0.0045)$ [kgm^2]
$g=9.81$ [m/s^2]

Simulations

Data

AR-Drone: m=0.456 [kg]
J=diag(0.0022,0.0025,0.0045) [kgm^2]
g=9.81 [m/s^2]

Reference trajectory: $\rho_r(t) = \begin{bmatrix} \cos t & \sin t & 1.5 + \sin t \end{bmatrix}^T$

R_r defined (from $\ddot{\rho}_r$ as presented earlier).

Technische Universiteit Eindhoven University of Technology

Simulations

Data

AR-Drone: m=0.456 [kg]
J=diag(0.0022,0.0025,0.0045) [kgm^2]
g=9.81 [m/s^2]

Reference trajectory: $\rho_r(t) = \begin{bmatrix} \cos t & \sin t & 1.5 + \sin t \end{bmatrix}^T$

R_r defined (from $\ddot{\rho}_r$ as presented earlier).

Initial conditions:

$$\rho(t_0) = \begin{bmatrix} -1 \\ 0.7 \\ 4 \end{bmatrix} \qquad R(t_0) = \begin{bmatrix} -0.25 & -0.433 & 0.866 \\ 0.533 & -0.808 & -0.25 \\ 0.808 & 0.34 & 0.433 \end{bmatrix}$$

$$\nu(t_0) = \begin{bmatrix} 0.1 \\ -0.8 \\ 0.7 \end{bmatrix} \qquad \omega(t_0) = \begin{bmatrix} -1 \\ 0.3 \\ -2 \end{bmatrix}.$$

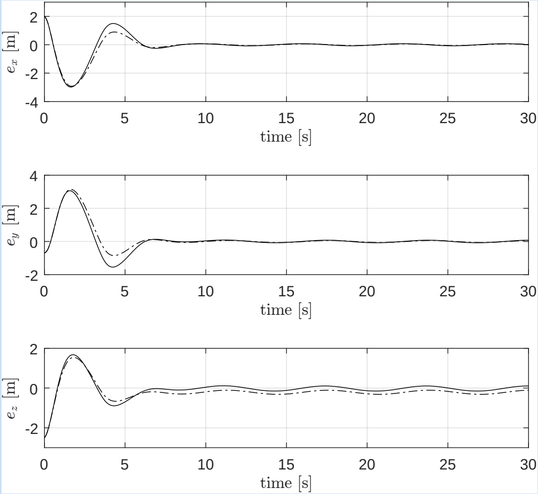
Simulations

Disturbances (more realistic)

- Mass discrepancy: $m = 0.456$[kg], $m_r = 0.48$[kg].
- Add model of sensors and actuators
  - sampling
  - delays
  - noisy measurements
- Use filtered measurements

Simulations

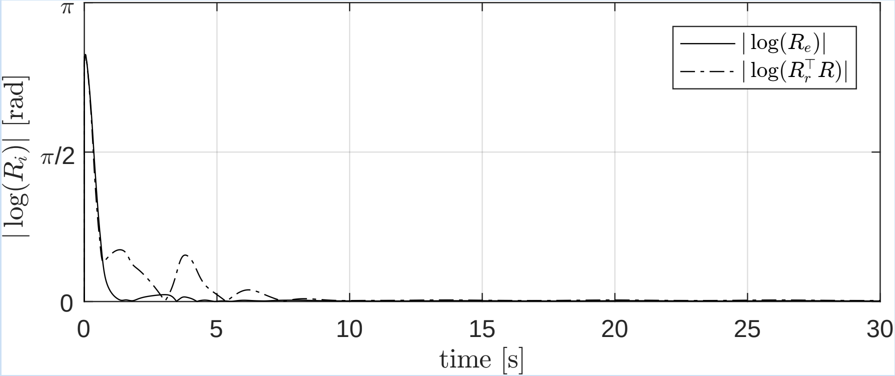
Results: Filtered errors $e = \rho_r(t) - \rho(t)$ in the inertial frame



- without integral control (dashed)
- with integral control (solid)

Simulations

Results: Attitude errors (metric: angle of rotation)



- with respect to the desired attitude, R_e, (solid),
- with respect to the reference attitude, $R_r^T R$, (dashed)

Experiments

[video]

Experiments

[video]

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances
  - difference in actual and expected mass

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances
  - difference in actual and expected mass
  - used sampled, delayed, and noisy measurements.

Conclusions

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances
  - difference in actual and expected mass
  - used sampled, delayed, and noisy measurements.
- Implemented on AR.Drone 2.0.

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances
  - difference in actual and expected mass
  - used sampled, delayed, and noisy measurements.
- Implemented on AR.Drone 2.0.
  - works well at low velocities

Conclusions

Conclusions

- Avoid singularities of Euler angles, ambiguity of quaternions, allows for large angular maneuvers.
- Explicitly took into account the constraint of non-zero total thrust in our controller design
- uniform almost global asymptotic stability on SE(3).
- Validated by simulations with added disturbances
  - difference in actual and expected mass
  - used sampled, delayed, and noisy measurements.
- Implemented on AR.Drone 2.0.
  - works well at low velocities
  - noticeable mismatch at high velocities

Future work

Future work

- Extend to model including velocity-dependent disturbance

Future work

- Extend to model including velocity-dependent disturbance
- Extend the state feedback controller to an output feedback controller (body-fixed velocity ν not available for measurement).