



SEP

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO  
Instituto Tecnológico de La Laguna

Torreón, Coah., 12 de diciembre/2017  
Dependencia: DEPI/CPCIE  
Oficio: DEPI/CPCIE/163/2017  
Asunto: Autorización de impresión de tesis.

C. Sergio López Hernández  
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA.  
PRESENTE

Después de haber sometido a revisión su trabajo de tesis titulado:

**"Propuesta de un Esquema de Control Neuronal Wavelet para un Quadrotor"**

Habiendo cumplido con todas las indicaciones que el Jurado revisor de tesis hizo, se le comunica que se le concede la autorización con número de registro **RIITEC: (21)-TMCIE-2017**, para que proceda a la impresión del mismo.

**ATENTAMENTE**

EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACIÓN



SECRETARÍA DE EDUCACIÓN PÚBLICA  
INSTITUTO TECNOLÓGICO DE LA LAGUNA

**DR. ARMANDO LONGORIA DE LA TORRE**  
Jefe de la División de Estudios de Posgrado e Investigación  
del Instituto Tecnológico de la Laguna

AIT/140



**DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

**“Propuesta de un Esquema de Control Neuronal  
Wavelet para un Quadrotor”**

POR

**Sergio López Hernández**

**TESIS**

**PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL  
GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

**DIRECTOR DE TESIS**

**Dr. Francisco Jurado Zamarripa**

**ISSN: 0188-9060**



**RIITEC: (21)-TMCIE-2017**

Torreón, Coahuila, México  
Diciembre 2017



Torreón, Coah., 07 de Noviembre de 2017

**DR. ARMANDO LONGORIA DE LA TORRE**  
**JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

Por medio de la presente, hacemos de su conocimiento que después de haber sometido a revisión el trabajo de tesis titulado:

**"Propuesta de un Esquema de Control Neuronal Wavelet para un Quadrotor"**

Desarrollado por el C. Sergio López Hernández, con número de control M1513049 y habiendo cumplido con todas las correcciones que se le indicaron, estamos de acuerdo que se le conceda la autorización de la fecha de examen de grado para que proceda a la impresión de la misma.

**ATENTAMENTE**  
EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACIÓN

**Dr. Francisco Jurado Zamarriza**  
Asesor/Director de Tesis

**Dr. Miguel A. Llamas Leal**  
Comité Tutorial

**Dr. Ricardo E. Campa Cocom**  
Comité Tutorial

**Dr. Alejandro E. Dzui López**  
Comité Tutorial



# TESIS

Sergio López Hernández

8 de diciembre de 2017

# Agradecimientos

# Abstract

In this work, the obtaining of the dynamic model of a quadrotor is presented, then we apply a desing direct model reference adaptive control to archive the control of the attitude, plus a classic PID to control the translation.

High-order recurrent neural networks (RHONN) are used, trained with an adaptive law called error filtering, as a solution alternative for the problem of tracking trajectories of nonlinear systems in continuous time, under a decentralized approach.

Then, a neuronal structure is developed considering functions wavelet as activation functions. Based on the recurrent wavelet neural networks of the first order (RWFON) a control law is designed using the technique of backstepping under the assumption of that states must be measurable.

Finally, to evaluate the performances of these schemes, simulation tests are developed on the dynamic model of a quadrotor UAV considering the parameters of the Q-ball 2 platform of Quanser, to then carry out experimental tests on the Q-ball 2 platform of Quanser.

# Índice general

|   |           |
|---|-----------|
| <b>1. Introducción</b>  | <b>1</b>  |
| 1.1. Estado del arte  | 1         |
| 1.2. Objetivos generales  | 3         |
| 1.3. Objetivos particulares   | 3         |
| 1.4. Organización del documento de tesis                                      | 3         |
| <b>2. Modelado de un <i>Quadrotor</i> UAV</b>                                 | <b>5</b>  |
| 2.1. Descripción del modelo dinámico  | 5         |
| 2.2. Modelo Newton-Euler  | 6         |
| 2.2.1. Modelado cinemático  | 6         |
| 2.2.2. Modelo dinámico  | 9         |
| <b>3. Controlador adaptable por modelo de referencia</b>                      | <b>15</b> |
| 3.1. Introducción   | 15        |
| 3.2. Diseño de un MRAC para un sistema MIMO                                   | 16        |
| 3.3. Diseño descentralizado MRAC directo aplicado a un <i>quadrotor</i>       | 19        |
| 3.4. Resultados de simulación   | 22        |
| <b>4. Redes neuronales recurrentes de alto orden (RHONNs)</b>                 | <b>25</b> |
| 4.1. Introducción   | 25        |
| 4.2. Estructura de la RHONN   | 26        |
| 4.3. Propiedades de aproximación de la RHONN                                  | 29        |
| 4.4. Algoritmo de aprendizaje   | 32        |
| 4.5. RHONN entrenada vía FE   | 32        |
| <b>5. Identificación y control neuronal de un <i>quadrotor</i> vía RHONNs</b> | <b>35</b> |
| 5.1. Introducción   | 35        |
| 5.2. Identificación neuronal descentralizada                                  | 35        |
| 5.2.1. Descentralización del modelo dinámico del <i>quadrotor</i>             | 36        |
| 5.2.2. Arquitectura de la red   | 36        |
| 5.2.3. Red neuronal descentralizada   | 37        |
| 5.3. Control <i>backstepping</i>  | 38        |
| 5.4. Controlador neuronal RHONN   | 41        |
| 5.4.1. Diseño del controlador neuronal RHONN                                  | 42        |
| 5.5. Control de movimiento traslacional (Cinemática inversa)                  | 45        |



|  |           |
|--|-----------|
| 5.6. Resultados de simulación . . . . .  | 46        |
| 5.6.1. Resultados de identificación . . . . .  | 46        |
| 5.6.2. Resultados de control . . . . .   | 48        |
| 5.7. Resultados experimentales para la identificación vía RHONNs . . . . .               | 50        |
| <b>6. Redes neuronales wavelet</b> . . . . .   | <b>59</b> |
| 6.1. Introducción . . . . .  | 59        |
| 6.2. Fundamentos de <i>wavelets</i> . . . . .  | 60        |
| 6.2.1. Funciones <i>wavelet</i> . . . . .  | 60        |
| 6.2.2. Transformada <i>wavelet</i> continua . . . . .                                    | 60        |
| 6.2.3. Análisis multiresolución . . . . .  | 61        |
| 6.3. Red neuronal <i>wavelet</i> recurrente de primer orden . . . . .                    | 62        |
| 6.3.1. Propiedades de aproximación para la RWFONN . . . . .                              | 64        |
| <b>7. Identificación y control neuronal <i>wavelet</i> de un quadrotor UAV</b> . . . . . | <b>67</b> |
| 7.1. Arquitectura de la red . . . . .  | 67        |
| 7.2. Identificación neuronal <i>wavelet</i> descentralizada . . . . .                    | 67        |
| 7.3. Controlador neuronal <i>wavelet</i> . . . . .                                       | 68        |
| 7.4. Resultados en simulación . . . . .  | 69        |
| 7.4.1. Resultados de identificación . . . . .  | 70        |
| 7.4.2. Resultados de control . . . . .   | 72        |
| 7.5. Resultados experimentales . . . . .   | 73        |
| <b>8. Conclusiones</b> . . . . .   | <b>81</b> |
| <b>Producción académica</b> . . . . .  | <b>83</b> |
| <b>Trabajos futuros</b> . . . . .  | <b>84</b> |

# Índice de figuras

|   |    |
|---|----|
| 2.1. Marcos de referencia. . . . .  | 7  |
| 3.1. Diagrama del MRAC para cada subsistema. . . . .  | 23 |
| 3.2. Desempeño del MRAC directo descentralizado para control de orientación. . . . .                  | 23 |
| 3.3. Desempeño del esquema de control PID descentralizado para seguimiento de trayectoria. . . . .    | 24 |
| 3.4. Evolución de las ganancias estimadas de <i>feedback</i> y <i>feedforward</i> . . . . .           | 24 |
| 4.1. Estructura de una red RNN. . . . .   | 27 |
| 4.2. Ejemplo gráfico de funciones de activación. . . . .  | 28 |
| 4.3. Estructura de una RHONN. . . . .   | 29 |
| 5.1. Diagrama a bloques del sistema (5.15)-(5.16). . . . .  | 38 |
| 5.2. Diagrama a bloques del sistema (5.15)-(5.16) introduciendo $\alpha(\chi)$ . . . . .              | 39 |
| 5.3. <i>Backstepping</i> de $-\alpha(\chi)$ a través del integrador. . . . .                          | 40 |
| 5.4. Diagrama del control RHONN descentralizado propuesto. . . . .                                    | 44 |
| 5.5. Diagrama del controlador RHONN descentralizado propuesto. . . . .                                | 46 |
| 5.6. Error de identificación RHONN descentralizado de las dinámicas de alabeo. . . . .                | 47 |
| 5.7. Error de identificación RHONN descentralizado de las dinámicas de cabeceo. . . . .               | 48 |
| 5.8. Error de identificación RHONN descentralizado de las dinámicas de guiñada. . . . .               | 49 |
| 5.9. Error de identificación RHONN descentralizado de las dinámicas traslacionales de $z$ . . . . .   | 49 |
| 5.10. Error de identificación RHONN descentralizado de las dinámicas traslacionales de $x$ . . . . .  | 50 |
| 5.11. Error de identificación RHONN descentralizado de las dinámicas traslacionales de $y$ . . . . .  | 50 |
| 5.12. Desempeño del esquema de control neuronal descentralizado propuesto. . . . .                    | 51 |
| 5.13. Vista superior del desempeño del esquema de control neuronal descentralizado propuesto. . . . . | 52 |
| 5.14. Error de seguimiento y señal de control para la dinámica rotacional de alabeo. . . . .          | 52 |
| 5.15. Error de seguimiento y señal de control para la dinámica rotacional de cabeceo. . . . .         | 53 |
| 5.16. Error de seguimiento y señal de control para la dinámica rotacional de guiñada. . . . .         | 53 |
| 5.17. Error de seguimiento y señal de control para la dinámica traslacional de $z$ . . . . .          | 54 |
| 5.18. Error de seguimiento y señal de control para la dinámica traslacional de $x$ . . . . .          | 54 |
| 5.19. Error de seguimiento y señal de control para la dinámica traslacional de $y$ . . . . .          | 55 |

|  |    |
|--|----|
| 5.20. Error de identificación RHONN descentralizado de las dinámicas de alabeo.                    | 55 |
| 5.21. Error de identificación RIIONN descentralizado de las dinámicas de cabeceo.                  | 56 |
| 5.22. Error de identificación RHONN descentralizado de las dinámicas de guiñada.                   | 56 |
| 5.23. Error de identificación RIIONN descentralizado de las dinámicas traslacio-<br>nales de $z$ . | 57 |
| 5.24. Error de identificación RIIONN descentralizado de las dinámicas traslacio-<br>nales de $x$ . | 57 |
| 5.25. Error de identificación RIIONN descentralizado de las dinámicas traslacio-<br>nales de $y$ . | 58 |
| 6.1. <i>Wavelet</i> Morlet (Gabor).  | 63 |
| 7.1. Error de identificación RWFONN descentralizado de las dinámicas de alabeo.                    | 69 |
| 7.2. Error de identificación RWFONN descentralizado de las dinámicas de cabeceo.                   | 69 |
| 7.3. Error de identificación RWFONN descentralizado de las dinámicas de guiñada.                   | 70 |
| 7.4. Error de identificación RWFONN descentralizado de las dinámicas trasla-<br>cionales de $z$ .  | 70 |
| 7.5. Error de identificación RWFONN descentralizado de las dinámicas trasla-<br>cionales de $x$ .  | 71 |
| 7.6. Error de identificación RWFONN descentralizado de las dinámicas trasla-<br>cionales de $y$ .  | 71 |
| 7.7. Desempeño del esquema de control neuronal propuesto.  | 72 |
| 7.8. Vista superior del desempeño del esquema de control neuronal propuesto.                       | 73 |
| 7.9. Error de seguimiento y señal de control para la dinámica de rotación de<br>alabeo.            | 73 |
| 7.10. Error de seguimiento y señal de control para la dinámica de rotación de<br>cabeceo.          | 74 |
| 7.11. Error de seguimiento y señal de control para la dinámica de rotación de<br>guiñada.          | 75 |
| 7.12. Error de seguimiento y señal de control para la dinámica de translación de $z$ .             | 75 |
| 7.13. Error de seguimiento y señal de control para la dinámica de translación de $x$ .             | 76 |
| 7.14. Error de seguimiento y señal de control para la dinámica de translación de $y$ .             | 76 |
| 7.15. Desempeño de la RWFONN descentralizada.  | 77 |
| 7.16. Vista superior del desempeño de la RWFONN descentralizada.                                   | 77 |
| 7.17. Vista lateral del desempeño de la RWFONN descentralizada.                                    | 78 |
| 7.18. Vista frontal del desempeño de la RWFONN descentralizada.                                    | 78 |
| 7.19. Desempeño de la RWFONN descentralizada para la trayectoria completa.                         | 79 |
| 7.20. Error de seguimiento y señal de control para $\psi$ en la plataforma experimental.           | 79 |
| 7.21. Error de seguimiento y señal de control para $z$ en la plataforma experimental.              | 80 |
| 7.22. Error de seguimiento para $x$ y $y$ en la plataforma experimental.                           | 80 |

# Índice de tablas

|  |    |
|--|----|
| 5.1. Parámetros de la plataforma Qball-2               | 47 |
| 5.2. MSEs para la identificación vía RHONNs            | 48 |
| 5.3. MSEs para el seguimiento                          | 51 |
| 5.4. MSEs de la identificación experimental vía RHONNs | 53 |
| 7.1. MSEs para la identificación vía RWFONNs           | 74 |
| 7.2. MSEs para el seguimiento                          | 74 |
| 7.3. MSEs para el seguimiento vía RWFONNs              | 77 |

# Capítulo 1

## Introducción

En el presente capítulo se resume lo referente al estado del arte respecto a la identificación y control neuronal de sistemas dinámicos.

### 1.1. Estado del arte

En los últimos años ha habido un avance muy importante en el desarrollo de vehículos aéreos no tripulados (*unmanned aerial vehicles*, UAVs por sus siglas en inglés) sobre todo para su uso en futuras aplicaciones civiles. Este tipo de vehículos se está empleando en tareas de búsqueda y rescate, **vigilancia comercial**, **exploración de edificios**, entre otras. Los UAVs son muy útiles cuando se busca acceder a zonas de difícil acceso o con un determinado nivel de peligro.

Las dinámicas del *quadrotor* UAV no solo son no lineales, sino que también están fuertemente acopladas y se trata de un sistema subactuado, características que pueden dificultar el control de dicha plataforma. En otras palabras, el *quadrotor* tiene seis grados de libertad (*degree-of-freedom*, DOF por sus siglas en inglés) con solo cuatro entradas de control que consisten en una fuerza de empuje vertical y tres pares rotacionales. Para resolver el problema de control de seguimiento del *quadrotor* UAV se han propuesto muchas técnicas donde el objetivo de control consiste en describir una trayectoria deseada en el espacio tridimensional manteniendo fijo el ángulo de guiñada.

Las redes neuronales artificiales (*artificial neural networks*, ANNs por sus siglas en inglés), representan una metodología presente en muchas disciplinas como: neurociencia, matemáticas, estática, física, ciencias computacionales e ingeniería. Las ANNs encuentran aplicación en campos muy diversos como modelado, análisis de series temporales, reconocimiento de patrones, procesamiento de señales e imágenes y en control, en virtud de una propiedad importante: La habilidad de aprender a partir datos de entrada, con o sin entrenamiento [Haykin, 1994].

Sin embargo, cuando se trata de sistemas dinámicos las ANNs clásicas no constituyen una buena opción, por esta razón el interés se ha orientado hacia el uso de redes neuronales

recurrentes (*Recurrent Neural Networks*, RNNs por sus siglas en inglés) que al incluir en su estructura lazos de realimentación facilitan el modelado, la identificación y el diseño de observadores para sistemas no lineales.

En [Kosmatopoulos et al., 1992], [Kosmatopoulos et al., 1995] y [Kosmatopoulos et al., 1997] se han estudiado las propiedades de aproximación y aprendizaje de una clase de RNNs conocidas como redes neuronales recurrentes de alto orden (*Recurrent High-Order Neural Networks*, RHONNs por sus siglas en inglés). Las propiedades de estabilidad y convergencia de una RHONN como modelo de sistemas dinámicos no lineales fueron también estudiadas, demostrando que con un número suficientemente grande de conexiones de alto orden entre neuronas se puede conseguir que el modelo RHONN sea capaz de aproximar un gran número de sistemas dinámicos complejos.

Algunos de los trabajos más sobresalientes enfocados al control de *quadrotor* UAVs son enlistados a continuación. En [Bonabdallah and Siegwart, 2005] se emplean las técnicas de *backstepping* y *sliding-modes* en el control de un *quadrotor*. En [Felix et al., 2005a] se utiliza la técnica de control por modelo de articulación cerebral (*Cerebellar Model Articulation Controller*, CMAC por sus siglas en inglés) combinada con ANNs.

En [Voos, 2007] se describe el desarrollo de un sistema de control basado en una combinación de ecuaciones de Riccati dependientes del estado (*Combination of State-Dependent Riccati Equations*, SDRE por sus siglas en inglés) combinado con ANNs y aplicado a un *quadrotor* UAV. En [Nicol et al., 2008] se propone un control adaptable con ANNs para estabilizar un *quadrotor* considerando perturbaciones; este método se compara con las técnicas adaptables de zona muerta y *e-modification*. En [Dierks and Jagannathan, 2008] se propone un controlador por realimentación de estados basado en una ANN, donde ésta última copia la dinámica no lineal de un *quadrotor*. En [Dierks and Jagannathan, 2010] se propone un controlador utilizando ANNs y la técnica de realimentación de estados, la ANN se utiliza para aprender en línea la dinámica completa del *quadrotor*, incluyendo incertidumbres y no linealidades. En [Bouhali and Boudjedir, 2011] se propuso un control neuronal adaptable basado en un nuevo observador aplicado a un *quadrotor*; este esquema fue implementado mediante el uso de dos ANNs del tipo *feedforward* conectadas en paralelo para cada subsistema en el *quadrotor*. En [Mohammadi and Shahri, 2013] se implementa un control adaptable descentralizado para la estabilización de la orientación y altura de un *quadrotor*. En [Frye and Provelce, 2014] se investiga la aplicación de la técnica de control de inversión directa utilizando una ANN para el aprendizaje de la dinámica en vuelo en sustentación (*hover*) de un *quadrotor*. En [Wang et al., 2016] se propone un método de control adaptable por modelo inverso (*adaptive inverse model control method*, AIMCM por sus siglas en inglés) para lograr la identificación y control de un *quadrotor* donde dos redes neuronales retropropagadas son usadas para lograr la identificación y control. En [Bakshi and Ramachandran, 2016] se utiliza la técnica de control adaptable por modelo de referencia (*Model Reference Adaptive Control*, MRAC por sus siglas en inglés), donde un emulador de la planta mediante una ANN predice de manera continua la próxima salida de la misma.

En [Antonio-Toledo et al., 2015] se utiliza una RHONN entrenada mediante el algo-

ritmo del filtro de Kalman extendido para identificar en línea la dinámica de los motores de un *quadrotor* donde el diseño del controlador se basa en el método de control óptimo inverso. En la literatura, pocos trabajos han sido reportados referentes al uso de una RHONN para la identificación y control en tiempo continuo de un *quadrotor* UAV. En cuanto a la identificación y control de un *quadrotor* usando el esquema RWFONN no se ha reportado ningún trabajo al respecto.

## 1.2. Objetivos generales

En el presente proyecto de investigación, se persigue como objetivo la propuesta de un esquema de ANN en tiempo continuo orientado a la identificación y control de una mini-nave aérea autónoma, en particular del tipo helicóptero de cuatro rotores denominado como *quadrotor*.

## 1.3. Objetivos particulares

- Implementar un esquema de identificación neuronal descentralizado en tiempo continuo sobre un *quadrotor* UAV bajo el recurso de RHONNs y RWFONNs empleando como algoritmo de entrenamiento la técnica de filtrado del error.
- Diseñar un controlador a partir del esquema de identificación neuronal descentralizado propuesto, y validar su desempeño en simulación, pretendiendo luego su implementación sobre un prototipo experimental.

## 1.4. Organización del documento de tesis

El siguiente documento de tesis está organizado de la siguiente manera:

- En el Capítulo 2 se describe el modelo matemático de un *quadrotor* con el recurso de Newton-Euler.
- En el Capítulo 3 se propone un MRAC directo descentralizado para el control de la orientación de un *quadrotor*.
- En el Capítulo 4 se introducen conceptos básicos sobre la identificación neuronal de sistemas dinámicos. Se explica lo correspondiente a la identificación neuronal mediante una arquitectura del tipo RHONN.
- En el Capítulo 5 se analiza el esquema de identificación neuronal RHONN aplicado a un *quadrotor* UAV y los resultados obtenidos mediante la implementación en simulación y en tiempo real.
- En el Capítulo 6 se analiza teoría de las funciones *wavelet* y sus propiedades, así como las propiedades de aproximación de las RWFONN.

- En el Capítulo 7 se analiza el esquema de identificación y control neuronal *wavelet* aplicado a un *quadrotor* UAV y los resultados obtenidos mediante la implementación en simulación y en tiempo real.
- En el Capítulo 8 se presentan las conclusiones del presente trabajo.



## Capítulo 2

# Modelado de un *Quadrotor* UAV

En el presente capítulo se describe la obtención del modelo de un *quadrotor* mediante el método de Newton-Euler. El modelo matemático en cuestión es de suma importancia ya que describe las dinámicas del *quadrotor*.

### 2.1. Descripción del modelo dinámico

La estructura básica del *quadrotor* se compone de una estructura en forma de cruz con cuatro hélices en sus extremos. Las hélices delantera y trasera giran en sentido antihorario mientras que las hélices izquierda y derecha giran en sentido horario. Esta configuración de direcciones de pares opuestos elimina la necesidad de un rotor de cola (necesario para la estructura del helicóptero estándar).

A pesar de que el *quadrotor* posee 6 grados de libertad, éste es sólo equipado con cuatro hélices, por lo que, no es posible alcanzar puntos deseados para todos los grados de libertad, pero sí un máximo de cuatro. Sin embargo, gracias a su estructura resulta fácil elegir las cuatro mejores variables controlables y separarlas para facilitar el control. A continuación se presenta la descripción de los cuatro movimientos básicos que permite al *quadrotor* alcanzar cierta altura y orientación:

- El desplazamiento en  $z$  (altitud,  $u_1$  [N]) se consigue aumentando (o disminuyendo) la velocidad de todas las hélices en la misma proporción, lo que produce una fuerza vertical con referencia al marco fijo al cuerpo que eleva o baja al *quadrotor*. Si el *quadrotor* está en posición horizontal luego la dirección vertical del marco inercial y la del marco fijo al cuerpo coinciden, de lo contrario, el empuje proporcionado genera aceleraciones verticales y horizontales en el marco inercial.
- El movimiento de alabeo (*roll*,  $u_2$  [N m]) se logra aumentando (o disminuyendo) la velocidad de la hélice izquierda y disminuyendo (o aumentando) la velocidad de la hélice derecha, lo que produce un par con respecto al eje  $x$  que hace girar al *quadrotor*. El empuje vertical total es el mismo que en sustentación (*hovering*), por lo tanto ésta acción conduce a una aceleración del ángulo de alabeo.

- El movimiento de cabeceo (*pitch*,  $u_3$  [N m]) es muy similar al de alabeo y se logra aumentando (o disminuyendo) la velocidad de la hélice trasera y disminuyendo (o aumentando) la velocidad de la hélice delantera, lo que produce un par con respecto al eje  $y$  que hace girar al *quadrotor*. El empuje vertical total es el mismo que en sustentación, por lo tanto, ésta acción sólo conduce a una aceleración del ángulo de cabeceo.
- El movimiento de guiñada (*yaw*,  $u_4$  [N m]) se logra aumentando (o disminuyendo) la velocidad de las hélices trasera y delantera y disminuyendo (o aumentando) la velocidad de las hélices derecha e izquierda. Lo que produce un par con respecto al eje  $z$  que hace girar el *quadrotor*. El movimiento en guiñada es generado debido al hecho de que las hélices izquierda y derecha giran en sentido horario mientras que la delantera y la trasera rotan en sentido antihorario. Luego, cuando el empuje vertical total está desbalanceado el *quadrotor* gira alrededor del eje  $z$ . El empuje vertical total es el mismo que en sustentación, por lo tanto, éste comando sólo conduce a una aceleración del ángulo de guiñada.

## 2.2. Modelo Newton-Euler

Esta sección presenta las ecuaciones básicas que describen la posición y orientación de un *quadrotor* como un cuerpo rígido giratorio de seis grados de libertad (6-GDL) utilizando la formulación de Newton-Euler. El modelo del *quadrotor* será obtenido de acuerdo a las ecuaciones cinemáticas y dinámicas.

### 2.2.1. Modelado cinemático

Para describir el movimiento en el espacio de un cuerpo rígido de 6-GDL se definen dos marcos de referencia [Nagaty et al., 2013]:

- Marco de referencia inercial de la tierra (Marco E)
- Marco de referencia fijo al cuerpo (Marco B)

El marco E ( $O_E, x_E, y_E, z_E$ ) se utiliza para definir la posición lineal ( $\Gamma^E$  [m]) y la posición angular ( $\Theta^E$  [rad]). En el marco B ( $O_B, x_B, y_B, z_B$ ) se definen las velocidades lineales ( $V^B$  [m/s]), las velocidades angulares ( $\omega^B$  [rad/seg]), las fuerzas ( $F^B$  [N]) y los pares ( $\tau$  [N m]). Estos marcos se muestran en la Figura 2.1.

La posición lineal

$$\Gamma^E = [x \ y \ z]^T \in \mathbb{R}^3 \quad (2.1)$$

se determina por las coordenadas del marco B con respecto al marco E.

La posición angular (orientación)

$$\Theta^E = [\phi \ \theta \ \psi]^T \in \mathbb{R}^3 \quad (2.2)$$

se define por la orientación del marco B respecto al marco E.

La matriz de rotación  $\mathbf{R}_{\mathbf{E}}$  se obtiene postmultiplicando las tres matrices de rotación básicas en el siguiente orden:

$$\begin{aligned}
 \mathbf{R}_{\mathbf{E}} &= \mathbf{R}(\psi, z)\mathbf{R}(\theta, y)\mathbf{R}(\phi, x) \\
 &= \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi & c_\phi \end{bmatrix} \\
 &= \begin{bmatrix} c_\psi s_\theta & -s_\psi c_\phi - c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi s_\theta c_\phi \\ s_\psi c_\theta & c_\psi c_\phi + s_\psi s_\theta s_\phi & -c_\psi s_\phi + s_\psi s_\theta c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \in \mathbb{R}^{3 \times 3}, \quad (2.3)
 \end{aligned}$$

donde  $\mathbf{R}(\psi, z) \in \mathbb{R}^{3 \times 3}$  representa una rotación alrededor del eje  $z$ ,  $\mathbf{R}(\theta, y) \in \mathbb{R}^{3 \times 3}$  representa una rotación alrededor del eje  $y$  y  $\mathbf{R}(\phi, x) \in \mathbb{R}^{3 \times 3}$  representa una rotación alrededor del eje  $x$ , donde  $c_k = \cos(k)$ ,  $s_k = \sin(k)$  y  $t_k = \tan(k)$ .

El vector de las velocidades lineales está dado por

$$\mathbf{V}^B = [u \ v \ w]^T \in \mathbb{R}^3 \quad (2.4)$$

y el vector de las velocidades angulares como

$$\boldsymbol{\omega}^B = [p \ q \ r]^T \in \mathbb{R}^3, \quad (2.5)$$

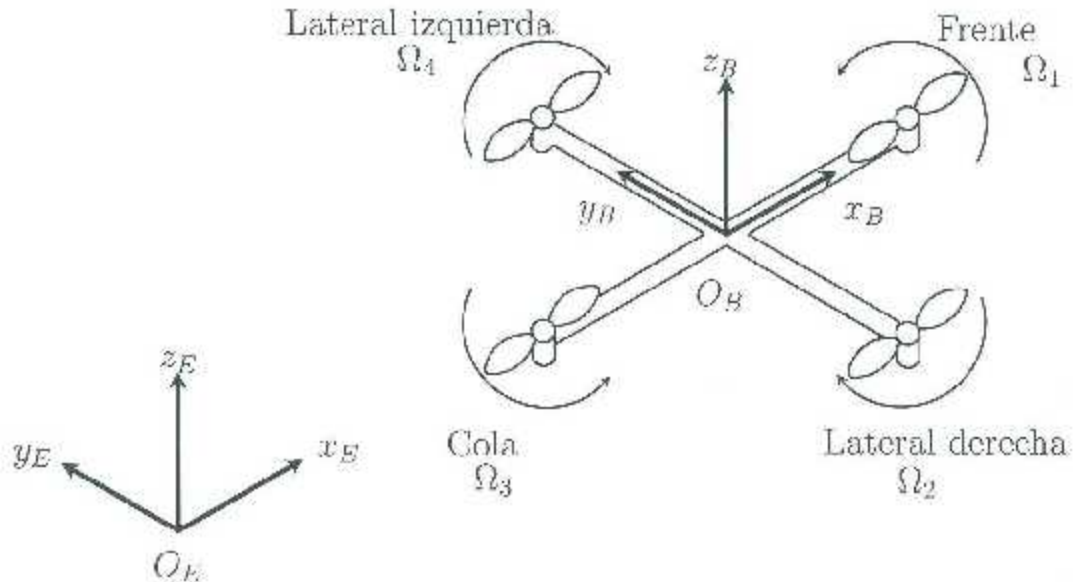


Figura 2.1: Marcos de referencia.

Para una representación que incluya las componentes de posición y orientación se combinan las cantidades lineales con las angulares, lo que da lugar al vector de posiciones generalizadas

$$\xi = [\Gamma^E \ \Theta^E]^T = [\dot{x} \ \dot{y} \ \dot{z} \ \dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T \in \mathbb{R}^6 \quad (2.6)$$

así como al vector de velocidades generalizadas

$$\nu = [V^B \ \omega^B]^T = [u \ v \ w \ p \ q \ r]^T \in \mathbb{R}^6 \quad (2.7)$$

La relación entre la velocidad lineal  $V^B$  respecto al marco B y la velocidad lineal  $V^E$  respecto al marco E involucra la matriz de rotación  $R_\Theta$  de la siguiente manera:

$$V^E = \dot{\Gamma}^E - R_\Theta V^B. \quad (2.8)$$

También es posible relacionar la velocidad angular  $\dot{\Theta}^E$  respecto al marco E con la velocidad angular  $\omega^B$  respecto al marco B mediante la matriz de transformación  $T_\Theta$ , i.e.,

$$\omega^B = T_\Theta^{-1} \dot{\Theta}^E \quad (2.9)$$

$$\dot{\Theta}^E = T_\Theta \omega^B \quad (2.10)$$

La matriz de transformación  $T_\Theta$  se obtiene resolviendo las velocidades angulares  $\dot{\Theta}^E$  en el marco B, i.e.,

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R(\dot{\phi}, x)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R(\dot{\phi}, x)^{-1} R(\theta, y)^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = T_\Theta^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (2.11)$$

$$T_\Theta^{-1} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix} \quad (2.12)$$

$$T_\Theta = \begin{bmatrix} 1 & s_\theta t_\theta & c_\theta t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\theta / c_\theta & c_\theta / c_\theta \end{bmatrix} \quad (2.13)$$

Luego, es posible expresar (2.8) y (2.10) en una sola ecuación como

$$J_\Theta = \begin{bmatrix} R_\Theta & 0_{3 \times 3} \\ 0_{3 \times 3} & T_\Theta \end{bmatrix} \in \mathbb{R}^{6 \times 6}, \quad (2.14)$$

la cual relaciona la derivada  $\dot{\xi}$  de las posiciones generalizadas respecto al marco E con las velocidades generalizadas  $\nu$  respecto al marco B como

$$\dot{\xi} = J_\Theta \nu, \quad (2.15)$$

donde la notación  $0_{3 \times 3}$  representa una submatriz de ceros de dimensión tres por tres.

### 2.2.2. Modelo dinámico

La dinámica es una rama de la mecánica que se encarga del estudio de los efectos de las fuerzas y pares en el movimiento de un cuerpo o sistemas de cuerpos. Existen diversos métodos para la obtención de las ecuaciones que describen las dinámicas de un cuerpo rígido de 6-GDL, en el presente trabajo se utiliza la formulación de Newton-Euler.

Las ventajas de formular las ecuaciones de movimiento con respecto al marco B son [Perez, 2009]:

- La matriz de inercias es invariante en el tiempo.
- La simetría del cuerpo es aprovechada para simplificar las ecuaciones.
- Las mediciones tomadas a bordo se pueden convertir fácilmente al marco B.
- Las entradas de control frecuentemente están dadas con respecto al marco B.

Lo anterior bajo los siguientes supuestos:

- El origen del marco B coincide con el centro de masa del cuerpo. De otra forma las ecuaciones que describen las dinámicas del cuerpo rígido se complican considerablemente.
- Los ejes del marco B coinciden con los ejes principales de inercia del cuerpo. En este caso la matriz  $J$  de inercias es diagonal.

Es conveniente definir el operador  $S(\cdot)$ , el cual crea una matriz antisimétrica. Para un vector

$$\mathbf{k} = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} \in \mathbb{R}^3, \quad (2.16)$$

la matriz antisimétrica de  $\mathbf{k}$  ( $S(\mathbf{k})$ ) se define como:

$$S(\mathbf{k}) = -S'(\mathbf{k}) = \begin{bmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3}. \quad (2.17)$$

La ecuación (2.19) muestra la derivada con respecto al tiempo de las componentes locales del movimiento del cuerpo de acuerdo al primer axioma de Euler de la segunda ley de Newton.

Se tiene que

$$m \ddot{\Gamma}^E = \mathbf{F}^E,$$

usando (2.3) y (2.8) resulta

$$m \frac{d}{dt} (R_{\Theta} V^B) = R_{\Theta} F^B,$$

derivando  $R_{\Theta} V^B$  respecto al tiempo se tiene

$$m \left( R_{\Theta} \dot{V}^B + \dot{R}_{\Theta} V^B \right) = R_{\Theta} F^B,$$

luego como la derivada de la matriz de rotación [Oliati-Saber, 2001]

$$\dot{R}_{\Theta} = R_{\Theta} S(\omega) \quad (2.18)$$

y usando la propiedad de una matriz antisimétrica (i.e.,  $S(\omega) \cdot = \omega \times \cdot$ ), se llega a

$$m R_{\Theta} \left( \dot{V}^B + \omega^B \times V^B \right) = R_{\Theta} F^B,$$

como  $R_{\Theta}$  se encuentra de ambos lados de la igualdad resulta

$$m \dot{V}^B + \omega^B \times (m V^B) = F^B, \quad (2.19)$$

donde  $m \in \mathbb{R}$  [Kg] es la masa del *quadrotor*,  $\ddot{V}^B \in \mathbb{R}^3$  [m/s<sup>2</sup>] representa la aceleración lineal con respecto al marco E,  $F^B \in \mathbb{R}^3$  [N] es el vector de fuerzas del *quadrotor* con respecto al marco E,  $\dot{V}^B \in \mathbb{R}^3$  [m/s<sup>2</sup>] es el vector de aceleraciones lineales con respecto al marco B y  $\dot{R}_{\Theta} \in \mathbb{R}^{3 \times 3}$  es la derivada de la matriz de rotación.

La ecuación (2.21) muestra la derivada con respecto al tiempo de las componentes angulares del movimiento del cuerpo.

Se tiene que

$$J \ddot{\Theta}^B = \tau^B$$

usando (2.13) y (2.10) resulta

$$J \frac{d}{dt} (T_{\Theta} \omega^B) = T_{\Theta} \tau^B$$

derivando  $T_{\Theta} \omega^B$  respecto al tiempo se tiene

$$J \left( T_{\Theta} \dot{\omega}^B - \dot{T}_{\Theta} \omega^B \right) = T_{\Theta} \tau^B$$

luego como la derivada de la matriz de transformación [Oliati-Saber, 2001]

$$\dot{T}_{\Theta} = T_{\Theta} S(\omega) \quad (2.20)$$

y usando la propiedad de una matriz antisimétrica (i.e.,  $S(\omega) \cdot = \omega \times \cdot$ ), se llega a

$$J T_{\Theta} \left( \dot{\omega}^B + \omega^B \times \omega^B \right) = T_{\Theta} \tau^B$$

como  $T_{\Theta}$  se encuentra de ambos lados de la igualdad resulta

$$J \dot{\omega}^B + \omega^B \times (J \omega^B) = \tau^B, \quad (2.21)$$

donde  $J \in \mathbb{R}^{3 \times 3}$  [N m s<sup>2</sup>] es la matriz de inercia del cuerpo con respecto al marco B,  $\dot{\Theta}^B \in \mathbb{R}^3$  [rad/s<sup>2</sup>] representa la aceleración angular del *quadrotor* con respecto al marco E,  $\omega^B \in \mathbb{R}^3$  es la aceleración angular del *quadrotor* con respecto al marco B y  $\tau^B \in \mathbb{R}^3$  [N m] son los pares ejercidos por el *quadrotor* con respecto al marco E.

Para describir el movimiento del cuerpo rígido de 6-DOF, se unen (2.19) y (2.21) para producir:

$$\begin{bmatrix} m I_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & J \end{bmatrix} \begin{bmatrix} \dot{V}^B \\ \dot{\omega}^B \end{bmatrix} + \begin{bmatrix} \omega^B \times m V^B \\ \omega^B \times J \omega^B \end{bmatrix} = \begin{bmatrix} F^B \\ \tau^B \end{bmatrix}, \quad (2.22)$$

en donde la notación  $I_{3 \times 3} \in \mathbb{R}^{3 \times 3}$  es la matriz identidad.

Luego, la dinámica se expresa con respecto a un marco híbrido

$$\zeta = [\dot{\Gamma}^E \ \omega^B]^T = [\dot{x} \ \dot{y} \ \dot{z} \ \dot{\varphi} \ q \ r]^T \in \mathbb{R}^6, \quad (2.23)$$

compuesto por las ecuaciones lineales con respecto al marco E y las ecuaciones angulares con respecto al marco B. Esta nueva referencia se adopta porque simplifica las ecuaciones de la dinámica combinadas con las entradas de control. Por lo tanto, las ecuaciones siguientes se expresarán con respecto al nuevo marco híbrido H.

Ya con el nuevo marco de referencia H se define un vector de fuerzas generalizadas

$$\Lambda = [F^B \ \tau^B]^T = [F_x \ F_y \ F_z \ \tau_x \ \tau_y \ \tau_z]^T \in \mathbb{R}^6, \quad (2.24)$$

el cual se puede dividir en tres componentes:

- El vector producido por la aceleración debido a la gravedad ( $g$  [m/s<sup>2</sup>])

$$G_H = \begin{bmatrix} F_G^E \\ \mathbf{0}_{3 \times 1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -mg \\ 0 \\ 0 \\ 0 \end{bmatrix} \in \mathbb{R}^6, \quad (2.25)$$

donde  $F_G^E$  [N] es el vector de fuerza gravitacional con respecto al marco E y  $\mathbf{0}_{3 \times 1}$  es un vector columna de ceros. El vector de gravedad (2.25) solo afecta a la parte lineal y no a la angular de las ecuaciones de movimiento, ya que se trata de una fuerza y no de un par.

- Los efectos giroscópicos debido al giro de las hélices. Ya que dos de las hélices giran en sentido horario y las otras dos restantes en sentido antihorario, existe un desequilibrio cuando la suma algebraica de las velocidades de los rotores es diferente de cero. Si, además, las velocidades de banqueo o cabeceo son también diferentes de cero, el *quadrotor* experimenta un par giroscópico de acuerdo con:

$$\begin{aligned}
 \mathbf{O}_H(\zeta) \boldsymbol{\Omega} &= \mathbf{O}_H(\boldsymbol{\nu}) \boldsymbol{\Omega} = \left[ - \sum_{k=1}^4 J_{RP} \left( \boldsymbol{\omega}^k \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) (-1)^k \Omega_k \right] \\
 &= \begin{bmatrix} \mathbf{0}_{3 \times 1} \\ J_{RP} \begin{bmatrix} -q \\ p \\ 0 \end{bmatrix} \end{bmatrix} \boldsymbol{\Omega} = J_{RP} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ q & -q & q & -q \\ -p & p & -p & p \\ 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{\Omega}, \quad (2.26)
 \end{aligned}$$

donde  $\mathbf{O}_H(\boldsymbol{\nu}) \in \mathbb{R}^{6 \times 4}$  es la matriz giroscópica de las hélices y  $J_{RP} \in \mathbb{R} [\text{K m s}^2]$  es el momento de inercia rotacional total alrededor de la hélice. Los efectos giroscópicos producidos por el giro de las hélices están relacionados con la parte angular de las ecuaciones.

La velocidad total  $\boldsymbol{\Omega}$  [rad/s] de las hélices y el vector  $\boldsymbol{\Omega}$  de las velocidades de las hélices utilizados en (2.26) están definidos como:

$$\boldsymbol{\Omega} = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \in \mathbb{R} \quad \boldsymbol{\Omega} = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \\ \Omega_4 \end{bmatrix} \in \mathbb{R}^4, \quad (2.27)$$

donde  $\Omega_1 \in \mathbb{R}$  [rad/s] es la velocidad angular de la hélice frontal,  $\Omega_2 \in \mathbb{R}$  [rad/s] es la velocidad angular de la hélice lateral derecha,  $\Omega_3 \in \mathbb{R}$  [rad/s] es la velocidad angular de la hélice trasera y  $\Omega_4 \in \mathbb{R}$  [rad/s] es la velocidad angular de la hélice lateral izquierda.

- Las fuerzas y pares producidos por las entradas principales del sistema son proporcionales al cuadrado de las velocidades de las hélices. Por lo tanto, la matriz de movimiento  $\mathbf{E}_H$  se multiplica por  $\boldsymbol{\Omega}^2$  para obtener el vector de movimiento



$$E_H(\xi) \Omega^2 = \begin{bmatrix} R_\Theta & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & I_{3 \times 3} \end{bmatrix} E_B \Omega^2 = \begin{bmatrix} (s_\psi s_\theta + c_\psi s_\theta c_\phi) u_1 \\ (-c_\psi s_\theta + s_\psi s_\theta c_\phi) u_1 \\ (c_\psi c_\theta) u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \quad (2.28)$$

donde  $u_1 \in \mathbb{R}$ ,  $u_2 \in \mathbb{R}$ ,  $u_3 \in \mathbb{R}$  y  $u_4 \in \mathbb{R}$  son el empuje vertical, momento en alabeo, momento en cabeceo y momento en guiñada respectivamente.

Por lo tanto, es posible reescribir (2.22) en forma matricial [Perez, 2009] como:

$$M_H \dot{\zeta} + C_H(\zeta) = G_H + O_H(\zeta) \Omega + E_H(\xi) \Omega^2 \quad (2.29)$$

donde  $\zeta \in \mathbb{R}^6$  es el vector de velocidades generalizadas con respecto al marco H.

$$M_H = M_B = \begin{bmatrix} m I_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{J} \end{bmatrix} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & J_x & 0 & 0 \\ 0 & 0 & 0 & 0 & J_y & 0 \\ 0 & 0 & 0 & 0 & 0 & J_z \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad (2.30)$$

es la matriz diagonal constante de inercias del sistema y

$$C_H(\zeta) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -S(\mathbf{J} \omega^B) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & J_x r & -J_y q \\ 0 & 0 & 0 & -J_x r & 0 & J_x p \\ 0 & 0 & 0 & J_y q & -J_x p & 0 \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad (2.31)$$

es la matriz de fuerzas centrífugas y de coriolis, ambas con respecto al marco H.

De (2.29) es posible despejar la derivada con respecto al tiempo del vector de velocidades generalizadas con respecto al marco H

$$\dot{\zeta} = M_H^{-1} (-C_H(\zeta) + G_H + O_H(\zeta) \Omega + E_H(\xi) \Omega^2), \quad (2.32)$$

para luego expresarlo como un sistema de ecuaciones de la siguiente forma:

$$\begin{aligned}
 \ddot{x} &= (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{u_1}{m} \\
 \ddot{y} &= (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{u_1}{m} \\
 \ddot{z} &= -g + (\cos \theta \cos \phi) \frac{u_1}{m} \\
 \dot{p} &= \frac{J_y - J_z}{J_x} q r - \frac{J_{TP}}{J_x} q \Omega + \frac{u_2}{J_x} \\
 \dot{q} &= \frac{J_z - J_x}{J_y} p r + \frac{J_{TP}}{J_y} p \Omega + \frac{u_3}{J_y} \\
 \dot{r} &= \frac{J_x - J_y}{J_z} p q + \frac{u_4}{J_z}
 \end{aligned} \tag{2.33}$$

Las entradas del sistema son proporcionales al cuadrado de las velocidades de las hélices, tal que:

$$\mathbf{U}_B(\Omega) = \mathbf{E}_B \Omega^2 = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ b l (\Omega_4^2 - \Omega_2^2) \\ b l (\Omega_3^2 - \Omega_1^2) \\ d(\Omega_1^2 + \Omega_4^2 - \Omega_2^2 - \Omega_3^2) \end{bmatrix}, \tag{2.34}$$

donde  $l \in \mathbb{R}$  [m] es la distancia del centro de la hélice al centro del *quadrotor*,  $b \in \mathbb{R}$  [N s<sup>2</sup>] es el coeficiente de empuje y  $d \in \mathbb{R}$  [N m s<sup>2</sup>] es el coeficiente de arrastre. De (2.34) es posible identificar una matriz constante  $\mathbf{E}_B \in \mathbb{R}^{4 \times 4}$ , la cual, multiplicada por  $\Omega^2$  genera el vector  $\mathbf{U}_B(\Omega) \in \mathbb{R}^4$  de entradas del sistema la cual resulta de la forma:

$$\mathbf{U}_B(\Omega) = \mathbf{E}_B \Omega^2 = \begin{bmatrix} b & b & b & b \\ 0 & -bl & 0 & bl \\ -bl & 0 & bl & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix}. \tag{2.35}$$

## Capítulo 3

# Controlador adaptable por modelo de referencia

En este capítulo se propone un esquema de control adaptable con el objetivo de controlar la orientación de un *quadrotor* UAV. Para lograr esto, el modelo no lineal del *quadrotor* se descompone en subsistemas. Así, se usa la técnica de MRAC directo para cada uno de los ángulos de orientación (*pitch*, *roll* y *yaw*) y un PID clásico se aplica a cada una de las coordenadas traslacionales ( $z$ ,  $x$  y  $y$ ).

### 3.1. Introducción

El control adaptable por modelo de referencia (*model reference adaptive control*, MRAC por sus siglas en inglés) es una de las principales técnicas para control adaptable. El modelo de referencia se elige para generar la trayectoria deseada  $y_m$ , que la salida de la planta  $y_p$  debe seguir. El error de seguimiento  $e_1 \triangleq y_p - y_m$  representa la desviación de la salida de la planta con respecto a la trayectoria deseada. El sistema en lazo cerrado que incluye una ley de control por realimentación ordinaria, consiste en la planta, un controlador  $C(\theta)$  y un mecanismo de ajuste que genera la estimación de parámetros de control  $\theta(t)$  en línea [Ioannou and Sun, 2013].

EL esquema MRAC puede ser catalogado como directo o indirecto y con leyes adaptables normalizadas y no normalizadas. En el MRAC directo, el vector  $\theta$  del controlador  $C(\theta)$  se actualiza directamente por una ley adaptable, mientras en el MRAC indirecto  $\theta$  se calcula en cada tiempo  $t$  resolviendo una cierta ecuación algebraica que relaciona  $\theta$  con los estimados en línea de los parámetros de la planta. Tanto en el MRAC directo como en el indirecto con leyes adaptables normalizadas, la forma de  $C(\theta)$ , motivado por el caso de parámetros conocidos, se mantiene sin cambios. El controlador  $C(\theta)$  se combina con una ley adaptable (o una ley adaptable y un ecuación algebraica en el caso indirecto) que se desarrolla independientemente siguiendo las técnicas de estimación paramétrica en línea. Este procedimiento de diseño permite a el usuario una clase amplia de leyes adaptables como son el gradiente, mínimos cuadrados y aquellos basados en en la técnica de diseño de Lyapunov SPR. Por otra parte, en el caso de esquemas MRAC con leyes adaptables no

normalizadas,  $C(\theta)$  se modifica para conducir a una ecuación de error cuya forma permite el uso de la técnica de diseño de Lyapunov SPR para generar la ley adaptable. En este caso el diseño de  $C(\theta)$  y la ley adaptable es más complicado tanto en el caso directo como en el indirecto, pero el análisis es mucho más simple y sigue una consideración de una sola función candidata de Lyapunov [Ioannou and Sun, 2013].

Este capítulo tiene como finalidad entender y evaluar la manera de trabajar las posibles parametrizaciones del modelo dinámico de un *quadrotor* UAV de manera descentralizada. Esto sirve de base para entender como se trabajan los modelos paramétricos en un *quadrotor* UAV y a su vez facilitara el desarrollo de los esquemas que posteriormente se abordarán.

### 3.2. Diseño de un MRAC para un sistema MIMO

Considerando un sistema no lineal de múltiples-entradas múltiples-salidas (MIMO, por sus siglas en inglés) de la forma [Lavretsky and Wise, 2012]

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{\Lambda}(\mathbf{u} + \mathbf{f}(\mathbf{x})), \quad (3.1)$$

donde  $\mathbf{x} \in \mathbb{R}^n$  es el estado del sistema,  $\mathbf{u} \in \mathbb{R}^m$  es la entrada de control, y  $\mathbf{B} \in \mathbb{R}^{n \times m}$  es la matriz de control, mientras que  $\mathbf{A} \in \mathbb{R}^{n \times n}$  y  $\mathbf{\Lambda} \in \mathbb{R}^{m \times m}$  son matrices constantes desconocidas. Luego, se establece que  $\mathbf{\Lambda}$  es diagonal, con sus elementos  $\lambda_i$  estrictamente positivos, y el par  $(\mathbf{A}, \mathbf{B}\mathbf{\Lambda})$  es controlable. La incertidumbre en  $\mathbf{\Lambda}$  es debido a fallas de control del modelo o errores de modelado, en ese sentido, pueden existir ganancias de control desconocidas o el diseñador pudo haber estimado incorrectamente la efectividad del control del sistema. La función vector posiblemente no lineal con parámetros desconocidos  $\mathbf{f}(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$  representa las incertidumbres emparejadas del sistema. Se hace la suposición de que cada componente individual de  $f_i(\mathbf{x})$  de  $\mathbf{f}(\mathbf{x})$  puede ser escrito como una combinación lineal de  $N$  funciones localmente Lipschitz-de base continua conocidas  $\varphi_i(\mathbf{x})$ , con coeficientes constantes desconocidos así,

$$\mathbf{f}(\mathbf{x}) = \mathbf{\Theta}^T \mathbf{\Phi}(\mathbf{x}) \quad (3.2)$$

donde  $\mathbf{\Theta} \in \mathbb{R}^{N \times m}$  es una matriz de coeficientes constantes desconocidos y  $\mathbf{\Phi}(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \dots, \varphi_N(\mathbf{x}))^T \in \mathbb{R}^N$  es el vector regresor [Ioannou and Fidan, 2006].

Es de interés diseñar una ley de control adaptable por realimentación de estados MIMO tal que el estado del sistema  $\mathbf{x}$  siga asintótica uniforme globalmente al estado  $\mathbf{x}_{ref} \in \mathbb{R}^n$  de el modelo de referencia.

$$\dot{\mathbf{x}}_{ref} = \mathbf{A}_{ref}\mathbf{x}_{ref} + \mathbf{B}_{ref}\mathbf{r}(t) \quad (3.3)$$

donde  $\mathbf{A}_{ref} \in \mathbb{R}^{n \times n}$  es Hurwitz,  $\mathbf{B}_{ref} \in \mathbb{R}^{n \times m}$ , y  $\mathbf{r}(t) \in \mathbb{R}^m$  es el vector de comando externo acotado. También es necesario que durante el seguimiento, todas las señales del sistema en lazo cerrado permanezcan uniformemente acotadas. Luego, dado cualquier comando acotado  $\mathbf{r}(t)$ , la entrada de control  $\mathbf{u}$  debe ser elegida tal que el error de seguimiento del estado

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_{ref} \quad (3.4)$$

tienda a cero asintótica uniforme globalmente, i.e.,

$$\lim_{t \rightarrow \infty} \|\mathbf{x} - \mathbf{x}_{ref}\| = 0. \quad (3.5)$$

si las matrices  $\mathbf{A}$  y  $\mathbf{\Lambda}$  fueran conocidas, se podría haber calculado y aplicado la ley de control ideal de ganancias fijas

$$\mathbf{u} = \mathbf{K}_x^T \mathbf{x} + \mathbf{K}_r^T \mathbf{r}(t) - \mathbf{\Theta}^T \Phi(\mathbf{x}) \quad (3.6)$$

y obtener el sistema en lazo cerrado

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\mathbf{\Lambda}\mathbf{K}_x^T)\mathbf{x} + \mathbf{B}\mathbf{\Lambda}\mathbf{K}_r^T \mathbf{r}(t). \quad (3.7)$$

Comparando (3.7) con la dinámica de referencia deseada (3.3), resulta que para la existencia de un controlador de la forma de (3.6), las ganancias de control ideales desconocidas,  $\mathbf{K}_x$  y  $\mathbf{K}_r$ , deben satisfacer la condición de emparejamiento

$$\begin{aligned} \mathbf{A} + \mathbf{B}\mathbf{\Lambda}\mathbf{K}_x^T &= \mathbf{A}_{ref}, \\ \mathbf{B}\mathbf{\Lambda}\mathbf{K}_r^T &= \mathbf{B}_{ref}. \end{aligned} \quad (3.8)$$

Suponiendo que se mantenga esta condición de emparejamiento, usando (3.6) produce el sistema en lazo cerrado el cual es exactamente el mismo que el modelo de referencia. Consecuentemente, para cualquier señal de entrada de referencia acotada  $\mathbf{r}(t)$ , el controlador de ganancias fijas (3.6) asegura un desempeño de seguimiento asintótico global y uniforme.

Suponiendo que las ganancias  $\mathbf{K}_x$  y  $\mathbf{K}_r$  existen, se considera la siguiente ley de control

$$\mathbf{u} = \hat{\mathbf{K}}_x^T \mathbf{x} - \hat{\mathbf{K}}_r^T \mathbf{r}(t) - \hat{\mathbf{\Theta}}^T \Phi(\mathbf{x}) \quad (3.9)$$

donde  $\hat{\mathbf{K}}_x \in \mathbb{R}^{n \times m}$ ,  $\hat{\mathbf{K}}_r \in \mathbb{R}^{m \times m}$ , y  $\hat{\mathbf{\Theta}} \in \mathbb{R}^{N \times m}$  son los estimados de las matrices ideales desconocidas  $\mathbf{K}_x$ ,  $\mathbf{K}_r$ , y  $\mathbf{\Theta}$ , respectivamente. Estos parámetros estimados se generarán en línea a través del análisis inverso de Lyapunov.

Sustituyendo (3.9) en (3.1), la dinámica del sistema en lazo cerrado puede ser escrita como

$$\dot{\mathbf{x}} = (\mathbf{A} + \mathbf{B}\mathbf{\Lambda}\hat{\mathbf{K}}_x^T)\mathbf{x} + \mathbf{B}\mathbf{\Lambda}(\hat{\mathbf{K}}_r^T \mathbf{r}(t) - (\hat{\mathbf{\Theta}} - \mathbf{\Theta})^T \Phi(\mathbf{x})). \quad (3.10)$$

Restando (3.3) de (3.10) y considerando la condición de emparejamiento (3.8), la dinámica de el error de seguimiento está dada por

$$\dot{\mathbf{e}} = \mathbf{A}_{ref} \mathbf{e} + \mathbf{B}\mathbf{\Lambda}(\tilde{\mathbf{K}}_x^T \mathbf{x} + \tilde{\mathbf{K}}_r^T \mathbf{r}(t) - \tilde{\mathbf{\Theta}}^T \Phi(\mathbf{x})), \quad (3.11)$$

con  $\tilde{\mathbf{K}}_x = \hat{\mathbf{K}}_x - \mathbf{K}_x$ ,  $\tilde{\mathbf{K}}_r = \hat{\mathbf{K}}_r - \mathbf{K}_r$ , y  $\tilde{\mathbf{\Theta}} = \hat{\mathbf{\Theta}} - \mathbf{\Theta}$  indicando el error de estimación paramétrico [Ioannou and Sun, 2013].

Con el fin de garantizar la estabilidad de la dinámica del error de seguimiento (3.11), se considera la función candidata de Lyapunov

$$V(\mathbf{e}, \tilde{\mathbf{K}}_x, \tilde{\mathbf{K}}_r, \tilde{\mathbf{\Theta}}) = \mathbf{e}^T \mathbf{P} \mathbf{e} + \text{tr}((\tilde{\mathbf{K}}_x^T \Gamma_x^{-1} \tilde{\mathbf{K}}_x + \tilde{\mathbf{K}}_r^T \Gamma_r^{-1} \tilde{\mathbf{K}}_r + \tilde{\mathbf{\Theta}}^T \Gamma_{\Theta}^{-1} \tilde{\mathbf{\Theta}}) \mathbf{\Lambda}) \quad (3.12)$$

donde  $\mathbf{P} = \mathbf{P}^T > 0$  es la solución de la ecuación de Lyapunov

$$\mathbf{P}\mathbf{A}_{ref} + \mathbf{A}_{ref}^T \mathbf{P} = -\mathbf{Q}$$

para cualquier  $Q = Q^T > 0$  y  $\text{tr}(\cdot)$  es el operador de la traza [Ioannou and Sun, 2013].

La derivada con respecto al tiempo  $\dot{V}$  de  $V$  está dada por

$$\dot{V} = \dot{e}^T P e + e^T P \dot{e} + 2 \text{tr}((\tilde{K}_x^T \Gamma_x^{-1} \dot{\tilde{K}}_x + \tilde{K}_r^T \Gamma_r^{-1} \dot{\tilde{K}}_r - \tilde{\Theta}^T \Gamma_\Theta^{-1} \dot{\tilde{\Theta}}) \Lambda),$$

el cual evaluado a lo largo de la trayectoria de (3.11) se convierte

$$\begin{aligned} \dot{V} = & (\mathbf{A}_{ref} e - B \Lambda (\tilde{K}_x^T x + \tilde{K}_r^T r(t) - \tilde{\Theta}^T \Phi(x)))^T P e \\ & + e^T P (\mathbf{A}_{ref} e + B \Lambda (\tilde{K}_x^T x + \tilde{K}_r^T r(t) - \tilde{\Theta}^T \Phi(x))) \\ & + 2 \text{tr}((\tilde{K}_x^T \Gamma_x^{-1} \dot{\tilde{K}}_x + \tilde{K}_r^T \Gamma_r^{-1} \dot{\tilde{K}}_r + \tilde{\Theta}^T \Gamma_\Theta^{-1} \dot{\tilde{\Theta}}) \Lambda). \end{aligned}$$

Considerando la ecuación de Lyapunov, expandiendo y reordenando términos produce expanding

$$\begin{aligned} \dot{V} = & -e^T Q e \\ & + (2e^T P B \Lambda \tilde{K}_x^T x + 2 \text{tr}(\tilde{K}_x^T \Gamma_x^{-1} \dot{\tilde{K}}_x \Lambda)) \\ & + (2e^T P B \Lambda \tilde{K}_r^T r(t) + 2 \text{tr}(\tilde{K}_r^T \Gamma_r^{-1} \dot{\tilde{K}}_r \Lambda)) \\ & - (2e^T P B \Lambda \tilde{\Theta}^T \Phi(x) - 2 \text{tr}(\tilde{\Theta}^T \Gamma_\Theta^{-1} \dot{\tilde{\Theta}} \Lambda)). \end{aligned}$$

Usando las propiedades para el operador de la traza [Ioannou and Sun, 2013], la derivada con respecto al tiempo  $\dot{V}$  de  $V$  se puede reescribir como

$$\begin{aligned} \dot{V} = & -e^T Q e \\ & + 2 \text{tr}(\tilde{K}_x^T (\Gamma_x^{-1} \dot{\tilde{K}}_x + x e^T P B) \Lambda) \\ & + 2 \text{tr}(\tilde{K}_r^T (\Gamma_r^{-1} \dot{\tilde{K}}_r + r(t) e^T P B) \Lambda) \\ & + 2 \text{tr}(\tilde{\Theta}^T (\Gamma_\Theta^{-1} \dot{\tilde{\Theta}} - \Phi(x) e^T P B) \Lambda). \end{aligned}$$

Así, seleccionando las leyes adaptables

$$\begin{aligned} \dot{\tilde{K}}_x &= -\Gamma_x x e^T P B, \\ \dot{\tilde{K}}_r &= -\Gamma_r r(t) e^T P B, \\ \dot{\tilde{\Theta}}_x &= \Gamma_\Theta \Phi(x) e^T P B, \end{aligned} \tag{3.13}$$

la derivada con respecto al tiempo  $\dot{V}$  de  $V$  luego resulta  $\dot{V} = -e^T Q e \leq 0$ . Por lo tanto, la dinámica del error en lazo cerrado es uniformemente estable. Así, el error de seguimiento  $e(t)$  y los errores de estimación paramétrica  $\tilde{K}_x(t)$ ,  $\tilde{K}_r(t)$ , y  $\tilde{\Theta}(t)$  están acotados uniformemente. Luego, las estimaciones paramétricas  $\hat{K}_x(t)$ ,  $\hat{K}_r(t)$ , y  $\hat{\Theta}(t)$  también están acotadas uniformemente. Ya que  $r(t)$  está acotado y  $\mathbf{A}_{ref}$  es Hurwitz, luego  $x_{ref}$  y  $\dot{x}_{ref}$  están acotados. Por lo tanto, el estado del sistema  $x$  está acotado uniformemente y la entrada

de control  $\mathbf{u}$  en (3.9) también está acotada. Esto último implica que  $\hat{\mathbf{x}}$  está acotado y así  $\hat{\mathbf{e}}$  está acotado.

Además, la segunda derivada con respecto al tiempo  $\ddot{V}$  de  $V$ , específicamente,

$$\ddot{V} = -2\mathbf{e}^T \mathbf{Q} \dot{\mathbf{e}} \quad (3.14)$$

está también acotada, y luego  $\dot{V}$  es uniformemente continua. Ya que  $V$  está acotada por abajo y  $\dot{V} \leq 0$ , usando el lema de Barbalat se obtiene  $\lim_{t \rightarrow \infty} \dot{V} = 0$ . De este modo se prueba que el error de seguimiento del estado  $\mathbf{e}$  tiende al origen global, uniforme, y asintóticamente, i.e.,  $\lim_{t \rightarrow \infty} \|\mathbf{x} - \mathbf{x}_{ref}\| = 0$ . Luego, el problema de seguimiento del comando MIMO para la dinámica del sistema (3.1) está resuelta y se ha formulado como sigue.

**Teorema 1** ([Lavretsky and Wise, 2012]). Dada la dinámica MIMO (3.1) con una incertidumbre de control  $\mathbf{A}$  y una función emparejada con parámetros desconocidos (3.2), el sistema MRAC (3.9)-(3.13) garantiza el desempeño de seguimiento asintótico uniforme y globalmente de la dinámica del modelo de referencia (3.3), llevado por cualquier comando acotado variante en el tiempo  $\mathbf{r}(t)$ . Además, todas las señales en el correspondiente sistema en lazo cerrado permanecen uniformemente acotadas en el tiempo.

### 3.3. Diseño descentralizado MRAC directo aplicado a un *quadrotor*

Una vez que se ha obtenido el modelo matemático del *quadrotor*, puede ser expresado en la forma que más convenga para su análisis. El primer paso hacia ese objetivo es la selección de un conjunto de variables de estado de la siguiente manera:

$$\begin{aligned} x_1 &= \phi, & x_2 &= \theta, & x_3 &= \psi, & x_4 &= x, & x_5 &= y, & x_6 &= z, \\ x_7 &= \dot{\phi}, & x_8 &= \dot{\theta}, & x_9 &= \dot{\psi}, & x_{10} &= \dot{x}, & x_{11} &= \dot{y}, & x_{12} &= \dot{z}. \end{aligned} \quad (3.15)$$

A partir del modelo matemático para el *quadrotor*, las ecuaciones en variables de estado

toman la forma general

$$\dot{x} = \begin{pmatrix} x_2 \\ \left(\frac{J_y - J_z}{J_x}\right) x_2 x_3 - \frac{J_{yz}}{J_x} \Omega x_4 + \frac{1}{J_x} u_1 \\ x_4 \\ \left(\frac{J_x - J_z}{J_y}\right) x_2 x_3 + \frac{J_{xz}}{J_y} \Omega x_2 + \frac{1}{J_y} u_2 \\ x_6 \\ \left(\frac{J_x - J_y}{J_z}\right) x_2 x_4 + \frac{1}{J_z} u_3 \\ x_8 \\ -g + (\cos x_1 \cos x_3) \frac{u_1}{m} \\ x_{10} \\ (\cos x_1 \sin x_3 \cos x_5 + \sin x_1 \sin x_5) \frac{u_1}{m} \\ x_{12} \\ (\cos x_1 \sin x_3 \sin x_5 - \sin x_1 \cos x_5) \frac{u_1}{m} \end{pmatrix} \quad (3.16)$$

El sistema (3.16) puede ser considerado como un conjunto de dos subsistemas, el sistema rotacional y el sistema translacional [Bouabdallah et al., 2004]. Así, se puede dividir como

$$\dot{x} = \underbrace{\begin{pmatrix} x_2 \\ \left(\frac{J_y - J_z}{J_x}\right) x_2 x_3 + \frac{1}{J_x} u_1 \\ x_4 \\ \left(\frac{J_x - J_z}{J_y}\right) x_2 x_3 + \frac{1}{J_y} u_2 \\ x_6 \\ \left(\frac{J_x - J_y}{J_z}\right) x_2 x_4 + \frac{1}{J_z} u_3 \\ \mathbf{0}_{6 \times 1} \end{pmatrix}}_{\text{Rotational Subsystem}} + \underbrace{\begin{pmatrix} \mathbf{0}_{6 \times 1} \\ x_8 \\ -g + (\cos x_1 \cos x_3) \frac{u_1}{m} \\ x_{10} \\ (\cos x_1 \sin x_3 \cos x_5 - \sin x_1 \sin x_5) \frac{u_1}{m} \\ x_{12} \\ (\cos x_1 \sin x_3 \sin x_5 - \sin x_1 \cos x_5) \frac{u_1}{m} \end{pmatrix}}_{\text{Translational Subsystem}} \quad (3.17)$$

donde  $\mathbf{0}_{6 \times 1} \in \mathbb{R}^{6 \times 1}$  es un vector de ceros. Los efectos giroscópicos son despreciados debido que los ángulos de Euler cambian ligeramente durante el vuelo.

Además, el subsistema de rotación se puede subdividir de la siguiente manera:

$$\dot{x}^1 = \begin{pmatrix} x_2 \\ \left(\frac{J_y - J_z}{J_x}\right) x_2 x_3 + \frac{1}{J_x} u_1 \end{pmatrix}, \quad (3.18)$$

$$\dot{x}^2 = \begin{pmatrix} x_4 \\ \left(\frac{J_x - J_z}{J_y}\right) x_2 x_3 + \frac{1}{J_y} u_2 \end{pmatrix}, \quad (3.19)$$

$$\dot{x}^3 = \begin{pmatrix} x_6 \\ \left(\frac{J_x - J_y}{J_z}\right) x_2 x_4 + \frac{1}{J_z} u_3 \end{pmatrix}, \quad (3.20)$$

con  $\dot{x}^1 = (\dot{x}_1 \quad \dot{x}_2)^T$ ,  $\dot{x}^2 = (\dot{x}_3 \quad \dot{x}_4)^T$ , y  $\dot{x}^3 = (\dot{x}_5 \quad \dot{x}_6)^T$ .



Con el fin de conseguir el objetivo de control de orientación para el *quadrotor* UAV vía técnica de diseño MRAC de manera descentralizada, (3.18)–(3.20) se parametrizan en la forma dada por (3.1), i.e., (3.18) toma la forma

$$\dot{\mathbf{x}}^1 = \mathbf{A}_1 \mathbf{x}^1 + \mathbf{B}_1 \Lambda_1 (u^1 + \Theta_1 \Phi_1(\mathbf{x})) \quad (3.21)$$

con

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{B}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

$\mathbf{x}^1 = (x_1 \ x_2)^T$ ,  $a_1 = (J_y - J_z)/J_x$ ,  $b_1 = l/J_x$ ,  $\Lambda_1 = b_1$ ,  $\Theta_1 = a_1/b_1$ ,  $\Phi_1(\mathbf{x}) = x_2 x_6$ , y  $u^1 = u_1$ .

Así,

$$\dot{\mathbf{x}}^2 = \mathbf{A}_2 \mathbf{x}^2 + \mathbf{B}_2 \Lambda_2 (u^2 + \Theta_2 \Phi_2(\mathbf{x})) \quad (3.22)$$

es para (3.19) donde  $\mathbf{A}_2 = \mathbf{A}_1$ ,  $\mathbf{B}_2 = \mathbf{B}_1$ ,  $\mathbf{x}^2 = (x_3 \ x_4)^T$ ,  $a_2 = (J_z - J_x)/J_y$ ,  $b_2 = l/J_y$ ,  $\Lambda_2 = b_2$ ,  $\Theta_2 = a_2/b_2$ ,  $\Phi_2(\mathbf{x}) = x_2 x_8$ , y  $u^2 = u_2$ .

Luego, para (3.20),

$$\dot{\mathbf{x}}^3 = \mathbf{A}_3 \mathbf{x}^3 + \mathbf{B}_3 \Lambda_3 (u^3 + \Theta_3 \Phi_3(\mathbf{x})) \quad (3.23)$$

con  $\mathbf{A}_3 = \mathbf{A}_1$ ,  $\mathbf{B}_3 = \mathbf{B}_1$ ,  $\mathbf{x}^3 = (x_5 \ x_6)^T$ ,  $a_3 = (J_x - J_y)/J_z$ ,  $b_3 = l/J_z$ ,  $\Lambda_3 = b_3$ ,  $\Theta_3 = a_3/b_3$ ,  $\Phi_3(\mathbf{x}) = x_2 x_4$ , y  $u^3 = u_3$ .

Se puede notar que, de (3.1), aunque  $\mathbf{f}(\mathbf{x})$  representa un modelo lineal en los parámetros, también llamado modelo paramétrico estático [Ioannou and Sun, 2013], aquí tanto los coeficientes constantes desconocidos como los vectores regresores están relacionados con cantidades escalares. Nuestra principal contribución, en contraste con otros trabajos reportados en la literatura, es el hecho de que los modelo parametrizados (3.21)–(3.23), aquí propuestos, satisfacen la dimensión para la parametrización (3.1); i.e., la aplicación del enfoque de diseño MRAC de [Lavretsky and Wise, 2012] es inmediata. Además, se puede observar que las fallas de control del modelo y los errores de modelado se pueden evaluar para cada subsistema.

A continuación, se presenta el diseño del modelo de referencia. La dinámica de referencia de *roll* está definida por la función de transferencia de segundo orden

$$\frac{x_{1ref}}{x_{1cmd}} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (3.24)$$

la cual representa el comportamiento comando-respuesta deseado del ángulo de *roll*, donde  $x_{1ref}$  es el ángulo de referencia de *roll*,  $x_{1cmd}$  es el ángulo de comando de *roll*,  $\omega_n$  es la frecuencia natural deseada, y  $\xi$  factor de amortiguamiento. en la forma de espacio de estados (3.3), la dinámica de *roll* de referencia está dada por

$$\begin{pmatrix} \dot{x}_{1ref} \\ \dot{x}_{2ref} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{pmatrix} \begin{pmatrix} x_{1ref} \\ x_{2ref} \end{pmatrix} + \begin{pmatrix} 0 \\ \omega_n^2 \end{pmatrix} x_{1cmd} \quad (3.25)$$

Debe notarse que, en general, dado  $(\mathbf{A}, \mathbf{B}, \Lambda, \mathbf{A}_{ref}, \mathbf{B}_{ref})$ , no hay garantía de que las ganancias ideales  $\mathbf{K}_x$  y  $\mathbf{K}_r$  existan de tal manera que las condiciones de emparejamiento (3.8) sean satisfechas. Sin embargo, a menudo en la práctica, la estructura de  $\mathbf{A}$  es conocida

y las matrices del modelo de referencia  $A_{ref}$  y  $B_{ref}$  se eligen de modo que (3.8) tenga al menos un par de soluciones ideales  $(K_x, K_r)$ .

De [User Manual QBall 2, 2014], con  $J_x = J_y = 0.03 \text{ kg m}^2$ ,  $J_z = 0.04 \text{ kg m}^2$ ,  $l = 0.2 \text{ m}$ ,  $m = 1.79 \text{ kg}$ , y seleccionando  $\omega_n^2 = 32\,653 \text{ rad/s}^2$  y  $\xi = 0.7$ , la condición de emparejamiento (3.8) se satisface con las ganancias ideales  $K_{x^1} = (-4.89795 \quad -1.2)^T$  y  $K_{r^1} = 4.89795$ .

Luego, resolviendo la ecuación de Lyapunov para

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 10 \end{pmatrix}, \quad (3.26)$$

así

$$P = \begin{pmatrix} 20.5931 & 0.0153 \\ 0.0153 & 0.6269 \end{pmatrix}. \quad (3.27)$$

Finalmente, después de algunas iteraciones, las ganancias para las leyes adaptables, de (3.13),  $\hat{K}_{x^1}$ ,  $\hat{K}_{r^1}$ , y  $\hat{\Theta}_{x^1}$  están dadas por

$$\Gamma_{x^1} = \begin{pmatrix} 9000 & 0 \\ 0 & 9000 \end{pmatrix} \quad (3.28)$$

y  $\Gamma_{r^1} = \Gamma_{\Theta^1} = 9000$ , respectivamente.

En este trabajo, debido a que se sigue el mismo procedimiento de diseño del MRAC para lograr la estabilización de los ángulos de *pitch* y *yaw*, solo se describe el procedimiento de diseño para la estabilización del ángulo de *roll*. Así, la misma matriz  $P$  se obtiene al considerar los mismos valores  $\omega_n$ ,  $\xi$ , y la matriz  $Q$ . Ahora bien, las ganancias adaptables están dadas por

$$\Gamma_{x^2} = \Gamma_{x^3} = \begin{pmatrix} 3430 & 0 \\ 0 & 3430 \end{pmatrix} \quad (3.29)$$

y  $\Gamma_{r^2} = \Gamma_{r^3} = \Gamma_{\Theta^2} = \Gamma_{\Theta^3} = 3430$ .

Para cada coordenada translacional se implementa un controlador PID [Dorf and Bishop, 2001]. Los valores de las ganancias del PID para el subsistema  $z$  están dadas como  $k_p = 40$  y  $k_i = k_d = 24$ . Para los subsistemas  $x$  y  $y$  estas ganancias están dadas como  $k_p = 2.5$ ,  $k_i = 0.001$ , y  $k_d = 3$ . La Figura 3.1 muestra el diagrama del esquema MRAC para cada subsistema de rotación.

### 3.4. Resultados de simulación

En esta sección se muestra el desempeño del esquema MRAC para el subsistema rotacional y un PID clásico para el sistema translacional, se le exigió al quadrotor que volara a una altura de 0.5 metros y que en el aire trazara un cuadrado en el plano horizontal de un metro por un metro.

En las Figuras 3.2-3.4 se muestran los resultados de simulación. La Figura 3.2 muestra el desempeño de el MRAC directo descentralizado para el control de orientación de un *quadrotor* UAV. La Figura 3.3 muestra el desempeño del esquema de control PID descentralizado para el control de translación para seguimiento de trayectoria. De la Figura 3.4

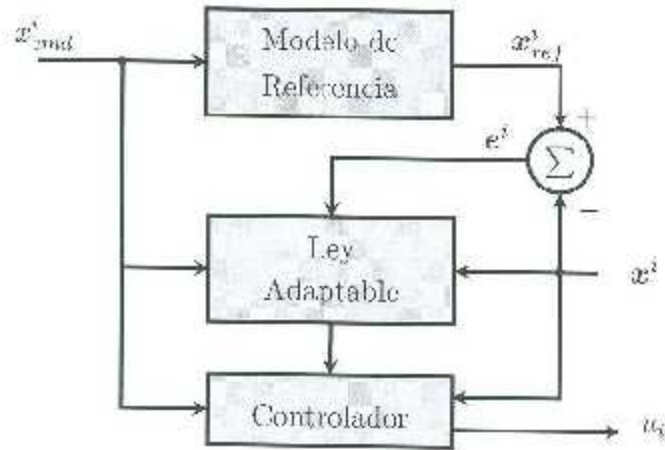


Figura 3.1: Diagrama del MRAC para cada subsistema.

se puede observar que las ganancias de *feedback* y *feedforward* tienden a converger a sus valores (desconocidos) ideales.

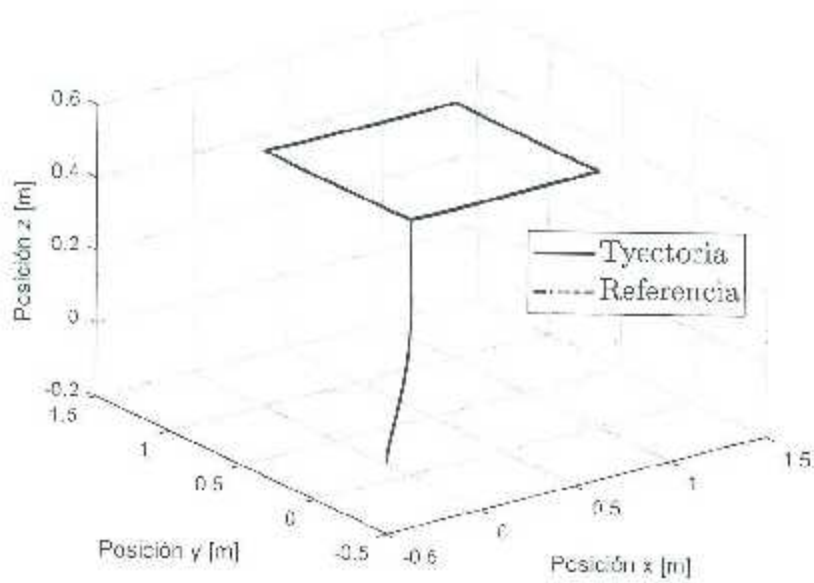


Figura 3.2: Desempeño del MRAC directo descentralizado para control de orientación.

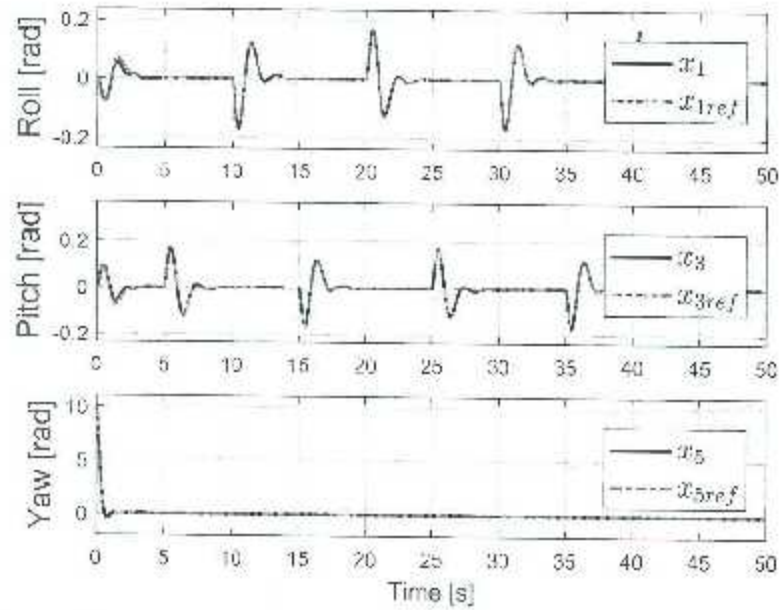


Figura 3.3: Desempeño del esquema de control PID descentralizado para seguimiento de trayectoria.

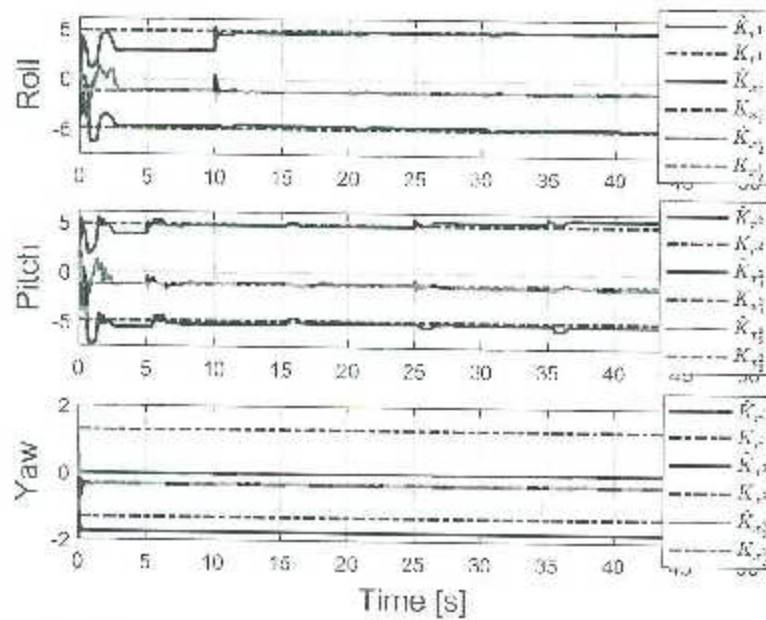


Figura 3.4: Evolución de las ganancias estimadas de *feedback* y *feedforward*.

## Capítulo 4

# Redes neuronales recurrentes de alto orden (RHONNs)

En este capítulo se aborda lo referente a la descripción de una RHONN así como su implementación para la identificación neuronal del comportamiento dinámico de un sistema no lineal en tiempo continuo. Además, se analiza la teoría básica del algoritmo de entrenamiento PE, el cual es una ley adaptable en función del producto de error de identificación y las interconexiones de alto orden.

### 4.1. Introducción

Los esquemas de identificación han sido desarrollados para obtener principalmente los valores de los parámetros desconocidos del sistema, los cuales comúnmente son considerados como constantes, a partir de sus variables medibles como lo son las entradas, las salidas y las posibles perturbaciones [Norgaard et al., 2003].

Dependiendo del nivel de conocimiento *a priori* de la planta, el problema de identificación se puede abordar de diferentes maneras. Si la identificación se basa exclusivamente en los datos medibles, suponiendo que no existe conocimiento acerca de la planta física, el proceso de identificación es denominado *modelado de caja negra*, para el cual resulta ventajoso utilizar ANN como esquemas de identificación. A diferencia de éste, el *modelado de caja blanca* es aquel en donde toda la información necesaria del sistema físico está disponible. Cuando se tiene cierto nivel de conocimiento acerca del sistema se emplea el denominado *modelado de caja gris*. Las ANN son una excelente herramienta en el aprendizaje de relaciones complejas a partir de un conjunto de datos, además de sus inherentes propiedades de adaptabilidad y aproximación; por estas razones han ganado un lugar importante en el proceso de modelado e identificación de sistemas dinámicos, en especial los que involucren no linealidades difíciles de determinar. Sin embargo, cuando se trata de sistemas dinámicos las ANN clásicas no constituyen una buena opción y es por esta razón que el interés se ha orientado hacia el uso de las RNN que, al incluir en su estructura lazos de realimentación facilitan, el modelado, la identificación y el diseño de observadores para sistemas no lineales.

Una RHONN es una de las estructuras de ANN para la identificación de sistemas que po-

seen parámetros variables en el tiempo. Además de ajustar los pesos sinápticos (parámetros de la red neuronal) en línea, tiene la capacidad de aproximar funciones no lineales. Dicha estructura puede ser implementada de dos maneras:

1. Modelo paralelo, en el cual el vector de entrada está en función del estado de la misma red

$$\dot{x}_i = -a_i x_i + \mathbf{w}_i^T \mathbf{z}(\mathbf{x}, \mathbf{u}),$$

2. Modelo serie-paralelo, en el cual el vector de entrada está en función del estado de la planta

$$\dot{x}_i = -a_i x_i + \mathbf{w}_i^T \mathbf{z}(\chi, \mathbf{u});$$

donde  $\{i = 1, 2, \dots, n\}$ ,  $x_i$  es el estado de la  $i$ -ésima neurona,  $\mathbf{x} = [x_1 \dots x_n]^T \in \mathbb{R}^n$ ,  $\chi$  es el vector de estados correspondientes a la planta,  $\mathbf{w}_i$  es el vector de pesos sinápticos y  $\mathbf{u}$  es el vector de entrada de la ANN.

## 4.2. Estructura de la RHONN

Las RNN se caracterizan por tener lazos de realimentación entre neuronas, como se ilustra en la Figura 4.1 lo que las distingue de las redes neuronales con prealimentación (*feedforward neural networks*) donde la salida de cada neurona es conectada solamente a las neuronas de la siguiente capa. En el caso más simple de una RNN tipo Hopfield [Hopfield, 1984], el estado de cada neurona se puede determinar por una ecuación diferencial de la forma [Kosmatopoulos et al., 1997]:

$$\dot{x}_i = -a_i x_i + b_i \sum_j w_{ij} y_j \quad (4.1)$$

donde  $x_i$  es el estado de la  $i$ -ésima neurona,  $a_i$  y  $b_i$  son constantes,  $w_{ij}$  es el peso sináptico conectado de la  $j$ -ésima entrada a la  $i$ -ésima neurona y  $y_j$  es la  $j$ -ésima entrada a la  $i$ -ésima neurona. Cada  $y_j$  es una entrada externa o el estado de una neurona pasado a través de una función sigmoide, i.e.,  $y_j = s(x_j)$  donde  $s(\cdot)$  representa una función sigmoide. En una RNN de segundo orden la entrada a la neurona no solo es la combinación lineal de sus componentes  $y_j$  sino también de sus productos  $y_j y_k$  para incluir interacciones de alto orden representadas por tripletas  $y_j y_k y_l$ , cuádruplas, etc., formando así las RHONN.

Considérese ahora una RHONN compuesta de  $n$  neuronas y  $m$  entradas. El estado de cada neurona es gobernado por una ecuación diferencial de la forma:

$$\dot{x}_i = -a_i x_i + b_i \sum_{k=1}^L w_{ik} \prod_{j \in I_k} y_j^{d_j(k)} \quad (4.2)$$

donde  $\{I_1, I_2, \dots, I_L\}$  son una colección de subconjuntos no ordenados para  $L = \{1, 2, \dots, m+n\}$ ,  $w_{ik}$  son los pesos sinápticos ajustables de la ANN,  $d_j(k)$  son enteros no negativos,  $k$

representa el número de interconexiones de alto orden,  $a_i$  y  $b_i$  son coeficientes reales. El estado de la  $i$ -ésima neurona es nuevamente representado por  $x_i$  y  $\mathbf{y} = [y_1 \ y_2 \ \dots \ y_{n-m}]^T$  es el vector de entradas a cada neurona definido como [Kosmatopoulos et al., 1997]:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \\ y_{n+1} \\ \vdots \\ y_{n+m} \end{bmatrix} = \begin{bmatrix} s(x_1) \\ \vdots \\ s(x_n) \\ u_1 \\ \vdots \\ u_m \end{bmatrix} \quad (4.3)$$

donde  $\mathbf{u} = [u_1 \ u_2 \ \dots \ u_m]^T$  es el vector de entradas externas a la red. La función  $s(\cdot)$  es creciente, diferenciable y es usualmente representada por una función sigmoideal de la forma:

$$s(x) = \frac{\alpha}{1 + e^{-\beta x}} - \gamma \quad (4.4)$$

donde los parámetros  $\alpha$  y  $\beta$  representan la cota y la creciente de la curva sigmoideal, respectivamente, y  $\gamma$  es una constante de crecimiento. En el caso especial donde  $\alpha = \beta = 1$  y  $\gamma = 0$  se obtiene una función logística y para los valores de  $\alpha = \beta = 2$  y  $\gamma = 1$  se obtiene una función tangente hiperbólica; las cuales son las funciones de activación más comúnmente usadas en aplicaciones de ANN (Ver Figura 4.2).

Para simplificar aún más el modelo de la RHONN se define un vector  $\mathbf{z}$  de dimensión  $l$ , tal que:

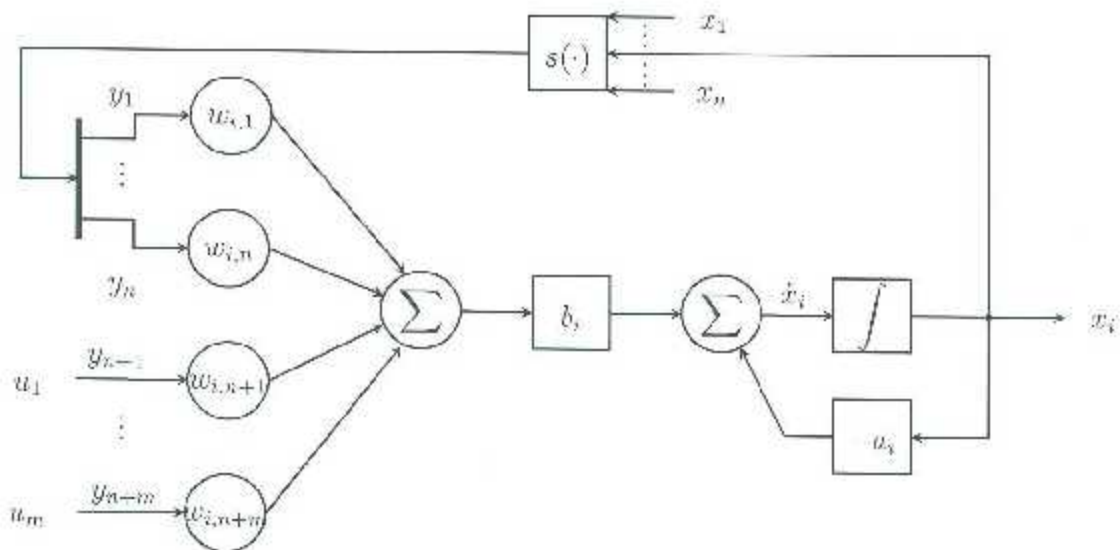


Figura 4.1. Estructura de una red RNN.

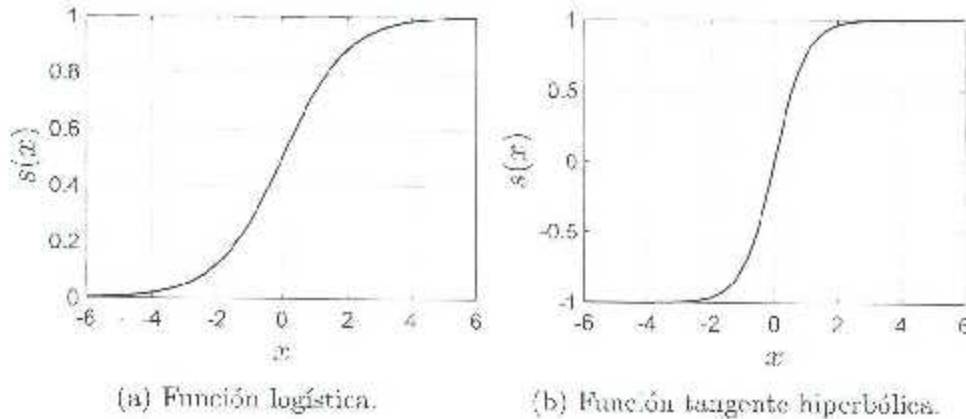


Figura 4.2: Ejemplo gráfico de funciones de activación.

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_L \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} y_j^{d_j(1)} \\ \prod_{j \in I_2} y_j^{d_j(2)} \\ \vdots \\ \prod_{j \in I_L} y_j^{d_j(L)} \end{bmatrix} \quad (4.5)$$

Luego, el modelo (4.2) se puede reescribir como:

$$\dot{x}_i = -a_i x_i + b_i \sum_{k=1}^L w_{ik} z_k \quad (4.6)$$

Más aún, si se define el vector de parámetros ajustables como  $\mathbf{w}_i = b_i [w_{i1} \ w_{i2} \ \dots \ w_{iL}]^T$ , luego (4.6) se convierte en

$$\dot{x}_i = a_i x_i + \mathbf{w}_i^T \mathbf{z} \quad (4.7)$$

Los vectores  $\{\mathbf{w}_i : i = 1, 2, \dots, n\}$  representan los pesos ajustables de la red, mientras que los coeficientes  $\{a_i : i = 1, 2, \dots, n\}$  son parte de la arquitectura fundamental de la red y son fijados durante el entrenamiento.

Para garantizar que cada neurona  $x_i$  sea estable en el sentido entrada-acotada/salida-acotada (*bounded input-bounded output*, BIBO por sus siglas en inglés) se deberá suponer que  $a_i > 0, \forall i = 1, 2, \dots, n$ . En el caso especial de un modelo Hopfield [Hopfield, 1984] en tiempo continuo, se tiene que  $a_i = 1/(R_i C_i)$ , donde  $R_i > 0$  y  $C_i > 0$  son la resistencia y la capacitancia, respectivamente, conectadas al  $i$ ésimo nodo de la red.

El comportamiento dinámico de la red en conjunto se describe expresando (4.6) en notación vectorial como:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{W}^T \mathbf{z}, \quad (4.8)$$

donde  $\mathbf{W} \in \mathbb{R}^{L \times n}$  y  $\mathbf{A} = \text{diag}\{-a_1, -a_2, \dots, -a_n\} \in \mathbb{R}^{n \times n}$  es una matriz diagonal. Siendo que  $a_i > 0 \forall i = 1, 2, \dots, n$  luego  $\mathbf{A}$  es una matriz de estabilidad (Hurwitz). El



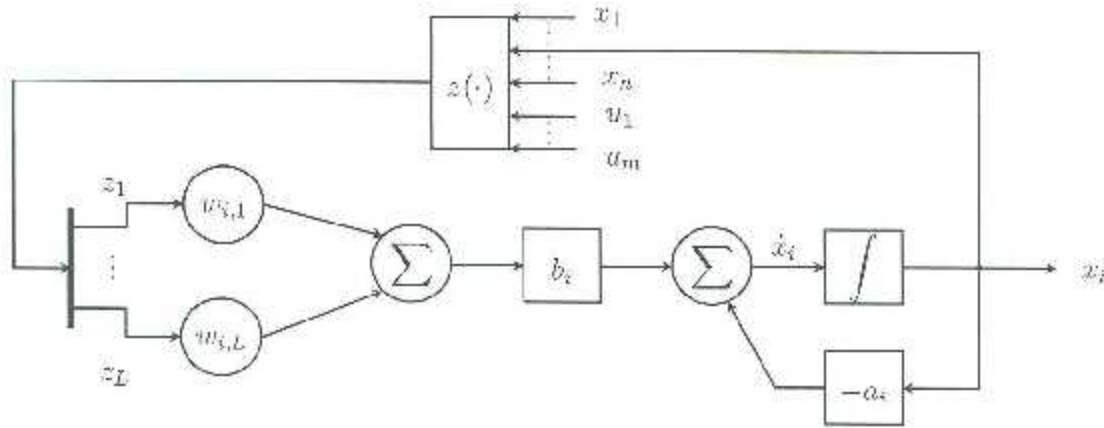


Figura 4.3: Estructura de una RHONN.

vector  $\mathbf{z}$  es una función tanto del estado  $\mathbf{x}$  de la ANN como de la entrada externa  $\mathbf{u}$ . La estructura RHONN se ilustra en la Figura 4.3.

### 4.3. Propiedades de aproximación de la RHONN

Se considera el problema de aproximación de un sistema dinámico no lineal cuyo comportamiento entrada-salida está dado por

$$\dot{\chi} = F(\chi, \mathbf{u}), \quad (4.9)$$

donde  $\chi \in \mathbb{R}^n$  es el estado del sistema,  $\mathbf{u} \in \mathbb{R}^m$  es la entrada del sistema y  $F: \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$  es un campo vectorial suave definido en un conjunto compacto  $\mathcal{Y} \subset \mathbb{R}^{n+m}$ . El problema de aproximación consiste en determinar si con un número suficiente de conexiones de alto orden existen pesos  $\mathbf{W}$  tales que el modelo RHONN (4.8) aproxime el comportamiento entrada-salida de un sistema dinámico arbitrario de la forma (4.9). Para tener un problema bien definido se supone que  $F(\cdot)$  es continua y satisface la condición local de Lipschitz, *i.e.*, asegurar la unicidad de la solución de modo que (4.9) tenga una única solución en el sentido de Carathéodory [Hale, 1969] y  $(\chi(t), \mathbf{u}(t)) \in \mathcal{Y}$  para todo  $t$  en algún intervalo de tiempo  $J_T = \{t: 0 \leq t \leq T\}$ . El intervalo  $J_T$  representa el periodo de tiempo sobre el cual la aproximación es efectuada. Con base en la suposición anterior se obtiene el siguiente resultado.

**Teorema 2.** [Kosmatopoulos et al., 1995] Suponiendo que el sistema real (4.9) y el modelo de identificación (4.8) están inicialmente en el mismo estado inicial  $\mathbf{x}(0) = \chi(0)$ , entonces para cualquier  $\varepsilon > 0$  y todo tiempo finito  $T > 0$  existe un elemento  $L$  y una matriz de valores desconocidos  $\mathbf{W}^* \in \mathbb{R}^{L \times n}$  tales que el estado  $\mathbf{x}(t)$  del modelo RHONN (4.8) con  $L$  conexiones de alto orden y valores de pesos  $\mathbf{W} = \mathbf{W}^*$  satisface

$$\sup_{0 \leq t \leq T} \|\mathbf{x}(t) - \chi(t)\| \leq \varepsilon$$

*Demostración.* [Kosmatopoulos et al., 1995] El comportamiento dinámico del modelo RHONN es descrito por (4.8)

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{W}^T \mathbf{z}$$

Sumando y restando  $\mathbf{A}\boldsymbol{\chi}$ , el sistema (4.9) puede ser reescrito como:

$$\dot{\boldsymbol{\chi}} = \mathbf{A}\boldsymbol{\chi} + \mathbf{G}(\boldsymbol{\chi}, \mathbf{u}) \quad (4.10)$$

donde  $\mathbf{G}(\boldsymbol{\chi}, \mathbf{u}) = \mathbf{F}(\boldsymbol{\chi}, \mathbf{u}) - \mathbf{A}\boldsymbol{\chi}$ . Dado que  $\mathbf{x}(0) = \boldsymbol{\chi}(0)$ , el error de identificación  $\boldsymbol{\xi} = \mathbf{x} - \boldsymbol{\chi}$  satisface

$$\dot{\boldsymbol{\xi}} = \mathbf{A}\boldsymbol{\xi} + \mathbf{W}^T \mathbf{z}(\mathbf{x}, \mathbf{u}) - \mathbf{G}(\boldsymbol{\chi}, \mathbf{u}) \quad (4.11)$$

con  $\boldsymbol{\xi}(0) = 0$ .

Suponiendo que  $(\boldsymbol{\chi}(t), \mathbf{u}(t)) \in \mathcal{Y} \forall t \in [0, T]$  donde  $\mathcal{Y}$  es un conjunto compacto de  $\mathbb{R}^{n+m}$  se tiene

$$\mathcal{Y}_\varepsilon = \{(\boldsymbol{\chi}, \mathbf{u}) \in \mathbb{R}^{n+m} : |(\boldsymbol{\chi}, \mathbf{u}) - (\boldsymbol{\chi}_y, \mathbf{u}_y)| \leq \varepsilon, (\boldsymbol{\chi}_y, \mathbf{u}_y) \in \mathcal{Y}\}. \quad (4.12)$$

Puede apreciarse que  $\mathcal{Y}_\varepsilon$  es también un subconjunto compacto de  $\mathbb{R}^{n+m}$  y  $\mathcal{Y} \subset \mathcal{Y}_\varepsilon$ . En otras palabras,  $\varepsilon$  es el grado requerido de aproximación. Dado que  $\mathbf{z}(\mathbf{x}, \mathbf{u})$  es una función continua que satisface la condición Lipschitz en el dominio compacto  $\mathcal{Y}_\varepsilon$ , i.e., existe una constante  $l$  tal que para todo  $(\mathbf{x}_1, \mathbf{u}), (\mathbf{x}_2, \mathbf{u}) \in \mathcal{Y}_\varepsilon$

$$|\mathbf{z}(\mathbf{x}_1, \mathbf{u}) - \mathbf{z}(\mathbf{x}_2, \mathbf{u})| \leq l|\mathbf{x}_1 - \mathbf{x}_2|. \quad (4.13)$$

La función  $\mathbf{W}^T \mathbf{z}(\mathbf{x}, \mathbf{u})$  debe satisfacer las condiciones del teorema de Stone-Weierstrass [Cotter, 1990] que dicta lo siguiente: Las funciones reales continuas definidas en un intervalo cerrado y acotado pueden ser aproximadas tanto como se quiera por un polinomio. Es decir, los polinomios de coeficientes reales son densos en el conjunto de las funciones continuas sobre un intervalo cerrado.

De (4.2) y (4.3) se puede ver que  $\mathbf{z}(\mathbf{x}, \mathbf{u})$  es un vector de productos de alto orden entre elementos de  $\mathbf{y}$ , el cual está formado por el vector  $\mathbf{x}$  preprocesado por una función sigmoideal  $s(\cdot)$  y el vector  $\mathbf{u}$ . El procesamiento de la entrada a través de una función invertible continua no afecta la habilidad de la ANN para aproximar funciones continuas [Cotter, 1990]. Sin embargo, se ha demostrado que si  $L$  es lo suficientemente grande, i.e., el número de pesos es lo suficientemente grande, entonces existen valores de pesos  $\mathbf{W} = \mathbf{W}^*$  tal que  $(\mathbf{W}^*)^T \mathbf{z}(\mathbf{x}, \mathbf{u})$  puede aproximar a  $\mathbf{G}(\boldsymbol{\chi}, \mathbf{u})$  en un dominio compacto para cualquier grado de precisión para todo  $(\mathbf{x}, \mathbf{u})$ . Por lo tanto, existe  $\mathbf{W} = \mathbf{W}^*$  tal que

$$\sup_{(\boldsymbol{\chi}, \mathbf{u}) \in \mathcal{Y}_\varepsilon} |(\mathbf{W}^*)^T \mathbf{z}(\mathbf{x}, \mathbf{u}) - \mathbf{G}(\boldsymbol{\chi}, \mathbf{u})| \leq \eta \quad (4.14)$$

donde  $\eta$  es una constante positiva.

La solución para la ecuación diferencial (4.11) puede ser expresada como:

$$\begin{aligned}
 \xi(t) &= \int_0^t \xi^{-A(t-\tau)} [(W^*)^t z(x(\tau), u(\tau)) - G(\chi(\tau), u(\tau))] d\tau \\
 &= \int_0^t \xi^{-A(t-\tau)} [(W^*)^t z(x(\tau), u(\tau)) - (W^*)^T z(\chi(\tau), u(\tau))] d\tau \\
 &\quad + \int_0^t \xi^{-A(t-\tau)} [(W^*)^T z(\chi(\tau), u(\tau)) - G(\chi(\tau), u(\tau))] d\tau. \quad (4.15)
 \end{aligned}$$

Dado que  $A$  es una matriz definida positiva, existe una constante positiva  $\alpha$  tal que  $\|\xi^{At}\| \leq \xi^{-\alpha t}$  y  $\alpha < L$ , donde  $L = l \|W^*\|$ . Por todo lo anterior,  $\eta$  puede elegirse como

$$\eta = \frac{(L - \alpha)\varepsilon}{2(\xi^{(L-\alpha)T} - 1)} > 0. \quad (4.16)$$

Considere ahora que  $(x(t), u(t)) \in \mathcal{Y}_\varepsilon, \forall t \in [0, T]$ . De (4.15), considerando la norma en ambos lados y usando (4.13), (4.14) y (4.16), las siguientes desigualdades se cumplen para  $t \in [0, T]$ :

$$\begin{aligned}
 |\xi(t)| &\leq \int_0^t \|\xi^{-A(t-\tau)}\| \cdot |(W^*)^T z(x(\tau), u(\tau)) - (W^*)^T z(\chi(\tau), u(\tau))| d\tau \\
 &\quad + \int_0^t \|\xi^{-A(t-\tau)}\| \cdot |(W^*)^T z(\chi(\tau), u(\tau)) - G(\chi(\tau), u(\tau))| d\tau \\
 &\leq \int_0^t \xi^{-\alpha(t-\tau)} L |\xi(\tau)| d\tau + \int_0^t \eta \xi^{-\alpha(t-\tau)} d\tau
 \end{aligned}$$

Usando el Lema de Bellman-Gronwall [Hale, 1969] que establece que existe una cota superior para las funciones no negativas que puedan acotarse por una función lineal de su integral, éste lema es de gran utilidad para probar la continuidad y unicidad de soluciones de ecuaciones diferenciales ordinarias; por lo anterior se tiene

$$\begin{aligned}
 |\xi(t)| &\leq \frac{\eta}{L - \alpha} (\xi^{(L-\alpha)t} - 1), \\
 |\xi(t)| &\leq \varepsilon/2 \quad (4.17)
 \end{aligned}$$

Ahora, bajo la suposición de que  $(x, u)$  no está contenida en  $\mathcal{Y}_\varepsilon, \forall t \in [0, T]$ , luego, por la continuidad de  $x(t)$  existe  $T^*$  donde  $0 \leq T^* < T$  tal que  $(x(T^*), u(T^*)) \in \partial\mathcal{Y}_\varepsilon$ , donde  $\partial\mathcal{Y}_\varepsilon$  expresa el acotamiento de  $\mathcal{Y}_\varepsilon$ . Realizando el mismo análisis para  $t \in [0, T^*]$  se obtiene que para este intervalo  $|x(t) - \chi(t)| < \varepsilon/2$ , por lo tanto, (4.17) se mantiene para todo  $t \in [0, T]$ . ■

El teorema anterior establece que, si un número lo suficientemente grande de conexiones es permitido en el modelo RHONN luego será posible aproximar cualquier sistema dinámico para cualquier grado de precisión. Este es estrictamente un resultado de existencia y no proporciona algún método constructivo para obtener los pesos  $W^*$  óptimos.

A continuación se considera el problema de aprendizaje en el ajuste adaptable de los pesos tal que el modelo RHONN identifique, en general, sistemas dinámicos.

## 4.4. Algoritmo de aprendizaje

En la presente sección se describe una ley ajustable de pesos bajo la suposición de que el sistema desconocido puede modelarse exactamente por una arquitectura del tipo RHONN (4.8) [Rovithakis and Christodoulou, 2012]. Este análisis se extiende en la siguiente sección para cubrir el caso en el que existe una diferencia entre el sistema y el modelo RHONN con valores de pesos óptimos. Esta diferencia es denominada como *error de modelado*.

Aunque la suposición de que no existen errores de modelado no es muy realista, el procedimiento de identificación de esta sección es útil por dos razones:

1. El análisis es más directo y fácil de entender;
2. Las técnicas desarrolladas para el caso en el que no existe error de modelado también son importantes en el diseño de leyes de adaptación de pesos en la presencia de errores de modelado.

Con base en la suposición de que no hay errores de modelado, existen vectores de pesos desconocidos  $w_i^*$  ( $i = 1, 2, \dots, n$ ) tales que cada estado  $\chi_i$  del sistema dinámico desconocido (4.9) satisface

$$\dot{\chi}_i = -a_i \chi_i + w_i^{*T} z(\chi, u), \quad \chi_i(0) = \chi_i^0 \quad (4.18)$$

donde  $\chi_i^0$  es el  $i$ -ésimo estado inicial del sistema. De aquí en adelante, por simplicidad en la notación los argumentos del campo vectorial  $z$  serán omitidos. Como es usual en los procedimientos de identificación de sistemas se supondrá que la entrada  $u(t)$  y el estado  $\chi(t)$  permanecen acotados para todo  $t \geq 0$ . Basándose en la definición (4.5),  $z(\chi, u)$  está acotado. A continuación se resume un enfoque seleccionado para su estructura simple para la estimación de sus parámetros desconocidos  $w_i^*$  del modelo RHONN, el cual requiere solo de  $n$  filtros y para el cual el cálculo computacional es más sencillo en comparación con el enfoque de filtrado del regresor [Rovithakis and Christodoulou, 2012].

## 4.5. RHONN entrenada vía FE

El desarrollo del presente esquema de identificación parte de la ecuación diferencial que describe al sistema desconocido, i.e.,

$$\dot{\chi}_i = -a_i \chi_i + w_i^{*T} z \quad (4.19)$$

Con base en (4.19) el identificador es elegido como,

$$\dot{\hat{x}}_i = -a_i \hat{x}_i - w_i^T z \quad (4.20)$$

donde  $\mathbf{w}_i$  es el estimado del vector desconocido  $\mathbf{w}_i^*$ . En este caso, el error de identificación delimitado como  $\xi_i = x_i - \chi_i$  satisface

$$\begin{aligned}\dot{\xi}_i &= \dot{x}_i - \dot{\chi}_i \\ &= -a_i x_i + \mathbf{w}_i^T \mathbf{z} - (-a_i \chi_i + \mathbf{w}_i^{*T} \mathbf{z}) \\ &= -a_i x_i + \mathbf{w}_i^T \mathbf{z} + a_i \chi_i - \mathbf{w}_i^{*T} \mathbf{z} \\ &= -a_i (x_i - \chi_i) + (\mathbf{w}_i^T - \mathbf{w}_i^{*T}) \mathbf{z},\end{aligned}$$

que puede reescribirse como

$$\dot{\xi}_i = -a_i \xi_i + \tilde{\mathbf{w}}_i^T \mathbf{z} \quad (4.21)$$

donde  $\tilde{\mathbf{w}} = \mathbf{w}_i - \mathbf{w}_i^*$ . Los pesos  $\mathbf{w}_i$ , para  $i = 1, 2, \dots, n$  son ajustados de acuerdo a la ley de aprendizaje [Kosmatopoulos et al., 1995]

$$\dot{\mathbf{w}}_i = -\Gamma_i \mathbf{z} \xi_i \quad (4.22)$$

donde la ganancia adaptable  $\Gamma_i \in \mathbb{R}^{L \times L}$  es una matriz definida positiva. En el caso especial en el que  $\Gamma_i = \gamma_i \mathbf{I}$ , donde  $\gamma_i > 0$  es un escalar, luego  $\Gamma_i$  en (4.22) puede ser reemplazada por  $\gamma_i$ , i.e.,

$$\dot{\mathbf{w}}_i = -\gamma_i \mathbf{z} \xi_i \quad (4.23)$$

Las ecuaciones (4.22) y (4.23) son denominadas FE. Las propiedades de estabilidad y convergencia de la ley ajustable de pesos dada por (4.22) son bien conocidas en la literatura del control adaptable [Narendra and Annaswamy, 2012].

El siguiente teorema demuestra las propiedades de convergencia del sistema de identificación en cuestión al emplear el método del gradiente en el ajuste de los pesos.

**Teorema 3.** [Kosmatopoulos et al., 1995] Considere el modelo RHONN con el algoritmo de entrenamiento FE dado por (4.20), cuyos pesos son ajustables de acuerdo a (4.22). Luego, para  $i = 1, 2, \dots, n$ ,

1.  $\xi_i, \tilde{\mathbf{w}}_i \in \mathcal{L}_\infty$  (i.e.,  $\xi_i$  y  $\tilde{\mathbf{w}}_i$  son uniformemente acotadas),
2.  $\lim_{t \rightarrow \infty} \xi_i(t) = 0$ .

*Demostración* [Kosmatopoulos et al., 1995]. Considere la función candidata de Lyapunov

$$V = \frac{1}{2} \sum_{i=1}^n (\xi_i^2 + \tilde{\mathbf{w}}_i^T \Gamma_i^{-1} \tilde{\mathbf{w}}_i). \quad (4.24)$$

Luego, usando (4.21)-(4.22) y bajo el hecho de que  $\dot{\tilde{\mathbf{w}}}_i = -\dot{\mathbf{w}}_i$ , la derivada  $\dot{V}$  de  $V$  satisface

$$\dot{V} = -\sum_{i=1}^n a_i \xi_i^2 \leq 0 \quad (4.25)$$

Comprobación:

$$\begin{aligned}
 \dot{V} &= \frac{1}{2} \sum_{i=1}^n \dot{\xi}_i^2 - \frac{1}{2} \sum_{i=1}^n \dot{\tilde{w}}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \\
 \dot{V} &= \sum_{i=1}^n \xi_i \dot{\xi}_i - \frac{1}{2} \sum_{i=1}^n \left[ \dot{\tilde{w}}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i - \tilde{w}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \right] \\
 \dot{V} &= \sum_{i=1}^n \xi_i \dot{\xi}_i + \frac{1}{2} \sum_{i=1}^n \left[ 2\tilde{w}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \right] \\
 \dot{V} &= \sum_{i=1}^n \left[ \xi_i \dot{\xi}_i - \tilde{w}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \right]
 \end{aligned}$$

Evaluando ésta última expresión a lo largo de las trayectorias de (4.21), se obtiene

$$\begin{aligned}
 \dot{V} &= \sum_{i=1}^n \left[ \xi_i (-a_i \xi_i + \tilde{w}_i^T z) + \tilde{w}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \right] \\
 \dot{V} &= \sum_{i=1}^n \left[ -a_i \xi_i^2 + \xi_i \tilde{w}_i^T z + \tilde{w}_i^T \Gamma_i^{-1} \dot{\tilde{w}}_i \right]
 \end{aligned}$$

Considerando la ley adaptable para los pesos sinápticos (4.22) y evaluando la expresión anterior se puede simplificar hasta obtener:

$$\begin{aligned}
 \dot{V} &= \sum_{i=1}^n \left[ -a_i \xi_i^2 + \xi_i \tilde{w}_i^T z + \tilde{w}_i^T \Gamma_i^{-1} (-\Gamma_i z \xi_i) \right] \\
 \dot{V} &= \sum_{i=1}^n \left[ -a_i \xi_i^2 - \xi_i \tilde{w}_i^T z - \xi_i \tilde{w}_i^T z \right] \\
 \dot{V} &= \sum_{i=1}^n -a_i \xi_i^2
 \end{aligned}$$

Ya que  $\dot{V} \leq 0$ , de (4.24) se tiene que  $\xi_i, \tilde{w}_i \in \mathcal{L}_\infty$  para  $i = 1, \dots, n$ . Usando este resultado en (4.21) se tiene que  $\dot{\xi}_i \in \mathcal{L}_\infty$  y además  $\xi_i \in \mathcal{L}_2$ , *i.e.*,  $\xi_i(t)$  es cuadrático integrable y por lo tanto, aplicando el lema de Barbalat [Zax, 2003], se tiene que  $\lim_{t \rightarrow \infty} \xi_i(t) = 0$ . ■

En [Rovithakis and Christodoulou, 2012] se propone otro esquema de identificación adaptable denominado RHONN con filtrado del regresor en tiempo continuo. Dicho esquema se apoya del filtrado del vector  $z$ , el cual es algunas veces denominado como el vector regresor. Usando ésta técnica de filtrado es posible obtener una muy simple expresión algebraica para el error, lo que permite la aplicación de procedimientos de optimización para el análisis y diseño de leyes de ajuste de pesos. Sin embargo, dicho método presenta la desventaja de que demanda una configuración compleja y alto costo computacional. Generalmente la dimensión del regresor será mucho más grande que la dimensión del sistema, *i.e.*,  $L > n$ ; lo que puede resultar muy costoso computacionalmente al emplear una gran cantidad de filtros. La RHONN con FE es de estructura simple, la cual requiere únicamente de  $n$  filtros, lo que conlleva a una menor carga computacional.

## Capítulo 5

# Identificación y control neuronal de un *quadrotor* vía RHONNs

En este capítulo se describe el procedimiento de identificación neuronal descentralizada del modelo matemático de un *quadrotor* empleando una arquitectura RHONN, entrenada con el algoritmo de FE, y cuyo proceso es validado mediante simulación para luego evaluar sobre la plataforma experimental Q-ball 2 de Quanser.

### 5.1. Introducción

Para identificar de manera descentralizada las dinámicas de un *quadrotor* se requiere descomponer el modelo no lineal completo en subsistemas, para luego diseñar una RHONN que identifique cada uno de los mismos. El proceso de diseño del identificador neuronal requiere de prueba y error para conseguir una arquitectura que entregue resultados satisfactorios, esto le atribuye cierta flexibilidad al diseño de la ANN. A pesar de que no es necesario el conocimiento completo del modelo del *quadrotor* para su identificación práctica lo es para evaluar su desempeño en simulación.

### 5.2. Identificación neuronal descentralizada

Un sistema se puede considerar como un conjunto de subsistemas interconectados, por lo que es posible diseñar identificadores y controladores considerando solamente las variables locales de cada subsistema en donde los efectos de interconexión pueden ser vistos como perturbaciones [Huang et al., 2003].

La estrategia de control descentralizado resulta ventajosa en el diseño de controladores de sistemas no lineales complejos de alto orden. En el presente estudio de tesis, siguiendo la estrategia en cuestión, se separaron las ecuaciones que describen la dinámica completa de un *quadrotor* agrupándose en subsistemas de segundo orden para cada una de las coordenadas de movimiento que describen las dinámicas traslacional y rotacional, sin embargo lo es para su evaluación en simulación.

### 5.2.1. Descentralización del modelo dinámico del *quadrotor*

En el Capítulo 2 se obtuvo el modelo dinámico de un *quadrotor* como se muestra en (2.33), el cual puede ser separado en  $N$  subsistemas de la forma [Sauchez et al., 2006]:

$$\dot{\chi}_2^i = f^i(\chi_1^i, \chi_2^i, u^i) + \nu^i(\chi, \dot{\chi}, U), \quad (5.1)$$

donde  $i = 1, 2, \dots, N$ ,  $f^i(\cdot) \in \mathbb{R}$  depende solo de variables locales,  $\nu^i(\cdot) \in \mathbb{R}$  representa los efectos de interconexión,  $\chi_1^i, \chi_2^i \in \mathbb{R}$  representan la posición y velocidad del  $i$ -ésimo subsistema, respectivamente,  $u^i \in \mathbb{R}$  es la entrada principal de cada subsistema,  $\chi \in \mathbb{R}^n$  es el vector de estados del sistema y  $U \in \mathbb{R}^m$  es el vector de entradas del sistema. De acuerdo a esto, las ecuaciones que gobiernan al sistema (2.33) se pueden representar en la ecuación de estado definiendo

$$\begin{aligned} x_1 = \phi, \quad x_3 = \theta, \quad x_5 = \psi, \quad x_7 = z, \quad x_9 = x, \quad x_{11} = y, \\ x_2 = \dot{\phi}, \quad x_4 = \dot{\theta}, \quad x_6 = \dot{\psi}, \quad x_8 = \dot{z}, \quad x_{10} = \dot{x}, \quad x_{12} = \dot{y}. \end{aligned} \quad (5.2)$$

Así, las dinámicas pueden escribirse en la forma descentralizada:

$$S_z \begin{cases} \dot{x}_7 = x_8, \\ \dot{x}_8 = (\cos x_1 \cos x_3) \frac{U_1}{m} - g; \end{cases} \quad (5.3)$$

$$S_x \begin{cases} \dot{x}_9 = x_{10}, \\ \dot{x}_{10} = (\cos x_1 \sin x_3 \cos x_5 - \sin x_1 \sin x_5) \frac{u_1}{m}; \end{cases} \quad (5.4)$$

$$S_y \begin{cases} \dot{x}_{11} = x_{12}, \\ \dot{x}_{12} = (\cos x_1 \sin x_3 \sin x_5 - \sin x_1 \cos x_5) \frac{u_1}{m}; \end{cases} \quad (5.5)$$

$$S_\phi \begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = \frac{J_y - J_z}{J_x} x_4 x_6 - \frac{J_{I\phi}}{J_x} x_4 \Omega + \frac{u_2}{J_x}; \end{cases} \quad (5.6)$$

$$S_\theta \begin{cases} \dot{x}_3 = x_4, \\ \dot{x}_4 = \frac{J_z - J_x}{J_y} x_2 x_6 + \frac{J_{I\theta}}{J_y} x_2 \Omega + \frac{u_3}{J_y}; \end{cases} \quad (5.7)$$

$$S_\psi \begin{cases} \dot{x}_5 = x_6, \\ \dot{x}_6 = \frac{(J_x - J_y)}{J_z} J_2 x_4 + \frac{u_4}{J_z}. \end{cases} \quad (5.8)$$

De esta manera, se han separado las ecuaciones que describen las dinámicas de un *quadrotor* agrupándose en subsistemas de segundo orden para cada una de las coordenadas de movimiento que describen las dinámicas traslacional y rotacional.

### 5.2.2. Arquitectura de la red

La topología o arquitectura de una ANN consiste en la organización y disposición de las neuronas en la misma, formando capas o agrupaciones de neuronas. Se distinguen tres tipos de capas: de entrada, de salida y ocultas. El problema de identificación del *quadrotor*



incluye la posición y velocidad de cada uno de los subsistemas, de esta manera cada ANN trabajará de forma independiente de las demás. Una característica que se debe tomar en cuenta al diseñar la ANN es la de mantener la arquitectura lo más simple posible, teniendo en mente facilitar el diseño de algún controlador.

### 5.2.3. Red neuronal descentralizada

Con el fin de identificar cada uno de los subsistemas (5.3)-(5.8), se propone la siguiente estructura para la RHONN:

$$\begin{aligned}\dot{x}_1^i &= -a_1^i x_1^i + w_{11}^i z_{11}^i(\chi_1^i) + x_2^i \\ \dot{x}_2^i &= -a_2^i x_2^i + w_{21}^i z_{21}^i(\chi_1^i) + w_{22}^i z_{22}^i(\chi_2^i) - w_{23}^i z_{23}^i(\chi_1^i, \chi_2^i) + u^i,\end{aligned}\quad (5.9)$$

la cual se puede expresar en forma compacta como:

$$\begin{aligned}\dot{x}_1^i &= -a_1^i x_1^i - (w_1^i)^T z_1^i + x_2^i \\ \dot{x}_2^i &= -a_2^i x_2^i - (w_2^i)^T z_2^i + u^i,\end{aligned}\quad (5.10)$$

donde  $i = 1, 2, 3, 4, 5, 6$  indica el subsistema,  $k = 1, 2, 3$  representa el número de interconexiones de alto orden,  $x_1^i$  y  $x_2^i$  son los estados de la  $i$ -ésima ANN que identifica a  $\chi_1^i$  y  $\chi_2^i$ , respectivamente, las cuales representan la posición y velocidad del  $i$ -ésimo subsistema,  $a_1^i$  y  $a_2^i$  son constantes positivas de la ANN descentralizada,  $z_k^i$  están en función de la posición y la velocidad de cada subsistema, incluyendo un término de alto orden como se muestra enseguida:

$$\begin{aligned}z_1^i &= \tanh(\chi_1^i), \\ z_2^i &= \tanh(\chi_2^i), \\ z_3^i &= \tanh(\chi_1^i) \tanh(\chi_2^i),\end{aligned}\quad (5.11)$$

$w_{jk}$  representa los pesos sinápticos de la ANN descentralizada,  $u^i$  es la entrada principal de cada subsistema.

Para cada  $i$ -ésimo subsistema el error de identificación para la posición es definido como

$$\xi_1^i = x_1^i - \chi_1^i \quad (5.12)$$

mientras que

$$\xi_2^i = x_2^i - \chi_2^i \quad (5.13)$$

es el error de identificación para la velocidad, tanto si se trata con las coordenadas traslacionales o rotacionales. Para actualizar los pesos sinápticos en línea la ley de aprendizaje adaptable está dada por  $\dot{w}_{jk}^i = -\gamma_{jk}^i z_k^i \xi_j^i$ , (i.e.,

$$\begin{aligned}\dot{w}_{11}^i &= -\gamma_{11}^i z_{11}^i \xi_1^i, \\ \dot{w}_{21}^i &= -\gamma_{21}^i z_{21}^i \xi_2^i, \\ \dot{w}_{22}^i &= -\gamma_{22}^i z_{22}^i \xi_2^i, \\ \dot{w}_{23}^i &= -\gamma_{23}^i z_{23}^i \xi_2^i.\end{aligned}\quad (5.14)$$

con  $\gamma_{jk}^i > 0$  como la ganancia adaptable.

### 5.3. Control *backstepping*

En [Khalil, 1996], se desarrolla un procedimiento recursivo para el diseño de una ley de control estabilizante para una clase de sistemas no lineales. El procedimiento de diseño es denominado como *backstepping*, el cual consiste en considerar primeramente la salida deseada y retroceder a través del sistema seleccionando valores deseables de los componentes de estado hasta que aparezca la entrada actual de control  $u(t)$ . Considérese el siguiente sistema

$$\dot{\chi} = f(\chi) + g(\chi)\xi \tag{5.15}$$

$$\dot{\xi} = u, \tag{5.16}$$

donde  $[\chi^T, \xi]^T \in \mathbb{R}^{n+1}$  es el estado y  $u \in \mathbb{R}$  es la entrada de control. Las funciones  $f: D \rightarrow \mathbb{R}^n$  y  $g: D \rightarrow \mathbb{R}^n$  son suaves en un dominio  $D \subset \mathbb{R}^n$  que contiene  $\chi = 0$  y  $f(0) = 0$ . Se desea diseñar una ley de control por realimentación de estados para estabilizar el origen ( $\chi = 0, \xi = 0$ ). Se suponen  $f$  y  $g$  como conocidas. Este sistema puede ser visto como una conexión en cascada de dos componentes, como se muestra en el diagrama a bloques de la Figura 5.1. El primer componente es (5.15) con  $\xi$  como entrada y el segundo componente es el integrador (5.16). Suponga que el componente (5.15) pueda ser estabilizado por una ley de control por realimentación de estados  $\xi = \alpha(\chi)$  con  $\alpha(0) = 0$ ; i.e., el origen de

$$\dot{\chi} = f(\chi) + g(\chi)\alpha(\chi)$$

es asintóticamente estable. Supóngase además que se conoce una función candidata de Lyapunov  $V(\chi)$  suave y definida positiva que satisface la desigualdad

$$\frac{\partial V}{\partial \chi} [f(\chi) + g(\chi)\alpha(\chi)] \leq -W(\chi) \quad \forall \chi \in D \tag{5.17}$$

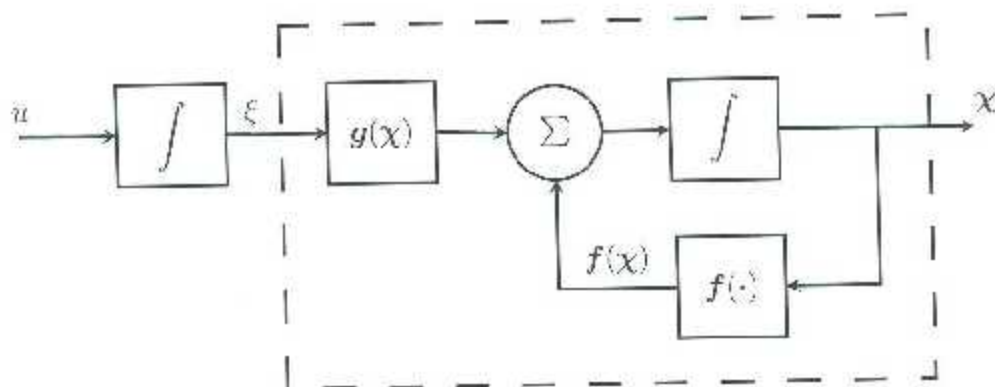


Figura 5.1: Diagrama a bloques del sistema (5.15)-(5.16).

donde  $W(\mathbf{x})$  es definida positiva. Sumando y restando  $\mathbf{g}(\mathbf{x})\alpha(\mathbf{x})$  en la parte derecha de la ecuación (5.15) se obtiene la representación equivalente

$$\begin{aligned}\dot{\mathbf{x}} &= [\mathbf{f}(\mathbf{x}) - \mathbf{g}(\mathbf{x})\alpha(\mathbf{x})] + \mathbf{g}(\mathbf{x})[\xi - \alpha(\mathbf{x})], \\ \dot{\xi} &= u;\end{aligned}$$

tal que los sistemas (5.15)-(5.16) son representados como en el diagrama a bloques de la Figura 5.2.

Al realizar el siguiente cambio de variables

$$\zeta = \xi - \alpha(\mathbf{x}) \quad (5.18)$$

resulta el sistema:

$$\begin{aligned}\dot{\mathbf{x}} &= [\mathbf{f}(\mathbf{x}) - \mathbf{g}(\mathbf{x})\alpha(\mathbf{x})] + \mathbf{g}(\mathbf{x})\zeta, \\ \dot{\zeta} &= u - \dot{\alpha}(\mathbf{x});\end{aligned} \quad (5.19)$$

el cual se muestra en la Figura 5.3. Al comparar las Figuras 5.2 y 5.3 se puede ver que en la última se exhibe un retroceso (un paso atrás o *backstepping*) de  $-\alpha(\mathbf{x})$  a través del integrador.

Suponiendo  $\mathbf{f}$ ,  $\mathbf{g}$  y  $\alpha$  como funciones conocidas, la derivada  $\dot{\alpha}$  puede calcularse mediante la expresión

$$\dot{\alpha} = \frac{\partial \alpha}{\partial \mathbf{x}} [\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\zeta]. \quad (5.20)$$

Definiendo

$$v = u - \dot{\alpha}, \quad (5.21)$$

el sistema se reduce a la conexión en cascada

$$\dot{\mathbf{x}} = [\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\alpha(\mathbf{x})] - \mathbf{g}(\mathbf{x})\zeta, \quad (5.22)$$

$$\dot{\zeta} = v. \quad (5.23)$$

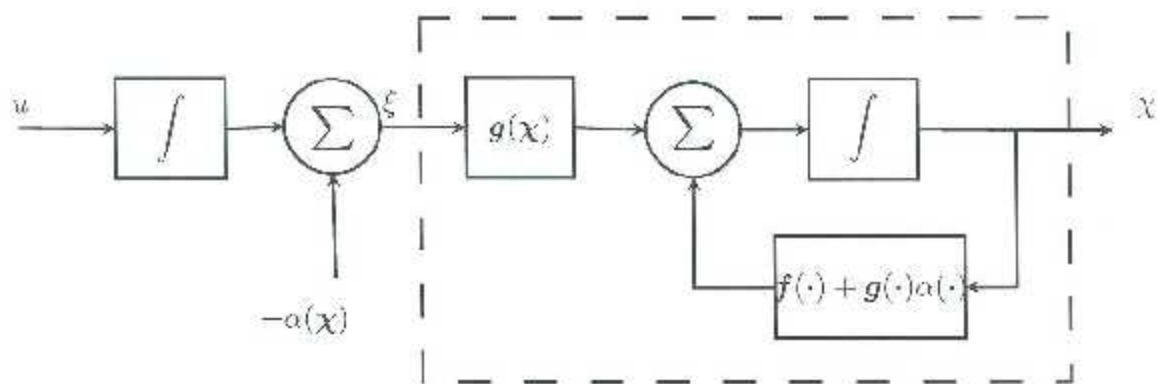
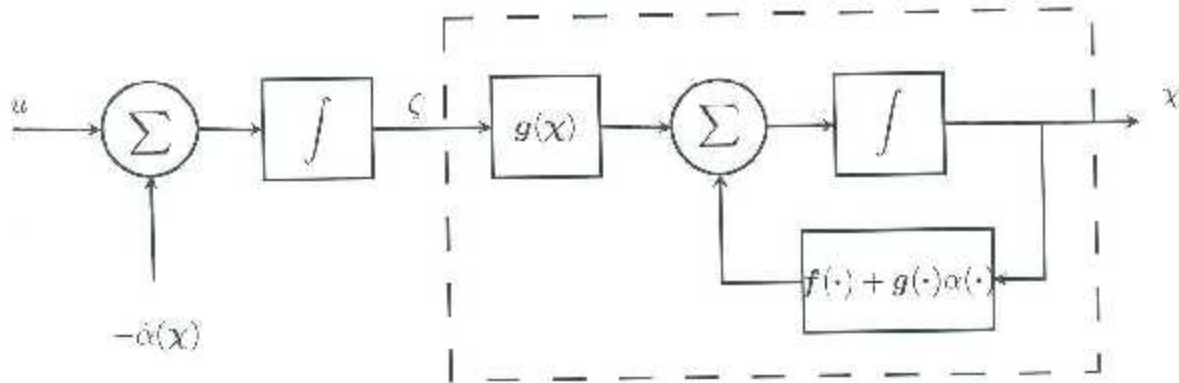


Figura 5.2: Diagrama a bloques del sistema (5.15)-(5.16) introduciendo  $\alpha(\mathbf{x})$ .


 Figura 5.3: *Backstepping* de  $-\alpha(x)$  a través del integrador.

El sistema (5.22)-(5.23) es similar al sistema (5.15)-(5.16), excepto que ahora la expresión (5.22) tiene un origen asintóticamente estable para cuando la entrada  $v$  es cero. Esta característica será explorada en el diseño de  $v$  para estabilizar completamente el sistema.

Proponiendo

$$V_c(x, \zeta) = V(x) + \frac{1}{2}\zeta^2 \quad (5.24)$$

como una función candidata de Lyapunov, derivando y evaluando a lo largo de las trayectorias (5.22)-(5.23) se obtiene

$$\dot{V}_c(x, \zeta) = \frac{\partial V}{\partial x} [f(x) + g(x)\alpha(x)] + \frac{\partial V}{\partial x} g(x)\zeta + \zeta v \leq -W(x) \left[ \frac{\partial V}{\partial x} \right] g(x)\zeta + \zeta v \quad (5.25)$$

Eligiendo

$$v = -\frac{\partial V}{\partial x} - k\zeta, \quad \text{para } k > 0, \quad (5.26)$$

conduce a

$$\dot{V}_c \leq -W(x) - k\zeta^2;$$

lo cual demuestra que el origen  $(x = 0, \zeta = 0)$  de (5.22)-(5.23) es asintóticamente estable. Además, ya que  $\alpha(0) = 0$  se concluye en (5.21) de (5.18) que el origen  $(x = 0, \xi = 0)$  de (5.15)-(5.16) es asintóticamente estable. Sustituyendo  $v$ ,  $\zeta$  y  $\alpha$  de (5.26), (5.18) y (5.20), respectivamente, se obtiene la ley de control por realimentación de estados

$$u = \frac{\partial \alpha}{\partial x} [f(x) + g(x)\xi] - \frac{\partial V}{\partial x} g(x) - k[\xi - \alpha(x)]. \quad (5.27)$$

Si todas las suposiciones se cumplen de manera global y  $V(x)$  es radialmente desacotada, se concluye que el origen es asintóticamente estable en forma global. Por todo lo anterior, se enuncia el siguiente lema:

**Lema 1.** Considérese el sistema (5.15)-(5.16). Sea  $\alpha(\chi)$  una ley de control estabilizadora por realimentación de estados para (5.15) con  $\alpha(0) = 0$  y  $V(\chi)$  una función de Lyapunov que satisface (5.24) para alguna función definida positiva  $W(\chi)$ . Luego, la ley de control por realimentación de estados (5.27) estabiliza el origen de (5.15)-(5.16) con  $V(\chi) + [\xi - \alpha(\chi)]^2/2$  como una función de Lyapunov. Además, si todas las suposiciones se cumplen en forma global y  $V(\chi)$  es radialmente desacotada el origen será asintóticamente estable en forma global.

## 5.4. Controlador neuronal RHONN

Partiendo de la arquitectura (5.10) para la RHONN, bajo la metodología *backstepping* se sintetiza el controlador neuronal donde las leyes adaptables para la actualización de los pesos están dadas por (5.14). Se busca diseñar una ley de control por realimentación  $u(t)$  para lograr que la salida del sistema siga una trayectoria deseada. El esquema de control neuronal descentralizado se basa en lo siguiente:

**Proposición 1.** [Félix et al., 2005b] Dados un vector de trayectorias deseadas  $\chi_d^i \in \mathbb{R}^n$ , un sistema dinámico con salida  $\chi^i \in \mathbb{R}^n$ , y una RHONN con salida  $x^i \in \mathbb{R}^a$ , entonces es posible establecer el error de seguimiento como:

$$|\chi^i - \chi_d^i| = |\chi^i - x^i| + |x^i - \chi_d^i|, \quad (5.28)$$

donde  $\chi^i$  y  $\chi_d^i$  es el vector de estado del subsistema y los estados deseados del subsistema, respectivamente;  $\chi^i - x^i$  es el error de identificación y  $x^i - \chi_d^i$  es el error de seguimiento de la salida de la RHONN.

Basándose en la proposición anterior es posible dividir el seguimiento del error en dos partes:

1. La minimización de  $|\chi^i - x^i|$ , la cual puede conseguirse por el algoritmo de identificación propuesto.
2. La minimización de  $|x^i - \chi_d^i|$ , la cual puede conseguirse por el algoritmo de control diseñado.

La minimización del error de seguimiento puede conseguirse mediante el diseño de un control basado en el modelo de la RHONN. En el diseño de la ley de control se propone utilizar la técnica de *backstepping* debido a que ofrece resolver problemas de estabilización, seguimiento y control robusto bajo condiciones menos restrictivas que las encontradas en otros métodos.

### 5.4.1. Diseño del controlador neuronal RHONN

Definiendo el error de identificación como  $\xi^i = \chi^i - x^i$  y el error de seguimiento de la salida de la RHONN como  $e^i = x^i - \chi_{2d}^i$  y definiendo el error de seguimiento de trayectoria como:

$$\tilde{\chi}^i = \xi^i + e^i, \quad (5.29)$$

Luego, la dinámica del error está dada por

$$\dot{\tilde{\chi}}^i = \dot{\xi}^i + \dot{e}^i. \quad (5.30)$$

De acuerdo al Teorema 2 [Kosmatopoulos et al., 1993] se propone la siguiente función de Lyapunov

$$V^i(\xi^i, \tilde{w}^i) = \frac{1}{2} \sum_{j=1}^n [(\xi_j^i)^2 + (\tilde{w}_j^i)^T (\gamma_j^i)^{-1} \tilde{w}_j^i]$$

luego derivando  $V^i(\xi^i, \tilde{w}^i)$  se tiene

$$\dot{V}^i(\xi^i, \tilde{w}^i) = \sum_{j=1}^n [\xi_j^i \dot{\xi}_j^i - (\tilde{w}_j^i)^T (\gamma_j^i)^{-1} \dot{\tilde{w}}_j^i].$$

De (4.21) y (4.23) se tiene

$$\begin{aligned} \dot{\xi}_j^i &= -a_j^i \xi_j^i - (\tilde{w}_j^i)^T z_j^i, \\ \dot{\tilde{w}}_j^i &= -\gamma_j^i z_j^i \xi_j^i, \end{aligned}$$

se llega a

$$\begin{aligned} \dot{V}^i(\xi^i, \tilde{w}^i) &= \sum_{j=1}^n (\xi_j^i (-a_j^i \xi_j^i + \tilde{w}_j^i z_j^i) + \tilde{w}_j^i (\gamma_j^i)^{-1} (-\gamma_j^i z_j^i \xi_j^i)) \\ &= \sum_{j=1}^n (-a_j^i (\xi_j^i)^2 + \tilde{w}_j^i z_j^i \xi_j^i - \tilde{w}_j^i z_j^i \xi_j^i) \\ &= \sum_{j=1}^n -a_j^i (\xi_j^i)^2 \leq 0. \end{aligned}$$

Considerando ahora el esquema neuronal (5.10) se procede con el diseño del controlador que estabiliza a  $e^i$ . Para esto, se expresa el mismo en función del error seguimiento de la RHONN, i.e.,

$$e^i = \begin{bmatrix} x_1^i - \chi_{2d}^i \\ x_2^i - \chi_{2d}^i \end{bmatrix} = \begin{bmatrix} e_1^i \\ e_2^i \end{bmatrix}, \quad (5.31)$$

así, la dinámica del error resulta

$$\begin{aligned}\dot{\tilde{e}}_1^i &= \dot{x}_1^i - \dot{\chi}_{1d}^i \\ \dot{\tilde{e}}_2^i &= \dot{z}_2^i - \dot{\chi}_{2d}^i\end{aligned}\quad (5.32)$$

luego de (5.10) y (5.10) se tiene

$$\dot{\tilde{e}}_1^i = -a_1^i \tilde{e}_1^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + \dot{e}_2^i \quad (5.33)$$

$$\dot{\tilde{e}}_2^i = -a_2^i \tilde{e}_2^i - a_2^i \chi_{2d}^i + (\mathbf{w}_2^i)^T \mathbf{z}_2^i + \dot{u}^i - \dot{\chi}_{2d}^i. \quad (5.34)$$

Del sistema escalar (5.33) se considera  $\dot{e}_2^i$  como entrada virtual y se procede a diseñar un control  $\dot{e}_2^i = \alpha_2^i$  para estabilizar el origen,  $\tilde{e}_1^i = 0$ .

Con

$$\dot{e}_2^i = \alpha_2^i = a_1^i \chi_{1d}^i - (\mathbf{w}_1^i)^T \mathbf{z}_1^i - c_1^i \tilde{e}_1^i \quad (5.35)$$

se cancelan los términos no lineales para obtener

$$\begin{aligned}\dot{\tilde{e}}_1^i &= -a_1^i \tilde{e}_1^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + \dot{e}_2^i \\ &= -a_1^i \tilde{e}_1^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + (a_1^i \chi_{1d}^i - (\mathbf{w}_1^i)^T \mathbf{z}_1^i - c_1^i \tilde{e}_1^i) \\ &= -a_1^i \tilde{e}_1^i - c_1^i \tilde{e}_1^i.\end{aligned}$$

Proponiendo ahora la función aumentada de Lyapunov  $V_1^i(\tilde{e}_1^i, \xi^i, \tilde{\mathbf{w}}^i) = V^i(\xi^i, \tilde{\mathbf{w}}^i) + \frac{1}{2}(\tilde{e}_1^i)^2$ , luego, la derivada con respecto al tiempo  $\dot{V}_1^i(\cdot)$  de  $V_1^i(\cdot)$  satisface

$$\begin{aligned}\dot{V}_1^i(\tilde{e}_1^i, \xi^i, \tilde{\mathbf{w}}^i) &= \dot{V}^i(\xi^i, \tilde{\mathbf{w}}^i) + \dot{\tilde{e}}_1^i \tilde{e}_1^i \\ &= \dot{V}^i(\xi^i, \tilde{\mathbf{w}}^i) + \tilde{e}_1^i (-a_1^i \tilde{e}_1^i - c_1^i \tilde{e}_1^i) \\ &= \dot{V}^i(\xi^i, \tilde{\mathbf{w}}^i) - (a_1^i + c_1^i)(\tilde{e}_1^i)^2, \quad \forall \tilde{e}_1^i, \xi^i, \tilde{\mathbf{w}}^i \in \mathbb{R}.\end{aligned}$$

Por lo tanto, el origen de  $\dot{\tilde{e}}_1^i = -a_1^i \tilde{e}_1^i - c_1^i \tilde{e}_1^i$  es globalmente exponencialmente estable. Para aplicar el *backstep* (retroceso) se usa el cambio de variable

$$\zeta_2^i = \tilde{e}_2^i - \alpha_2^i = \tilde{e}_2^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + c_1^i \tilde{e}_1^i. \quad (5.36)$$

Despejando  $\tilde{e}_2^i$  de (5.36) resulta

$$\tilde{e}_2^i = \zeta_2^i + \alpha_2^i = \zeta_2^i + a_1^i \chi_{1d}^i - (\mathbf{w}_1^i)^T \mathbf{z}_1^i - c_1^i \tilde{e}_1^i.$$

Sustituyendo la identidad anterior en (5.33) se tiene

$$\begin{aligned}\dot{\tilde{e}}_1^i &= -a_1^i \tilde{e}_1^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + \tilde{e}_2^i \\ &= -a_1^i \tilde{e}_1^i - a_1^i \chi_{1d}^i + (\mathbf{w}_1^i)^T \mathbf{z}_1^i + (\zeta_2^i + a_1^i \chi_{1d}^i - (\mathbf{w}_1^i)^T \mathbf{z}_1^i - c_1^i \tilde{e}_1^i),\end{aligned}$$

lo que conduce a

$$\dot{\tilde{e}}_1^i = -a_1^i \tilde{e}_1^i - c_1^i \tilde{e}_1^i + \zeta_2^i. \quad (5.37)$$

Para continuar con el diseño del controlador es necesario obtener las derivadas temporales de (5.35) y (5.36) como se muestra enseguida

$$\begin{aligned}\dot{\alpha}_2^i &= a_1^i \lambda_{2a}^i - \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i - c_1^i \epsilon_1^i, \\ &= a_1^i \lambda_{2a}^i - \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i - c_1^i (-a_2^i c_1^i - c_1^i \epsilon_1^i + \zeta_2^i),\end{aligned}\quad (5.38)$$

$$\begin{aligned}\dot{\zeta}_2^i &= \dot{\epsilon}_2^i - \dot{\alpha}_2^i, \\ &= -a_2^i c_2^i - a_2^i \lambda_{2a}^i + (\mathbf{w}_2^i)^T \mathbf{z}_2^i + u^i - \dot{\lambda}_{2a}^i - a_1^i \lambda_{2a}^i \\ &\quad + \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i + c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i).\end{aligned}\quad (5.39)$$

De (5.37) y (5.39) el sistema toma la forma

$$\dot{\epsilon}_1^i = -a_1^i c_1^i - c_1^i \epsilon_1^i - \zeta_2^i. \quad (5.40)$$

$$\begin{aligned}\dot{\zeta}_2^i &= -a_2^i c_2^i - a_2^i \lambda_{2a}^i - a_2^i \lambda_{2a}^i + (\mathbf{w}_2^i)^T \mathbf{z}_2^i + u^i - \dot{\lambda}_{2a}^i - a_1^i \lambda_{2a}^i \\ &\quad + \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i + c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i).\end{aligned}\quad (5.41)$$

Considerando

$$V_2^i(\boldsymbol{\xi}^i, \bar{\mathbf{w}}^i, c_1^i, \zeta_2^i) = \dot{V}^i(\boldsymbol{\xi}^i, \bar{\mathbf{w}}^i) + \frac{1}{2}(\epsilon_1^i)^2 + \frac{1}{2}(\zeta_2^i)^2,$$

como una función de Lyapunov aumentada se tiene

$$\begin{aligned}\dot{V}_2^i(\cdot) &= \dot{V}^i(\boldsymbol{\xi}^i, \bar{\mathbf{w}}^i) + c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i) + \zeta_2^i \left[ -a_2^i c_2^i - a_2^i \lambda_{2a}^i - a_2^i \lambda_{2a}^i + (\mathbf{w}_2^i)^T \mathbf{z}_2^i \right. \\ &\quad \left. + u^i - \dot{\lambda}_{2a}^i - a_1^i \lambda_{2a}^i + \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i + c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i) \right] \\ &= \dot{V}^i(\boldsymbol{\xi}^i, \bar{\mathbf{w}}^i) - a_1^i (c_1^i)^2 - c_1^i (\epsilon_1^i)^2 - a_2^i (\zeta_2^i)^2 - \zeta_2^i \left[ \epsilon_1^i - a_2^i c_2^i - a_2^i \lambda_{2a}^i + (\mathbf{w}_2^i)^T \mathbf{z}_2^i \right. \\ &\quad \left. + u^i - \dot{\lambda}_{2a}^i - a_1^i \lambda_{2a}^i + \frac{d(\mathbf{w}_1^i)^T}{dt} \mathbf{z}_1^i + \frac{d(\mathbf{z}_1^i)^T}{dt} \mathbf{w}_1^i + c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i) \right].\end{aligned}$$

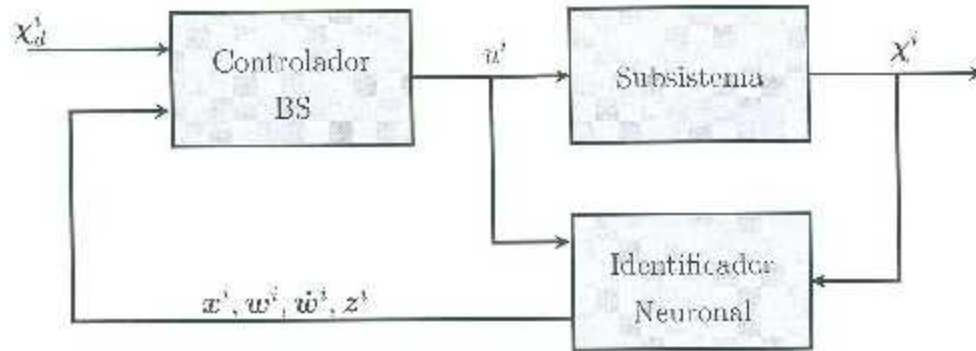


Figura 5.4: Diagrama del control RHONN descentralizado propuesto.



Luego, y eligiendo la entrada de control como

$$\begin{aligned} w^i = & -\epsilon^i + a_2^i \alpha_2^i + a_2^i \chi_{2d} - (w_2^i)^T z_2^i + \dot{\chi}_{2d} + a_1^i \chi_{2d} \\ & - \frac{d(w_1^i)^T}{dt} z_1^i + \frac{d(z_1^i)^T}{dt} w_1^i - c_1^i (-a_1^i \epsilon_1^i - c_1^i \epsilon_1^i + \zeta_2^i) - c_2^i \zeta_2^i \end{aligned} \quad (5.42)$$

produce

$$\dot{V}_2^i(\cdot) - \dot{V}^i(\xi, \bar{w}^i) - (a_1^i + c_1^i)(\epsilon_1^i)^2 - (a_2^i + c_2^i)(\zeta_2^i)^2 \leq 0.$$

Por lo anterior, el origen es globalmente asintóticamente estable.

En la Figura 5.4 se muestra el diagrama de control neuronal propuesto por subsistemas.

## 5.5. Control de movimiento traslacional (Cinemática inversa)

A diferencia de la orientación y altitud ( $z$ ) del *quadrotor*, el movimiento traslacional a lo largo de las coordenadas  $x$  y  $y$  no puede ser directamente controlado mediante alguna de las cuatro entradas de control ( $u_1, u_2, u_3$  o  $u_4$ ). Por otra parte, los desplazamientos a lo largo de las coordenadas  $x$  y  $y$  se pueden controlar a través de los ángulos de alabeo (*roll*) y cabeceo (*pitch*). Los ángulos  $\phi_d$  y  $\theta_d$  para alabeo y cabeceo, respectivamente, pueden ser calculados de las ecuaciones de movimiento traslacional (5.4)-(5.5), como sigue [Mahony et al., 2012]:

$$\begin{aligned} \dot{x}_{1d} &= (\sin x_{1d} \sin x_5 + \cos x_{1d} \sin x_{3d} \cos x_5) \frac{u_1}{m}, \\ \dot{x}_{12} &= (-\sin x_{1d} \cos x_5 + \cos x_{1d} \sin x_{3d} \sin x_5) \frac{u_1}{m}. \end{aligned} \quad (5.43)$$

De la consideración del vuelo en sustentación (*hover*) para el *quadrotor*, empleando el recurso de linealización de ángulos pequeños luego se tiene  $\sin \phi_d \equiv \phi_d$ ,  $\sin \theta_d = \theta_d$  y  $\cos \phi_d \equiv \cos \theta_d \equiv 1$ , lo que da lugar a

$$\begin{aligned} \dot{x}_{1d} &= (x_{1d} \sin x_5 + x_{3d} \cos x_5) \frac{u_1}{m}, \\ \dot{x}_{12} &= (-x_{1d} \cos x_5 + x_{3d} \sin x_5) \frac{u_1}{m}; \end{aligned}$$

la cual puede ser reescrita en forma vectorial como:

$$\begin{bmatrix} \sin x_5 & \cos x_5 \\ -\cos x_5 & \sin x_5 \end{bmatrix} \begin{bmatrix} x_{1d} \\ x_{3d} \end{bmatrix} = \frac{m}{u_1} \begin{bmatrix} \dot{x}_{1d} \\ \dot{x}_{12d} \end{bmatrix},$$

de donde multiplicando por la izquierda por

$$\begin{bmatrix} \sin x_5 & -\cos x_5 \\ \cos x_5 & \sin x_5 \end{bmatrix}$$

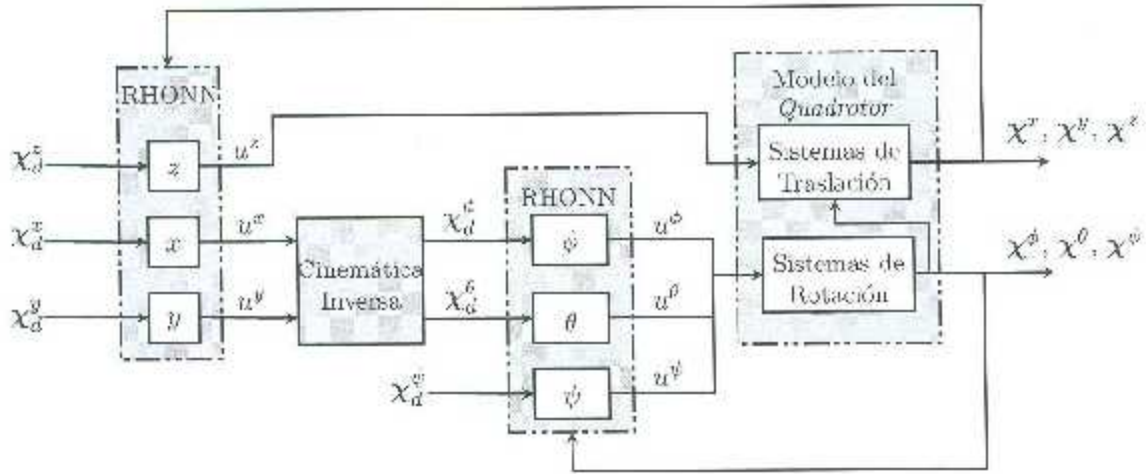


Figura 5.5: Diagrama del controlador RHONN descentralizado propuesto.

resulta

$$\begin{bmatrix} \dot{x}_{1d} \\ \dot{x}_{3d} \end{bmatrix} = \frac{m}{u_1} \begin{bmatrix} \sin x_5 & -\cos x_5 \\ \cos x_5 & \sin x_5 \end{bmatrix} \begin{bmatrix} \dot{x}_{10d} \\ \dot{x}_{12d} \end{bmatrix}. \quad (5.44)$$

Los valores de  $\phi_d$  y  $\theta_d$  tienen que ser limitados a un rango entre  $\pm 20^\circ$  para satisfacer el criterio de ángulos pequeños, lo que se puede establecer mediante una función saturación. Así, la cinemática inversa (5.44) se utiliza para conseguir el control del *quadrotor* a lo largo de las coordenadas traslacionales  $x$  y  $y$ .

En la Figura 5.5 se muestra el diagrama completo del controlador RHONN descentralizado propuesto.

## 5.6. Resultados de simulación

En la presente sección se discuten los resultados en simulación obtenidos de la identificación y control simultánea del *quadrotor* vía RHONNs usando el esquema neuronal (5.10), las leyes de actualización de pesos (5.11) y la ley de control (5.42). En la Tabla 5.1 se exhiben los parámetros proporcionados por Quanser para el *quadrotor* QBall-2 [User Manual QBall 2, 2014], los cuales son utilizados en la simulación con la finalidad de considerar un modelo más aproximado al de la plataforma experimental.

### 5.6.1. Resultados de identificación

En las Figuras 5.6 a la 5.11 se muestran el desempeño en simulación de la identificación neuronal de la dinámica de los subsistemas (5.3)-(5.8), a través del error de identificación. Las condiciones iniciales consideradas para la trayectoria a identificar son  $\phi_0 = 0$ ,  $\theta_0 = 0$ ,

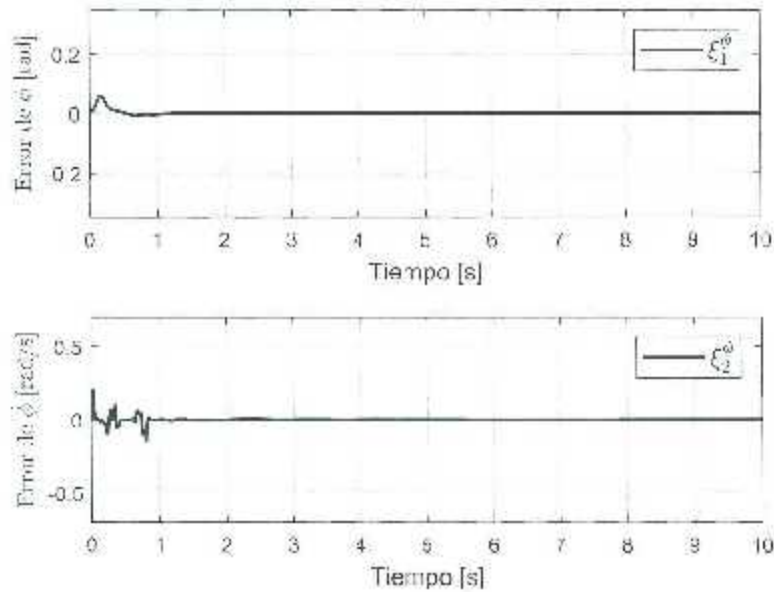


Figura 5.6: Error de identificación RHONN descentralizado de las dinámicas de alabeo.

$\psi_0 = 0$ ,  $\dot{\psi}_0 = 0$ ,  $x_0 = 0$  y  $y_0 = 0$ , las condiciones iniciales de los estados de las NN son  $x_j^i = 0$  para todo  $i = 1, 2, 3, 4, 5, 6$  y  $j = 1, 2$ . Se utilizaron las mismas ganancias  $\gamma$  para todos los subsistemas, las cuales fueron establecidas como  $\gamma_{11}^i = \gamma_{12}^i = \gamma_{21}^i = \gamma_{22}^i = 10000000$ . Las constantes fueron definidas como  $a_1^i = a_2^i = 100$  y  $b_1^i = b_2^i = 1$ .

En las Figuras 5.6-Figura 5.11 se exhiben los errores de identificación de posición y velocidad para los subsistemas de  $S_\psi$ ,  $S_\theta$ ,  $S_\phi$ ,  $S_z$ ,  $S_x$  y  $S_y$ , respectivamente.

Tabla 5.1: Parámetros de la plataforma Qball-2

| Parámetro | Valor                  |
|-----------|------------------------|
| $K$       | 120 N                  |
| $\omega$  | 15 rad/s               |
| $J_z$     | 0.03 Kg m <sup>2</sup> |
| $J_y$     | 0.03 Kg m <sup>2</sup> |
| $J_x$     | 0.04 Kg m <sup>2</sup> |
| $m$       | 1.79 Kg                |
| $K_v$     | 4 N·m                  |
| $l$       | 0.2 m                  |

Tabla 5.2: MSEs para la identificación vía RHONNs

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.003892 | 0.003905 | 0.008178 | 0.003341 | 0.003908 | 0.003905 |

### 5.6.2. Resultados de control

En las Figuras 5.14 a la 5.19 se muestran los errores de seguimiento y los pares aplicados a cada subsistema. En la Figura 7.18 se muestra la trayectoria descrita por el *quadrotor* y la trayectoria de referencia. Las señales de referencia para el control del *quadrotor* son  $\chi_{1d}^x = 1$  [m],  $\chi_{1d}^y = 0.1745 \sin(t/5)$  [rad],  $\chi_{1d}^z = Am \cos(ft)$  [m] y  $\chi_{1d}^\psi = Am \sin(ft)$  [m], con  $Am = 0.5$  [m] y  $f = \pi/12.5$  [rad/s]. La señal de referencia para el subsistema de guiñada se eligió como una sinusoidal de  $10^\circ$  de amplitud, esto con la finalidad de mostrar que puede trabajar con una referencia diferente de cero.

En las Figuras 5.14-5.19 se exhiben los errores de seguimiento de trayectoria y la señal de control para los subsistemas de  $S_\phi$ ,  $S_\theta$ ,  $S_z$ ,  $S_\psi$ ,  $S_x$  y  $S_y$ , respectivamente, en éstas se puede observar que a pesar de la desviación considerable al inicio de la descripción de la trayectoria circular para cuando alcanza la altura de referencia, el error de seguimiento es aceptable.

Para evaluar el desempeño del esquema RHONN se consideró el error cuadrático medio (Mean Square Error, MSE por sus siglas en inglés). La Tabla 5.2 muestra los MSEs de la identificación neuronal para cada uno de los subsistemas mientras que la Tabla 5.3 muestra los MSEs del seguimiento de trayectoria de cada subsistema.

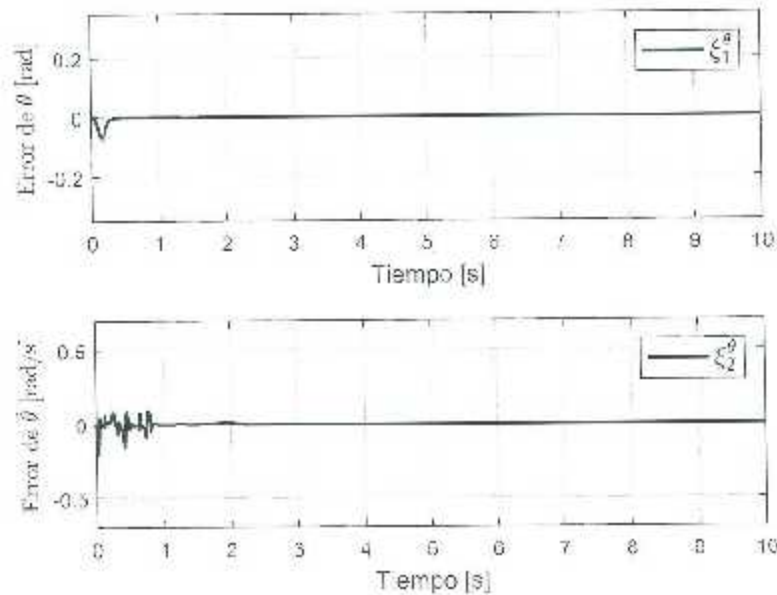


Figura 5.7: Error de identificación RHONN descentralizado de las cinámicas de cabeceo.

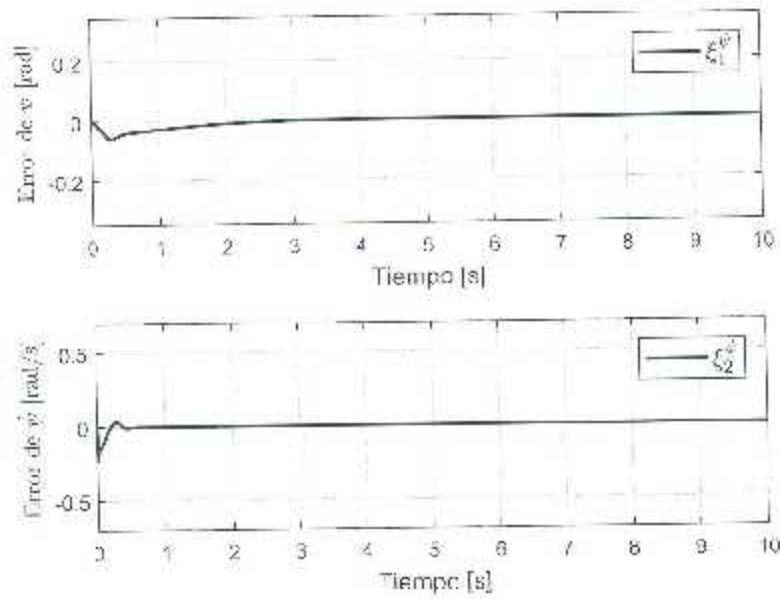


Figura 5.8: Error de identificación RHONN descentralizado de las dinámicas de guiñada.

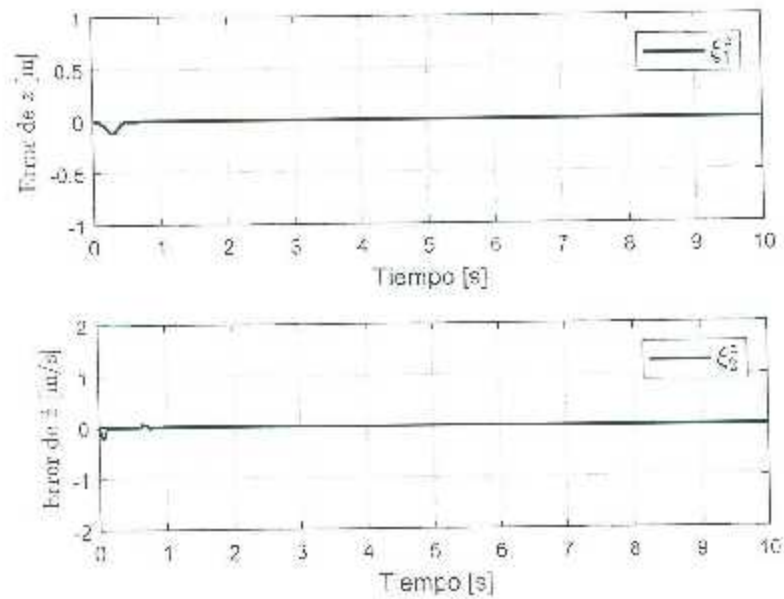


Figura 5.9: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $z$ .

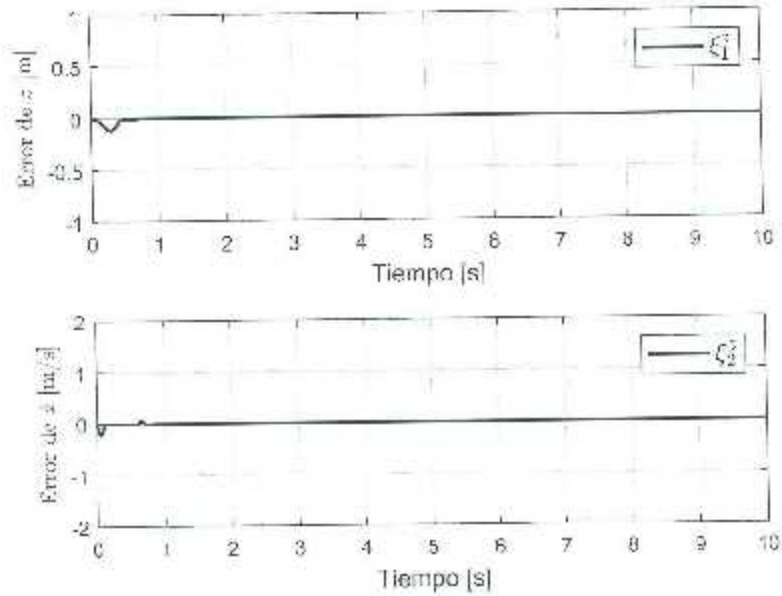


Figura 5.10: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $x$ .

## 5.7. Resultados experimentales para la identificación vía RHONNs

En esta sección se muestran los resultados de la identificación experimental del *quadrotor* vía RHONN. Esta identificación se llevó a cabo en el laboratorio de vehículos móviles

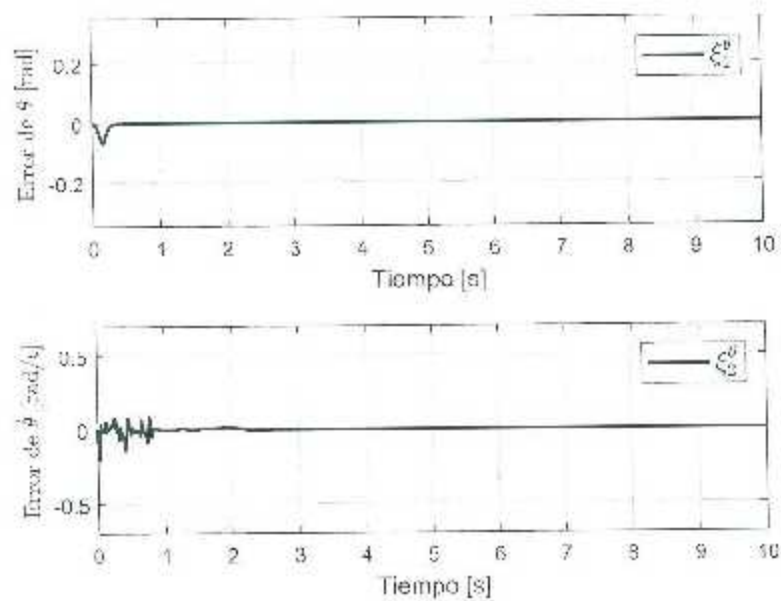


Figura 5.11: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $y$ .

Tabla 5.3: MSEs para el seguimiento

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.004629 | 0.006181 | 0.007953 | 0.111574 | 0.068455 | 0.022092 |

en el Q-ball 2 de Quanser.

En las Figuras 5.20 a la 5.25 se muestran el desempeño en la plataforma experimental de la identificación de la dinámica de los subsistemas (5.3)-(5.8), a través de sus errores de identificación. Las condiciones iniciales consideradas para la trayectoria a identificar son  $\phi_0 \approx 0$ ,  $\theta_0 \approx 0$ ,  $\psi_0 \approx 0$ ,  $z_0 \approx 0.29$ ,  $x_0 \approx 0$  y  $y_0 \approx 0$ , las condiciones iniciales de los estados de las RHONNs son  $x_j^i = 0$  para todo  $i = \phi, \theta, \psi, z, x, y$  y  $j = 1, 2$ . Se utilizaron las mismas ganancias  $\gamma$  para los subsistemas angulares, i.e.,  $\gamma_{11}^i = \gamma_{12}^i = \gamma_{21}^i = \gamma_{22}^i = 100000$  para  $i = \phi, \theta, \psi$ , y las mismas ganancias  $\gamma$  para los subsistemas traslacionales i.e.,  $\gamma_{11}^i = \gamma_{12}^i = \gamma_{21}^i = \gamma_{22}^i = 12000$  para  $i = z, x, y$ . Las constantes  $a_j^i$  son elegidas como  $a_1^i = a_2^i = 100$  así como las constantes  $b_j^i$  como  $b_1^i = b_2^i = 1$ , estas constantes se utilizaron para todos los subsistemas. Cabe destacar que en las pruebas experimentales el *quadrotor* empieza a emprender el vuelo alrededor de los cinco segundos, dependiendo de la prueba. Como advertencia, en la identificación de las dinámicas rotacionales requiere ganancias  $\gamma$  más grandes en comparación con las ganancias establecidas para la identificación de las dinámicas traslacionales.

En las Figuras 5.20-5.25 se muestra los errores de identificación de las dinámicas para los subsistemas  $S_\phi$ ,  $S_\theta$ ,  $S_\psi$ ,  $S_z$ ,  $S_x$  y  $S_y$ , respectivamente, en la cual se puede ver como al identificador le cuesta un poco más de tiempo llegar a un error aceptable.

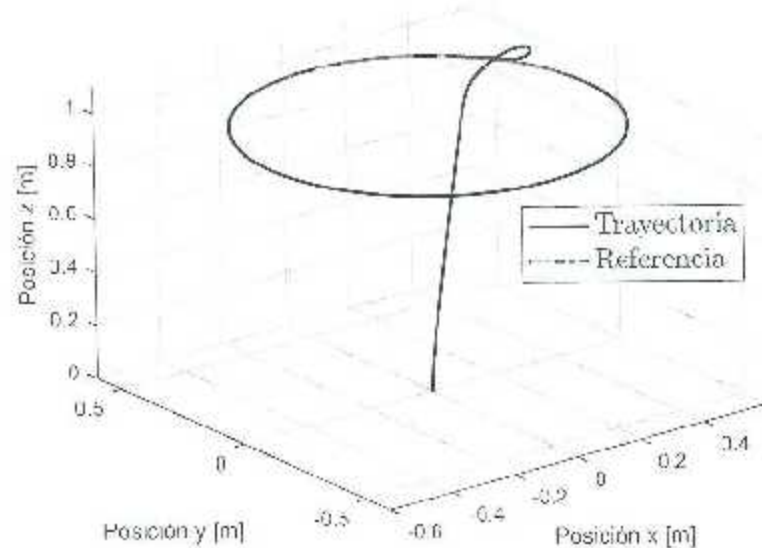


Figura 5.12: Desempeño del esquema de control neuronal descentralizado propuesto.

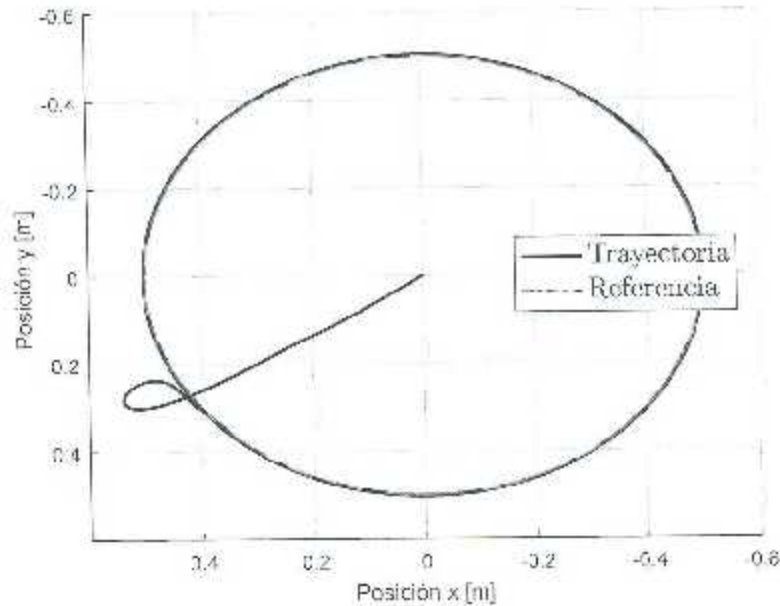


Figura 5.13: Vista superior del desempeño del esquema de control neuronal descentralizado propuesto.

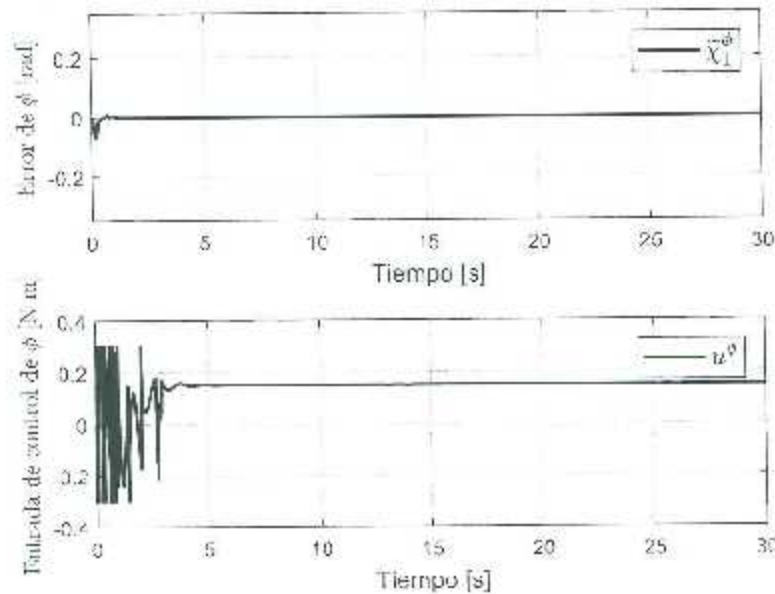


Figura 5.14: Error de seguimiento y señal de control para la dinámica rotacional de alabeo.

La Tabla 5.4 exhibe los MSEs de la identificación neuronal de cada uno de los subsistemas al ejecutar una trayectoria circular con un tiempo de vuelo de 35 segundos. Estos resultados difieren de los obtenidos en simulación por el hecho de que en los resultados experimentales solo se realiza identificación, y de esta manera el controlador no está in-



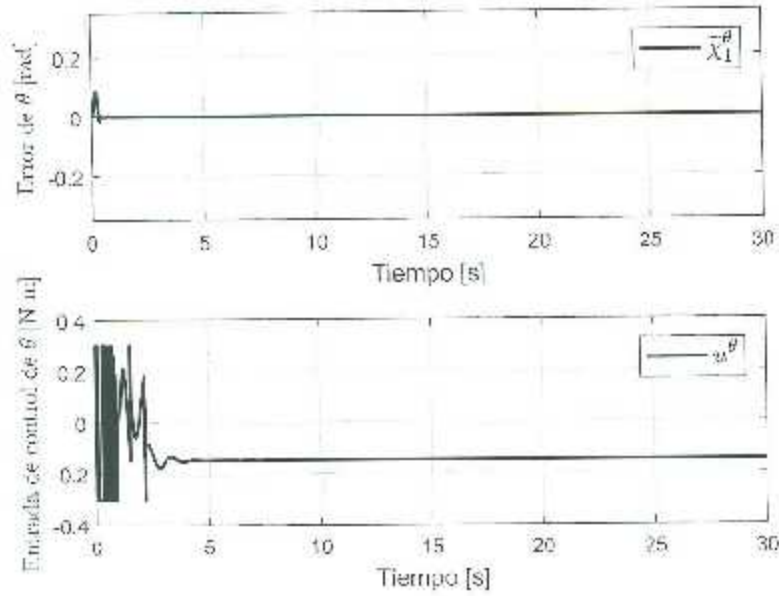


Figura 5.15: Error de seguimiento y señal de control para la dinámica rotacional de cabeceo fuenciado por el identificador.

Tabla 5.4: MSEs de la identificación experimental vía RHONNs

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.003680 | 0.006215 | 0.003659 | 0.010686 | 0.010956 | 0.011103 |

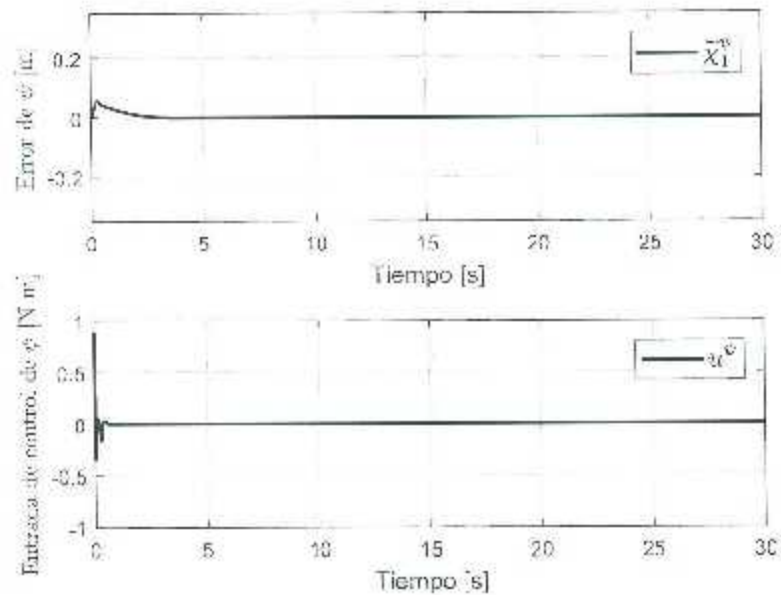


Figura 5.16: Error de seguimiento y señal de control para la dinámica rotacional de guiñada.

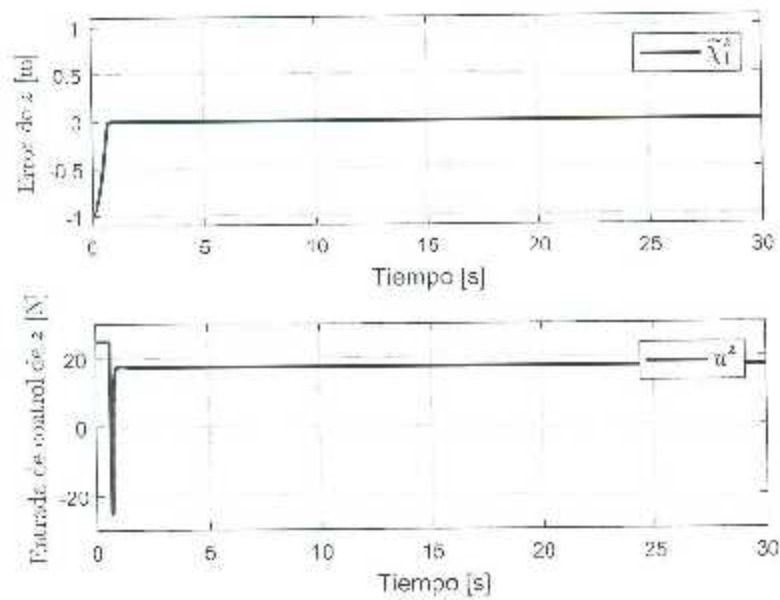


Figura 5.17: Error de seguimiento y señal de control para la dinámica translacional de  $z$ .

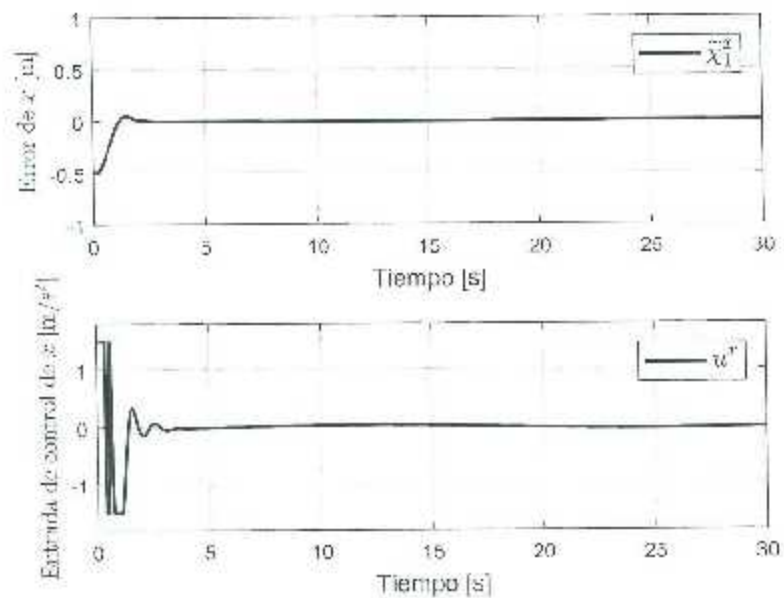


Figura 5.18: Error de seguimiento y señal de control para la dinámica translacional de  $x$ .

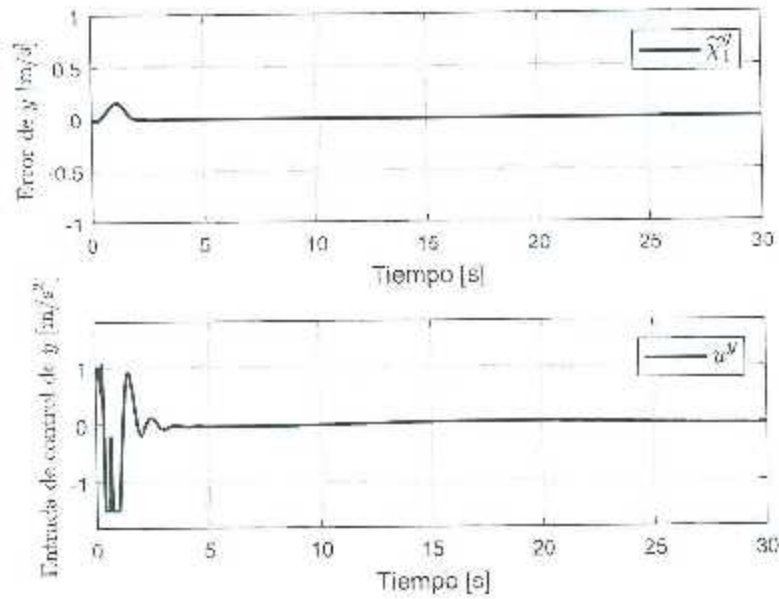


Figura 5.19: Error de seguimiento y señal de control para la dinámica translacional de  $y$ .

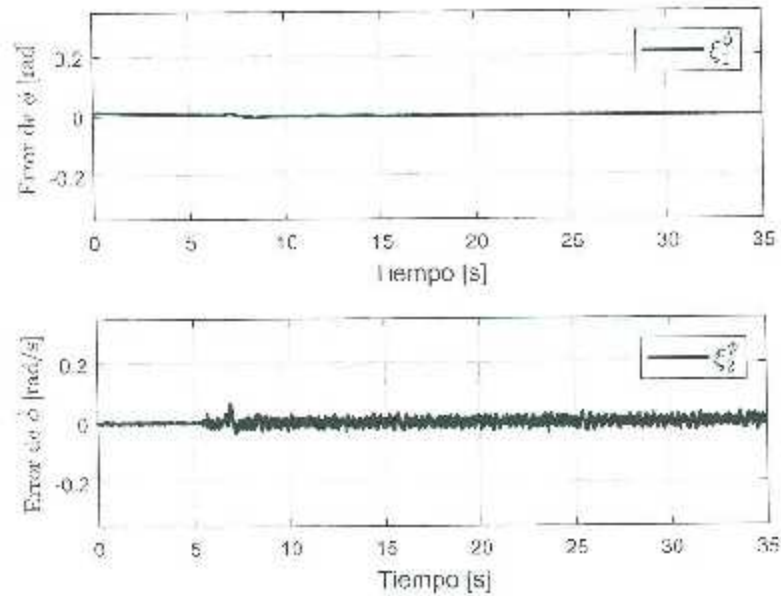


Figura 5.20: Error de identificación RHONN descentralizado de las dinámicas de alabeo.

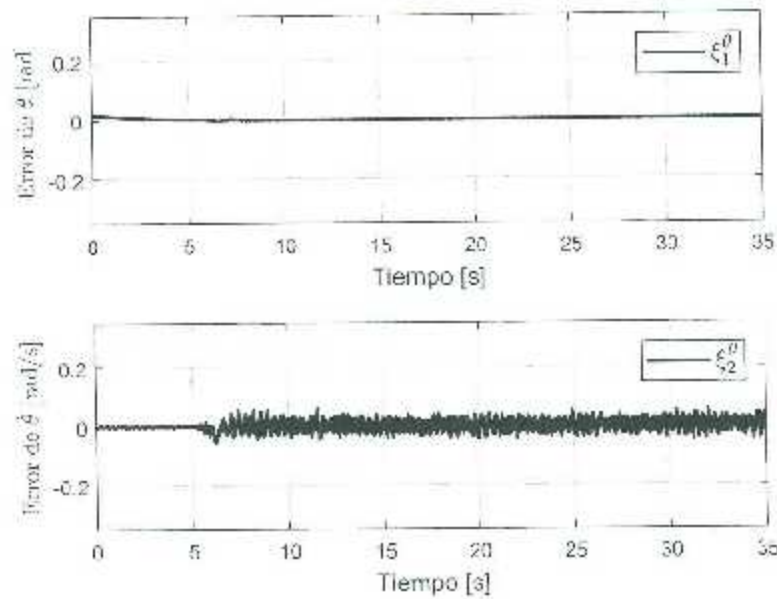


Figura 5.21: Error de identificación RHONN descentralizado de las dinámicas de cabeceo.

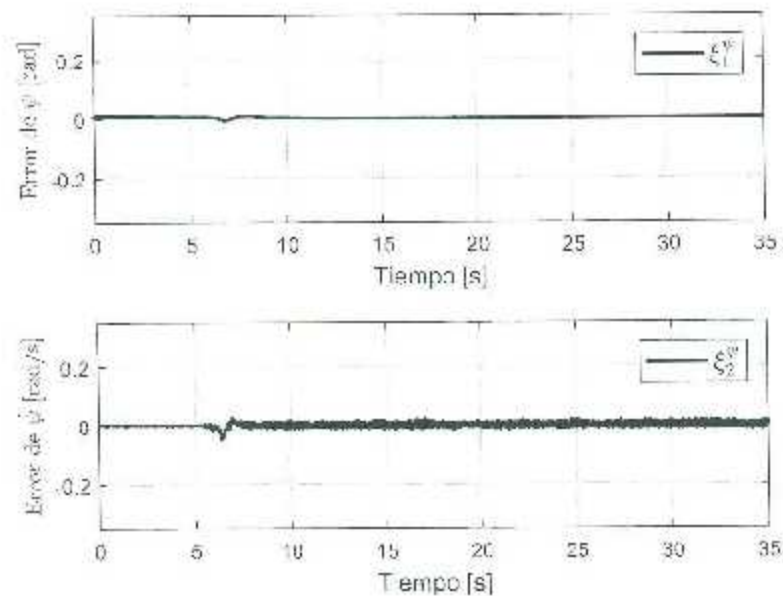


Figura 5.22: Error de identificación RHONN descentralizado de las dinámicas de guiñada.

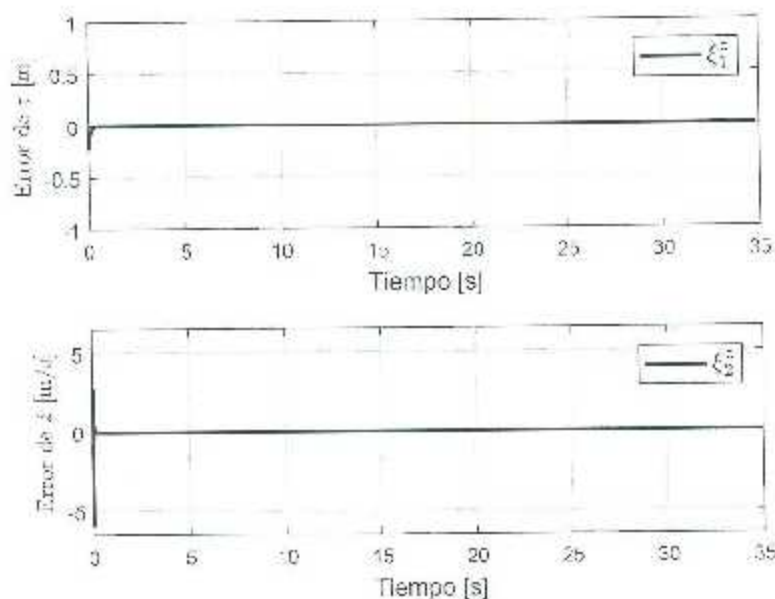


Figura 5.23: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $z$ .

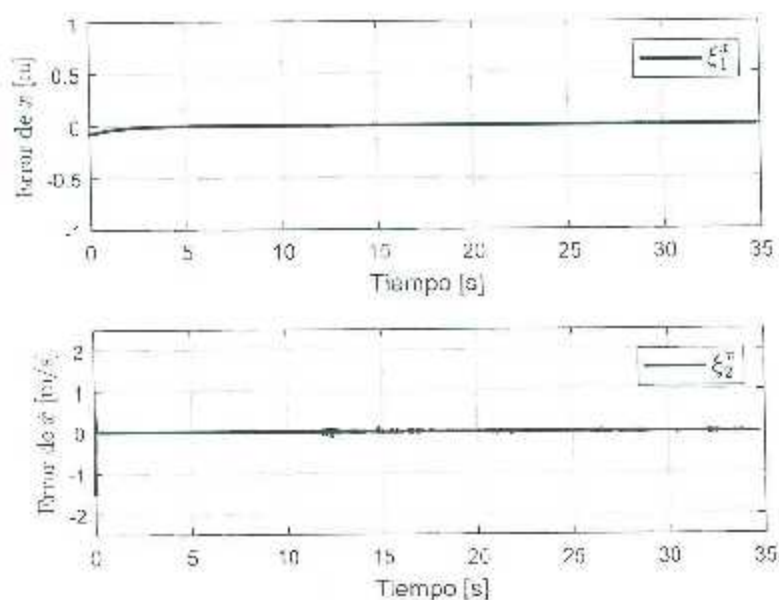


Figura 5.24: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $x$ .

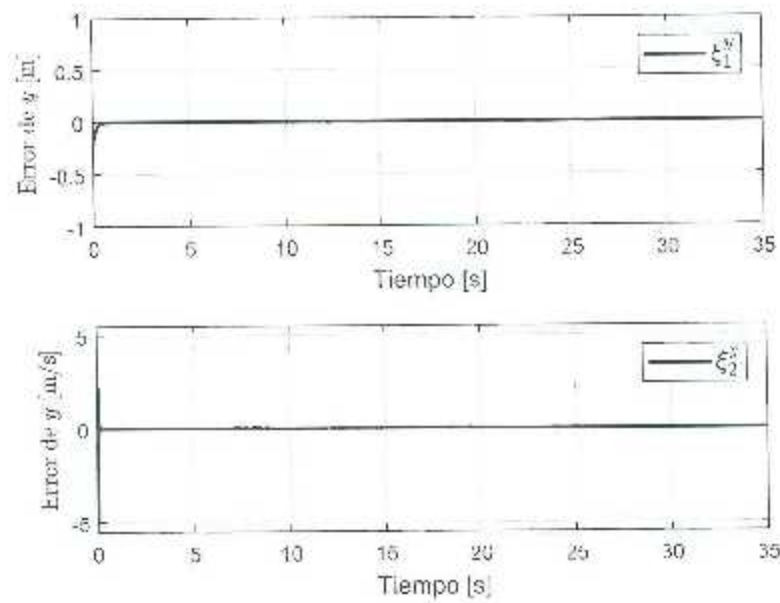


Figura 5.25: Error de identificación RHONN descentralizado de las dinámicas traslacionales de  $y$ .

# Capítulo 6

## Redes neuronales wavelet

En este capítulo se resume lo referente a la teoría de las funciones *wavelet* y sus propiedades. Además, se analiza lo referente a la estructura de las redes neuronales *wavelet* recurrentes de primer orden (*Recurrent Wavelet First Order Neural Network*, RWFONN por sus siglas en inglés) así como sus propiedades de aproximación.

### 6.1. Introducción

Las redes neuronales *wavelet* (*Wavelet Neural Networks*, WNNs por sus siglas en inglés) son una estructura que combina el concepto de funciones *wavelet* con el recurso de ANNs con el objetivo de mejorar el desempeño en la identificación [Vázquez and Jurado, 2014]- [Wai and Chang, 2002]. A diferencia de las funciones de activación sigmoidea utilizadas en las ANNs convencionales, en las WNNs se usan funciones de activación *wavelet*. Las WNNs han demostrado ser mejores que las ANN en el sentido de que su estructura puede proporcionar más potencial para analizar el mapeo en las relaciones existentes entre entradas y salidas de un sistema dinámico [Vázquez and Jurado, 2014]. Las WNNs poseen al mismo tiempo las ventajas de una ANN y su capacidad de aprendizaje sumada a la habilidad de una función *wavelet* para descomponer una señal. Los sistemas de control basados en WNNs, han sido adoptados ampliamente para el control de sistemas dinámicos complejos debido a sus propiedades de aprendizaje rápido y su capacidad de convergencia [Lio et al., 2001], [Wei and Chang, 2004]. Por otra parte, las redes neuronales recurrentes *wavelet* (*Recurrent Wavelet Neural Network*, RWNNs por sus siglas en inglés), las cuales combinan las propiedades de velocidad de respuesta de las RNN y la velocidad de convergencia de las WNN, han sido propuestas para identificar y controlar el comportamiento dinámico de diferentes sistemas no lineales con muy buenos resultados [Yoo et al., 2006]- [El-Sousy, 2013].

El uso de las WNNs y RWNNs no es un tema nuevo en la literatura, sin embargo, la aplicación de estas estructuras implica el uso de múltiples capas con múltiples neuronas.

## 6.2. Fundamentos de *wavelets*

A continuación se presentan los fundamentos teóricos para el desarrollo de las WNNs, las características de las funciones *wavelet* y sus principales propiedades.

### 6.2.1. Funciones *wavelet*

El análisis *wavelet* consiste en la descomposición de una señal en un conjunto jerárquico de aproximaciones y características (rasgos) que permiten representar casi fielmente a una determinada señal original. En cada nivel de la jerarquía se obtiene una señal que aproxima y otra que representa los rasgos de dicha función para lo cual se toman en cuenta las frecuencias bajas que permiten aproximar a la señal original mientras que la señal que muestra los rasgos corresponde a las componentes de alta frecuencia. Las *wavelets* son funciones que permiten descomponer una señal en distintas componentes de frecuencia y después analizar cada una en una resolución acorde a su escala. La teoría de las funciones *wavelets* fue desarrollada en distintos campos del conocimiento como matemáticas, física e ingeniería eléctrica.

Para que una función  $\psi(\cdot)$  definida sobre el eje real  $(-\infty, \infty)$  sea considerada una función *wavelet* deberá satisfacer las siguientes propiedades [Daubechies, 1992]:

1. La integral de  $\psi(\cdot)$  de  $(-\infty, \infty)$  es cero.
2. La integral de la raíz cuadrada de  $\psi(\cdot)$  de  $(-\infty, \infty)$  es la unidad.
3. La condición de admisibilidad deberá mantenerse, i.e.,

$$C_\psi \equiv \int_0^\infty \frac{|\psi(f)|^2}{f} \text{ satisfacc } 0 < C_\psi < \infty. \quad (6.1)$$

Si las tres condiciones anteriores son satisfechas, entonces la señal a ser analizada puede ser reconstruida a partir de su transformada *wavelet* continua.

### 6.2.2. Transformada *wavelet* continua

La transformada *wavelet* continua (*Continuous Wavelet Transform*, CWT por sus siglas en inglés) [Daubechies, 1992] es utilizada para hacer la transformación de una señal  $x(\cdot)$  definida en tiempo continuo; por lo tanto, los parámetros  $\lambda$  y  $t$  en (6.2) se utilizan para crear una familia *wavelet* que varíe en el dominio del tiempo continuo. La intención de la transformada es que, dada una dilatación  $\lambda$  y una traslación  $t$  sobre la función *wavelet*  $\psi$ , se calcule el coeficiente de amplitud que hace de  $\phi_{\lambda,t}$  la mejor aproximación sobre la señal  $x(\cdot)$ , i.e., integrar el producto de la señal con la función *wavelet*:

$$\langle x, \psi_{\lambda,t} \rangle = \int_{-\infty}^{\infty} \psi_{\lambda,t}(u)x(u)du. \quad (6.2)$$



Variando el coeficiente  $\lambda$  la función *wavelet* aproxima de mejor manera a la señal a analizar, *i.e.*, de un valor de la dilatación a otro la función puede ser reconstruida. Realizando la variación de  $t$  la naturaleza de la señal se puede desplazar en el eje del tiempo. La colección de coeficientes  $\{\langle x, \psi_{\lambda t} \rangle \mid \lambda > 0, -\infty < t < \infty\}$  es llamada CWT de  $x(\cdot)$ . Un hecho fundamental sobre la CWT es que conserva toda la información de la señal original  $x(\cdot)$ . Si la función *wavelet*  $\psi(\cdot)$  satisface (6.1) y además si

$$\int_{-\infty}^{\infty} x^2(t) dt < \infty$$

es satisfecha, entonces la señal  $x(\cdot)$  puede ser reconstruida a partir de su CWT aplicando la transformada inversa

$$x(t) = \frac{1}{C_{\psi}} \int_0^{\infty} \left[ \int_{-\infty}^{\infty} \langle x, \psi_{\lambda u} \rangle \psi_{\lambda u}(u) du \right] \frac{d\lambda}{\lambda^2},$$

por lo tanto  $x(\cdot)$  y su CWT son dos representaciones de la misma entidad. La CWT presenta a  $x(\cdot)$  en una nueva forma que permite obtener más información que de otra manera no se podría conocer con facilidad.

### 6.2.3. Análisis multiresolución

El análisis multiresolución (*Multiresolution Analysis*, MRA por sus siglas en inglés) muestra como una base *wavelet* ortonormal puede usarse como una herramienta para describir matemáticamente el incremento de la información necesaria para pasar de una aproximación básica a una aproximación de alta resolución.

**Definición 1.** [Mallat, 1989] Un MRA es una secuencia  $\nu_l$  de subespacios de  $\mathcal{L}^2(\mathbb{R})$  que implica:

(i)  $\{0\} \subset \dots \subset \nu_0 \subset \nu_1 \subset \dots \subset \mathcal{L}^2(\mathbb{R})$ ,

(ii)  $\bigcap_{l \in \mathbb{Z}} \nu_l = \{0\}$ .

(iii)  $\text{span} \bigcup_{l \in \mathbb{Z}} \nu_l = \mathcal{L}^2(\mathbb{R})$ ,

(iv)  $x(t) \in \nu_l \iff x(2^{-l}t) \in \nu_0$ ,

(v)  $x(t) \in \nu_0 \iff x(t-k) \in \nu_0$  para todo  $k \in \mathbb{Z}$ .

(vi)  $\{\phi(t-k)\}_{k \in \mathbb{Z}}$  es una base ortonormal para  $\nu_0$  donde  $\phi \in \nu_0$  es llamada función de escalamiento.

De (iv), donde  $\{\phi(2^l t - k)\}_{k \in \mathbb{Z}}$  es una base ortogonal para el espacio  $\nu_l$ , por lo tanto,  $\{\phi_{l,k}\}_{k \in \mathbb{Z}}$  forma una base ortonormal para  $\nu_l$  donde

$$\phi_{l,k} = 2^{l/2} \phi(2^l t - k).$$

Para obtener el MRA de  $\{\nu_l\}$  en  $\mathcal{L}^2(\mathbb{R})$  con una función de escalamiento  $\phi(\cdot)$ , la función *wavelet* asociada  $\psi(\cdot)$  se obtiene de la siguiente forma:

- Para cada  $l \in \mathbb{Z}$  se define una función  $\varpi_l$  para ser complemento ortogonal de  $\nu_l$  en  $\nu_{l+1}$ , i.e.

$$\nu_{l+1} = \nu_l \oplus \varpi_l$$

con  $\varpi$  satisfaciendo  $\varpi_l \perp \varpi_{l'}$  si  $l \neq l'$ .

- Para algún  $l_0 < l$  donde  $\nu_{l_0} \subset \nu_l$  se obtiene

$$\begin{aligned} \nu_l &= \nu_{l-1} \oplus \varpi_{l-1} \\ &= \nu_{l-2} \oplus \varpi_{l-2} \oplus \varpi_{l-1} \\ &\vdots \\ &= \nu_{l_0} \oplus_{k=l_0}^{l-1} \varpi_k \end{aligned}$$

donde cada  $\varpi_l$  satisface  $\varpi_l \subset \varpi_{l'}$  para  $l < l'$  y  $\varpi_l \perp \varpi_{l'}$  para  $l \neq l'$ .

- De (ii) y (iii) el MRA  $\{\varpi_l\}$  forma una base ortogonal para  $\mathcal{L}^2(\mathbb{R})$ ,

$$\mathcal{L}^2(\mathbb{R}) = \bigoplus_{l \in \mathbb{Z}} \varpi_l.$$

- Una función  $\psi \in \varpi_0$  la cual hace que  $\{\psi(t - k)\}_{k \in \mathbb{Z}}$  sea una base ortogonal en  $\varpi_0$  es llamada una función *wavelet*. Considerando a  $\{\psi_{l,k}\}_{k \in \mathbb{Z}}$  como una base ortonormal de  $\varpi_l$ , entonces  $\psi_{l,k} = 2^{l/2} \psi(2^l t - k)$ .

El símbolo  $\oplus$  o  $\bigoplus$  representa una suma directa;  $U \oplus V := \{u + v : u \in U, v \in V\}$ , por mostrar un ejemplo;  $\perp$  denota contradicción.

### 6.3. Red neuronal *wavelet* recurrente de primer orden

La red neuronal *wavelet* recurrente de primer orden es una ANN con una simple conexión de primer orden y es la estructura más simple para una RHONN, esta última considerada una expansión de los modelos Hopfield [Hopfield, 1984] y Cohen-Grossberg [Cohen

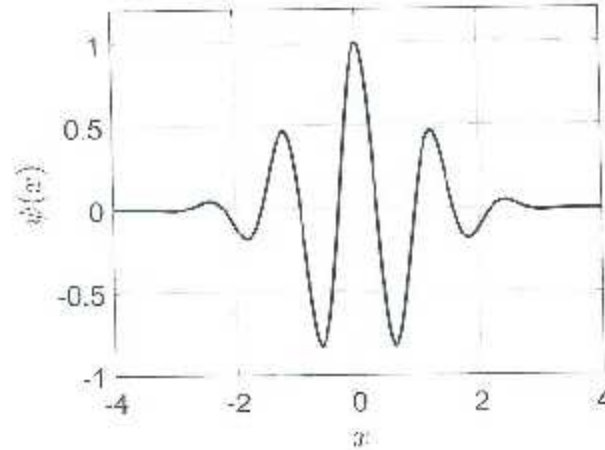


Figura 6.1: Wavelet Morlet (Gabor).

and Grossberg, 1983] de primer orden. La función de activación sigmoidal  $s(\cdot)$  es reemplazada por la versión real modificada de la función *wavelet* Morlet [Addison et al., 2002] y [Mi et al., 2005] de la forma

$$\psi = e^{-x^2/\beta} \cos(\lambda x) \quad (6.3)$$

con parámetros  $\beta$  y  $\lambda$  representando a la expansión y dilatación, respectivamente. La Figura 6.1 muestra la representación gráfica de una función *wavelet* Morlet.

Considerando que los términos de alto orden  $y_n y_i$  (Sección 4.2) no serán utilizados, se reemplaza el vector  $z$  en (4.6) por

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \\ y_{n+1} \\ \vdots \\ y_{n+m} \end{bmatrix} = \begin{bmatrix} \psi(x_1) \\ \vdots \\ \psi(x_n) \\ u_1 \\ \vdots \\ u_m \end{bmatrix}, \quad (6.4)$$

el cual es similar a (4.3), con la diferencia de que se ha reemplazado la función  $s(\cdot)$  por la función *wavelet*  $\psi$ ; por lo anterior, el modelo RIONN puede ser reescrito como:

$$\hat{x}_i = a_i x_i + b_i \sum_{k=1}^L w_{ik} y_k. \quad (6.5)$$

El vector de parámetros ajustables es definido por  $\mathbf{w}_i = b_i [w_{i1} \ w_{i2} \ \dots \ w_{iL}]^T$ , por lo que (6.5) se convierte en

$$\hat{x}_i = a_i x_i + \mathbf{w}_i^T \mathbf{y}, \quad (6.6)$$

donde el vector  $w_k$ , para  $k = 1, 2, \dots, L$  representa los pesos ajustables de la NN mientras los coeficientes  $a_i$  para  $i = 1, 2, \dots, n$  son parte de la arquitectura de la NN y son fijos durante el entrenamiento. De aquí en adelante se entenderá que al hablar de  $y$  se habla de (6.4). En el presente estudio, la estructura en la forma (6.6) es denominada RWFONN.

El comportamiento dinámico de la red en conjunto se describe expresando (6.5) en notación vectorial como:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} - \mathbf{W}^T \mathbf{y}, \quad (6.7)$$

donde  $\mathbf{x} = [x_1 \dots x_n]^T \in \mathbb{R}^n$ ,  $\mathbf{W} \in \mathbb{R}^{L \times n}$  y  $\mathbf{A} = \text{diag}\{-a_1, -a_2, \dots, -a_n\} \in \mathbb{R}^{n \times n}$  es una matriz diagonal. Siendo que  $a_i > 0 \forall i = 1, 2, \dots, n$  luego  $\mathbf{A}$  es una matriz de estabilidad (matriz Hurwitz). El vector  $\mathbf{y}$  es una función tanto del estado  $\mathbf{x}$  de la NN como de la entrada externa  $\mathbf{u}$ .

### 6.3.1. Propiedades de aproximación para la RWFONN

El comportamiento dinámico de un sistema puede ser expresado mediante la siguiente expresión

$$\dot{\chi} = F(\chi, \mathbf{u}), \quad (6.8)$$

donde  $\mathbf{u} \in \mathbb{R}^m$  es la entrada para el sistema,  $\chi \in \mathbb{R}^n$  es el estado del sistema y  $F: \mathbb{R}^{n+m} \rightarrow \mathbb{R}^n$  es un campo vectorial suave definido sobre un conjunto compacto  $\mathcal{Y} \subset \mathbb{R}^{n+m}$  donde  $m$  y  $n$  son constantes.

El problema de aproximación consiste en determinar, usando una función de activación *wavelet* para tiempo continuo, si existen pesos  $\mathbf{W}$  tal que (6.7) aproxime el comportamiento entrada-salida de un sistema dinámico arbitrario de la forma (6.8). Suponga que  $F(\cdot)$  es continua y satisface la condición local Lipschitz tal que (6.8) tiene una única solución y  $(\chi(t), \mathbf{u}(t)) \in \mathcal{Y}$  para todo  $t$  en algún intervalo de tiempo  $J_T = t: 0 \leq t \leq T$  donde  $J_T$  representa el periodo de tiempo sobre el cual se desempeña la aproximación. Basado en las consideraciones anteriores y tomando como referencia el Teorema 2 [Kosmatopoulos et al., 1995], se puede enunciar el siguiente teorema, el cual es estrictamente un resultado existencial y no provee ningún método constructivo para obtener el correcto valor de los pesos sinápticos  $w_k^*$ ; lo que sí establece es que si existe un suficiente número de pesos sinápticos en (6.7) entonces, será posible aproximar cualquier sistema dinámico para cualquier grado de exactitud.

**Teorema 4.** Suponga que el sistema (6.8) y el modelo RWFONN (6.7) están inicialmente en el mismo estado  $\chi(0) = \mathbf{x}(0)$ . Entonces, para cualquier  $\epsilon > 0$  y cualquier valor finito  $T > 0$  existe un entero  $L$  y una matriz  $\mathbf{W} \in \mathbb{R}^{L \times n}$  tales que el estado  $\mathbf{x}(t)$  del modelo RWFONN (6.7), con  $L$ , siendo el número de pesos con valores  $\mathbf{W} = \mathbf{W}^*$ , satisfacen

$$\sup_{0 \leq t \leq T} |\mathbf{x}(t) - \chi(t)| \leq \epsilon$$

*Demostración.* (Siguiendo las mismas líneas de [Kosmatopoulos et al., 1995]) El comportamiento dinámico del modelo RWPONN es descrito por

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{W}^T \mathbf{y}. \quad (6.9)$$

Suponiendo que la matriz  $\mathbf{A}\mathbf{x}$  es definida positiva, entonces, se garantiza el acotamiento entrada-salida establecido para cada neurona  $x_i$ . Sumando y restando  $\mathbf{A}\mathbf{x}$ , el sistema (6.8) puede ser reescrito como

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{G}(\mathbf{x}, \mathbf{u}) \quad (6.10)$$

donde  $\mathbf{G}(\mathbf{x}, \mathbf{u}) = \mathbf{F}(\mathbf{x}, \mathbf{u}) - \mathbf{A}\mathbf{x}$ . Dado que  $\mathbf{x}(0) = \hat{\mathbf{x}}(0)$ , el error de identificación  $\boldsymbol{\xi} = \mathbf{x} - \hat{\mathbf{x}}$  satisface

$$\dot{\boldsymbol{\xi}} = \mathbf{A}\boldsymbol{\xi} + \mathbf{W}^T \mathbf{y}(\mathbf{x}, \mathbf{u}) - \mathbf{G}(\mathbf{x}, \mathbf{u}) \quad (6.11)$$

con  $\boldsymbol{\xi}(0) = 0$ .

Suponiendo que  $(\mathbf{x}(t), \mathbf{u}(t)) \in \mathcal{Y} \forall t \in [0, T]$  donde  $\mathcal{Y}$  es un conjunto compacto de  $\mathbb{R}^{n+m}$ , se tiene

$$\mathcal{Y}_\varepsilon = \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^{n+m} : |(\mathbf{x}, \mathbf{u}) - (\mathbf{x}_y, \mathbf{u}_y)| \leq \varepsilon, (\mathbf{x}_y, \mathbf{u}_y) \in \mathcal{Y}\}.$$

Puede apreciarse que  $\mathcal{Y}$  es también un subconjunto compacto de  $\mathbb{R}^{n+m}$  y  $\mathcal{Y} \subset \mathcal{Y}_\varepsilon$ , donde  $\varepsilon$  es el grado requerido de aproximación. Por lo tanto  $\mathbf{y}$  es una función continua que satisface la condición Lipschitz

$$|\mathbf{y}(\mathbf{x}_1, \mathbf{u}) - \mathbf{y}(\mathbf{x}_2, \mathbf{u})| \leq l|\mathbf{x}_1 - \mathbf{x}_2|, \quad (6.12)$$

en el dominio compacto  $\mathcal{Y}_\varepsilon$ .

La función  $\mathbf{W}^T \mathbf{y}(\mathbf{x}, \mathbf{u})$  satisface las condiciones del teorema de Stone-Weierstrass [Stone, 1937] y que puede, sin embargo, aproximar sobre un dominio compacto cualquier función continua. El preprocesamiento de la entrada a través de una función invertible continua no afecta la habilidad de la NN para aproximar funciones continuas. Sin embargo, se ha demostrado que si  $L$  es lo suficientemente grande, i.e. el número de pesos es lo suficientemente grande, entonces existen valores de pesos  $\mathbf{W} = \mathbf{W}^*$  tales que  $(\mathbf{W}^*)^T \mathbf{y}(\mathbf{x}, \mathbf{u})$  puede aproximar a  $\mathbf{G}(\mathbf{x}, \mathbf{u})$  en un dominio compacto con cualquier grado de precisión para todo  $(\mathbf{x}, \mathbf{u})$ . Por lo tanto, existen  $\mathbf{W} = \mathbf{W}^*$  tales que

$$\sup_{(\mathbf{x}, \mathbf{u}) \in \mathcal{Y}_\varepsilon} |(\mathbf{W}^*)^T \mathbf{y}(\mathbf{x}, \mathbf{u}) - \mathbf{G}(\mathbf{x}, \mathbf{u})| \leq \eta \quad (6.13)$$

donde  $\eta$  es una constante.

La solución para la ecuación diferencial (6.11) puede ser expresada como:

$$\begin{aligned}
 \xi(t) &= \int_0^t \xi^{-A(t-\tau)} [(W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau)) - G(x(\tau), \mathbf{u}(\tau))] d\tau \\
 &= \int_0^t \xi^{-A(t-\tau)} [(W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau)) - (W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau))] d\tau \\
 &+ \int_0^t \xi^{-A(t-\tau)} [(W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau)) - G(x(\tau), \mathbf{u}(\tau))] d\tau. \quad (6.14)
 \end{aligned}$$

donde cada  $A$  es una matriz definida positiva. Existe un valor para  $\alpha$  tal que  $\|\xi^{-At}\| \leq \xi^{-\alpha t}$  y  $\alpha < L$ , donde  $L = l \|W^*\|$ . Por todo lo anterior  $\eta$  puede elegirse como:

$$\eta = \frac{(L - \alpha)\varepsilon}{2(\xi^{(L-\alpha)T} - 1)} > 0. \quad (6.15)$$

Considere ahora que  $(x(t), u(t)) \in \mathcal{Y}_\varepsilon, \forall t \in [0, T]$ . De (6.14), considerando la norma en ambos lados y usando (6.12), (6.13), (6.15), las siguientes desigualdades se cumplen para  $t \in [0, T]$ :

$$\begin{aligned}
 |\xi(t)| &\leq \int_0^t \|\xi^{-A(t-\tau)}\| \cdot |(W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau)) - (W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau))| d\tau \\
 &+ \int_0^t \|\xi^{-A(t-\tau)}\| \cdot |(W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau)) - (W^*)^T \mathbf{y}(x(\tau), \mathbf{u}(\tau))| d\tau \\
 &\leq \int_0^t \xi^{-\alpha(t-\tau)} L |\xi(\tau)| d\tau + \int_0^t \eta \xi^{-\alpha(t-\tau)} d\tau
 \end{aligned}$$

Usando el lema de Bellman-Gronwall [Bellman et al., 1943] y [Gronwall, 1919], se tiene

$$\begin{aligned}
 |\xi(t)| &\leq \frac{\eta}{L - \alpha} (\xi^{(L-\alpha)t} - 1), \\
 |\xi(t)| &\leq \varepsilon/2 \quad (6.16)
 \end{aligned}$$

Suponga que  $(x, u)$  no está contenida en  $\mathcal{Y}_\varepsilon, \forall t \in [0, T]$ . Luego, por la continuidad de  $x$  existe un  $T^*$  donde  $0 < T^* \leq T$  tal que  $(x(T^*), u(T^*)) \in \partial\mathcal{Y}_\varepsilon$ , donde  $\partial\mathcal{Y}_\varepsilon$  denota la frontera de  $\mathcal{Y}_\varepsilon$ . Realizando el mismo análisis para  $t \in [0, T^*]$  en el intervalo  $|x(t) - \chi(t)| \leq \varepsilon/2$ , (6.16) se sigue cumpliendo para todo  $t \in [0, T]$ . ■

# Capítulo 7

## Identificación y control neuronal *wavelet* de un *quadrotor* UAV

En este capítulo se presentan los resultados obtenidos tanto en simulación como experimentalmente referentes a la identificación y control de un *quadrotor* UAV. La identificación y control neuronal se lleva a cabo por medio de una RWFONN bajo un enfoque descentralizado.

### 7.1. Arquitectura de la red

En la definición de la arquitectura del identificador neuronal se parte del esquema (5.9), donde la diferencia entre ambas estructuras neuronales radica en el hecho de reemplazar la función sigmoideal  $s(\cdot)$  por una función *wavelet*  $\psi(\cdot)$ , lo que conlleva a no considerar más términos de alto orden ya que la función *wavelet* es del tipo ortogonal.

### 7.2. Identificación neuronal *wavelet* descentralizada

Para conseguir la identificación de cada una de las dinámicas de los subsistemas del *quadrotor* (5.3)-(5.8), se propone el siguiente esquema RWFONN:

$$\begin{aligned} \dot{x}_1^i &= -a_1^i x_1^i + w_1^i y_1^i(\chi_1^i) + \dot{x}_2^i \\ \dot{x}_2^i &= -a_2^i x_2^i + w_2^i y_2^i(\chi_2^i) - u^i \end{aligned} \quad (7.1)$$

donde  $i = 1, 2, 3, 4, 5, 6$  indica cada uno de los subsistemas,  $x_1^i$  y  $x_2^i$  son los estados de la  $i$ -ésima NN que identifica a  $\chi_1^i$  y  $\chi_2^i$ , respectivamente, las cuales representan la posición y velocidad del  $i$ -ésimo subsistema,  $a_1^i$  y  $a_2^i$  son constantes positivas,  $w_1^i$  y  $w_2^i$  representan los pesos sinápticos de la ANN,  $u^i$  es la entrada principal de cada subsistema,  $y_1^i$  y  $y_2^i$  están en función de la posición y la velocidad, respectivamente, de cada subsistema, como se muestra enseguida:

$$\begin{aligned} y_1^i &= e^{-(x_1)^2/\beta_1} \cos(\lambda_1 \tilde{x}_1^i) \\ y_2^i &= e^{-(x_2)^2/\beta_2} \cos(\lambda_2 \tilde{x}_2^i), \end{aligned}$$

con  $\beta_1 = 20$ ,  $\lambda_1 = 0.01$  para  $y_1^i$  y  $\beta_2 = 20$ ,  $\lambda_2 = 0.01$  para  $y_2^i$ .

Para cada  $i$ -ésimo subsistema el error de identificación es definido como:

$$\xi_1^i = x_1^i - \chi_1^i \quad (7.2)$$

para la posición y

$$\xi_2^i = x_2^i - \chi_2^i \quad (7.3)$$

para la velocidad, pudiendo ser estas lineales o angulares. Para actualizar los pesos sinápticos en línea la ley de aprendizaje adaptable está dada como

$$\begin{aligned} \dot{w}_1^i &= -\gamma_1^i y_1^i \xi_1^i, \\ \dot{w}_2^i &= -\gamma_2^i y_2^i \xi_2^i, \end{aligned} \quad (7.4)$$

con  $\gamma_1^i > 0$  y  $\gamma_2^i > 0$  como ganancias adaptables.

### 7.3. Controlador neuronal *wavelet*

Utilizando el esquema de identificación neuronal (7.1) y las leyes adaptables (7.4) que actualizan los pesos del identificador, luego se puede aplicar el mismo procedimiento para el diseño de la ley de control obtenida en la Sección 5.4. Luego, la ley de control queda establecida como

$$\begin{aligned} u^i &= -c_1^i + \alpha_2^i \alpha_2^i + \alpha_2^i \chi_{2d} - w_2^i y_2^i - \dot{\chi}_{2d} + \alpha_1^i \chi_{2d} \\ &\quad - \frac{dw_1^i}{dt} y_1^i - \frac{dy_1^i}{dt} w_1^i - c_1^i (-\alpha_1^i \epsilon_1^i - c_1^i \epsilon_1^i - \zeta_1^i) - c_2^i \zeta_2^i, \end{aligned} \quad (7.5)$$

con  $c_1^i$  definido como el error de seguimiento de la salida de la RWFONN,  $c_1^i > 0$  como ganancias de control y  $\alpha_2^i$  dada como

$$\alpha_2^i = \alpha_1^i \chi_{1d} - w_1^i y_1^i - c_1^i \epsilon_1^i. \quad (7.6)$$

Los diagramas de control utilizados para la identificación y control RWFONN son los mismos que los utilizados en el esquema de identificación y control neuronal descentralizado RHONN, (Figura 5.4) y (Figura 5.5), con una función *wavelet* como función de activación.



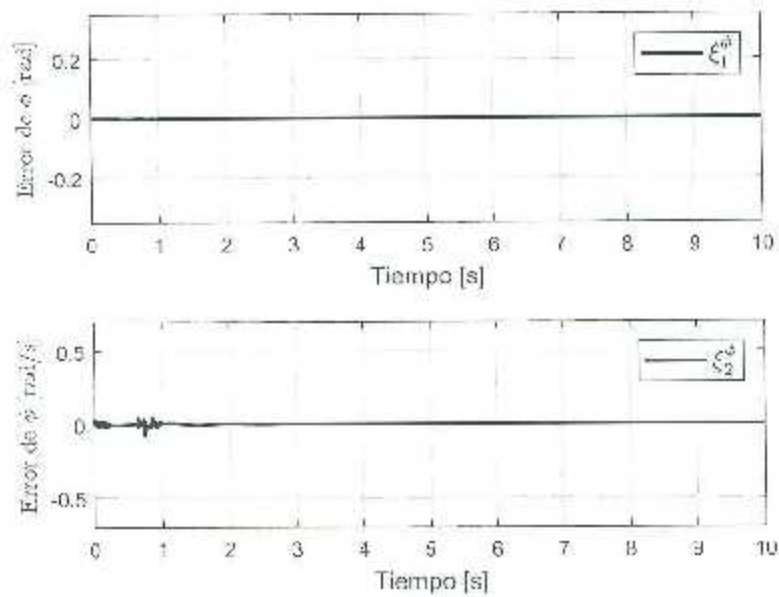


Figura 7.1: Error de identificación RWFONN descentralizado de las dinámicas de alabeo.

## 7.4. Resultados en simulación

A continuación se discuten los resultados en simulación obtenidos de la identificación y control simultáneo de las dinámicas del *quadrotor* vía RWFONN. La Tabla 5.1 muestra los parámetros proporcionados por Quanser para el *quadrotor* Qball-2 [User Manual QBall 2, 2014], los cuales fueron considerados para la presente tarea.

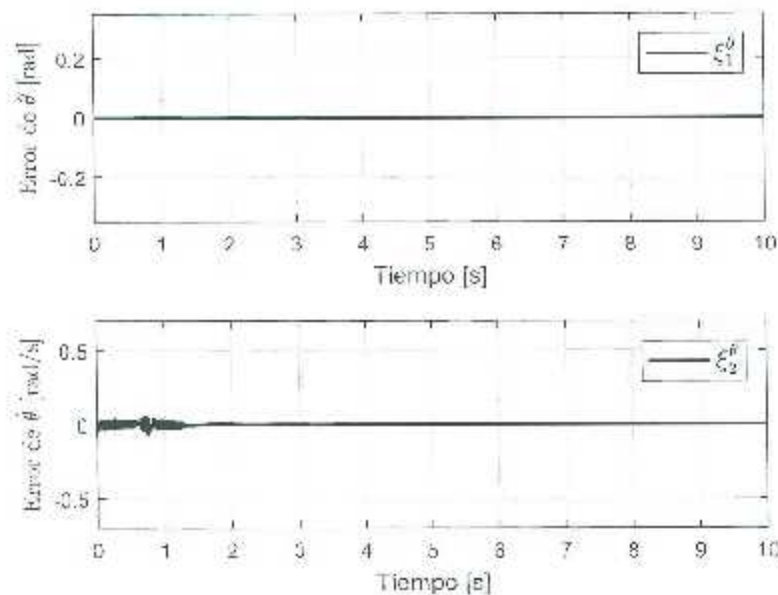


Figura 7.2: Error de identificación RWFONN descentralizado de las dinámicas de cabeceo.

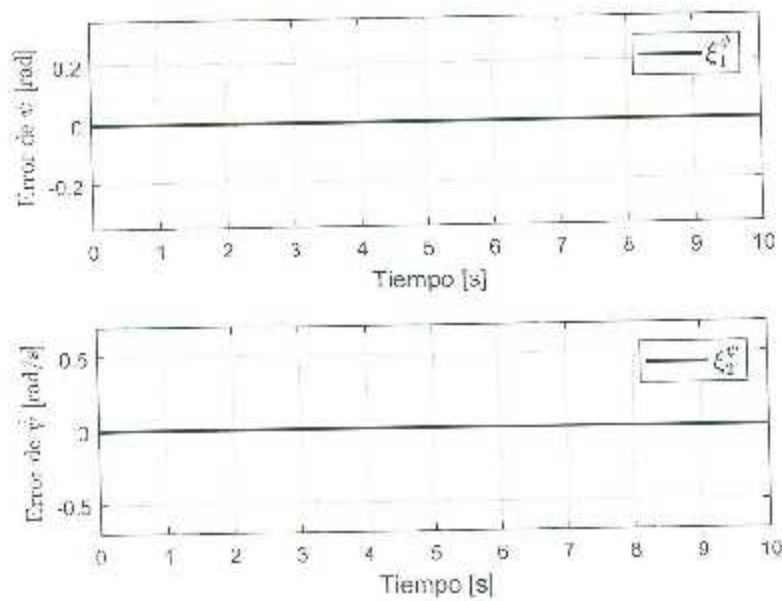


Figura 7.3: Error de identificación RWFONN descentralizado de las dinámicas de guiñada.

### 7.4.1. Resultados de identificación

De las Figuras 7.1 a la 7.6 se exhibe el desempeño en simulación de la identificación de la dinámica de los subsistemas (5.3)-(5.8). Las condiciones iniciales consideradas para la trayectoria a describir en la identificación de las dinámicas del sistema son  $\phi_0 = 0$ ,  $\theta_0 = 0$ ,

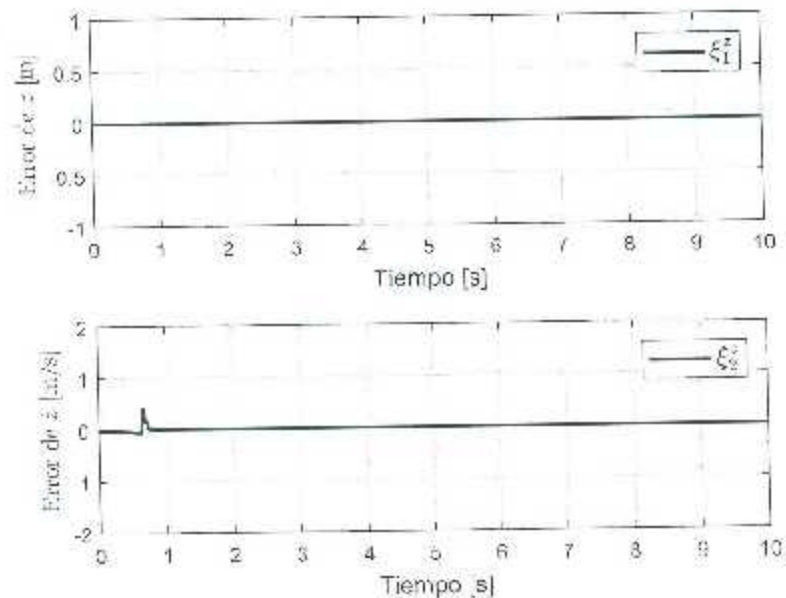


Figura 7.4: Error de identificación RWFONN descentralizado de las dinámicas traslacionales de  $z$ .

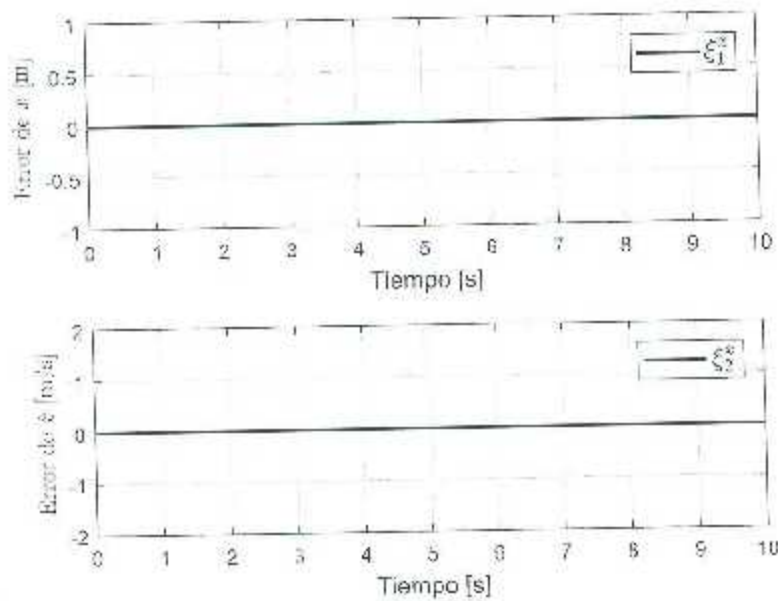


Figura 7.5: Error de identificación RWFONN descentralizado de las dinámicas translacionales de  $x$ .

$\psi_0 = 0$ ,  $z_0 = 0$ ,  $x_0 = 0$  y  $y_0 = 0$  en tanto que las condiciones iniciales de los estados de las RWFONNs son  $x_j^i = 0$  para todo  $i = \phi, \theta, \psi, z, x, y$  y  $j = 1, 2$ . Las ganancias  $\gamma$  utilizadas son  $\gamma_j^z = \gamma_j^x = \gamma_j^y = \gamma_j^\psi = 5500$ ,  $\gamma_j^\phi = 25000$  y  $\gamma_j^\theta = 20000$  para  $j = 1, 2$ . Los

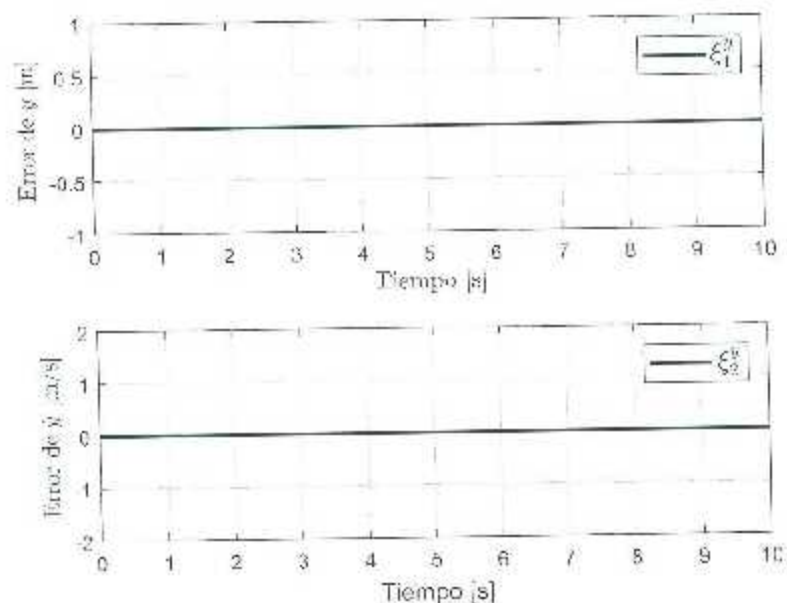


Figura 7.6: Error de identificación RWFONN descentralizado de las dinámicas translacionales de  $y$ .

valores para las constantes se fijaron como  $a_1^i = a_2^i = 100$  y  $b_1^i = b_2^i = 1$  para cada uno de los subsistemas.

En las Figuras 7.1-7.6 se observan los errores de identificación neuronal de posición y velocidad para los subsistemas  $S_\phi$ ,  $S_\theta$ ,  $S_\psi$ ,  $S_{z_1}$ ,  $S_x$  y  $S_y$ , respectivamente, los cuales muestran resultados superiores a los obtenidos en el esquema de identificación RHONN descentralizado, esto al no presentar una desviación inicial tan pronunciada.

## 7.4.2. Resultados de control

En las Figuras 7.9 a la 7.14 se muestran los errores de seguimiento y los pares generados para cada subsistema. En la Figura 7.7 se muestra la trayectoria descrita por el *quadrotor* UAV contra la de referencia. Las señales de referencia para el control del *quadrotor* son  $\chi_{1d}^z = 1$  [m],  $\chi_{1d}^y = 0.1745 \sin(t/5)$  [rad],  $\chi_{1d}^x = Am \cos(ft)$  [m] y  $\chi_{1d}^x = Am \sin(ft)$  [m], con  $Am = 0.5$  [m] y  $f = \pi/12.5$  [rad/s].

En las Figuras 7.13 - 7.14 se muestran los errores de seguimiento de trayectoria de posición y señales de control para los subsistemas  $S_\phi$ ,  $S_\theta$ ,  $S_\psi$ ,  $S_{z_1}$ ,  $S_x$  y  $S_y$ , respectivamente, en la cual se puede observar que los errores de seguimiento se mantiene acotado durante el vuelo.

Para evaluar el desempeño del esquema RWFONN se consideró el MSE. La Tabla 7.1 muestra los MSEs de la identificación neuronal de cada uno de los subsistemas, mientras que la Tabla 7.2 muestra los MSEs del seguimiento de trayectoria de cada subsistema.

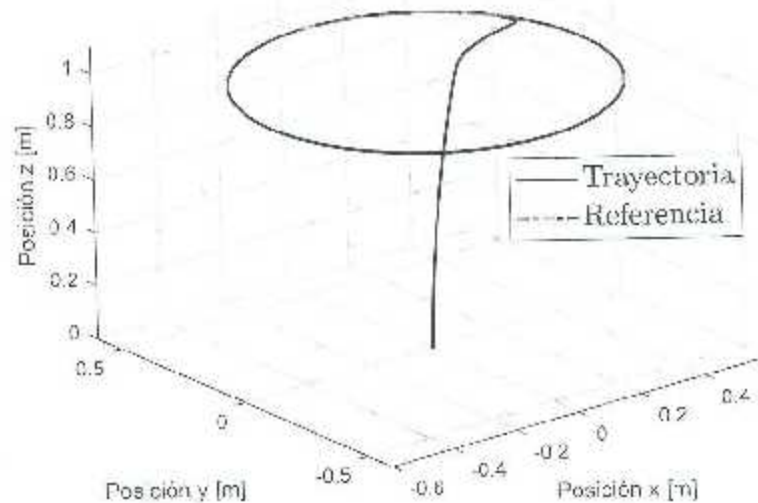


Figura 7.7: Desempeño del esquema de control neuronal propuesto.

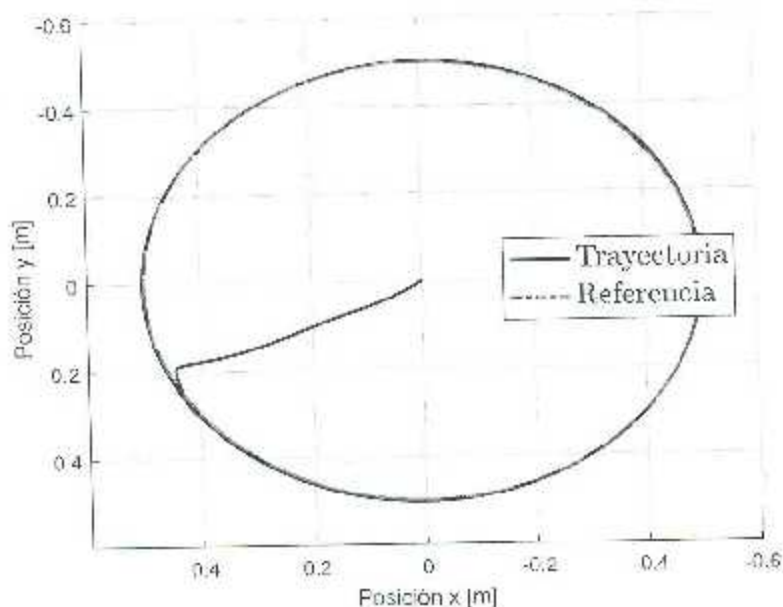


Figura 7.8: Vista superior del desempeño del esquema de control neuronal propuesto.

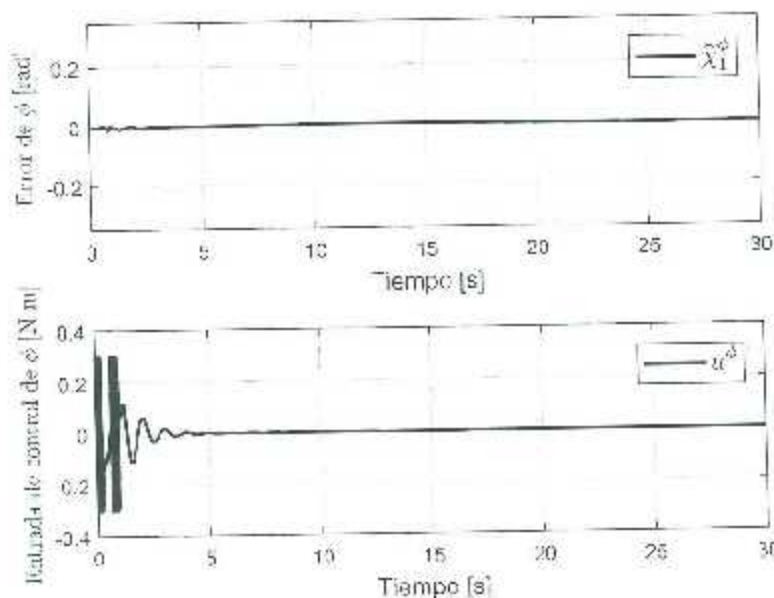


Figura 7.9: Error de seguimiento y señal de control para la dinámica de rotación de alabeo.

## 7.5. Resultados experimentales

En la presente sección se discuten los resultados experimentales obtenidos de la identificación y control del *quadrotor* UAV vía RWFONN usando el esquema neuronal (7.1) y las leyes de actualización de pesos (7.4).

Tabla 7.1: MSEs para la identificación vía RWFONNs.

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.000085 | 0.000112 | 0.000097 | 0.001289 | 0.000213 | 0.000171 |

Tabla 7.2: MSEs para el seguimiento.

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.002553 | 0.003593 | 0.000274 | 0.111253 | 0.075974 | 0.004960 |

Cabe destacar que en los resultados mostrados a continuación solo se consiguió el control vía RWFONNs de los subsistemas  $S_\phi$ ,  $S_z$ ,  $S_x$  y  $S_y$ , en tanto que para los subsistemas de  $S_\theta$  y  $S_\psi$  se utilizaron otros controladores que nos permitieron volar el *quadrotor* UAV. Las señales de referencia para el control del *quadrotor* son  $\chi_{1d}^z = 0.5$  [m],  $\chi_{1d}^\psi = 0$  [rad],  $\chi_{1d}^x = 0$  [m] y  $\chi_{1d}^y = 0$  [m].

En las Figuras 7.20 a la 7.22 se exhibe el desempeño de los controladores neuronales aplicados a las dinámicas de los subsistemas (5.5)-(5.8), a través del error de seguimiento y su señal de control. La Figura 7.19 muestra la trayectoria descrita por el *quadrotor* UAV contra la referencia. Las condiciones iniciales consideradas para la trayectoria a identificar son  $\phi_0 \approx 0$ ,  $\theta_0 \approx 0$ ,  $\psi_0 \approx 0$ ,  $z_0 \approx 0.29$  [m],  $x_0 \approx 0$  y  $y_0 \approx 0$ , las condiciones iniciales de los estados de las NN son  $x_j^i = 0$  para todo  $i = 3, 4, 5, 6$  y  $j = 1, 2$ . Las ganancias  $\gamma$

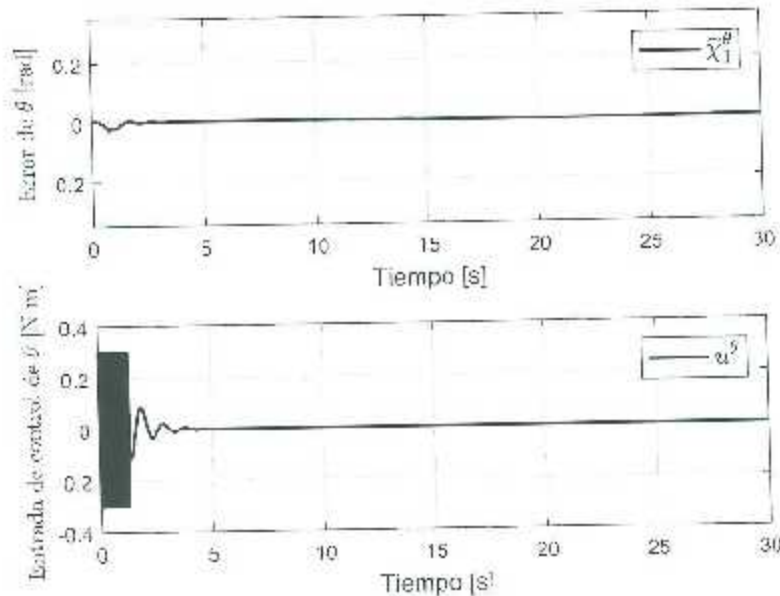


Figura 7.10: Error de seguimiento y señal de control para la dinámica de rotación de cabeceo.

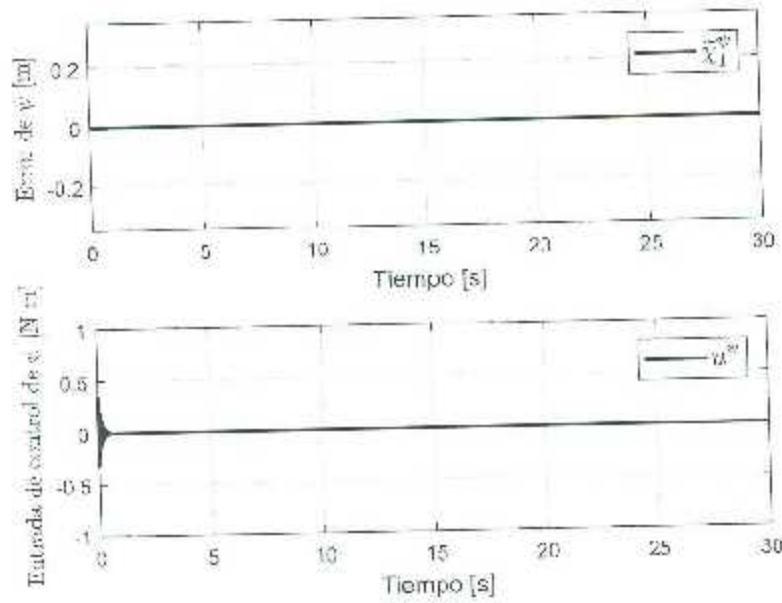


Figura 7.11: Error de seguimiento y señal de control para la dinámica de rotación de guiñada.

utilizadas fueron establecidas como  $\gamma_j^i = 3000$  para  $i = \psi, z, x, y$  y  $j = 1, 2$ . Los valores de las constantes fueron definidas como  $a_1^i = a_2^i = 100$  y  $b_1^i = b_2^i = 1$ .

En las Figuras 7.20 - 7.22 se exhiben los errores de seguimiento de trayectoria de posición y señal de control para la dinámica de los subsistemas  $S_\psi$ ,  $S_z$ ,  $S_x$  y  $S_y$ . En estas se

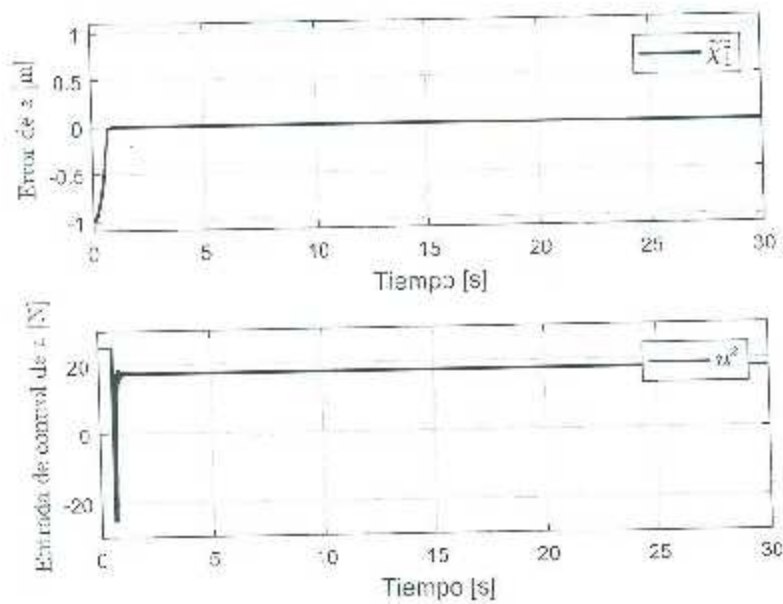


Figura 7.12: Error de seguimiento y señal de control para la dinámica de translación de  $z$ .

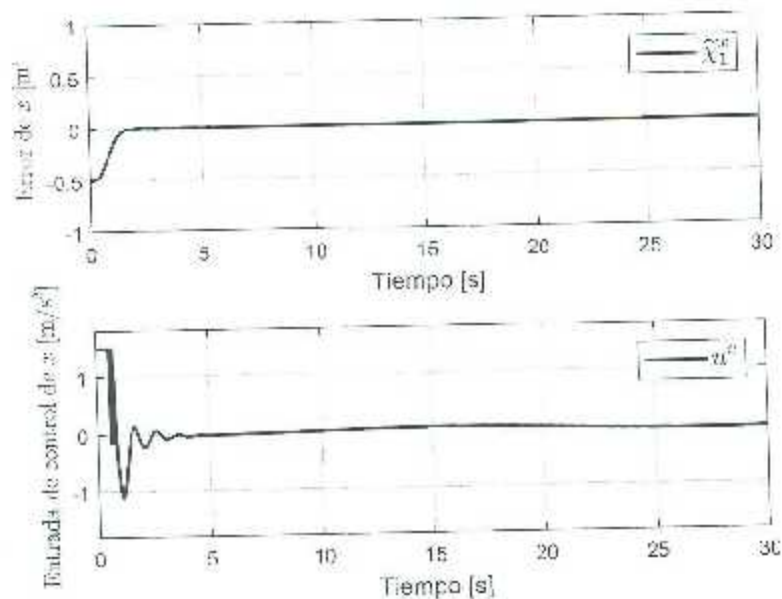


Figura 7.13: Error de seguimiento y señal de control para la dinámica de translación de  $x$ .

puede observar que las señales de control se saturan y son muy demandantes, lo que consume considerablemente los actuadores del *quadrotor* y mengua la duración de la batería. El error de posición de guiñada casi alcanza los 0.1 radianes, el error de seguimiento translacional de  $x$  no rebasa los 4 centímetros, en cambio, el error de seguimiento translacional de  $y$  mantiene un *offset* de 20 centímetros.

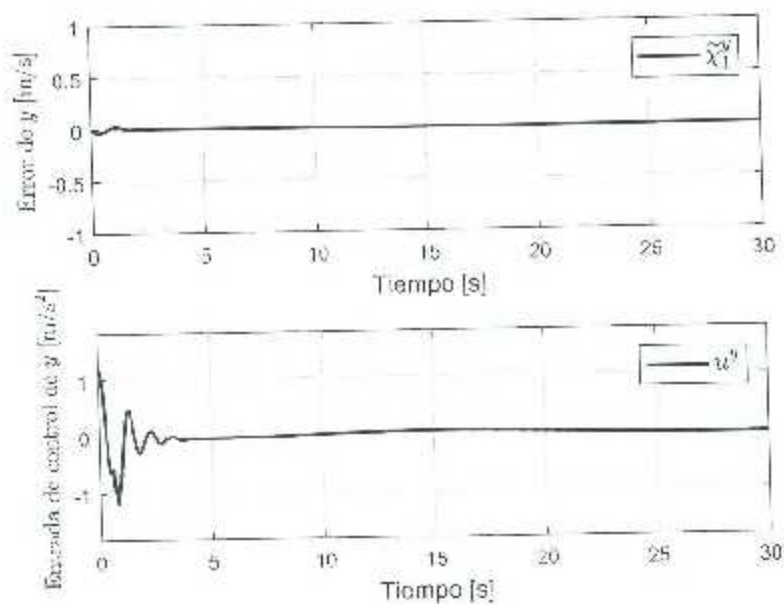


Figura 7.14: Error de seguimiento y señal de control para la dinámica de translación de  $y$ .



La Tabla 7.3 muestra los MSEs de seguimiento de trayectoria de cada uno subsistema.

Tabla 7.3: MSEs para el seguimiento vía RWFONNs

| Subsistema | $\phi$   | $\theta$ | $\psi$   | $z$      | $x$      | $y$      |
|------------|----------|----------|----------|----------|----------|----------|
| MSE        | 0.012305 | 0.019967 | 0.038071 | 0.156589 | 0.018176 | 0.130199 |

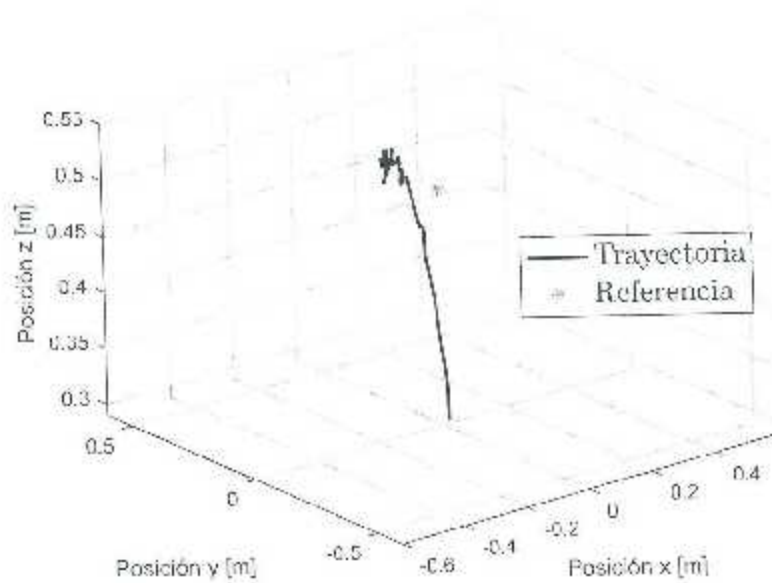


Figura 7.15: Desempeño de la RWFONN descentralizada.

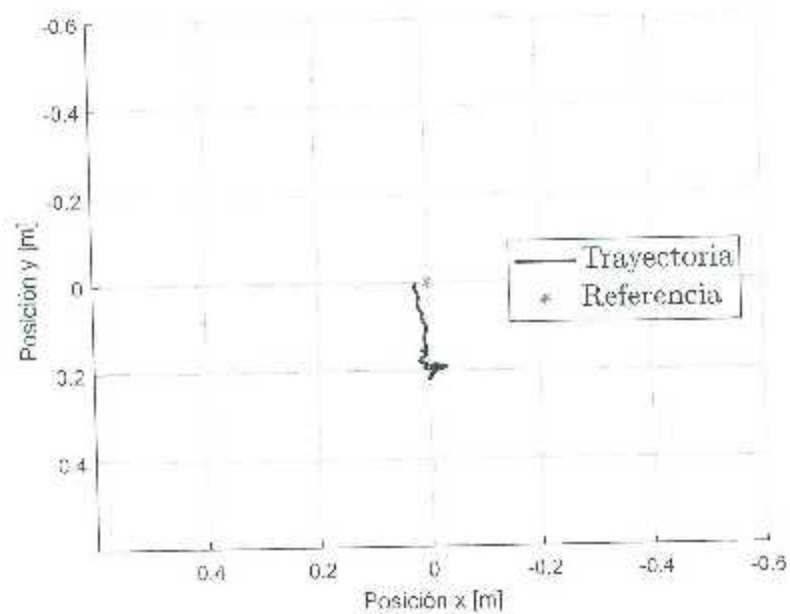


Figura 7.16: Vista superior del desempeño de la RWFONN descentralizada.

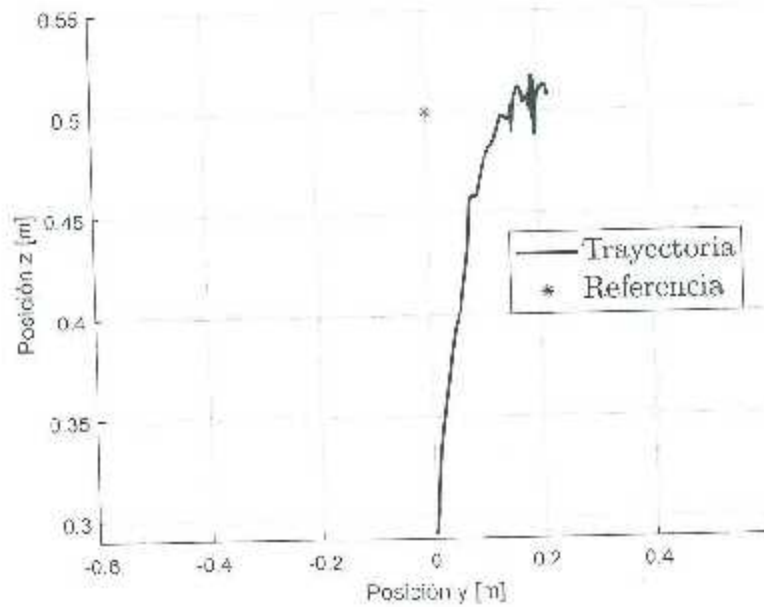


Figura 7.17: Vista lateral del desempeño de la RWFONN descentralizada.

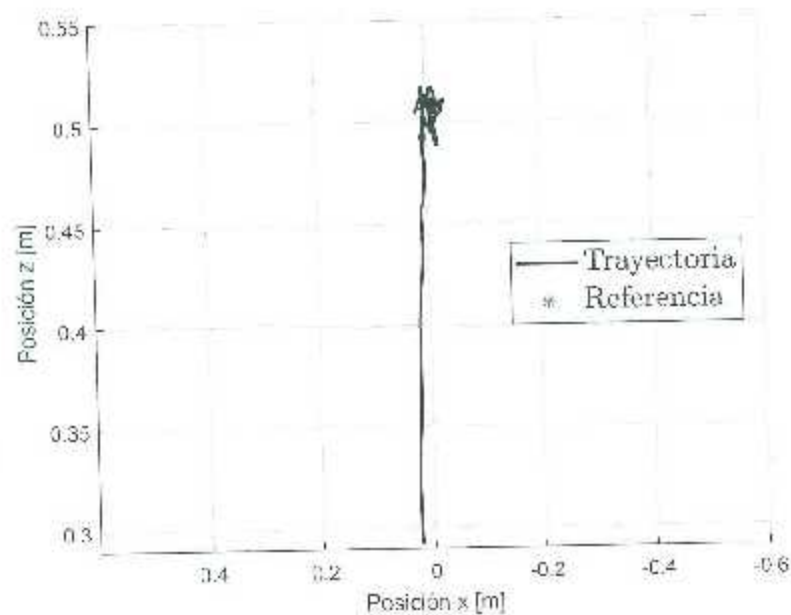


Figura 7.18: Vista frontal del desempeño de la RWFONN descentralizada.

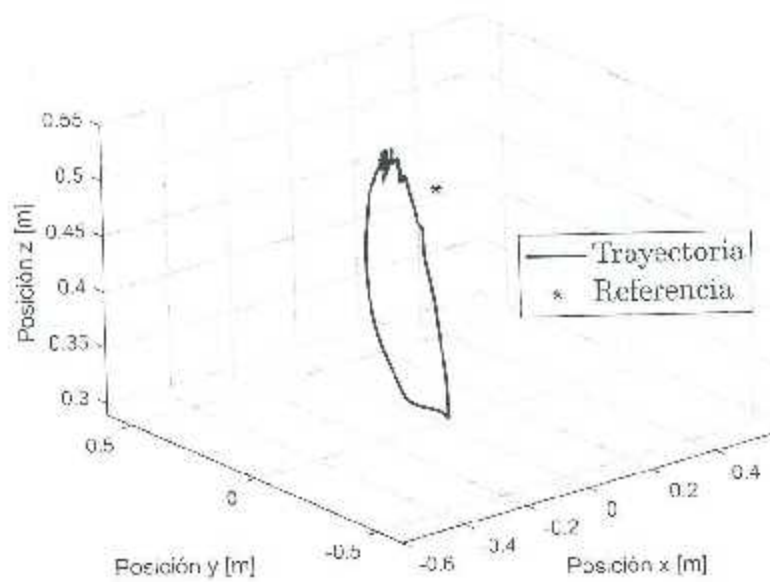


Figura 7.19: Desempeño de la RWFONN descentralizada para la trayectoria completa.

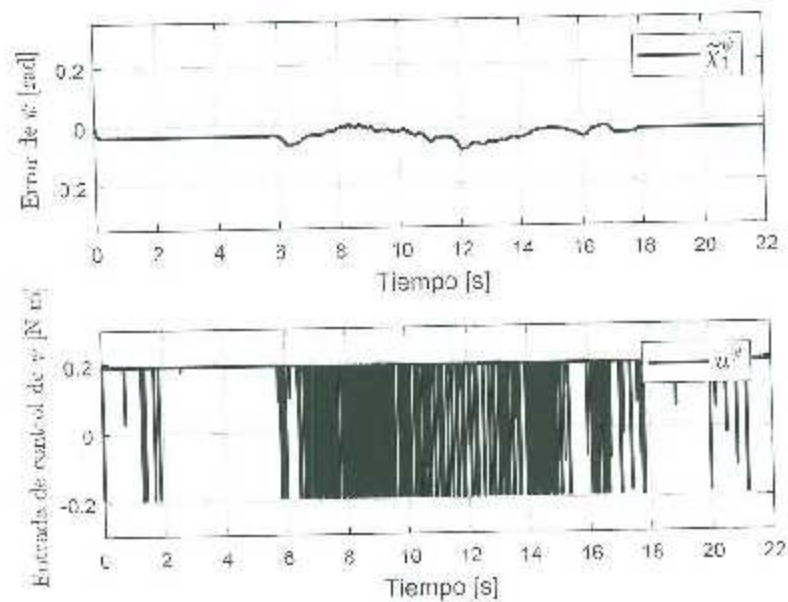


Figura 7.20: Error de seguimiento y señal de control para  $\psi$  en la plataforma experimental.

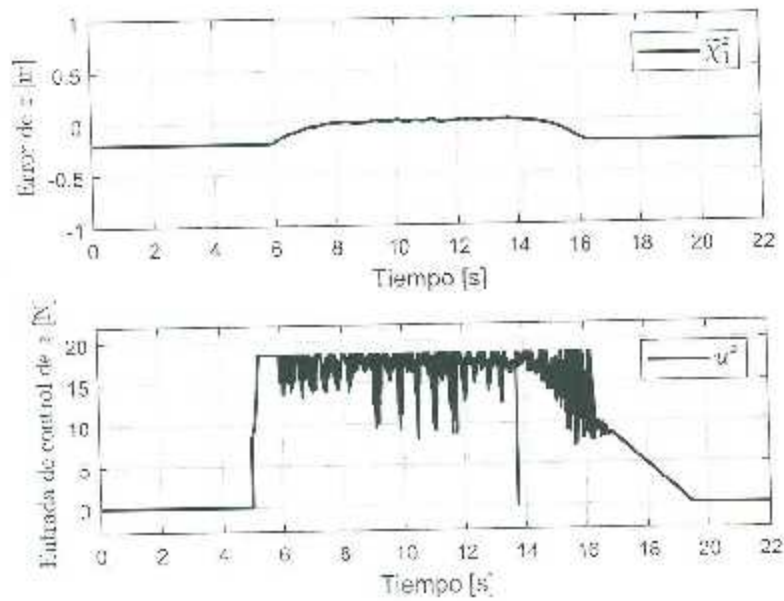


Figura 7.21: Error de seguimiento y señal de control para  $z$  en la plataforma experimental.

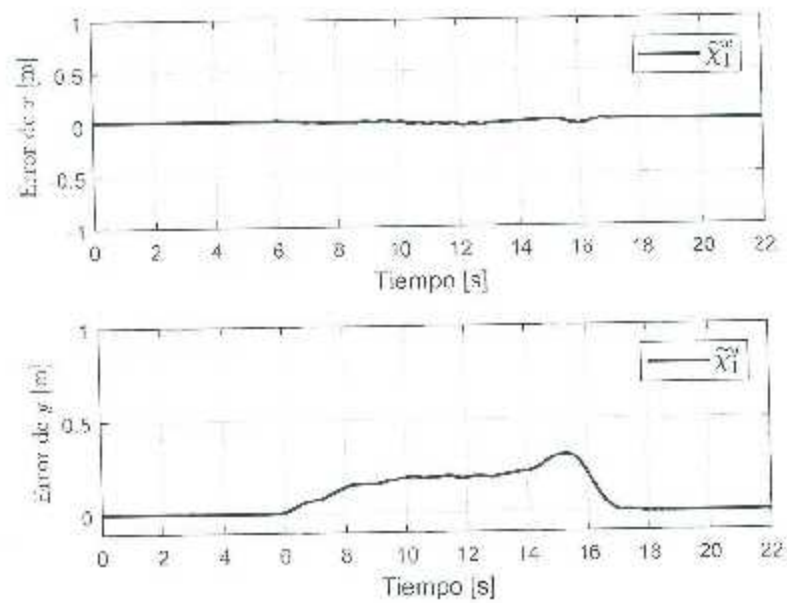


Figura 7.22: Error de seguimiento para  $x$  y  $y$  en la plataforma experimental.

# Capítulo 8

## Conclusiones

En el presente trabajo de tesis se trabajó con la implementación de esquemas de identificación y control neuronal descentralizado aplicados a una mini-nave aérea tipo helicóptero de cuatro rotores, también llamada *quadrotor*.

Los identificadores neuronales vía RHONN presentan una cresta o pico de magnitud considerable, en comparación con el error en estado estacionario que puede llegar a alcanzar. Esta cresta se presenta en los primeros instantes de tiempo en los que el identificador trabaja, y ocurre incluso igualando las condiciones iniciales de la dinámica a identificar con los estados del identificador neuronal. En cuanto a la implementación del esquema RHONN, las ganancias  $\gamma_j^i$  requeridas para la identificación de los subsistemas rotacionales ( $S_\phi$ ,  $S_\theta$  y  $S_\psi$ ) son mayores que para los subsistemas traslacionales ( $S_z$ ,  $S_x$  y  $S_y$ ).

El esquema de identificación neuronal *wavelet* es superior en todo sentido al esquema de identificación RHONN, responde rápido y mantiene un error de identificación muy pequeño, además requiere un valor de las ganancias  $\gamma_j^i$  pequeño para tener un buen desempeño en comparación con el identificador RHONN. El hecho de que el esquema RHONN requiera valores grandes de las  $\gamma_j^i$  puede afectar de manera negativa al controlador.

El desempeño de los controladores en simulación en general fue bueno, pero una incorrecta sintonización de las ganancias  $\gamma_j^i$  pueden resultar en un mal funcionamiento del controlador (se puede apreciar ruido en la señal de control). Luego los valores ideales de las ganancias  $\gamma_j^i$  para obtener el mejor desempeño de identificación no siempre dará los mejores resultados ya aplicando el control basado en este identificador neuronal.

La implementación práctica de la identificación y control del *quadrotor* UAV vía RWFONN arrojó un buen desempeño de control para las coordenadas de  $z$  y  $x$ , sin embargo en la coordenada  $y$  se presentó un *offset* de 20 centímetros en todo momento durante el vuelo, el cual no se consiguió eliminar. El desempeño del control de  $\psi$  no fue el ideal, ya que el error de seguimiento se mantenía apenas por debajo de los 5 grados, además que su señal de control era un poco errática. No fue posible lograr el control de los subsistemas de  $\phi$  y  $\theta$ , lo anterior pudiera implicar que al esquema RWFONN le es complicado controlar la coordenadas rotacionales.

Las leyes de control arrojadas por los controladores basados en RWFONN son muy demandantes, esto hace que el *quadrotor* tenga un menor tiempo de vuelo e incluso menos vuelos. En una primera instancia se puede concluir que se requieren más pruebas experimentales para lograr que el esquema RWFONN trabaje de manera correcta para cada uno de los subsistemas a la vez, esto con el fin de encontrar los parámetros correctos que arrojen un buen desempeño en la identificación y control neuronal.

Se debe destacar, además, que al usarse el mismo diseño de identificación y control para cada uno de los subsistemas se facilita mucho la implementación en simulación, tanto para el esquema RWFONN como para el RHONN. Más aún, las mismas ganancias de control  $g_j$  que se usaron para el esquema RHONN dieron también los mejores resultados para el esquema RWFONN.

# Producción académica

Durante el desarrollo del presente trabajo de tesis se sometieron algunos artículos, trabajos a congresos. Hasta el momento se tiene la siguiente productividad.

Aceptados:

- Decentralized Direct MRAC for Attitude Control of a Quadrotor UAV. 2017 14th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE).
- Decentralized Control of the Quadrotor's 6 DOF. The International Conference on Mechatronics, Electronics and Automotive Engineering 2017 (ICMEAE)

## Trabajos futuros

Aun queda mucho por hacer en cuanto a estos esquemas se refiere, se puede destacar como trabajos futuros el dar continuidad a lo que en este trabajo se presenta, haciendo una comparativa del esquema de identificación y control vía RHONN con el esquema de RWFONN aplicado al *quadrotor*, tanto en simulación como en la plataforma experimental. A pesar de que se demostró que el esquema RWFONN es superior en lo que concierne a la identificación de la dinámica del *quadrotor* es necesaria la comparativa para la parte de control.

En cuanto a la arquitectura de los identificadores neuronales aplicados al *quadrotor* se tiene en mente una nueva propuesta para descentralizar la dinámica del *quadrotor*, la cual consiste en tomar la parte de la dinámica de  $\phi$  y  $\theta$  y unirlos con sus respectivas dinámicas subactuadas  $y$  y  $z$ . Esto implicará tener dos subsistemas de cuarto orden y dos subsistemas de segundo orden, lo que representa un desafío a la hora del diseño de los controladores.



# Bibliografía

- [Addison et al., 2002] Addison, P., Watson, J., and Feng, T. (2002). Low-oscillation complex wavelets. *Journal of Sound and Vibration*, 254(4):733–762.
- [Antonio-Toledo et al., 2016] Antonio Toledo, M. E., Sanchez, E. N., and Alanís, A. Y. (2016). Robust neural decentralized control for a quadrotor uav. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 714–719. IEEE.
- [Bakshi and Ramachandran, 2016] Bakshi, N. A. and Ramachandran, R. (2016). Indirect model reference adaptive control of quadrotor uavs using neural networks. In *Intelligent Systems and Control (ISCO), 2016 10th International Conference on*, pages 1–6. IEEE.
- [Bellman et al., 1943] Bellman, R. et al. (1943). The stability of solutions of linear differential equations. *Duke Math. J.*, 10(4):643–647.
- [Bouabdallah et al., 2004] Bouabdallah, S., Murriero, P., and Siegwart, R. (2004). Design and control of an indoor micro quadrotor. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, volume 5, pages 4393–4398 Vol.5.
- [Bouabdallah and Siegwart, 2005] Bouabdallah, S. and Siegwart, R. (2005). Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In *Robotics and Automation, 2005. ICRA 2005, Proceedings of the 2005 IEEE International Conference on*, pages 2247–2252. IEEE.
- [Bouhali and Boudjedir, 2011] Bouhali, O. and Boudjedir, H. (2011). Neural network control with neuro-sliding mode observer applied to quadrotor helicopter. In *Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on*, pages 24–28. IEEE.
- [Cohen and Grossberg, 1983] Cohen, M. A. and Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Transactions on Systems, Man, and Cybernetics*, (5):815–826.
- [Cotter, 1990] Cotter, N. E. (1990). The stone-weierstrass theorem and its application to neural networks. *IEEE Transactions on Neural Networks*, 1(4):290–295.
- [Daubechies, 1992] Daubechies, I. (1992). *Ten lectures on wavelets*. SIAM.

- [Dierks and Jagannathan, 2008] Dierks, T. and Jagannathan, S. (2008). Neural network output feedback control of a quadrotor uav. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 3633–3639. IEEE.
- [Dierks and Jagannathan, 2010] Dierks, T. and Jagannathan, S. (2010). Output feedback control of a quadrotor uav using neural networks. *IEEE Transactions on Neural Networks*, 21(1):53–66.
- [Dorf and Bishop, 2001] Dorf, R. and Bishop, R. (2001). *Modern Control Systems*. World student series. Prentice Hall.
- [El-Sousy, 2013] El-Sousy, F. F. (2013). Intelligent optimal recurrent wavelet Elman neural network control system for permanent-magnet synchronous motor servo drive. *IEEE Transactions on Industrial Informatics*, 9(4):1986–2003.
- [Felix et al., 2005a] Felix, R. A., Sanchez, E. N., and Loukianov, A. G. (2005a). Avoiding controller singularities in adaptive recurrent neural control. *IFAC Proceedings Volumes*, 38(1):109–114.
- [Felix et al., 2005b] Felix, R. A., Sanchez, E. N., and Loukianov, A. G. (2005b). Avoiding controller singularities in adaptive recurrent neural control. *IFAC Proceedings Volumes*, 38(1):109–114. 16th IFAC World Congress.
- [Frye and Provence, 2014] Frye, M. T. and Provence, R. S. (2014). Direct inverse control using an artificial neural network for the autonomous hover of a helicopter. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 4121–4122. IEEE.
- [Gronwall, 1919] Gronwall, T. H. (1919). Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, pages 292–296.
- [Hale, 1969] Hale, J. (1969). *Ordinary differential equations*. Pure and applied mathematics. Wiley-Interscience.
- [Haykin, 1992] Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.
- [Ho et al., 2001] Ho, D. W., Zhang, P.-A., and Xu, J. (2001). Fuzzy wavelet networks for function learning. *IEEE Transactions on Fuzzy Systems*, 9(1):200–211.
- [Hopfield, 1984] Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences of the United States of America*, 81(10):3088–3092.
- [Huang et al., 2003] Huang, S., Tan, K. K., and Lee, T. H. (2003). Decentralized control design for large-scale systems with strong interconnections using neural networks. *IEEE Transactions on Automatic Control*, 48(5):805–810.

- 
- [Ioannou and Fidan, 2006] Ioannou, P. and Fidan, B. (2006). *Adaptive Control Tutorial*. Advances in Design and Control. Society for Industrial and Applied Mathematics.
- [Ioannou and Sun, 2013] Ioannou, P. and Sun, J. (2013). *Robust Adaptive Control*. Dover Publications.
- [Khalil, 1996] Khalil, H. K. (1996). Nonlinear systems. *Prentice-Hall, New Jersey*, 2(5):5-1.
- [Kosmatopoulos et al., 1997] Kosmatopoulos, E. B., Christodoulou, M. A., and Ioannou, P. A. (1997). Dynamical neural networks that ensure exponential identification error convergence. *Neural Networks*, 10(2):299–314.
- [Kosmatopoulos et al., 1992] Kosmatopoulos, E. B., Ioannou, P. A., and Christodoulou, M. A. (1992). Identification of nonlinear systems using new dynamic neural network structures. In *[1992] Proceedings of the 31st IEEE Conference on Decision and Control*, pages 20–25 vol.1.
- [Kosmatopoulos et al., 1995] Kosmatopoulos, E. B., Polycarpou, M. M., Christodoulou, M. A., and Ioannou, P. A. (1995). High-order neural network structures for identification of dynamical systems. *IEEE Transactions on Neural Networks*, 6(2):422–431.
- [Lavretsky and Wise, 2012] Lavretsky, E. and Wise, K. (2012). *Robust and Adaptive Control: With Aerospace Applications*. Springer London.
- [Mahony et al., 2012] Mahony, R., Kumar, V., and Corke, P. (2012). Multirotor aerial vehicles. *IEEE Robotics and Automation magazine*, 20(32).
- [Mallat, 1989] Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693.
- [Mi et al., 2005] Mi, X., Ren, H., Ouyang, Z., Wei, W., and Ma, K. (2005). The use of the mexican hat and the morlet wavelets for detection of ecological patterns. *Plant Ecology*, 179(1):1–19.
- [Mohammadi and Shahri, 2013] Mohammadi, M. and Shahri, A. M. (2013). Decentralized adaptive stabilization control for a quadrotor nav. In *Robotics and Mechatronics (ICRoM), 2013 First RSI/ISM International Conference on*, pages 288–292. IEEE.
- [Nagaty et al., 2013] Nagaty, A., Saeedi, S., Thibault, C., Seto, M., and Li, H. (2013). Control and navigation framework for quadrotor helicopters. *Journal of intelligent & robotic systems*, pages 1–12.
- [Narendra and Annaswamy, 2012] Narendra, K. S. and Annaswamy, A. M. (2012). *Stable adaptive systems*. Prentice-Hall.
- [Nicol et al., 2008] Nicol, C., Macnab, C., and Ramirez-Serrano, A. (2008). Robust neural network control of a quadrotor helicopter. In *Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on*, pages 001233–001238. IEEE.
-

- [Norgaard et al., 2003] Norgaard, M., Ravn, O., Poulsen, N., and Hansen, L. (2003). *Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook*. Springer London.
- [Olfati-Saber, 2001] Olfati-Saber, R. (2001). *Nonlinear control of underactuated mechanical systems with application to robotics and aerospace vehicles*. PhD thesis, Massachusetts Institute of Technology.
- [Perez, 2009] Perez, T. (2009). *Ship Motion Control: Course Keeping and Roll Stabilisation Using Rudder and Fins*. Springer London.
- [Rovithakis and Christodoulou, 2012] Rovithakis, G. A. and Christodoulou, M. A. (2012). *Adaptive control with recurrent high-order neural networks: Theory and industrial applications*. Springer, London.
- [Sanchez et al., 2006] Sanchez, E. N., Gaytan, A., and Saad, M. (2006). Decentralized neural identification and control for robotics manipulators. In *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pages 1614–1619.
- [Stone, 1937] Stone, M. H. (1937). Applications of the theory of boolean rings to general topology. *Transactions of the American Mathematical Society*, 41(3):375–481.
- [User Manual QBall 2, 2014] User Manual QBall 2, Q. (2014). *User Manual QBall 2 for QUARC*. Ontario, Canada. Quanser Inc.
- [Vázquez and Jurado, 2014] Vázquez, L. A. and Jurado, F. (2014). Continuous-time decentralized wavelet neural control for a 2 dof robot manipulator. In *Electrical Engineering, Computing Science and Automatic Control (CCE), 2014 11th International Conference on*, pages 1–6. IEEE.
- [Vázquez et al., 2015] Vázquez, L. A., Jurado, F., and Alanís, A. Y. (2015). Decentralized identification and control in real-time of a robot manipulator via recurrent wavelet first-order neural network. *Mathematical Problems in Engineering*, 2015.
- [Voos, 2007] Voos, H. (2007). Nonlinear and neural network-based control of a small four-rotor aerial robot. In *Advanced intelligent mechatronics, 2007 IEEE/ASME International Conference on*, pages 1–6. IEEE.
- [Vázquez and Jurado, 2014] Vázquez, L. A. and Jurado, F. (2014). Continuous-time decentralized wavelet neural control for a 2 dof robot manipulator. In *2014 11th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pages 1–6.
- [Wai and Chang, 2004] Wai, R.-J. and Chang, H.-H. (2004). Backstepping wavelet neural network control for indirect field-oriented induction motor drive. *IEEE Transactions on Neural Networks*, 15(2):367–382.

- [Wang et al., 2016] Wang, Y., Zhang, H., and Han, D. (2016). Neural network adaptive inverse model control method for quadrotor uav. In *Control Conference (CCC), 2016 35th Chinese*, pages 3653–3658. IEEE.
- [Yoo et al., 2006] Yoo, S. J., Park, J. D., and Chci, Y. H. (2006). Adaptive dynamic surface control of flexible-joint robots using self-recurrent wavelet neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(6):1342–1355.
- [Zak, 2003] Zak, S. H. (2003). *Systems and control*, volume 174. Oxford University Press. New York.