Dynamics and Control of Quadrotor UAV

F.L. Lewis, NAI

Moncrief-O'Donnell Chair, UTA Research Institute (UTARI)
The University of Texas at Arlington, USA

and

Hai-Long Pei

Director, Key Laboratory of Autonomous Systems and Network Control, MoE
South China University of Technology, Guangzhou

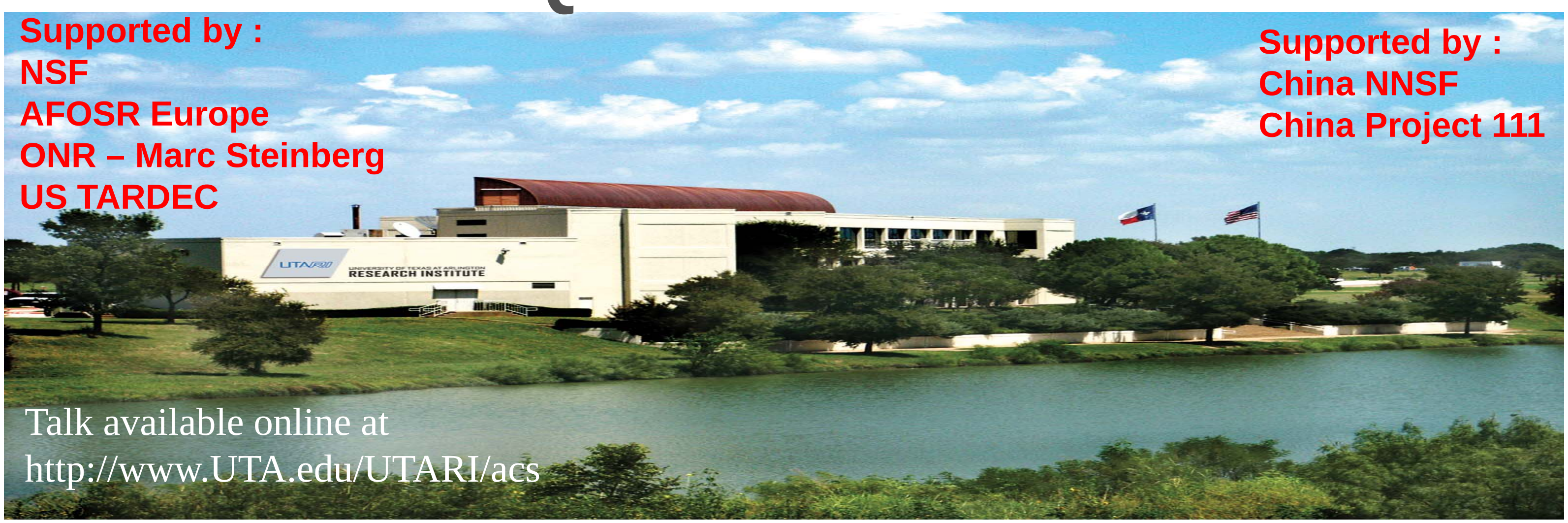
UTARI UNIVERSITY OF TEXAS AT ARLINGTON RESEARCH INSTITUTE

The University of Texas ARLINGTON

F.L. Lewis, NAI

Moncrief-O'Donnell Chair, UTA Research Institute (UTARI)
The University of Texas at Arlington, USA
and
Qian Ren Consulting Professor, State Key Laboratory of Synthetical Automation for Process Industries
Northeastern University, Shenyang, China

Dynamics and Control of Quadrotor UAV



J Intell Robot Syst (2009) 56:127–151
DOI 10.1007/s10846-009-9331-0

Backstepping Approach for Controlling a Quadrotor Using Lagrange Form Dynamics

Abhijit Das · Frank Lewis · Kamesh Subbarao

Published in IET Control Theory and Applications
Received on 2nd January 2008
Revised on 7th August 2008
doi: 10.1049/iet-cta:20080002

IET Control Theory Appl., 2009, Vol. 3, Iss. 3, pp. 303–314

ISSN 1751-8644

Dynamic inversion with zero-dynamics stabilisation for quadrotor control

A. Das[1] *K. Subbarao*[2] *F. Lewis*[1]

[1]*Automation and Robotics Research Institute, The University of Texas at Arlington, 7300 Jack Newell Blvd. S., Fort Worth, TX, 76118, USA*

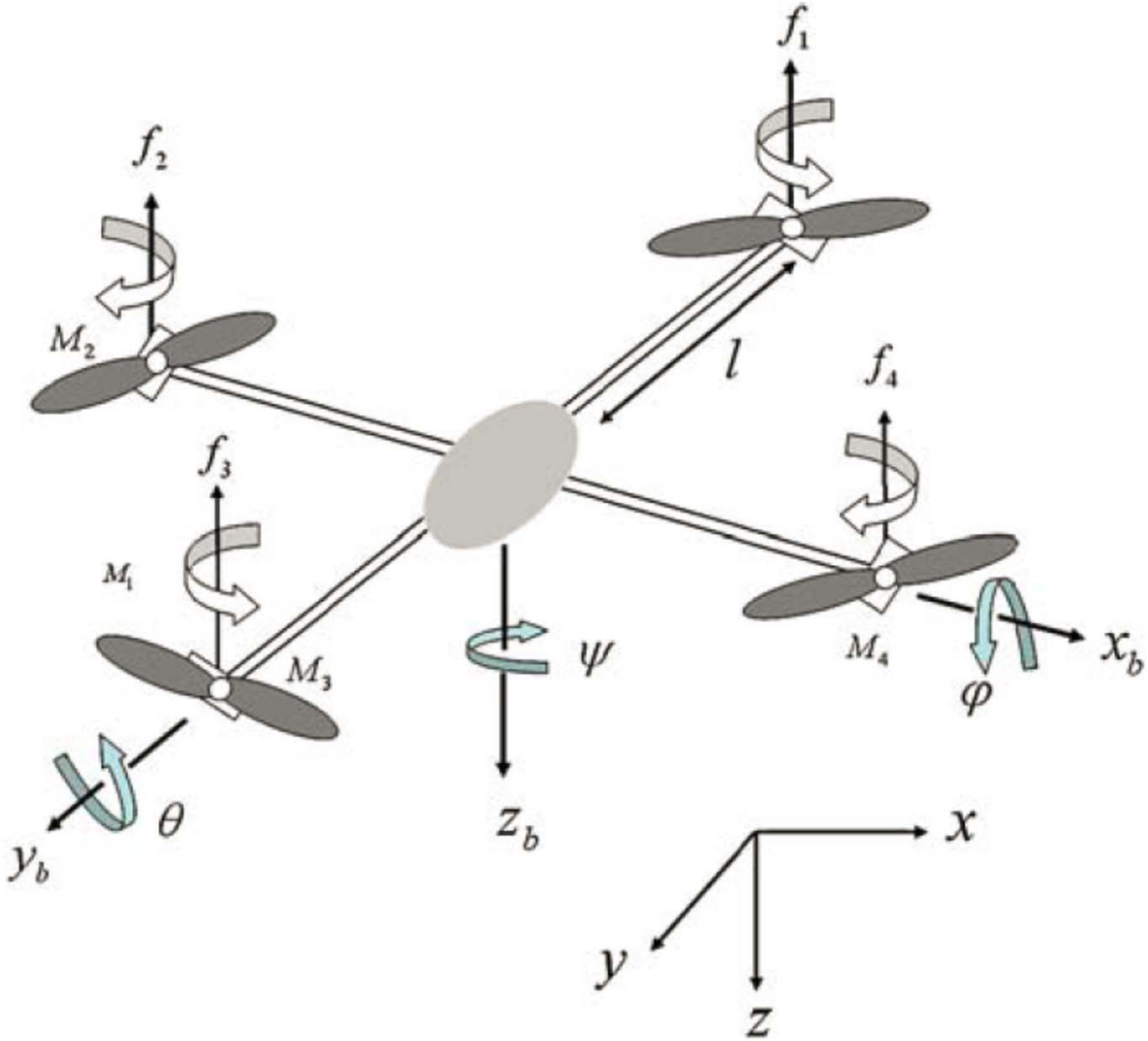
[2]*Department of Mechanical and Aerospace Engineering, The University of Texas at Arlington, 500 W. First Street, Arlington, TX, 76019, USA*

E-mail: adas@arri.uta.edu

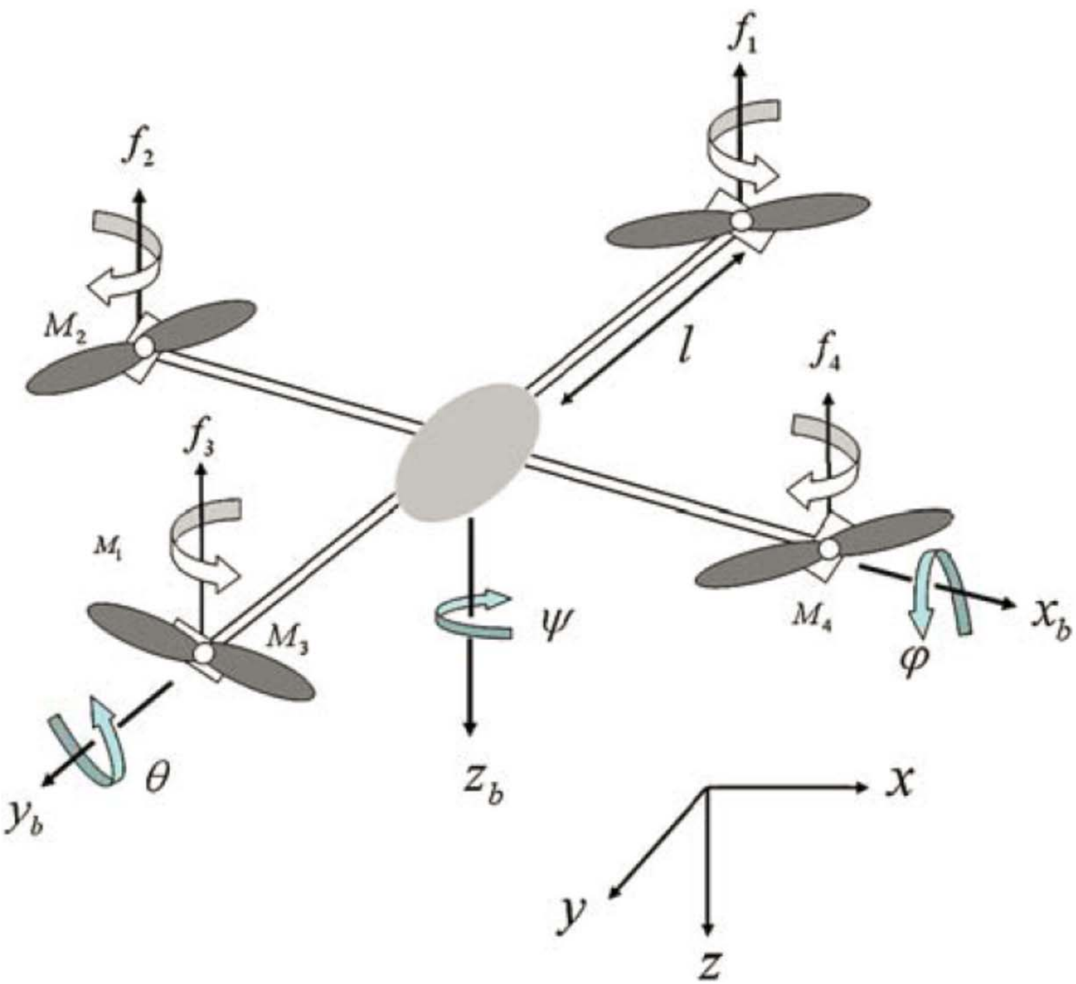
AR Drone Parrot

Crazyflie

3D Robotics Octocopter



f_1
f_2
M_2
l
f_4
f_3
M_1
ψ
M_3
M_4
x_b
φ
θ
y_b
z_b
x
y
z



Body Axes
Vs. earth-fixed axes

The Quadrotor States

$$X = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \end{bmatrix}$$

Position – navigational states (x, y, z)

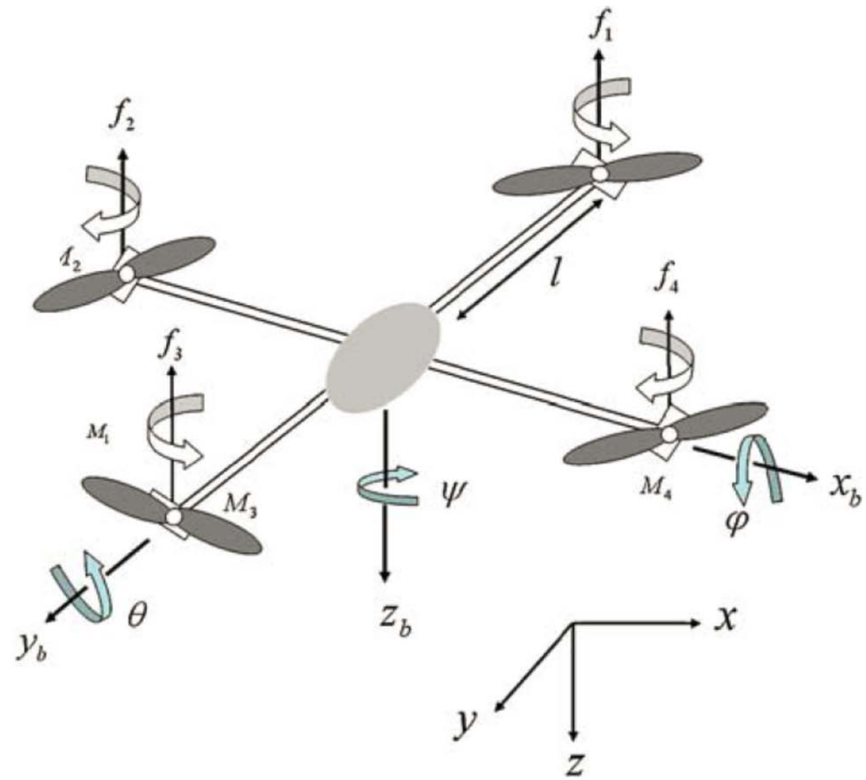
Roll, Pitch, yaw (ϕ, θ, ψ) — Angular position – attitudes

Control distribution from 4 actuator rotors to lift and torques

$$\begin{pmatrix} u \\ \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 \\ l & 0 & -l & 0 \\ 0 & -l & 0 & l \\ c & -c & c & -c \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix}}_{f} = Mf$$

Lift $\quad u$

torques $\quad \tau = \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix}$



The Quadrotor States

$$X = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \end{bmatrix}$$

Position – navigational states (x, y, z)

Roll (ϕ)

Pitch (θ)

yaw (ψ)

Angular position – attitudes (ϕ, θ, ψ)

The Quadrotor Controls

Lift u

torques $\tau = \begin{bmatrix} \tau_{\varphi} \\ \tau_{\theta} \\ \tau_{\psi} \end{bmatrix}$

6 states and 4 controls = under-actuated system

Position states- navigation states

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Attitude states

$$\eta = \begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix}$$

Quadrotor equations of motion

Position subsystem

$$m\ddot{\xi} = u \begin{pmatrix} -\sin\theta \\ \cos\theta \sin\varphi \\ \cos\theta \cos\varphi \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix}$$

Angle subsystem

$$J\ddot{\eta} = -C\left(\eta, \dot{\eta}\right)\dot{\eta} + \tau$$

Position subsystem

$$m\ddot{\xi} = u\begin{pmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} + F$$

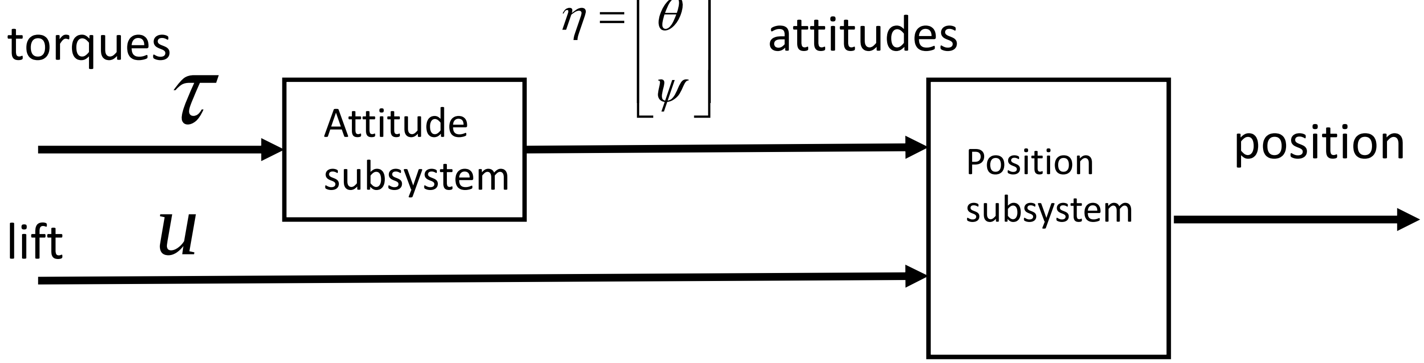
Attitude subsystem

$$J\ddot{\eta} = -C\left(\eta, \dot{\eta}\right)\dot{\eta} + \tau$$

Virtual control input for position subsystem

$$F = u\begin{pmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \end{pmatrix}$$

$$\eta = \begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix}$$



$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Backstepping Control Design

$$m\ddot{\xi} = u\begin{pmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} + F = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} + F_d - (F_d - F)$$

$$m\ddot{\xi} = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} + F_d - \tilde{F}$$

Where ideal virtual force input is F_d

And force mismatch is $\tilde{F} = F_d - F$

$$J\ddot{\eta} = -C\left(\eta, \dot{\eta}\right)\dot{\eta} + \tau$$

Backstepping Control Design

1. Pick desired virtual force F_d to make position dynamics track desired positions ξ_d

$$m\ddot{\xi} = \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} + F_d - \tilde{F}$$

2. Pick actual control – the torques τ - to make force error $\tilde{F}$ go to zero

$$J\ddot{\eta} = -C\left(\eta, \dot{\eta}\right)\dot{\eta} + \tau$$

An Inverse Kinematics problam

Given F_d find required attitude angles and lift

$$\begin{pmatrix} -u_d \sin\theta_d \\ u_d \cos\theta_d \sin\varphi_d \\ u_d \cos\theta_d \cos\varphi_d \end{pmatrix} = \hat{F}_d$$

define $$a = u_d \sin\theta_d,\, b = u_d \cos\theta_d$$

then $$\begin{pmatrix} -a \\ b\,\sin\varphi_d \\ b\,\cos\varphi_d \end{pmatrix} = \hat{F}_d(t) \equiv \begin{pmatrix} f_{x_d}(t) \\ f_{y_d}(t) \\ f_{z_d}(t) \end{pmatrix}$$

So that

$$a = -f_{x_d}(t)$$

$$b = \sqrt{f_{y_d}^2(t) + f_{z_d}^2(t)}$$

Then compute

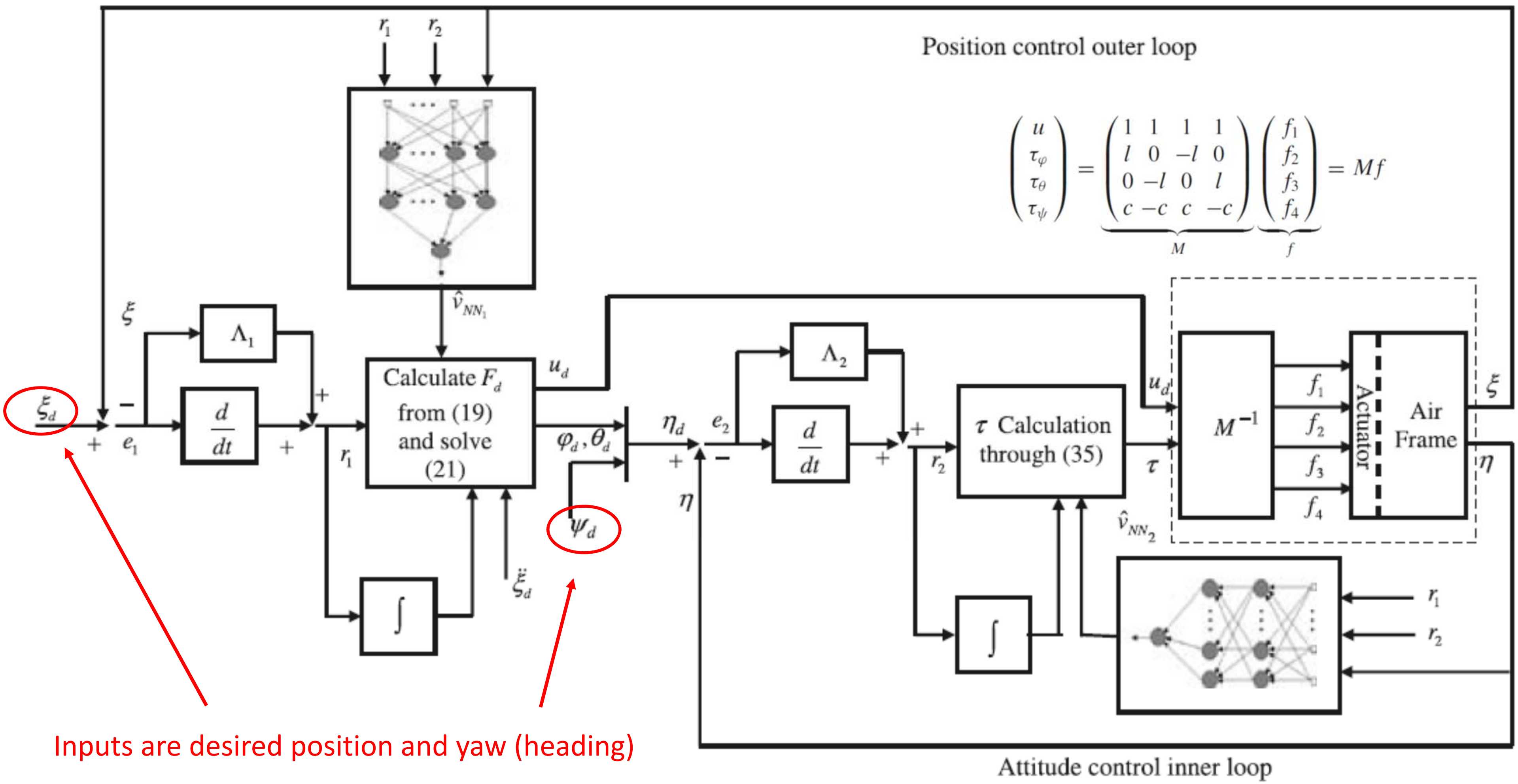
$$u_d = \sqrt{a^2 + b^2}$$

$$\theta_d = \tan^{-1}\left(\frac{a}{b}\right)$$

$$\varphi_d = \tan^{-1}\left(\frac{f_{y_d}}{f_{z_d}}\right)$$

Note that ψ is not involved here!

Backstepping Controller- 2 loops



Inputs are desired position and yaw (heading)


聚龍苑
聚龙苑休闲吧