

# Trabajo Práctico: Helicóptero vertical de un solo eje.

Elías Álvarez

Carrera de Ing. Electrónica

Universidad Católica Nuestra Señora de la Asunción  
Asunción, Paraguay

Email: elias.alvarez@universidadcatolica.edu.py

Tania Romero

Carrera de Ing. Electrónica

Universidad Católica Nuestra Señora de la Asunción  
Asunción, Paraguay

Email: tania.romero@universidadcatolica.edu.py

**Resumen**—En este trabajo práctico se estudia el control de altura de una planta experimental basada en un sistema de propulsión vertical mediante un motor brushless controlado electrónicamente. La planta consiste en un cuerpo móvil guiado mecánicamente en el eje vertical, cuya posición es medida mediante un sensor de distancia láser y regulada a través de una señal PWM aplicada a un controlador electrónico de velocidad (ESC).

A partir del modelado físico y matemático del sistema, se desarrollan e implementan distintas estrategias de control vistas a lo largo de la materia Automatizaciones. Se analizan controladores clásicos, tales como el PID y métodos de diseño basados en el lugar de las raíces, diagramas de Bode, síntesis directa y ubicación arbitraria de polos. Asimismo, se aborda el control en el espacio de estados, incorporando control integral para sistemas de seguimiento y el uso de estimadores de estado tanto de predicción como de actualización.

Finalmente, se diseñan y evalúan controladores óptimos que incluyen integrador y estimadores basados en el filtro de Kalman, culminando en la implementación de un regulador LQG. Los distintos enfoques de control son comparados en términos de desempeño dinámico, estabilidad y robustez frente a perturbaciones y variaciones paramétricas, utilizando resultados obtenidos mediante ensayos experimentales sobre la planta real.

**Index Terms**—control de altura, sistemas dinámicos, identificación de sistemas, control PID, control en espacio de estados, LQG, PSOC, TFMiNi, ESC, PWM

## I. INTRODUCCIÓN

El control de sistemas propulsados verticalmente constituye un problema clásico dentro de la ingeniería de control, particularmente cuando la dinámica presenta inestabilidad inherente, no linealidades y limitaciones físicas severas. En este trabajo se aborda la regulación y seguimiento de referencia de una planta experimental de tercer orden, caracterizada por un comportamiento inestable en lazo abierto, presencia de polos complejos dominantes, un integrador y ceros que afectan significativamente la respuesta dinámica del sistema.

Más allá del modelo teórico, la planta real introduce múltiples no idealidades: saturaciones estrictas del actuador (1100–1700  $\mu$ s por razones de seguridad térmica y estructural), fricción variable debida a imperfecciones mecánicas en los rieles de guiado, variación del punto de operación asociada a la descarga progresiva de la batería, dinámica propia del motor y del ESC, y ruido de medición proveniente del sensor láser utilizado. Estas condiciones convierten el problema en un banco de pruebas exigente para evaluar la robustez y aplicabilidad práctica de distintas estrategias de control.

Un aspecto central del trabajo fue la obtención de un modelo lineal representativo del sistema alrededor del punto de operación de hover, incluyendo la identificación experimental del esfuerzo de equilibrio y la caracterización del ruido de medición y proceso. La calidad de este modelo resultó determinante para el desempeño de los métodos de control modernos, particularmente aquellos basados en espacio de estados y estimación óptima.

Sobre esta misma planta se implementaron y compararon distintas filosofías de diseño, desde enfoques clásicos hasta control óptimo con estimación de estados, evaluando su desempeño no sólo en simulación sino también mediante validación experimental directa. Esta comparación permite analizar en condiciones reales las ventajas, limitaciones y requerimientos de modelado de cada metodología.

El objetivo del trabajo no se limita a la obtención de un regulador funcional, sino a estudiar la relación entre complejidad del método, dependencia del modelo y desempeño obtenido en un sistema físico con restricciones reales.

## II. PLANTEAMIENTO DEL PROBLEMA

Los sistemas de regulación de altura basados en propulsión vertical constituyen un problema particularmente exigente desde el punto de vista del control automático, debido a la combinación de inestabilidad inherente, dinámica no lineal y restricciones físicas severas del actuador. La planta experimental considerada en este trabajo presenta comportamiento inestable en lazo abierto, dinámica de tercer orden con integrador y ceros que condicionan la respuesta transitoria, incluyendo la presencia de un cero en el semiplano derecho que impone limitaciones fundamentales en el desempeño alcanzable.

Desde el punto de vista físico, el sistema consiste en un cuerpo móvil guiado en el eje vertical cuyo equilibrio depende del balance entre la fuerza gravitatoria y el empuje generado por un motor brushless accionado mediante un ESC. Sin embargo, el comportamiento real se ve influenciado por múltiples no idealidades: saturaciones estrictas del esfuerzo de control por razones de seguridad térmica (1100–1700  $\mu$ s), variaciones del punto de operación asociadas a la descarga de la batería, fricción no uniforme en los rieles de guiado, dinámica propia del conjunto motor–ESC y ruido de medición proveniente del sensor láser de distancia.

Se realizaron desarrollos físico-matemáticos iniciales con el objetivo de modelar el sistema a partir de principios dinámicos

fundamentales. No obstante, la complejidad aerodinámica del empuje, junto con la dinámica interna del motor y del ESC, dificultaron la obtención de un modelo analítico suficientemente representativo, lo que motivó la adopción de técnicas de identificación experimental alrededor del punto de operación de hover.

El desafío central radica entonces en obtener un modelo lineal que capture adecuadamente la dinámica dominante del sistema y permita diseñar controladores implementables en tiempo real sobre hardware embebido. El firmware desarrollado integra múltiples estrategias de control dentro del mismo sistema, permitiendo la modificación dinámica de coeficientes, la adquisición de datos en lazo abierto y cerrado, y la transmisión en tiempo real de las señales de esfuerzo y altura mediante comunicación UART para su análisis posterior.

En este contexto, el problema abordado no se limita a la estabilización del sistema, sino que implica analizar la relación entre calidad del modelado, complejidad de la estrategia de control y desempeño experimental obtenido sobre una planta física con restricciones reales.

### III. OBJETIVOS

#### III-A. *Objetivo general*

Desarrollar, analizar e implementar un sistema de control de altura para una planta experimental basada en un motor brushless, aplicando y comparando distintas estrategias de control automático estudiadas en la materia Automatizaciones, a partir del modelado físico y matemático del sistema y su validación experimental sobre la planta real.

#### III-B. *Objetivos específicos*

- Caracterizar físicamente la planta experimental y describir su comportamiento dinámico a partir de sus componentes mecánicos, eléctricos y de sensado.
- Desarrollar un modelo físico y matemático del sistema que represente adecuadamente la dinámica vertical de la planta y sirva como base para el diseño de controladores.
- Identificar experimentalmente los parámetros relevantes del modelo, considerando las no idealidades propias del sistema real.
- Diseñar e implementar controladores clásicos de altura, incluyendo el controlador PID y métodos de diseño basados en el lugar de las raíces, diagramas de Bode, síntesis directa y ubicación arbitraria de polos.
- Diseñar controladores en el espacio de estados para sistemas de seguimiento, incorporando control integral para la eliminación del error estacionario.
- Implementar y evaluar estimadores de estado tanto de tipo predictivo como de actualización, analizando su influencia en el desempeño del sistema.
- Diseñar y aplicar estrategias de control óptimo, incluyendo el uso de integrador y estimadores basados en el filtro de Kalman, culminando en la implementación de un regulador LQG.

- Comparar el desempeño de las distintas estrategias de control implementadas en términos de estabilidad, respuesta transitoria, error en régimen permanente y robustez frente a perturbaciones y variaciones paramétricas.
- Validar experimentalmente los resultados obtenidos mediante ensayos sobre la planta real, contrastando el comportamiento observado con las predicciones del modelo.

## IV. CARACTERIZACIÓN FÍSICA DE LA PLANTA

#### IV-A. *Descripción general de la planta*

La planta desarrollada corresponde a un sistema mecatrónico de **un grado de libertad**, cuyo movimiento está restringido a la **dirección vertical**. El principio de funcionamiento se basa en la generación de empuje aerodinámico mediante un **motor brushless con hélice**, controlado electrónicamente, que permite regular la altura de un cuerpo móvil guiado mecánicamente.

El sistema fue concebido como una planta experimental para el análisis y diseño de estrategias de control en altura, incorporando sensado directo de posición y actuadores eléctricos de rápida respuesta.

#### IV-B. *Estructura física de la planta*

La planta experimental fue diseñada y construida específicamente para el desarrollo de las prácticas de control de altura previstas en el presente trabajo. El diseño de la estructura respondió a la necesidad de disponer de una altura útil suficiente para la correcta evaluación de las distintas estrategias de control, manteniendo al mismo tiempo un esquema constructivo simple, robusto y de fácil implementación. Durante el desarrollo del proyecto se atravesaron distintas etapas de diseño, las cuales se describen con mayor detalle en el Apéndice D.

Inicialmente, la planta fue concebida con una altura total de aproximadamente 80 cm. Sin embargo, dicha dimensión resultó insuficiente para la realización de todas las prácticas propuestas. En consecuencia, y por recomendación del profesor, se decidió extender la estructura hasta alcanzar una altura total de 165 cm, permitiendo una altura útil de movimiento del cuerpo móvil de aproximadamente 134 cm.

La estructura se apoya sobre una base de madera de dimensiones  $50 \times 45$  cm y un espesor aproximado de 2 cm, la cual proporciona estabilidad al conjunto. En la parte superior se dispone de un techo de madera de  $50 \times 50$  cm y un espesor de 0,6 cm, que actúa como elemento de cierre y soporte estructural. La base y el techo se encuentran unidos mediante tres columnas verticales de madera, cuya función principal es aportar rigidez al conjunto y limitar las deformaciones de la estructura.

El **movimiento vertical** del sistema se guía mediante tres rieles metálicos dispuestos en paralelo, construidos a partir de vigas de metal de aproximadamente 0,6 cm de diámetro. Estas vigas presentan deformaciones inherentes al material y a su longitud, por lo que las columnas de madera cumplen un rol fundamental en evitar que dichas deformaciones se acentúen durante el funcionamiento del sistema.

El **cuerpo móvil** se desplaza a lo largo de los rieles mediante piezas impresas en 3D de tipo abrazadera. Estas abrazaderas poseen un diámetro aproximado de 1,5 cm, superior al de las



Figura 1. Vista general de la planta física.

vigas, con el objetivo de permitir cierto grado de libertad angular y evitar perturbaciones en el movimiento vertical causadas por las deformaciones de los rieles. Las abrazaderas se unen al cuerpo móvil mediante un sistema de tornillos, arandelas y tuercas, funcionando como articulaciones tipo “muñeca”, que facilitan el guiado sin generar atascamientos.



Figura 2. Diseño 3D final de la abrazadera con agarre tipo “muñeca”.

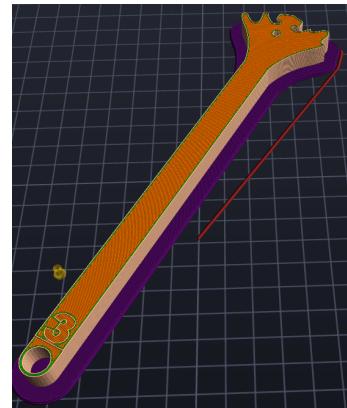


Figura 3. Diseño 3D final del brazo de unión del cuerpo móvil.

El cuerpo móvil está compuesto por dos **soportes principales**: un soporte superior que constituye la base de montaje del motor brushless y un soporte inferior que sostiene el conjunto estructural. Ambos soportes se encuentran unidos mediante tres brazos impresos en 3D, de aproximadamente 1 cm de altura, 1,3 cm de ancho y 23 cm de longitud. La unión de los distintos componentes del cuerpo móvil se realiza mediante roscas de aproximadamente 3 cm de diámetro y tuercas, evitando el uso de tornillería adicional. El conjunto incluye además un soporte para la batería, en cuya base se encuentra montado el ESC, equipado con dos disipadores térmicos laterales. La masa total del cuerpo móvil es de aproximadamente 360 g.

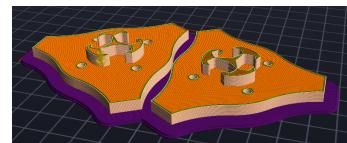


Figura 4. Diseño 3D final del soporte principal del cuerpo móvil (soporte superior e inferior).

Con el fin de proteger el sistema ante caídas y movimientos bruscos, se incorporaron **elementos de seguridad adicionales**. En la parte inferior de la estructura se dispuso papel burbuja enrollado alrededor de los rieles, actuando como amortiguación ante impactos. Asimismo, se añadieron tubos de PVC tanto en la base como en la parte superior de la estructura, los cuales funcionan como topes mecánicos que limitan el recorrido del cuerpo móvil y evitan colisiones con la base o el techo. Adicionalmente, se incorporó una cuerda de seguridad destinada a restringir levantamientos excesivos durante las prácticas, reduciendo el riesgo de movimientos abruptos.

Finalmente, se añadió una cinta métrica a lo largo de la estructura con el objetivo de **facilitar la visualización** del desplazamiento vertical y permitir una referencia directa de la altura durante el funcionamiento del sistema.

El diseño de la estructura **priorizó** la rigidez del cuerpo móvil, de modo que pueda soportar tanto su propio peso como eventuales caídas desde alturas cercanas a un metro. Asimismo, se buscó una solución de fácil construcción, utilizando materia-

les de adquisición accesible en el contexto local y adaptándose a los recursos disponibles y al tiempo de desarrollo del trabajo. La visibilidad del movimiento y el correcto funcionamiento mecánico del sistema fueron considerados aspectos clave para su utilización como planta experimental en este trabajo práctico.

#### IV-C. Cuerpo móvil

- **Masa total móvil:**

$$m = 0.360 \text{ kg}$$

(incluye motor, hélice, soporte, cableado y elementos solidos al movimiento).

- **Tipo de movimiento:** traslación puramente vertical.
- **Altura inicial típica:** aproximadamente 12.5 cm.
- **Altura máxima disponible en la estructura:** aproximadamente 134 cm.

La ausencia de contrapesos implica que el sistema depende exclusivamente del empuje generado por la hélice para vencer la fuerza gravitatoria y los efectos de rozamiento.

#### IV-D. Sistema de actuación (propulsión)

El sistema de actuación está compuesto por:

- **Motor:** brushless A2212/5T, 2450 KV [1].



Figura 5. Motor

- **Hélice:**

- Diámetro: 25 cm.



Figura 6. Hélice

- **Controlador electrónico (ESC) [2]:**

- Corriente continua: 40 A.
- Corriente máxima de corta duración: 55 A.

- **Batería:** LiPo 3S, [3].



Figura 7. ESC 40A

- Tensión inicial típica: 12.5 V.
- La tensión disminuye de forma apreciable durante la operación, dependiendo del régimen de empuje y la duración de la práctica.



Figura 8. Bateria LiPo 3S Ovonic

El empuje generado por el sistema depende fuertemente del comando aplicado, de la hélice y del voltaje instantáneo de la batería, lo que introduce una **no linealidad significativa** en la planta.

#### IV-E. Señal de control

- **Tipo de señal:** PWM tipo servo.
- **Frecuencia:** 50 Hz.
- **Rango:**  $1000 \mu\text{s} - 2000 \mu\text{s}$ .

Esta señal actúa como la **entrada manipulada** del sistema, regulando indirectamente el empuje generado por el motor y la hélice a través del ESC.

#### IV-F. Sistema de sensado

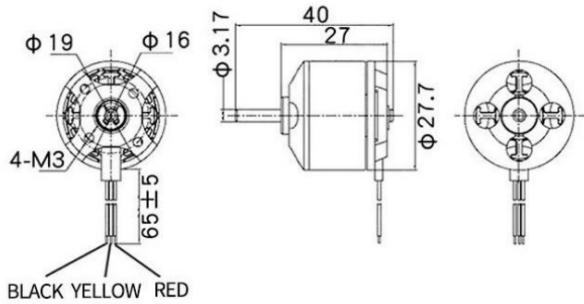
- **Sensor de altura:** TFMini Plus [4].
- **Variable medida:** posición vertical del cuerpo móvil  $z(t)$ .
- **Frecuencia de lectura:** configurable, típicamente en el rango de 1 Hz hasta 1000 Hz, según la configuración utilizada durante las distintas prácticas.

La señal de medición presenta efectos de **ruido, cuantización y latencia**, propios del sistema de sensado y del procesamiento digital, los cuales deben ser considerados tanto en el diseño del sistema de control como en el tratamiento de la señal medida.

#### IV-G. Variables del sistema

- **Entrada del sistema:**

$$u(t) = \text{PWM} \in [1000, 2000] \mu\text{s}$$



MOTOR PERFORMANCE DATA (性能参数) :

MODEL	KV (rpm/V)	Voltage (V)	Prop	Load Current (A)	Pull (g)	Power (W)	Efficiency (g/W)	Lipo Cell	Weight (g) Approx
	930	1060		9.8	660	109	6.1		60
	1000	1047		15.6	885	173	5.1	3S	61
A2212	1400	11.1	9050	19.0	910	210	4.3		61
	2200	6030		21.5	732	239	3.1	2-3S	62
	2450	6.3		25.2	815	280	2.9		62

Figura 9. Características del motor A2212/5T, 2450KV [1]



Figura 10. Sensor óptico de distancia TFMini Plus.

#### ■ Salida del sistema:

$$y(t) = z(t)$$

#### ■ Disturbios principales:

- Variaciones del voltaje de la batería durante la operación.
- Rozamiento mecánico en las guías.
- Perturbaciones aerodinámicas externas.
- Vibraciones estructurales.

#### IV-H. Limitaciones físicas y no idealidades

La planta presenta las siguientes características no ideales:

- **Saturación del actuador**, limitada por el rango de PWM y la corriente máxima del ESC.
- **Dinámica no instantánea del empuje**, asociada a la respuesta del ESC, del motor y de la hélice.
- **Variabilidad paramétrica**, principalmente debida a la caída de tensión de la batería bajo carga.

#### V. DIAGRAMA DEL SISTEMA

La Figura 11 muestra el diagrama de bloques del sistema de control de altura en lazo cerrado. La referencia de altura  $z_{ref}$  es comparada con la señal de salida medida  $z(t)$ , generando el error  $e(t)$ , el cual es procesado por el controlador.

El controlador representa el algoritmo de control implementado, pudiendo corresponder a un controlador estudiado en el semestre. A partir del error, el controlador genera la señal de control  $u(t)$ , que constituye el esfuerzo aplicado al sistema.

La señal  $u(t)$  actúa sobre el actuador, conformado por el motor brushless y la hélice, el cual convierte la señal de control en empuje mecánico. Dicho empuje excita la planta física, cuya dinámica vertical determina la altura real del sistema  $y(t)$ .

La altura es medida mediante un sensor TFMini Plus, que proporciona la señal  $z(t)$  utilizada para cerrar el lazo de control. En el diagrama se indican explícitamente las fuentes de ruido asociadas tanto al controlador como al sensor, reflejando las perturbaciones y no idealidades presentes en el sistema real.

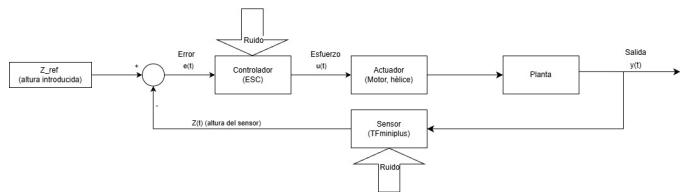


Figura 11. Diagrama de bloques de Control.

La Figura 12 muestra un diagrama de bloques que representa la implementación física y funcional completa del sistema de control de altura. En dicho diagrama se identifican claramente los siguientes subsistemas:

- la cadena de energía (batería → ESC → motor),
- la cadena de control (MATLAB → PSOC → PWM),
- la planta física correspondiente al sistema bajo control,
- y la cadena de medición y realimentación (planta → sensor → PSOC → MATLAB).

#### V-A. Flujo de energía (parte electro-energética)

La batería utilizada es una LiPo de tres celdas (3S), con tensión nominal de 11,1 V. Cada celda presenta una tensión máxima de 4,2 V, por lo que la batería completamente cargada alcanza:

$$V_{max} = 12,6 \text{ V}$$

Con el objetivo de preservar la integridad química de la batería y evitar degradación prematura, el rango operativo se restringe aproximadamente a:

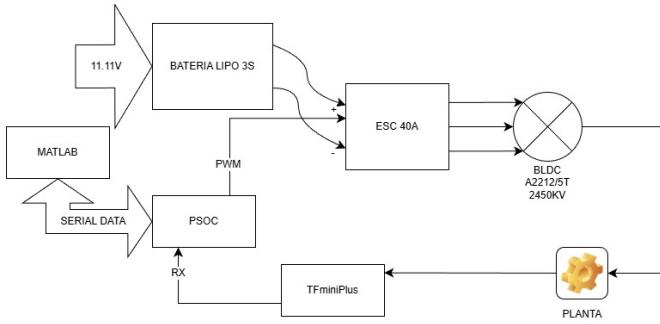


Figura 12. Diagrama de bloques, implementación física.

$$V \in [11.5, 12.5] \text{ V}$$

Evitar descargas por debajo de 11,5 V resulta fundamental, ya que tensiones inferiores pueden producir daño irreversible en las celdas LiPo.

La batería alimenta directamente al controlador electrónico de velocidad (ESC) de 40 A, el cual convierte la tensión continua en señales trifásicas moduladas para accionar el motor brushless.

El ESC incorpora disipadores térmicos integrados sobre los dispositivos de potencia (MOSFETs), cuya función es evacuar el calor generado durante la commutación y conducción de corriente. Dado que el sistema opera en un régimen de potencia relativamente elevado para el tamaño del conjunto, la gestión térmica resulta crítica para evitar sobrecalentamientos que puedan producir fallas o reducción de eficiencia.

La presencia de disipadores mejora la transferencia térmica hacia el ambiente, aumentando la confiabilidad del sistema durante los intervalos de operación de aproximadamente 5 min continuos.

A medida que la batería se descarga, se observa una disminución progresiva de la tensión disponible, lo que impacta directamente en la capacidad de generación de empuje del motor. Este fenómeno es perceptible incluso en aplicaciones aeronáuticas (como drones), donde la pérdida de tensión se traduce en menor capacidad de sustentación.

En la práctica experimental, el tiempo de operación continua es del orden de 5 min como máximo. Esto se debe a que la batería utilizada posee una capacidad limitada para el nivel de potencia demandado por el sistema.

El tiempo típico de recarga completa es aproximadamente 1,5 h, lo que introduce una restricción significativa en la repetibilidad de los ensayos experimentales, constituyendo un factor limitante en el desarrollo de la práctica.

En esta etapa del sistema no se realiza acción de control propiamente dicha, sino únicamente conversión y transferencia de potencia hacia el actuador.

#### V-B. Flujo de control (parte de mando)

El PSoC actúa como el controlador embebido del sistema. A partir de los algoritmos de control implementados (PID, control

en espacio de estados, LQG, entre otros), el PSoC genera una señal PWM de tipo servo que es enviada al ESC.

Esta señal PWM representa la variable de control  $u(t)$  del sistema y determina el nivel de empuje aplicado al motor. Para asegurar una correcta referencia eléctrica y el funcionamiento adecuado del sistema, el PSoC y el ESC comparten una conexión de tierra común (GND).

El sistema MATLAB se comunica con el PSoC mediante una interfaz de datos seriales, lo que permite:

- enviar referencias de altura,
- modificar parámetros de control,
- recibir y visualizar datos del sistema en tiempo real.

En este esquema, MATLAB cumple una función de supervisión, ajuste y análisis experimental, mientras que el PSoC ejecuta el control en tiempo real.

#### V-C. Medición y realimentación (cierre del lazo)

La planta física, al desplazarse verticalmente, genera una altura real que es medida mediante el sensor de distancia láser TFmini Plus. Este sensor entrega la medición de altura al PSoC a través de su interfaz de recepción (RX).

La señal medida es utilizada por el PSoC para calcular el error entre la referencia y la salida real del sistema, cerrando de esta manera el lazo de control de altura. Adicionalmente, los datos medidos pueden ser enviados a MATLAB para su visualización, almacenamiento y análisis experimental.

## VI. MODELADO FÍSICO DEL SISTEMA

#### VI-A. Variables y convenciones

Se definen a continuación las variables y convenciones utilizadas para el modelado del sistema:

- Eje vertical  $z$  [m], definido positivo hacia arriba.
- Masa móvil:

$$m = 0.360 \text{ kg}$$

- Aceleración de la gravedad:

$$g = 9.81 \text{ m/s}^2$$

- Peso del cuerpo móvil:

$$mg = 3.924 \text{ N}$$

- Entrada del sistema: señal PWM tipo servo a 50 Hz,

$$u \in [1000, 2000] \mu\text{s}$$

Si bien el rango eléctrico nominal del protocolo PWM se encuentra entre 1000 y 2000  $\mu\text{s}$ , durante la operación en vuelo la señal se restringe intencionalmente al intervalo

$$u \in [1100, 1700] \mu\text{s}$$

Esta limitación surge de criterios de confiabilidad experimental. En ensayos previos realizados con ESCs de menor capacidad nominal (30 A), se observaron fallas térmicas al operar en valores superiores a aproximadamente 1600  $\mu\text{s}$  bajo carga sostenida.

Dado que no se realizaron mediciones directas de corriente ni caracterizaciones térmicas detalladas del conjunto motor-ESC-hélice, se adoptó un margen de seguridad conservador que evita la operación prolongada en regímenes de alta demanda energética.

Asimismo, el límite inferior de  $1100 \mu s$  se fija con el objetivo de evitar regiones cercanas a la detención del motor, donde pueden presentarse comportamientos fuertemente no lineales y pérdida abrupta de sustentación.

En consecuencia, el modelo identificado y las estrategias de control desarrolladas se consideran válidos únicamente dentro de este rango operativo seguro.

- Normalización de la entrada:

$$\hat{u} = \text{sat}(u - u_0, 1100 - u_0, 1700 - u_0)$$

Siendo  $u_0$  el valor de PWM necesario para generar un empuje equivalente al peso del sistema, es decir, la condición de equilibrio vertical estacionario.

- Salida medida:

$$y = z$$

correspondiente a la altura del cuerpo móvil medida mediante el sensor TFMini Plus, cuya resolución efectiva se encuentra en el orden de centímetros.

*VI-A1. Modelo físico simplificado:* Desde un punto de vista físico, el movimiento vertical del cuerpo móvil puede describirse, en primera aproximación, mediante:

$$\dot{z} = v$$

$$m\dot{v} = T - mg$$

Este modelo corresponde al caso ideal sin disipación, en el cual la dinámica posición–empuje presenta una estructura de doble integración.

En una aproximación más realista, puede incorporarse el efecto de rozamiento de las guías mediante un término viscoso proporcional a la velocidad:

$$m\dot{v} = T - mg - bv$$

donde  $b$  [N s/m] representa un coeficiente equivalente de fricción viscosa.

La inclusión de este término modifica la estructura ideal de doble integrador, convirtiéndola en un sistema con integración amortiguada. Desde el punto de vista de función de transferencia continua, el doble polo en el origen deja de ser estrictamente doble, introduciéndose un amortiguamiento mecánico que desplaza uno de los polos hacia el semiplano izquierdo.

Por lo tanto, la presencia de un único integrador dominante en el modelo simplificado resulta coherente con la física del sistema cuando se consideran pérdidas mecánicas.

*VI-A2. Modelo dinámico equivalente del actuador (BLDC):* A efectos de análisis lineal, el conjunto ESC–motor BLDC puede aproximarse mediante un modelo promedio equivalente al de un motor DC de imanes permanentes:

$$v(t) = R i(t) + L \dot{i}(t) + K_e \omega(t)$$

$$J \dot{\omega}(t) = K_t i(t) - B \omega(t) - \tau_L(t)$$

Este modelo introduce una dinámica electromecánica adicional respecto del modelo puramente mecánico de la masa móvil, justificando la aparición de un polo adicional en la relación entrada–salida.

En muchos casos, la constante de tiempo eléctrica  $\tau_e = L/R$  es considerablemente menor que la mecánica, permitiendo despreciar  $L$  y obtener una dinámica dominante de segundo orden asociada al actuador.

*VI-A3. Función de transferencia continua equivalente:* La representación continua obtenida a partir del modelo identificado puede expresarse inicialmente como:

$$G(s) = \frac{-0.12107(s - 14)(s + 10.62)}{(s + 0.0002797)(s^2 + 5.61s + 14.02)}$$

El polo ubicado en  $s = -0.0002797$  corresponde a una dinámica extremadamente lenta respecto de las restantes constantes de tiempo del sistema.

Considerando que:

- la resolución del sensor de altura se encuentra en el orden de centímetros,
- no se dispone de mediciones de alta precisión submilimétrica,
- el identificador polinomial puede ajustar polos muy cercanos al origen para capturar pequeñas tendencias de deriva,

se interpreta que dicho polo próximo a cero no representa una dinámica física real dominante, sino un posible efecto de sobreajuste (*overfitting*) del procedimiento de identificación.

En consecuencia, y en coherencia con el modelo físico que contempla amortiguamiento mecánico, se adopta la siguiente simplificación:

$$G(s) = \frac{-0.12107(s - 14)(s + 10.62)}{s(s^2 + 5.61s + 14.02)}$$

En esta forma:

- El polo en el origen representa el carácter integrador dominante de la posición vertical.
- El segundo orden  $s^2 + 5.61s + 14.02$  modela la dinámica electromecánica del actuador.
- Los ceros se interpretan como parámetros de ajuste que capturan efectos agregados del actuador, discretización y linealización alrededor del punto de operación.

**VI-A4. Obtención del modelo mediante identificación en MATLAB:** La función de transferencia utilizada en este trabajo no se obtuvo exclusivamente a partir de un modelo físico teórico, sino mediante un proceso de **identificación experimental**.

En particular, se empleó la herramienta *System Identification Toolbox* de MATLAB, utilizando registros experimentales previamente adquiridos del sistema en lazo abierto (señal de mando PWM y altura medida por el sensor). Dichos datos fueron importados al entorno de identificación y organizados en un objeto *iddata*, definiendo la entrada como  $u(t)$  y la salida como  $y(t) = z(t)$ , con el tiempo de muestreo correspondiente al sistema de adquisición.

Sobre ese conjunto de datos se ajustó un modelo lineal SISO en tiempo continuo, seleccionando una estructura paramétrica compatible con la dinámica esperada del sistema, y estimando sus parámetros mediante los algoritmos de optimización incluidos en MATLAB.

Finalmente, el modelo resultante fue validado comparando su respuesta con los datos medidos (ajuste temporal y análisis de residuales), y se adoptó como representación equivalente continua la función de transferencia  $G(s)$  presentada en la sección siguiente.

**VI-A5. Validez del modelo:** El modelo adoptado constituye una aproximación coherente con la física del sistema y adecuada para el diseño de control dentro del rango operativo  $[1100, 1700] \mu s$ .

Fuera de dicho intervalo, el sistema presenta comportamientos no lineales significativos (aerodinámica, fricción no lineal, saturaciones y posibles limitaciones térmicas del actuador) que no son capturados por el modelo lineal simplificado.

## VII. MÉTODOS CLÁSICOS DE CONTROL

### VII-A. Introducción

Los métodos clásicos de control se fundamentan en el análisis de sistemas lineales mediante funciones de transferencia y herramientas del dominio de la frecuencia y del plano complejo.

En este enfoque, la dinámica del sistema se describe como:

$$G(s) = \frac{Y(s)}{U(s)} \quad \text{o en tiempo discreto} \quad G(z) = \frac{Y(z)}{U(z)}$$

El diseño del controlador se realiza modificando la función de transferencia del lazo abierto:

$$L(s) = C(s)G(s) \quad \text{o} \quad L(z) = C(z)G(z)$$

con el objetivo de garantizar:

- Estabilidad en lazo cerrado.
- Desempeño transitorio adecuado.
- Error estacionario reducido.
- Robustez frente a incertidumbre.

Los métodos clásicos utilizados en este trabajo incluyen:

- Controlador PID.
- Diseño por Lugar de Raíces.

- Diseño mediante Respuesta en Frecuencia (Bode).
- Síntesis Directa (Truxal-Ragazzini).

### VII-B. Modelo de la Planta

El diseño clásico parte de la función de transferencia identificada:

$$G(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}}$$

Tiempo de muestreo, depende de la experiencia.

### VII-C. Fundamentos Matemáticos del Control Clásico

**VII-C1. Estabilidad en Lazo Cerrado:** El sistema en lazo cerrado se expresa como:

$$G_{cl}(z) = \frac{C(z)G(z)}{1 + C(z)G(z)}$$

La estabilidad discreta requiere que:

$$|z_i| < 1 \quad \forall i$$

donde  $z_i$  son los polos de  $G_{cl}(z)$ .

**VII-C2. Lugar de Raíces:** Los polos del lazo cerrado satisfacen:

$$1 + C(z)G(z) = 0$$

El Lugar de Raíces describe la trayectoria de estos polos al variar la ganancia del controlador.

**VII-C3. Márgenes de Estabilidad:** A partir del diagrama de Bode del lazo abierto  $L(z)$  se definen:

- Margen de ganancia.
- Margen de fase.
- Frecuencia de cruce de ganancia.

Estos parámetros cuantifican la estabilidad relativa y robustez del sistema.

### VII-D. Controlador PID

Durante las etapas iniciales de diseño se intentó sintonizar un controlador PID utilizando la herramienta PID Tuner de MATLAB. Sin embargo, los resultados obtenidos no fueron satisfactorios para la planta bajo estudio, principalmente debido a la complejidad de la dinámica identificada, la presencia de retardos efectivos asociados al sistema de actuación (ESC-motor) y las limitaciones físicas del actuador. En particular, el desempeño obtenido presentaba respuestas lentas o esfuerzos de control excesivos, incompatibles con la implementación experimental.

Ante esta situación, se optó por utilizar una formulación de controlador PID basada en el método propuesto por Åström, el cual permite un mayor control sobre la estructura del controlador y sobre el compromiso entre rapidez, amortiguamiento y esfuerzo de control. Este enfoque resultó más adecuado para la planta identificada y permitió obtener respuestas dinámicas satisfactorias en simulación y en la práctica.

**VII-D1. Formulación del PID de Åström:** La estructura del controlador PID de Åström se implementa en forma discreta y separa explícitamente las acciones proporcional, integral y derivativa. El término proporcional se define como:

$$P(t) = K(b u_c(t) - y(t))$$

donde  $K$  es la ganancia proporcional y  $b$  es un parámetro que determina qué fracción de la referencia se introduce en la acción proporcional, permitiendo reducir el sobreimpulso ante cambios bruscos de referencia.

La acción derivativa se implementa mediante un filtro de primer orden, cuya ecuación en tiempo discreto es:

$$D(kh) = \frac{T_d}{T_d + Nh} D(kh-h) - \frac{KT_d N}{T_d + Nh} (y(kh) - y(kh-h))$$

donde  $T_d$  es la constante de tiempo derivativa,  $N$  es el parámetro que limita el ancho de banda del término derivativo, y  $h$  es el período de muestreo.

La acción integral se describe mediante:

$$I(kh + h) = I(kh) + \frac{K_h}{T_i} e(kh)$$

donde  $T_i$  es la constante de tiempo integral y  $e(kh) = u_c(kh) - y(kh)$  es el error de control.

La señal de control total se obtiene como la suma de los tres términos:

$$u(kh) = P(kh) + I(kh) + D(kh)$$

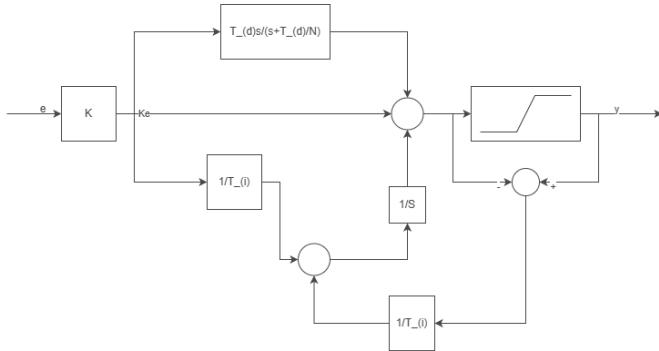


Figura 13. Diagrama de bloques, Astrom - Antiwindup [5]. Aplicada a la planta.

**VII-D2. Criterios de sintonización:** La sintonización de los parámetros del controlador se realizó de forma iterativa, utilizando simulaciones y observando tanto la respuesta del sistema como el esfuerzo de control. Los criterios adoptados fueron los siguientes:

- **Ganancia proporcional  $K$ :** se incrementó progresivamente hasta aproximar el sistema al umbral de inestabilidad, con el objetivo de obtener un transitorio rápido y una respuesta ágil.
- **Acción integral  $T_i$ :** se incorporó posteriormente para eliminar el error en régimen permanente, ajustando su valor de manera que el tiempo de establecimiento resultara razonable sin introducir oscilaciones significativas.

- **Acción derivativa  $T_d$ :** se añadió con el fin de reducir el sobreimpulso y mejorar el amortiguamiento del sistema.
- **Parámetro  $N$ :** se ajustó para limitar la amplificación de ruido del término derivativo, probando distintos valores hasta observar una señal excesivamente sensible al ruido de medición.

Durante todo el proceso de sintonización se monitoreó cuidadosamente el esfuerzo de control. Como condición de diseño, se impuso que la variación de la señal PWM no superara aproximadamente  $10\mu\text{s}$  por centímetro de incremento en la altura, garantizando de esta manera que el actuador no entrara en saturación ni se expusiera la planta a condiciones potencialmente dañinas.

**VII-D3. Resultados:** El controlador PID basado en el método de Åström permitió obtener un comportamiento dinámico estable, con un compromiso adecuado entre rapidez de respuesta, amortiguamiento y esfuerzo de control. En comparación con los resultados obtenidos mediante el PID Tuner, esta metodología ofreció mayor flexibilidad y un mejor ajuste a las particularidades de la planta identificada, resultando adecuada para su implementación experimental (Figuras 14 y 15).

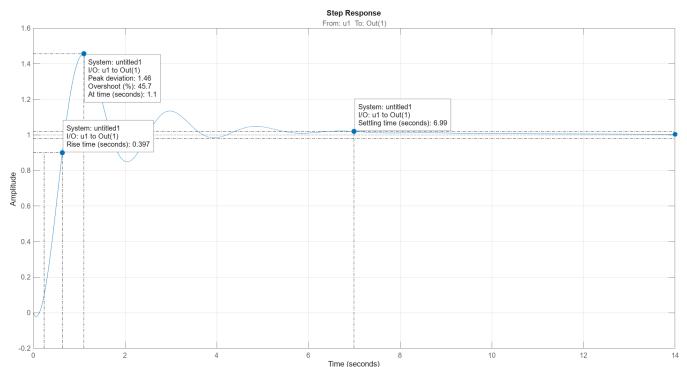


Figura 14. La respuesta al escalón con el controlador PID.

#### VII-D4. Práctica:

$$K_p = [2.5], \quad T_i = [5], \quad T_d = [0.1], \quad N = [3]$$

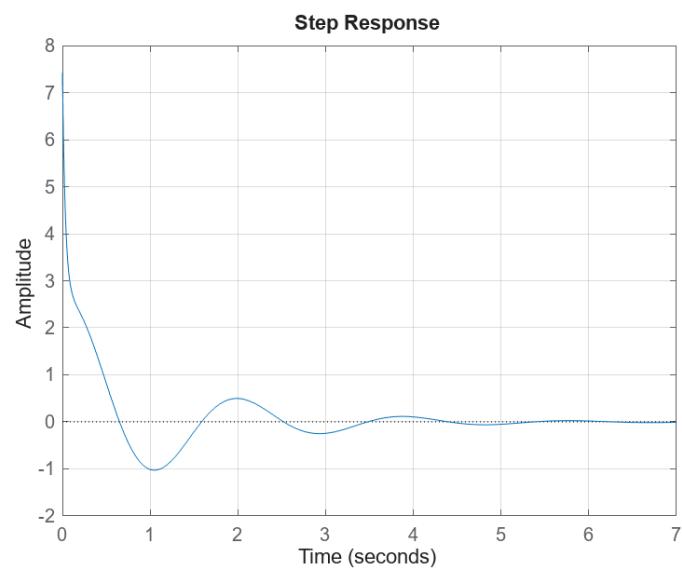


Figura 15. El esfuerzo del con el controlador PID.

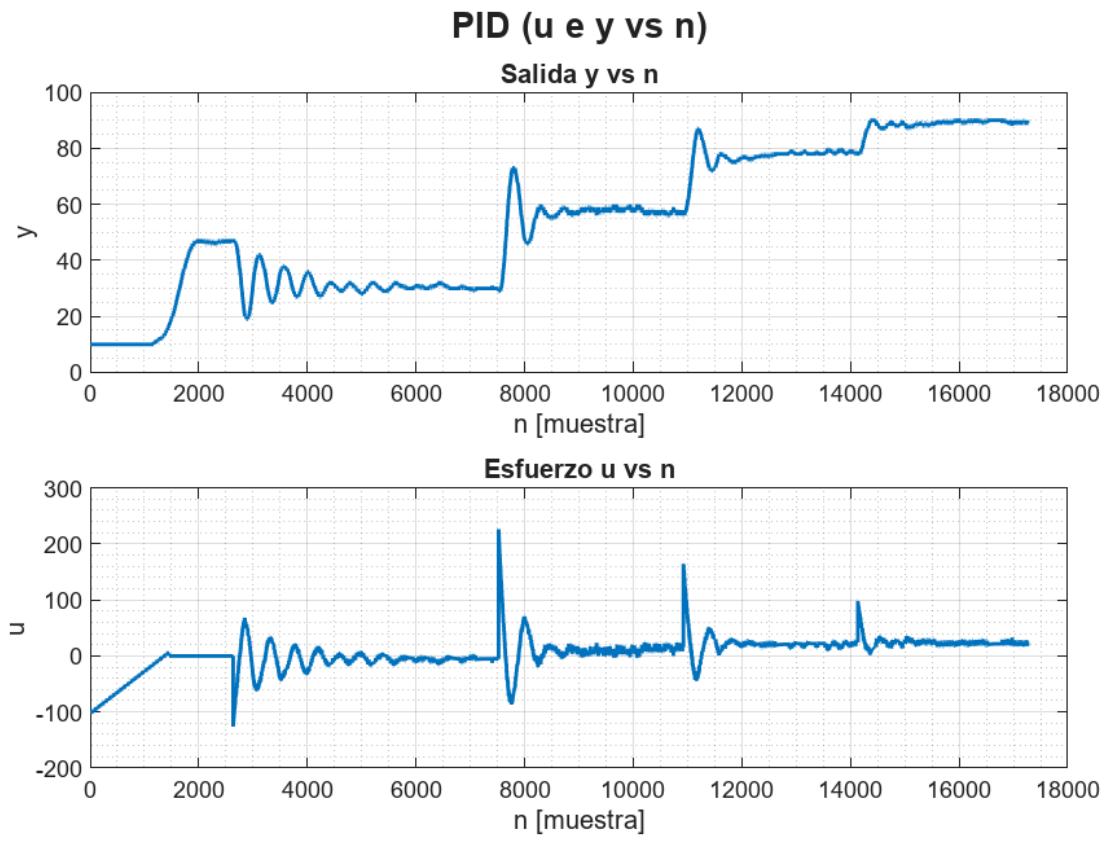


Figura 16. Implementación práctica del PID.

## VII-E. Diseño por Lugar de Raíces

Se analiza la ecuación característica:

$$1 + C(z)G(z) = 0$$

Dado que la planta identificada (ecuación ??) presenta polos ubicados fuera del círculo unitario, el sistema en lazo abierto resulta inestable en el dominio discreto. En consecuencia, cualquier diseño de control debe garantizar que los polos del lazo cerrado queden estrictamente dentro del círculo unitario para asegurar estabilidad interna.



Figura 17. Ubicación de los polos de la planta sin compensar en el plano  $z$ . Se observa que al menos uno de ellos se encuentra fuera del círculo unitario, lo que implica inestabilidad discreta.

El objetivo del diseño consistió en modificar la dinámica del sistema mediante compensación, de modo que:

- todos los polos del lazo cerrado queden dentro del círculo unitario,
- se logre un compromiso adecuado entre rapidez de respuesta y amortiguamiento,
- el esfuerzo de control permanezca dentro de límites físicamente realizable.

### VII-E1. Elección de la estructura del compensador:

Para estabilizar el sistema se adoptó una estructura de tipo **lag-lead** (atraso–adelanto). Esta configuración permite actuar simultáneamente sobre la estabilidad relativa y el desempeño en régimen permanente.

El término *lead* (adelanto) se empleó para aumentar el margen de fase y desplazar los polos dominantes del lazo cerrado hacia regiones del plano  $z$  asociadas con mayor amortiguamiento y mejor desempeño transitorio. Por otro lado, el término *lag* (atraso) permitió ajustar la ganancia en bajas frecuencias, mejorando el comportamiento estacionario sin comprometer significativamente la estabilidad.

La adecuada ubicación de ceros permitió modificar la geometría del lugar de raíces, atrayendo las ramas hacia

la región estable del plano discreto, mientras que los polos adicionales modelaron el compromiso dinámico requerido.

**VII-E2. Determinación de la ganancia  $K$ :** Una vez definida la estructura del compensador, se analizó el lugar de raíces del sistema compensado. La determinación manual de la ganancia  $K$  resultó particularmente sensible, ya que pequeños incrementos en su valor provocaban que las trayectorias de los polos abandonaran el círculo unitario antes de satisfacer las especificaciones dinámicas deseadas.

Esta sensibilidad está directamente relacionada con la naturaleza inestable de la planta y con la fuerte dependencia de la ubicación de los polos del lazo cerrado respecto a la ganancia del compensador.

El compensador finalmente adoptado fue:

$$C(z) = -0.0173 \frac{(z - 1.0140)(z - 0.5)}{(z - 0.9522)(z - 0.9894)} \quad (1)$$

**VII-E3. Ajuste mediante Optimization-Based Tuning:** Con el fin de sistematizar el proceso de ajuste y garantizar el cumplimiento simultáneo de múltiples especificaciones (estabilidad, rapidez y esfuerzo de control), se utilizó la herramienta Optimization-Based Tuning de MATLAB.

Este enfoque permitió:

- definir directamente especificaciones temporales (tiempo de establecimiento, sobreimpulso, etc.),
- ajustar automáticamente los parámetros del compensador,
- verificar la estabilidad del sistema en el dominio discreto.

El resultado fue un compensador lag–lead cuyos parámetros fueron obtenidos mediante optimización numérica, asegurando que los polos del lazo cerrado se mantengan dentro del círculo unitario y que el desempeño temporal cumpla con los objetivos establecidos para la planta experimental.

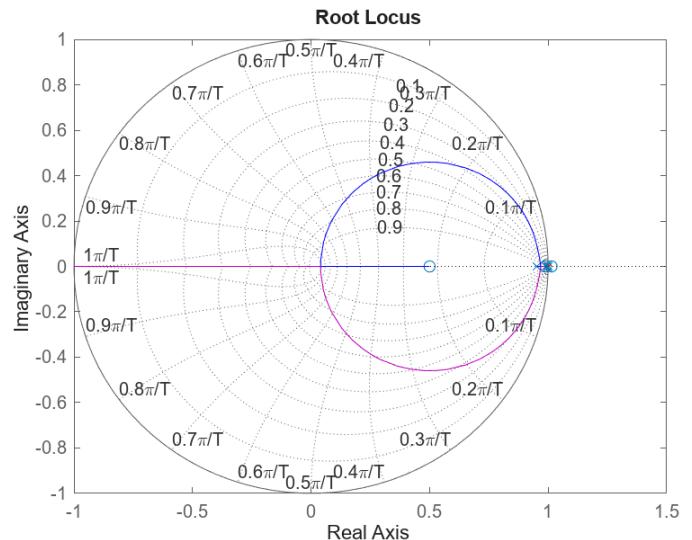


Figura 18. Ubicación de los polos del sistema compensado en el plano  $z$ . Se verifica que todos ellos se encuentran dentro del círculo unitario, garantizando estabilidad discreta.

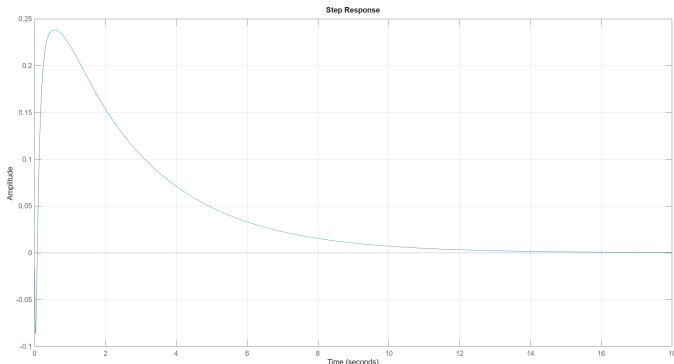


Figura 19. Esfuerzo de control en lazo cerrado con el compensador diseñado mediante Lugar de Raíces.

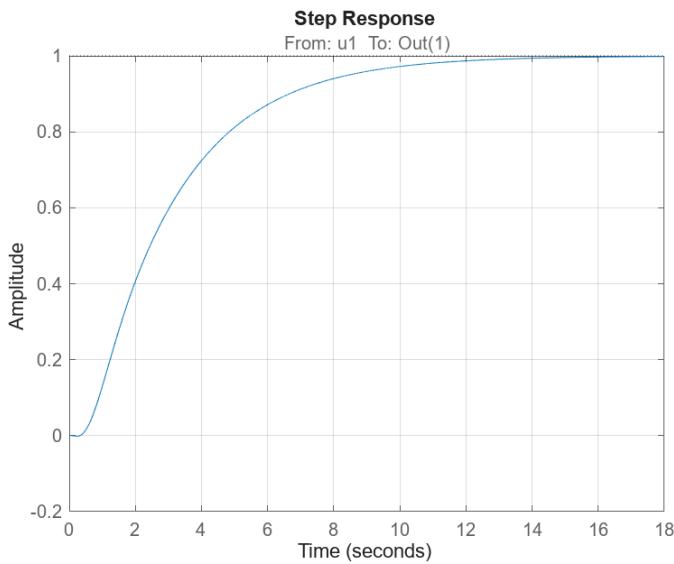


Figura 20. Respuesta temporal del sistema en lazo cerrado con el compensador diseñado.

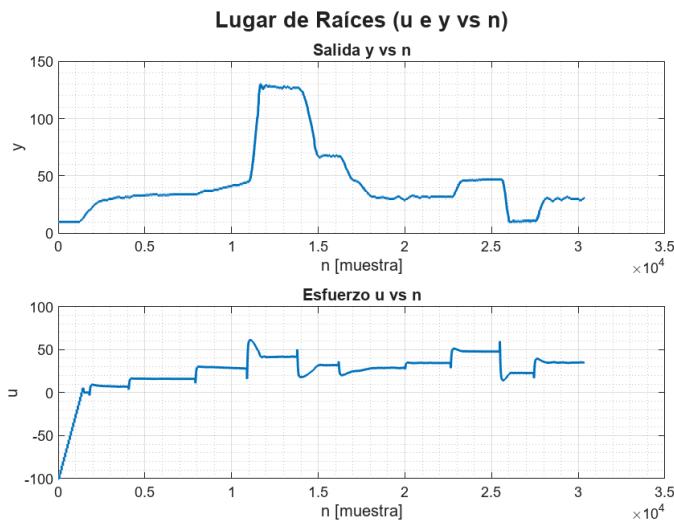


Figura 21. Esfuerzo y altura de la implementación práctica de lugar de raíces

#### VII-E4. Práctica: Objetivos:

- Ubicar polos dentro del círculo unitario.
- Controlar amortiguamiento.
- Ajustar velocidad de respuesta.

## VII-F. Diseño por Respuesta en Frecuencia

Se analiza el lazo abierto:

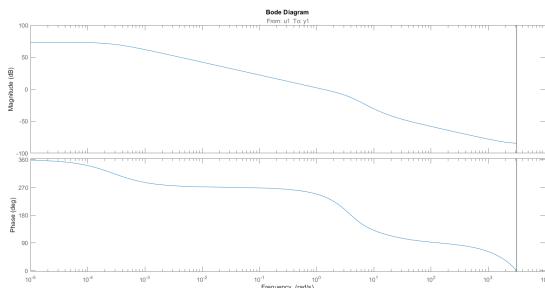


Figura 22. Respuesta en frecuencia del lazo abierto del sistema identificado sin compensación.

Para el diseño del controlador basado en el método de respuesta en frecuencia se utilizó directamente el modelo discreto de la planta obtenido mediante identificación experimental, mostrado en la Fig. 22. En particular, se trabajó con la función de transferencia discreta  $G(z)$  dada en (??), la cual representa el comportamiento dinámico del sistema en el rango de operación considerado.

El modelo identificado fue introducido en el entorno de diseño de MATLAB mediante la herramienta `controlSystemDesigner`, lo que permitió analizar la respuesta en frecuencia del sistema y realizar el diseño del controlador de manera interactiva. A partir de esta herramienta se obtuvieron los diagramas de Bode del lazo abierto, así como los márgenes de ganancia y de fase asociados.

**VII-F1. Análisis del sistema sin compensar:** En la Fig. 23, correspondiente al sistema sin compensación, se observa el diagrama de Bode del lazo abierto conformado únicamente por la planta identificada. A partir de dicho análisis se determinan los márgenes de estabilidad iniciales, los cuales permiten evaluar la estabilidad relativa y la robustez del sistema frente a variaciones paramétricas.

El sistema presenta un margen de fase positivo, lo que indica estabilidad en lazo cerrado para valores reducidos de ganancia, aunque con un compromiso limitado en términos de rapidez de respuesta y amortiguamiento. Asimismo, la pendiente del módulo en la región de cruce de ganancia evidencia la presencia de múltiples polos dominantes, coherentes con la dinámica de orden superior identificada y asociada al conjunto motor-ESC-hélice.

**VII-F2. Diseño del compensador:** Con base en el análisis previo, se procedió al diseño de un compensador con el objetivo de mejorar el desempeño dinámico del sistema, manteniendo márgenes de estabilidad adecuados. El diseño se realizó ajustando la estructura y los parámetros del controlador directamente sobre el diagrama de Bode del lazo abierto, utilizando la herramienta gráfica provista por MATLAB.

Los criterios de diseño considerados fueron:

- incrementar el margen de fase para mejorar el amortiguamiento del sistema,

- fijar una frecuencia de cruce que permita un compromiso adecuado entre rapidez de respuesta y robustez,
- limitar la amplificación de ruido a altas frecuencias y el esfuerzo de control.

**VII-F3. Análisis del sistema compensado:** La Fig. 24 muestra el diagrama de Bode del lazo abierto una vez incorporado el controlador diseñado. Se observa un aumento del margen de fase y un ajuste controlado del margen de ganancia, lo cual indica una mejora en la estabilidad relativa del sistema.

Los valores obtenidos a partir de los diagramas de Bode son los siguientes:

### Sistema sin compensación (Fig. 23):

- Margen de ganancia: 13 dB,
- Margen de fase: 61,700°.

### Sistema compensado (Fig. 24):

- Margen de ganancia: 10.6 dB,
- Margen de fase: 52,900°.

La validación del diseño se realizó mediante el análisis de la respuesta temporal en lazo cerrado. La respuesta al escalón del sistema compensado presenta un comportamiento estable, con un sobreimpulso moderado y un tiempo de establecimiento acorde a los objetivos del trabajo. Asimismo, la señal de control se mantiene dentro de valores aceptables, evitando saturaciones prolongadas del actuador.

En este caso, la compensación se realizó mediante un controlador puramente proporcional, definido como:

$$C(z) = K_p, \quad K_p = 1.308$$

Por lo tanto, el lazo abierto queda dado por  $L(z) = K_p G(z)$ . La acción del controlador proporcional consiste en escalar la magnitud de la respuesta en frecuencia del lazo abierto, desplazando la frecuencia de cruce y, en consecuencia, modificando los márgenes de ganancia y de fase observados. No se introducen polos ni ceros adicionales, por lo que no se realiza una compensación dinámica de fase; la mejora del desempeño se logra exclusivamente mediante el ajuste de la ganancia.

La incorporación de polos adicionales introduciría retardos en la respuesta y una reducción del margen de fase, mientras que la adición de ceros podría generar un adelanto de fase a costa de un incremento significativo del esfuerzo de control. Dado que en la simulación no se dispone de una estimación fiable del esfuerzo del actuador, no resulta posible ponderar adecuadamente estos efectos. Por este motivo, se descartó la inclusión de polos, ceros o acción integral en esta etapa, con el fin de preservar la integridad del sistema físico durante la implementación experimental.

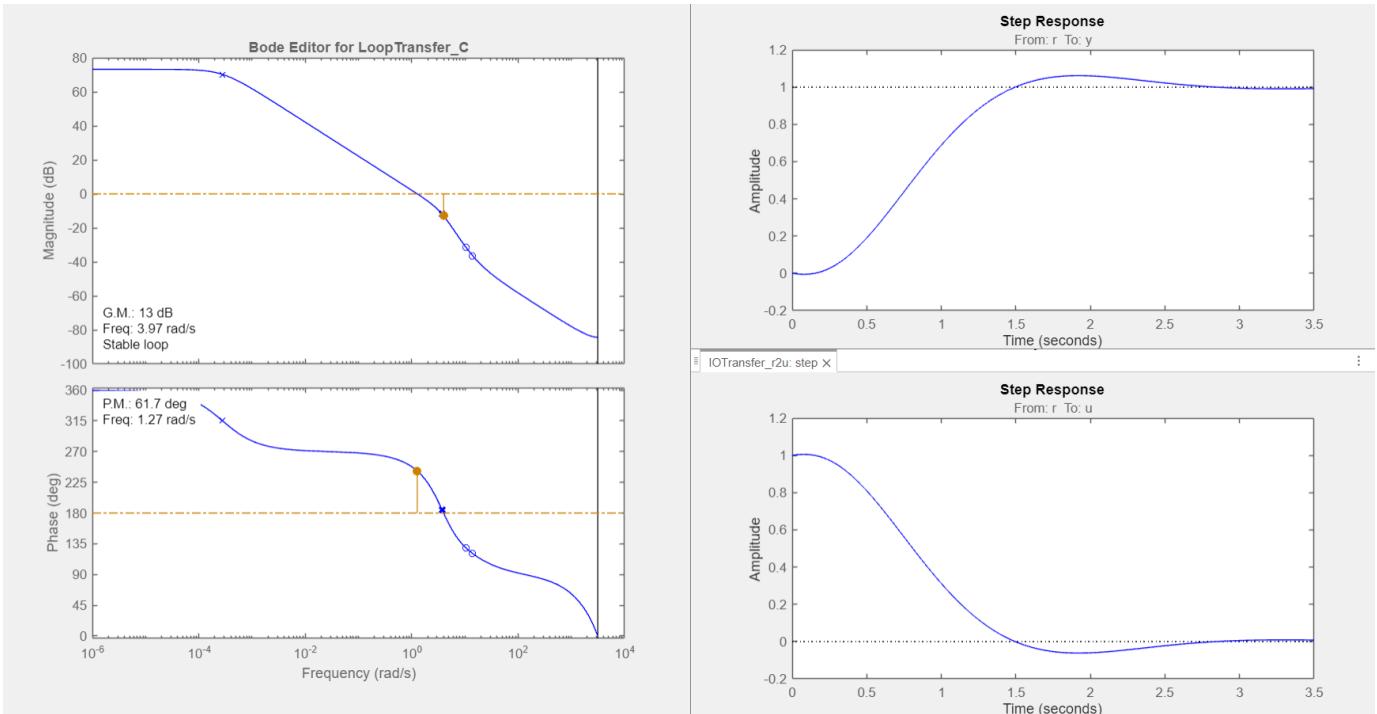


Figura 23. Diagrama de Bode y respuestas temporales del sistema con compensación.

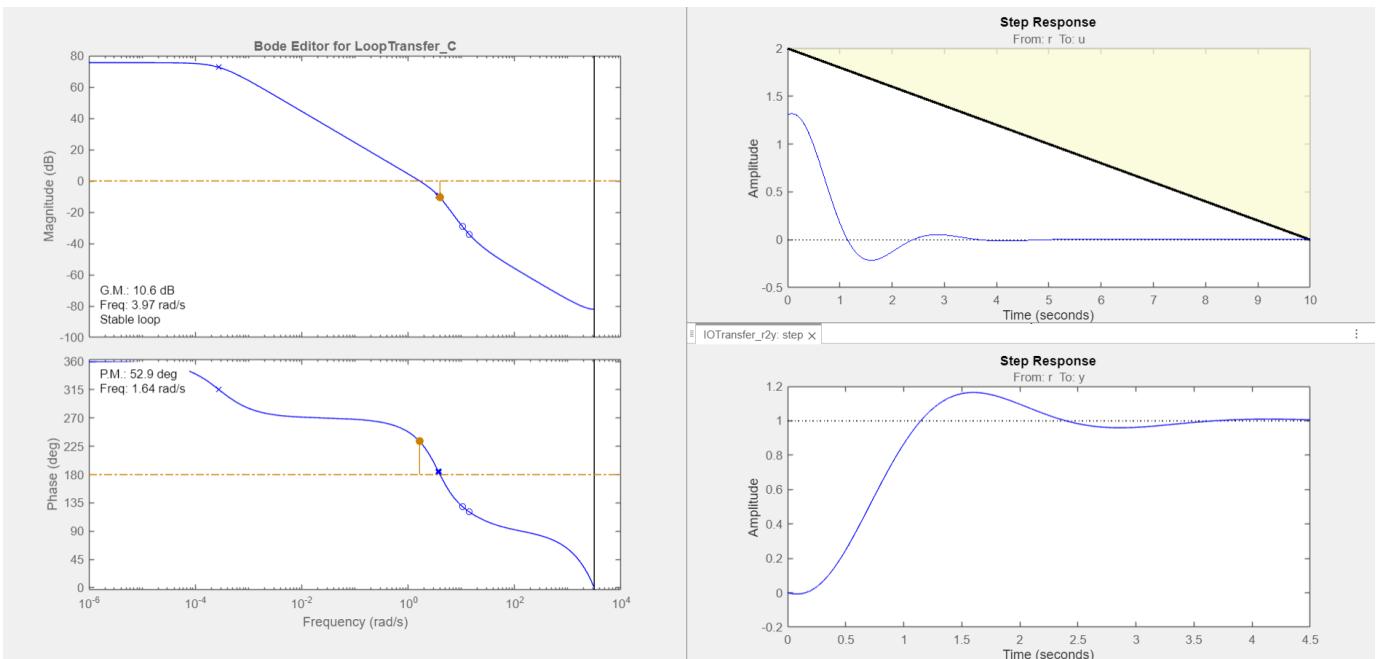


Figura 24. Diagrama de Bode y respuestas temporales del sistema con compensación.

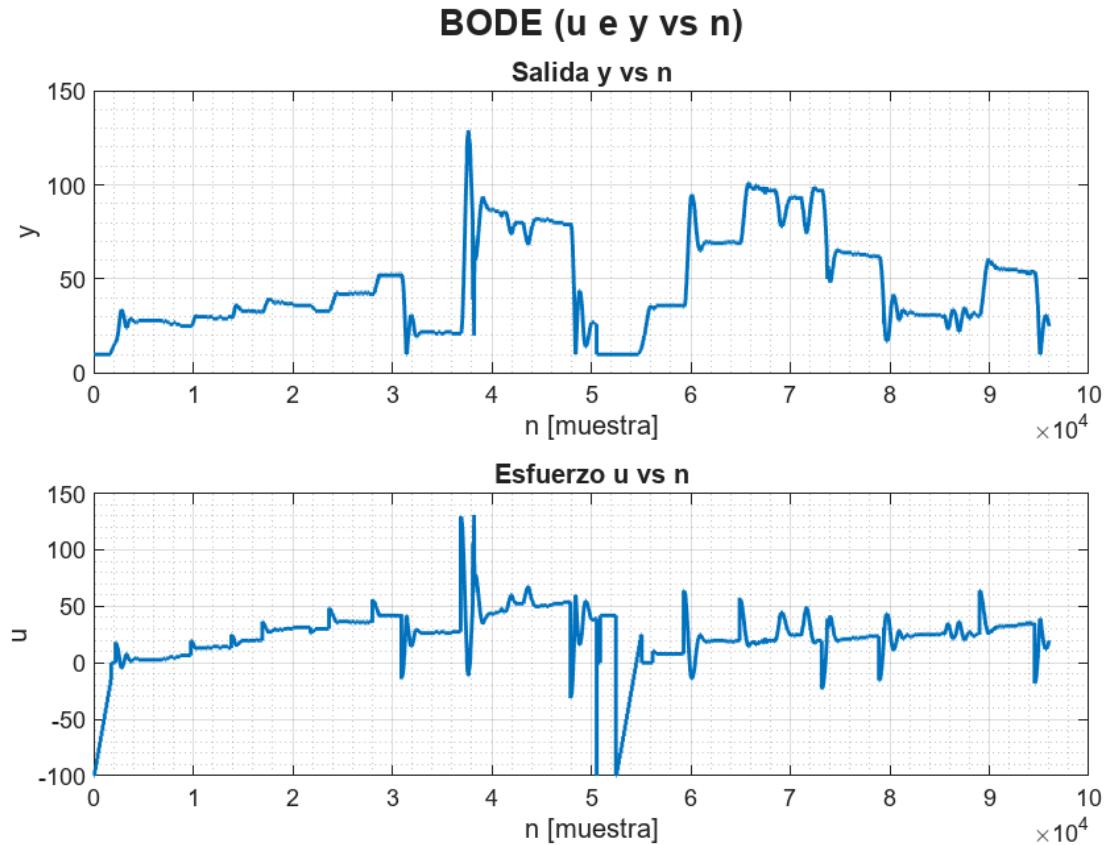


Figura 25. Figura de la respuesta del bode, altura y esfuerzo en n muestras

Compensador usado:

$$C_{Bode} = 1.3082$$

Tiempo de muestreo:

$$T_s = 0.0001$$

## VII-H. Síntesis Directa

**VII-H1. Síntesis directa (Truxal–Ragazzini):** Se parte del modelo discreto identificado de la planta:

$$G_{ZAS}(z) = \frac{-0.0001205 z^{-1} + 0.0002415 z^{-2} - 0.0001209 z^{-3}}{1 - 2.994 z^{-1} + 2.989 z^{-2} - 0.9944 z^{-3}} \quad (2)$$

El método de Truxal–Ragazzini consiste en especificar explícitamente una dinámica deseada en lazo cerrado  $G_{cl}(z)$  y obtener el controlador a partir de la relación:

$$G_{cl}(z) = \frac{C(z)G_{ZAS}(z)}{1 + C(z)G_{ZAS}(z)} \implies C(z) = \frac{1}{G_{ZAS}(z)} \frac{G_{cl}(z)}{1 - G_{cl}(z)} \quad (3)$$

Este procedimiento implica una inversión explícita del modelo de la planta, por lo que el diseño depende fuertemente de la exactitud del modelo identificado.

**VII-H1a. Método 1: respuesta deadbeat:** Como primera aproximación se adoptó una dinámica deseada del tipo *deadbeat*, definida por:

$$G_{cl}(z) = z^{-1}. \quad (4)$$

Esta elección implica que la salida alcance el valor deseado en un único período de muestreo, anulando el error en el menor tiempo posible.

Reemplazando en (3) se obtiene:

$$C_1(z) = \frac{1}{G_{ZAS}(z)} \frac{z^{-1}}{1 - z^{-1}} = \frac{1}{G_{ZAS}(z)} \frac{1}{z - 1}. \quad (5)$$

Se observa que el controlador resultante contiene explícitamente la inversa de la planta y un polo adicional en  $z = 1$ , lo que anticipa posibles problemas de magnitud del esfuerzo de control.

**VII-H2. Método 2: Deadbeat ripple-free:** Como alternativa se evaluó la variante *ripple-free*, cuyo controlador obtenido es:

$$C_2(z) = \frac{10.43038 z^2 - 2.7635 z + 3.3344}{-0.33662 z^2 - 0.66338 z + 1}. \quad (6)$$

Multiplicando numerador y denominador por  $(-1)$  y normalizando el coeficiente líder del denominador, se obtiene:

$$C_2(z) = \frac{-30.9856 z^2 + 8.2096 z - 9.9055}{z^2 + 1.9707 z - 2.9707}. \quad (7)$$

Para implementación digital resulta conveniente expresarlo en términos de  $z^{-1}$ :

$$C_2(z) = \frac{-30.9856 + 8.2096 z^{-1} - 9.9055 z^{-2}}{1 + 1.9707 z^{-1} - 2.9707 z^{-2}}. \quad (8)$$

**VII-H3. Resultados y Limitaciones Prácticas:** Las simulaciones mostraron que ambos controladores demandan esfuerzos de control extremadamente elevados, alcanzando valores del orden de:

$$|u_{\max}| \sim 10^{29},$$

lo cual excede ampliamente las capacidades del actuador físico.

En la implementación real, la señal de control corresponde a una señal PWM tipo servo a 50 Hz acotada en el rango:

$$u \in [1000, 2000] \mu s. \quad (9)$$

La magnitud desproporcionada del esfuerzo se explica por:  
■ La inversión explícita del modelo  $G_{ZAS}(z)$ .

- La presencia de polos cercanos a  $z = 1$  en la planta.
- Alta sensibilidad a pequeñas incertidumbres del modelo.
- Cancelaciones exactas requeridas por el diseño.

En particular, la inversión de dinámicas cercanas al borde del círculo unitario produce amplificaciones significativas en la señal de control, haciendo que el diseño sea extremadamente sensible a variaciones como cambios en la tensión de batería, fricción, efectos aerodinámicos y dinámica no modelada del conjunto ESC–motor–hélice.

Por estas razones, si bien la síntesis directa resulta valiosa desde el punto de vista conceptual y didáctico, no se considera viable para implementación experimental en la planta real.

En consecuencia, para la etapa práctica se priorizan estrategias de menor orden y mayor robustez, que contemplen explícitamente las limitaciones del actuador y la saturación de la señal de control. Ecuación general:

$$C(z) = \frac{1}{G(z)} \frac{G_{cl}(z)}{1 - G_{cl}(z)}$$

Dinámica deseada:

$$G_{cl}(z) = [\text{Completar}]$$

Observaciones:

- Sensibilidad al modelo.
- Elevado esfuerzo de control.
- Limitaciones prácticas.

## VII-I. Comparación entre Métodos Clásicos

- PID: **[Completar fortalezas y debilidades]**
- Lugar de Raíces: **[Completar]**
- Bode: **[Completar]**
- Síntesis Directa: **[Completar]**

Observación general:

**[Completar: limitaciones estructurales del enfoque clásico]**

## VIII. MÉTODOS MODERNOS DE CONTROL

### VIII-A. Introducción

A diferencia de los métodos clásicos basados en funciones de transferencia y análisis en frecuencia, los métodos modernos de control se fundamentan en la representación en espacio de estados del sistema dinámico.

En este enfoque, la dinámica se describe mediante:

$$x_{k+1} = Ax_k + Bu_k \quad (10)$$

$$y_k = Cx_k + Du_k \quad (11)$$

donde  $x_k \in \mathbb{R}^n$  es el vector de estados,  $u_k$  la entrada de control y  $y_k$  la salida medida.

Este formalismo permite:

- Diseñar realimentación directa de estados.
- Ubicar polos del sistema de manera sistemática.
- Formular problemas de control óptimo.
- Incorporar estimadores de estado.

*VIII-A1. Modelo en espacio de estados continuo:* Las matrices del modelo continuo en representación estado–espacio son:

$$F = \begin{bmatrix} -5.6102 & -3.5055 & -0.0314 \\ 4.0000 & 0 & 0 \\ 0 & 0.0312 & 0 \end{bmatrix}$$

$$G = \begin{bmatrix} 16 \\ 0 \\ 0 \end{bmatrix}$$

$$H = [-0.0076 \quad 0.0064 \quad 8.9975]$$

$$J = [0]$$

*VIII-A2. Modelo en Espacio de Estados:* El modelo discreto obtenido:

$$A = e^{FT_s}, \quad B = F^{-1}(e^{FT_s} - I)G, \quad C = H,$$

$$D = J$$

Tiempo de muestreo depende de la práctica

*VIII-A3. Análisis de Controlabilidad y Observabilidad:*  
Matriz de controlabilidad:

$$\mathcal{C} = [G \quad FG \quad F^2G \quad \dots \quad F^{n-1}G]$$

$$\text{rank}(\mathcal{C}) = [3]$$

Matriz de observabilidad:

$$\mathcal{O} = \begin{bmatrix} H \\ HF \\ HF^2 \\ \vdots \\ HF^{n-1} \end{bmatrix}$$

$$\text{rank}(\mathcal{O}) = [3]$$

Conclusión estructural:

Dado que el rango de ambas matrices coincide con el orden del sistema, se concluye que el modelo es completamente controlable y completamente observable.

Desde el punto de vista físico, esto implica que:

- Existe una combinación adecuada de la señal de entrada que permite influenciar todos los estados internos del sistema.
- La salida medida contiene información suficiente para reconstruir completamente el vector de estados mediante un observador.

En consecuencia, el modelo identificado resulta estructuralmente apto para el diseño de control por realimentación de estados, ubicación de polos, LQR y estimación de estados mediante observador de Luenberger o filtro de Kalman.

No obstante, la validez práctica de dicha síntesis en tiempo discreto depende de que el tiempo de muestreo  $T_s$  sea suficientemente pequeño para capturar la dinámica relevante del sistema, evitando aliasing. En particular, se asume que las componentes significativas de la señal de salida y de las perturbaciones se encuentran por debajo de la frecuencia de Nyquist  $f_N = \frac{1}{2T_s}$ , y que el acondicionamiento analógico (filtrado anti-alias) es consistente con esta hipótesis.

### VIII-B. Ubicación Arbitraria de Polos

### VIII-C. Realimentación de Estados y Estimación

*VIII-C1. Realimentación de Estados:* Considerando el modelo discreto del sistema:

$$x_{k+1} = Ax_k + Bu_k$$

$$y_k = Cx_k$$

se propone una ley de control por realimentación de estados:

$$u_k = -Kx_k$$

lo que conduce a la dinámica en lazo cerrado:

$$x_{k+1} = (A - BK)x_k$$

El diseño por ubicación arbitraria de polos consiste en determinar la matriz de ganancia  $K$  tal que:

$$\lambda(A - BK) = \{p_1, p_2, \dots, p_n\}$$

siendo  $|p_i| < 1$  condición necesaria para estabilidad discreta.

La determinación de  $K$  puede realizarse mediante el método de Ackermann o utilizando la función `place()` de MATLAB, siempre que el sistema sea completamente controlable.

**VIII-C2. Estimador de Estados:** En situaciones donde no todos los estados son medibles, se introduce un estimador para reconstruir el vector de estados a partir de la entrada y la salida medida.

El estimador discreto se define como:

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + L(y_k - \hat{y}_k)$$

donde:

$$\hat{y}_k = C\hat{x}_k$$

La dinámica del error de estimación:

$$e_k = x_k - \hat{x}_k$$

queda gobernada por:

$$e_{k+1} = (A - LC)e_k$$

Por lo tanto, la convergencia del estimador depende de la ubicación de los autovalores de la matriz  $A - LC$ , que pueden ser fijados arbitrariamente siempre que el sistema sea observable:

$$\lambda(A - LC) = \{p_{obs,1}, \dots, p_{obs,n}\}$$

**VIII-C3. Principio de Separación:** Cuando se combinan realimentación de estados y estimación, la ley de control adopta la forma:

$$u_k = -K\hat{x}_k$$

y la dinámica total del sistema presenta autovalores dados por la unión de los polos del controlador y los polos del estimador:

$$\lambda_{total} = \lambda(A - BK) \cup \lambda(A - LC)$$

Este resultado, conocido como principio de separación, permite diseñar independientemente el controlador y el estimador.

#### VIII-D. Control Integrador

**VIII-D1. Ubicación arbitraria de polos con Integrador:** En esta sección se implementa el procedimiento propuesto en Ogata (Ec. 6.19) para incorporar acción integral al sistema en espacio de estados, utilizando únicamente un integrador externo y realimentación de estados estimados.

La planta discreta utilizada (orden  $n = 3$ ) se obtuvo mediante discretización por ZOH con período de muestreo:

$$T_s = 0.02 \text{ s} \quad (50 \text{ Hz})$$

**VIII-D1a. Sistema aumentado con integrador:** Para eliminar el error en régimen permanente frente a referencias constantes, se define el estado integral:

$$v_{k+1} = v_k + (r_k - y_k)$$

$$\text{donde } y_k = Cx_k.$$

Siguiendo el desarrollo de Ogata, se construye el sistema aumentado:

$$\hat{A} = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

y se ubican los polos deseados del sistema aumentado:

$$p_{ctrl} = \{0.95 \pm 0.15j, 0.98\}, \quad p_i = 0.96$$

De esta forma se obtienen las ganancias equivalentes:

$$K_2 = [0.7820721 \quad -0.5678709 \quad 0.3779423]$$

$$K_1 = 0.1469358$$

donde:

-  $K_2$  actúa sobre el estado estimado  $\hat{x}_k$ . -  $K_1$  actúa sobre el estado integral  $v_k$ .

La ley de control implementada es:

$$u_k = K_1 v_k - K_2 \hat{x}_k$$

Los polos del sistema aumentado cerrado fueron correctamente ubicados (según place) en:

$$p_{K_{21}} = 13$$

(lo que confirma la correcta asignación interna en MATLAB).

**VIII-D1b. Diseño del observador:** Se diseñaron dos variantes de observador con los mismos polos deseados:

$$p_{obs} = \{0.8 \pm 0.25j, 0.9\}$$

VIII-D1c. *Observador predictor*: Dinámica del error:

$$A - L_{\text{pred}}C$$

Ganancia obtenida:

$$L_{\text{pred}} = \begin{bmatrix} 167.3535 \\ 253.6822 \\ 86.0310 \end{bmatrix}$$

$$p_{L_{\text{pred}}} = 12$$

VIII-D1d. *Observador actual*: Dinámica del error:

$$A - L_{\text{actual}}CA$$

Ganancia obtenida:

$$L_{\text{actual}} = \begin{bmatrix} 126.8411 \\ 172.0621 \\ 45.1301 \end{bmatrix}$$

$$p_{L_{\text{actual}}} = 13$$

VIII-D1e. *Estructura completa del sistema aumentado cerrado*: Para analizar explícitamente la dinámica del sistema completo (planta + integrador + control), el script construye el modelo aumentado:

$$A_{\text{aug}} = \begin{bmatrix} A & B \\ K_2 - K_2A - K_1CA & 1 - K_2B - K_1CB \end{bmatrix}$$

$$B_{\text{aug}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ K_1 \end{bmatrix}$$

$$C_{\text{aug}} = [C \quad 0]$$

Este sistema representa la dinámica cerrada resultante de aplicar la ley de control integral derivada mediante Ogata 6.19.

Además, para comparar la dinámica del error de estimación, se construye el sistema:

$$A_{\text{obs}} = A - L_{\text{actual}}CA$$

y se superponen ambos mapas de polos mediante `pzmap`.

VIII-D2. *Resultados de simulación*: Se evaluó el comportamiento del sistema para una referencia escalón de 25 unidades, comparando:

- Observador predictor
- Observador actual
- Esfuerzo de control
- Evolución del estado integral

Las figuras correspondientes se muestran a continuación:

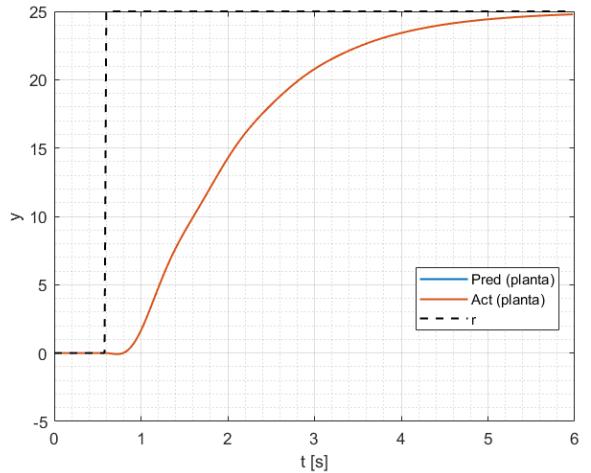


Figura 26. Respuesta temporal  $y(t)$  con integrador — comparación predictor vs actual.

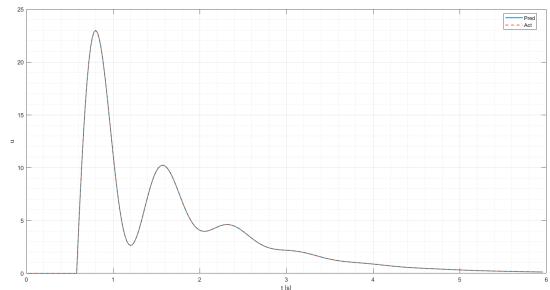


Figura 27. Esfuerzo de control  $u(t)$  con integrador.

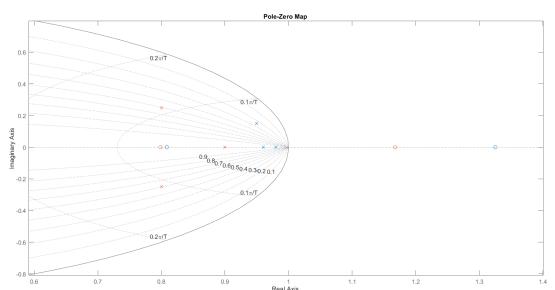


Figura 28. Mapa de polos: planta, sistema aumentado y dinámica del observador.

VIII-D3. *Resultados experimentales (implementacion practica)*: Ademas de las simulaciones, se realizaron ensayos sobre la planta real implementando el controlador con accion integral y las dos variantes de observador (predictor y actual). Las siguientes figuras muestran los resultados practicos registrados para una referencia tipo escalon, utilizando la misma configuracion de ganancias reportada previamente.

VIII-D4. *Discusión*: El método de Ogata 6.19 permite incorporar acción integral sin modificar la estructura original

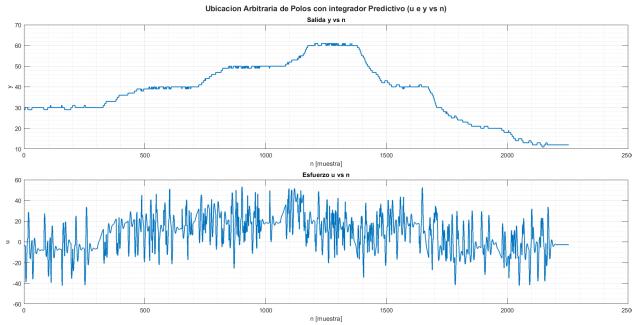


Figura 29. Resultado experimental con observador predictor (implementacion practica).

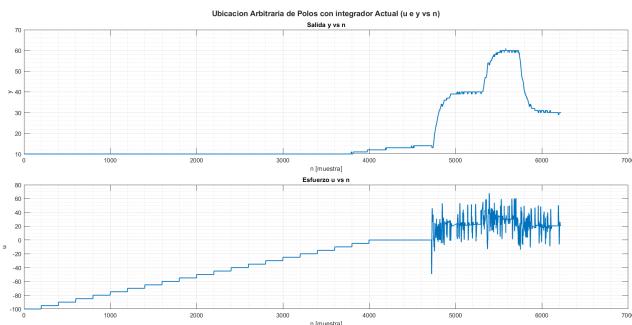


Figura 30. Resultado experimental con observador actual (implementacion practica).

de la planta, utilizando un estado adicional y realimentación adecuada.

Se observa que:

- El integrador elimina el error en régimen permanente.
- La dinámica del observador es más rápida que la del sistema controlado.
- Ambos esquemas (predictor y actual) resultan estables, con polos dentro del círculo unidad.
- El sistema aumentado conserva estabilidad global.

El código completo utilizado para esta sección se incluye en el Apéndice ??.

#### VIII-E. Control Óptimo (LQR)

Se diseñó un regulador óptimo discreto de tipo LQR a partir del modelo en espacio de estados discretizado mediante retención de orden cero (ZOH) con tiempo de muestreo:

$$T_s = 0.01 \text{ s}$$

El objetivo del diseño es minimizar el funcional de costo cuadrático infinito:

$$J = \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k)$$

donde  $Q \succeq 0$  penaliza la energía de los estados y  $R \succ 0$  penaliza el esfuerzo de control.

**VIII-E0a. Matrices de ponderación:** Se adoptó una sintonización práctica basada en escalas típicas del experimento: un cambio de referencia del orden de  $\Delta y \approx 20$  y una restricción del mando aproximadamente  $|u| \lesssim 300$  (en unidades del actuador). El diseño penaliza principalmente la salida mediante  $C^T C$ , agregando un término pequeño para asegurar buena condición numérica:

$$Q = w_y (C^T C) + 10^{-8} I_n, \quad R = \frac{w_u}{u_{\max}}$$

con  $w_y = 20$ ,  $w_u = 500$  y  $u_{\max} = 300$ . Para el modelo discretizado utilizado, las matrices resultantes fueron:

$$Q = \begin{bmatrix} 6.8188 \times 10^{-3} & -6.9883 \times 10^{-3} & 7.0544 \times 10^{-3} \\ -6.9883 \times 10^{-3} & 7.1620 \times 10^{-3} & -7.2298 \times 10^{-3} \\ 7.0544 \times 10^{-3} & -7.2298 \times 10^{-3} & 7.2982 \times 10^{-3} \end{bmatrix}$$

$$R = 1.6667$$

**VIII-E0b. Ecuación de Riccati discreta y ganancia óptima:** La solución del problema se obtiene resolviendo la ecuación de Riccati discreta:

$$P = A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A + Q$$

La ganancia óptima resulta:

$$K = (R + B^T P B)^{-1} B^T P A$$

Para el modelo discretizado se obtuvo:

$$K = [0.5119095 \quad -0.4937470 \quad 0.4765571]$$

**VIII-E0c. Polos del sistema en lazo cerrado:** La estabilidad se verifica mediante los polos del sistema:

$$A_{cl} = A - BK$$

Los polos obtenidos fueron:

$$0.9728458 + 0.0327300 j$$

$$\lambda(A - BK) = 0.9728458 - 0.0327300 j$$

$$0.9663930$$

Se observa que todos los polos se encuentran dentro del círculo unidad, garantizando estabilidad discreta.

**VIII-E0d. Referencia al apéndice de implementación:** El código completo utilizado para:

- cargar y discretizar la planta,
- definir  $Q$  y  $R$  mediante parámetros *knobs* ( $w_y$ ,  $w_u$ ),
- calcular  $K$  y verificar controlabilidad,
- diseñar observadores (predictor y actual) por ubicación de polos,
- calcular el prefiltro  $N_{bar}$  para seguimiento,
- y simular el lazo con/sin saturación, con/sin ruido, y en doble precisión y float32,

se incluye en el Apéndice A.

*VIII-E1. Simulación del LQR con observador, saturación y ruido:* Con el fin de aproximar el comportamiento del sistema real y evitar conclusiones optimistas, se implementó un entorno de simulación que contempla:

- **Observador de estados** en dos variantes:
  - *Predictor*: corrige usando  $y_k$ .
  - *Actual*: corrige usando  $y_{k+1}$ .
- **Prefiltro de referencia**  $N_{\text{bar}}$  para mejorar el seguimiento de referencia en lazo cerrado (estructura  $u_k = N_{\text{bar}}r_k - K\hat{x}_k$ ).
- **Saturación** del actuador con límite  $\pm u_{\text{máx}}$  para representar las restricciones del PWM/ESC.
- **Ruido** configurable para emular incertidumbres del experimento:
  - Ruido de medición:  $y_{\text{meas}} = y_{\text{true}} + v_y$ , con  $v_y \sim \mathcal{N}(0, \sigma_y^2)$ .
  - Ruido del actuador (jitter) y cuantización:  $u_{\text{app}} = \text{sat}(\text{quant}(u_{\text{cmd}} + v_u))$ .
  - Ruido de proceso:  $x_{k+1} = Ax_k + Bu_k + w_k$  (configurable como prueba de estrés).
- **Comparación numérica** entre simulación en doble precisión y `float32` para anticipar efectos de implementación embocada.

En particular, para evitar comparaciones sesgadas, las secuencias de ruido se pre-generaron y se reutilizaron de forma idéntica tanto en doble precisión como en `float32`.

*VIII-E1a. Observadores y ubicación de polos:* Los observadores se diseñaron mediante ubicación arbitraria de polos en el plano- $z$ , definiendo polos deseados:

$$p_{\text{obs}} = \{0.8 \pm 0.25j, 0.9\}$$

y obteniendo:

$$K_{e,\text{pred}} = \text{place}(A^T, C^T, p_{\text{obs}})^T \quad K_{e,\text{act}} = \text{place}(A^T, (CA)^T, p_{\text{obs}})^T$$

*VIII-E1b. Prefiltro  $N_{\text{bar}}$ :* Para mejorar el seguimiento de referencia (SISO), se calculó un prefiltro  $N_{\text{bar}}$  tal que la ganancia estática equivalente sea adecuada, utilizando la rutina auxiliar `refi`:

$$u_k = N_{\text{bar}} r_k - K \hat{x}_k$$

*VIII-E1c. Diagnósticos para evitar autoengaño:* Se incluyeron verificaciones directas sobre el ruido injectado:

$$e_y(k) = y_{\text{meas}}(k) - y_{\text{true}}(k)$$

y se reportaron métricas como  $\text{RMS}(e_y)$  y conteo de saturaciones en  $u$ , para confirmar que las perturbaciones y límites realmente están actuando sobre el lazo.

*VIII-E1d. Referencia al apéndice:* El script completo de simulación (incluyendo funciones locales `sim_obs_loop`, `plot_zplane`, `refi` y el *fallback* `dlqr_iter_nolic`) se incluye en el Apéndice A.

### VIII-F. Filtro de Kalman

**VIII-F1. Modelo con Ruido:** Para incorporar incertidumbre y modelar explícitamente la presencia de perturbaciones no modeladas y ruido del sensor, se adopta la siguiente representación estocástica discreta:

$$x_{k+1} = Ax_k + Bu_k + w_k$$

$$y_k = Cx_k + v_k$$

donde:

- $w_k \sim \mathcal{N}(0, Q)$  representa el ruido de proceso, asociado a una dinámica no modelada, perturbaciones aerodinámicas y simplificaciones del modelo identificado.
- $v_k \sim \mathcal{N}(0, R)$  representa el ruido de medición, proveniente del sensor láser de distancia.

**VIII-F1a. Estimación de  $R$  (ruido de medición):** La varianza del ruido de medición se obtuvo mediante ensayos empíricos, midiendo la dispersión de la señal del sensor con la planta en reposo. Siendo  $\sigma_v$  la desviación estándar medida (en cm), se adopta:

$$R = \sigma_v^2$$

En los ensayos realizados se obtuvo:

$$\sigma_v = 2.043 \text{ cm} \implies R = 4.174 \text{ cm}^2$$

**VIII-F1b. Parametrización y sintonización de  $Q$  (ruido de proceso):** La matriz de covarianza del ruido de proceso se parametrizó como:

$$Q = q I_n$$

donde  $q$  es un escalar positivo ajustable e  $I_n$  es la matriz identidad de dimensión  $n$ . El valor de  $q$  se determinó mediante consistencia estadística de la innovación normalizada (ver Apéndice A).

El valor óptimo obtenido fue:

$$q = 50.8022 \times 10^{-3} \implies Q = q I_n$$

**VIII-F2. Filtro de Kalman en régimen permanente:** Se utilizó un estimador de Kalman discreto en su variante *current estimator* (`kalman(..., 'current')`). La ganancia en régimen permanente  $L$  se obtiene a partir de la solución estacionaria  $P$  de la ecuación de Riccati discreta:

$$P = APA^T - APC^T (CPC^T + R)^{-1} CPA^T + Q$$

y la ganancia queda:

$$L = APC^T (CPC^T + R)^{-1}$$

La ganancia obtenida en MATLAB para el modelo discretizado fue:

$$L = \begin{bmatrix} 97.4968 \\ 192.3014 \\ 94.7820 \end{bmatrix}$$

**VIII-F2a. Polos del observador:** La estabilidad del observador se verifica a partir de los polos del sistema estimador. Para el *current estimator*, la dinámica del error queda determinada por  $A - LCA$ , por lo que se reportan los polos:

$$\begin{aligned} & 0.9730274 + 0.0308455 j \\ \lambda(A - LCA) = & 0.9730274 - 0.0308455 j \\ & 0.9694430 \end{aligned}$$

**VIII-F3. Control integral y realimentación de estados (LQI):** Con el objetivo de eliminar el error en régimen permanente ante referencias tipo escalón, se incorporó un integrador de error. Definiendo el estado integral  $\xi_k$ :

$$\xi_{k+1} = \xi_k + (r_k - y_k)$$

Se diseñó una ley de control tipo LQI:

$$u_k = -K_x \hat{x}_k + K_i \xi_k$$

donde  $\hat{x}_k$  proviene del estimador de Kalman.

Los valores obtenidos mediante `dlqr` (con pesos heurísticos) fueron:

$$K_x = [16.5343 \quad -15.8046 \quad 15.1271]$$

$$K_i = 3.2067$$

**VIII-F3a. Polos de la planta y del lazo cerrado:** Los polos de la planta discretizada (sin control) fueron:

$$\begin{aligned} & 0.9999972 \\ \lambda(A) = & 0.9720409 + 0.0241158 j \\ & 0.9720409 - 0.0241158 j \end{aligned}$$

Los polos del lazo cerrado del sistema aumentado (planta + integrador + control) fueron:

$$\begin{aligned} & 0.9735939 + 0.0318308 j \\ & 0.9735939 - 0.0318308 j \\ \lambda(A_{cl}) = & 0.9671966 \\ & 0 \end{aligned}$$

**VIII-F4. Sistema aumentado usado para `pzmap`:** Además del cálculo teórico de polos del lazo cerrado, en el script de validación se armó explícitamente un *sistema aumentado discreto* para analizar la estabilidad en el plano- $z$  con `pzmap`. Este sistema corresponde a la dinámica conjunta de planta e integrador, con la ley de control LQI aplicada.

En el código se definieron las matrices del sistema aumentado como:

$$A_{\text{aug}} = \begin{bmatrix} A & B \\ K_x - K_x A - K_i C A & 1 - K_x B - K_i C B \end{bmatrix}$$

$$B_{\text{aug}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ K_i \end{bmatrix}$$

$$C_{\text{aug}} = [C \quad 0] \quad D_{\text{aug}} = 0$$

y se construyó el modelo:

$$\text{sysDaug} = \text{ss}(A_{\text{aug}}, B_{\text{aug}}, C_{\text{aug}}, 0, \text{Ts}).$$

Finalmente, se utilizó `pzmap(sysDaug)` para visualizar los polos del sistema aumentado en el plano- $z$  y verificar que la dinámica resultante permanezca estable (polos dentro del círculo unidad) en el caso nominal.

**VIII-F4a. Nota:** Esta construcción se incluye tal cual se implementa en el script, con el objetivo de replicar el análisis práctico realizado y obtener directamente el mapa de polos asociado a la estructura planta + integrador + control.

**VIII-F5. Nota práctica sobre saturación:** En la implementación práctica se aplica saturación al mando  $u_k$  para respetar los límites del actuador (PWM). La presencia de saturación puede degradar el cumplimiento exacto de la dinámica diseñada y producir integrador acumulado, por lo que en firmware se complementa con estrategias de anti-windup cuando corresponde.

**VIII-F6. Resultados con ruido y validación:** Para evaluar el desempeño se simuló el sistema incluyendo tanto ruido de medición (como dispersión del sensor) como ruido de proceso (dinámica no modelada). Se comparó:

- Respuesta de salida con ruido, y seguimiento de referencia.
- Esfuerzo de control requerido en presencia de ruido.
- Estabilidad del sistema mediante mapa de polos y ceros.

Las siguientes figuras muestran los resultados obtenidos:

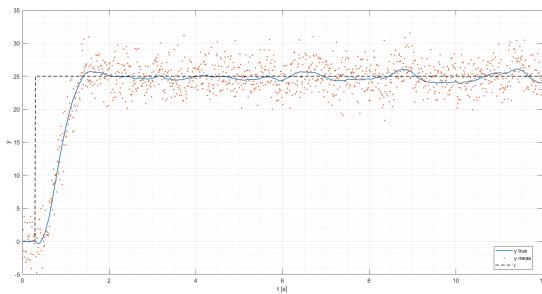


Figura 31. Respuesta temporal de la salida en presencia de ruido (medición y proceso) y comparación con la referencia.

El código completo utilizado para calcular  $L$ ,  $K_x$ ,  $K_i$ , los polos de la planta, los polos del observador y los polos del lazo cerrado se incluye en el Apéndice B.

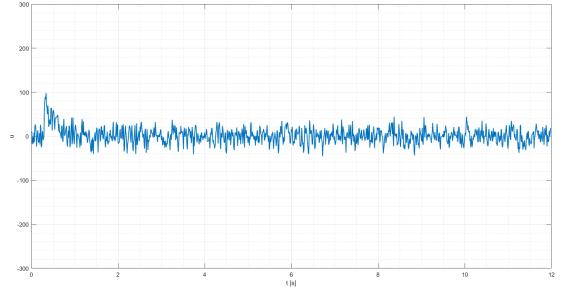


Figura 32. Esfuerzo de control (señal de mando) en presencia de ruido.

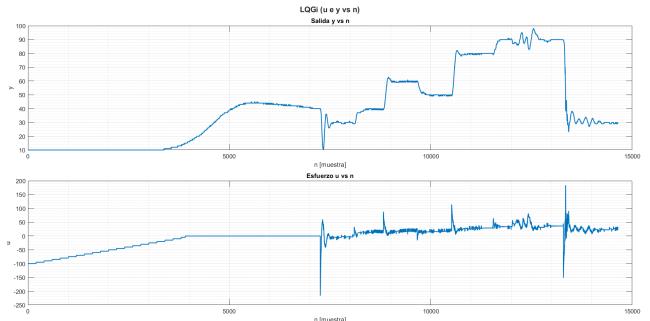


Figura 33. Resultado experimental obtenido con la implementación práctica sobre la planta real.

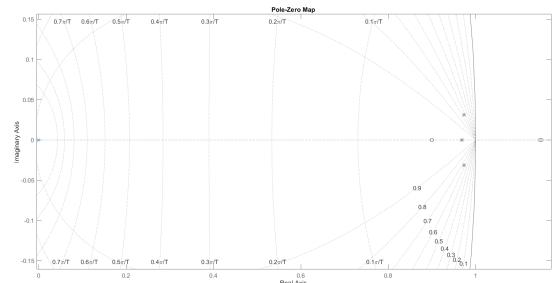


Figura 34. Mapa de polos y ceros. En naranja se muestran los polos del observador y en azul los polos asociados a la planta/sistema.

## IX. CONCLUSIONES

### APÉNDICE A PARÁMETROS DEL CONTROLADOR

### APÉNDICE B CÓDIGOS MATLAB

### APÉNDICE C ESPACIO DE ESTADO CON INTEGRADOR)

En este apéndice se incluye el script completo utilizado para implementar el método de Ogata 6.19 con un integrador externo (accion integral) y dos variantes de observador en tiempo discreto: predictor y actual. El código calcula las ganancias  $K_1$  y  $K_2$  mediante `place` sobre el sistema aumentado, diseña los observadores con los polos  $p_{\text{obs}}$ , y genera las simulaciones y graficos asociados.

```

1 % % =====
2 % OGATA 6.19 – SOLO INTEGRADOR
3 % + Observador PREDICTOR y ACTUAL
4 % (Planta discreta desde .mat, orden 3)
5 % % =====
6 close all; clear; clc

7 % % =====
8 % 1) CARGA + DISCRETIZACION
9 % % =====
10 S = load('planta_(1).mat');

11 if isfield(S,'plantaC')
12 plantaC = S.plantaC;
13 elseif isfield(S,'sysC')
14 plantaC = S.sysC;
15 else
16 error('No _encuentro _"plantaC" _ni_ "sysC" _dentro _de _planta __(1).mat');
17 end

18 Ts = 1/50; % ajusta si queres
19 sysD = c2d(plantaC, Ts, 'zoh');
20 [A,B,C,D] = ssdata(ss(sysD));

21 n = size(A,1);
22 if n ~= 3
23 error('Esta _plantilla _asume _planta _de _orden _3._n
24 = %d', n);
25 end

26 % Asegurar SISO (una salida)
27 if size(C,1) ~= 1
28 C = C(1,:);
29 end

30 fprintf('Ts= %.9f\n n=%d\n', Ts, n);

31 % % =====
32 % 2) POLOS DESEADOS (FIJOS)
33 % % =====
34 p_ctrl = [0.95 + 0.15i, 0.95 - 0.15i, 0.98]; % control
35 p_obs = [0.8 + 0.25i, 0.8 - 0.25i, 0.9];
36 %p_obs = [0.4 + 0.25i, 0.4 - 0.25i, 0.6]; % observador
37 p_i = 0.96; % polo integrador

38 % % =====
39 % 3) OGATA 6.19 (K1 y K2) TAL CUAL
40 % % =====
41 m = 1;

42 Ahat = [A B; zeros(m,n+m)];

```

```

51 Bhat = [zeros(n,m); eye(m)];
52 Chat = [C zeros(1,m)];
53
54 polos_i = [p_ctrl p_i]; % 3 polos control
55 + 1 integrador
56 [Khat,pKhat] = place(Ahat, Bhat, polos_i);
57
58 Aux = [A-eye(size(A)) B;
59 C*A C*B];
60
61 K2K1 = (Khat + [zeros(1,n) eye(m)]) / Aux;
62 K2 = K2K1(1,1:n); % sobre xhat (1
63 x3)
64 K1 = K2K1(1,n+1:end); % sobre v (1
65 x1)
66
67 fprintf('\nK2_='); disp(K2);
68 fprintf('K1_='); disp(K1); fprintf('tpK21_=');
69 disp(pKhat);
70
71 % % =====
72 % 4) OBSERVADORES (PRED / ACT) – mismo
73 p_obs
74 % % =====
75 [L_pred, pL_pred] = place(A', C', p_obs);
76 L_pred = L_pred.%; % predictor: A – L*C
77 [L_actual, pL_actual] = place(A', (C*A)', p_obs);
78 L_actual = L_actual.%; % actual: A – L*C*A
79
80 fprintf('L_pred_='); disp(L_pred); fprintf('
81 tpL_pred_='); disp(pL_pred);
82 fprintf('\nL_actual_='); disp(L_actual); fprintf('
83 tpL_actual_='); disp(pL_actual);
84
85 % % =====
86 % 5) SIMULACION (SOLO CON INTEGRADOR)
87 % % =====
88 N = 300;
89 t = 0:Ts:(N-1)*Ts;
90
91 r = ones(1,N)*25;
92 r(1:30) = 0;
93
94 % ruido en medicion (igual que tu estilo)
95 w = [zeros(1,75), 0.01*ones(1,N-75)];
96
97 % ----- PREDICTIVO CON integrador -----
98
99 X_p = zeros(n,N); % estado real
100 Xh_p = zeros(n,N); % estado estimado
101 V_p = zeros(1,N); % integrador
102 U_p = zeros(1,N); % control
103
104 % ----- ACTUAL CON integrador -----
105 X_a = zeros(n,N);
106 Xh_a = zeros(n,N);

```

```

97 V_a    = zeros(1,N);
98 U_a    = zeros(1,N);
99
100 for k = 1:N-1
101 rk = r(k);
102
103 % % ===== 1) PREDICTIVO + integrador (
104 % MISMA FORMA) =====
105 y_p      = C*X_p(:,k);
106 V_p(k+1) = V_p(k) + (rk - y_p);
107 % integrador real
108 u_p      = K1*V_p(k+1) - K2*Xh_p(:,k);
109 X_p(:,k+1) = A*X_p(:,k) + B*u_p; % + w(k);
110 Xh_p(:,k+1) = A*Xh_p(:,k) + B*u_p + L_pred*(y_p - C*Xh_p(:,k));
111 U_p(k)    = u_p;
112
113 % % ===== 2) ACTUAL + integrador (MISMA
114 % FORMA) =====
115 y_a      = C*X_a(:,k);
116 V_a(k+1) = V_a(k) + (rk - y_a);
117 u_a      = K1*V_a(k+1) - K2*Xh_a(:,k);
118 X_a(:,k+1) = A*X_a(:,k) + B*u_a; % + w(k);
119 y_next   = C*X_a(:,k+1); % + w(k);
120 % misma muestra de ruido
121 z_a      = A*Xh_a(:,k) + B*u_a;
122 Xh_a(:,k+1) = z_a + L_actual*(y_next - C*z_a);
123 U_a(k)    = u_a;
124 end
125
126 % % =====
127 % 6) PLOTS RAPIDOS
128 % % =====
129 figure('Name','y(t)_-_integrador_+_predictor_vs_
actual');
130 plot(t, (C*X_p).', t, (C*X_a).', t, r, 'k--','LineWidth'
1.2);
131 grid on; grid minor;
132 xlabel('t_[s]'); ylabel('y');
133 legend('Pred_(planta)','Act_(planta)',r,'Location','
best');
134
135 figure('Name','u(t)_-_integrador_+_predictor_vs_
actual');
136 plot(t(1:end-1), U_p(1:end-1), 'LineWidth',1.4);
137 hold on;
138 plot(t(1:end-1), U_a(1:end-1), '--', 'LineWidth'
1.4);
139 grid on; grid minor;
140 xlabel('t_[s]'); ylabel('u');
141 legend('Pred','Act','Location','best');
142
143 figure('Name','v(t)_-_integrador_(estado_integral)');
144 plot(t, V_p, 'LineWidth',1.4); hold on;
145 plot(t, V_a, '--', 'LineWidth',1.4);
146 grid on; grid minor;
147 xlabel('t_[s]'); ylabel('v');
148 legend('Pred','Act','Location','best');
149
150 % % =====
151 % 7) MAPA DE POLOS (resumen)
152 % % =====
153 figure('Name','Z-plane_-_planta,_aumentado_e_
info_obs','Position',[100 100 1000 400]);
154 subplot(1,3,1); zgrid; hold on; grid on; box on;
155 p_planta = eig(A);
156 plot(real(p_planta), imag(p_planta), 'ko','MarkerSize'
8,'LineWidth',1.4);
157 title('Planta_(A)'); xlabel('Re\{z\}'); ylabel('Im\{z\}');
158 ;
159
160 subplot(1,3,2); zgrid; hold on; grid on; box on;
161 p_aug   = eig(Ahat);
162 p_aug_cl = eig(Ahat - Bhat*Khat);
163 plot(real(p_aug),   imag(p_aug),   'm^','
MarkerSize',8,'LineWidth',1.4);
164 plot(real(p_aug_cl), imag(p_aug_cl), 'c*','
MarkerSize',10,'LineWidth',1.6);
165 title('Aumentado:_Ahat_y_Ahat-BhatKhat'); xlabel
('Re\{z\}'); ylabel('Im\{z\}');
166
167 subplot(1,3,3); zgrid; hold on; grid on; box on;
168 p_obs_pred = eig(A - L_pred*C);
169 p_obs_act  = eig(A - L_actual*C*A);
170 plot(real(p_obs_pred), imag(p_obs_pred), 'bs','
MarkerSize',9,'LineWidth',1.5);
171 plot(real(p_obs_act),  imag(p_obs_act),  'rx','
MarkerSize',9,'LineWidth',1.5);
172 title('Obs_error'); xlabel('Re\{z\}'); ylabel('Im\{z\}');
173 legend('A-LC_(pred)','A-LCA_(act)','Location','
best');
174
175 sgtitle('Ogata_6.19_-_Solo_integrador_+_predictor
/actual');
176
177 Aaug = [A,B;...
178 K2-K2*A-K1*C*A,1-K2*B-K1*C*B];
179 Baug = [0;0;0;K1];
180 Caug = [C,0];
181 D = 0;
182 sysDaug = ss(Aaug,Baug,Caug,D,Ts);
183 sysObs = ss((A-L_actual*C*A),B,C,D,Ts);
184 figure; pzmap(sysDaug); grid on; zgrid; hold on;
185 pzmap(sysObs);

```

Listing 1. OGATA 6.19 - SOLO INTEGRADOR + Observador PREDICTOR y ACTUAL (planta orden 3).

## APÉNDICE

En este apéndice se incluye el script MATLAB utilizado para: (i) cargar y discretizar la planta identificada, (ii) definir las

ponderaciones  $Q$  y  $R$  del LQR, (iii) calcular la ganancia optima  $K$  (via `dlqr/dare`), (iv) disenar observadores (predictor y actual) por ubicacion arbitraria de polos, (v) calcular el prefiltro  $N_{bar}$  para seguimiento de referencia, y (vi) simular el lazo con/sin saturacion y con/sin ruido (incluyendo comparacion double vs float32).

```

1 close all; clear; clc
2
3 % % =====
4 % 1) CARGA + DISCRETIZACION
5 % % =====
6 S = load('planta_(1).mat');
7
8 if isfield(S,'plantaC')
9 plantaC = S.plantaC;
10 elseif isfield(S,'sysC')
11 plantaC = S.sysC;
12 else
13 error('No_encuentro_>"plantaC"< ni_>"sysC"< dentro_
de_planta_(1).mat');
14 end
15
16 Ts = 1/100; % sample time
17 sysD = c2d(plantaC, Ts, 'zoh');
18 [A,B,C,D] = ssdata(ss(sysD));
19 n = size(A,1);
20
21 fprintf('Ts=%.9f,\n n=%d\n', Ts, n);
22 disp('A='); disp(A); disp('B='); disp(B); disp('C=');
23 disp(C); disp('D='); disp(D);
24
25 % % =====
26 % 2) PARAMETROS + PESOS (LQR)
27 % % =====
28 % Observador (z-plane)
29 p_obs = [0.8 + 0.25i, 0.8 - 0.25i, 0.9];
30
31 % Objetivo practico: step ~20 y u limitado a
32 +/-300
33 r_step = 20;
34 u_max = 300;
35
36 % knobs
37 wy = 20; % subir => mas seguimiento (
38 mas agresivo)
39 wu = 500; % subir => menos esfuerzo (
40 mas timido)
41
42 % Q y R coherentes
43 Q = wy*(C'*C) + 1e-8*eye(n);
44 R = wu/(u_max);
45
46 % % =====
47 % 3) GANANCIAS: LQR + OBSERVADOR +
48 Nbar
49
50 % % =====
51 rc = rank(ctrb(A,B));
52 if rc < n
53 error('El_par_(A,B)_NO_es_controlable_(rank=%
d,<,n=%d).', rc, n);
54 end
55
56 % --- LQR discreto (con fallback) ---
57 if exist('dlqr','file') == 2
58 [K, P, e_cl] = dlqr(A, B, Q, R);
59 elseif exist('dare','file') == 2
60 [P,~,~] = dare(A,B,Q,R);
61 K = (R + B'*P*B)\(B'*P*A);
62 e_cl = eig(A - B*K);
63 else
64 [P, K, e_cl, info] = dlqr_iter_nolic(A,B,Q,R);
65 fprintf('DLQR_sin_toolbox:_iters=%d,_err=%.3e\n
', info.iters, info.err);
66 end
67
68 fprintf('LQR:>eig(A-BK)=\n'); disp(e_cl);
69
70 % --- Observador (place) ---
71 Ke_pred = place(A', C', p_obs).'; % predictor: A - Ke*C
72 Ke_act = place(A', (C*A)', p_obs).'; % "actual": A - Ke*C*A
73
74 % --- Nbar (SISO) ---
75 if size(C,1) ~= 1
76 error('Tu_C_no_es_SISO_(tiene_%d_salidas)._
Elegi_una_fila_de_C.', size(C,1));
77 end
78 [~,~,Nbar] = refi(A, B, C, K);
79 fprintf('Nbar=% .6g\n', Nbar);
80
81 % % =====
82 % 4) SIMULACION: SIN RUIDO vs CON RUIDO
83 (double y single)
84 % % =====
85 N = 200;
86 ulim_sat = u_max;
87 ulim_inf = Inf;
88
89 % referencia
90 r = zeros(1,N);
91 r(2:end) = r_step;
92
93 % ----- RUIDO (config) -----
94 rng(1); % repetible
95
96 noise.enable = true;
97 noise.sigma_y = 2; % ruido de medicion
98 noise.sigma_u = 2.0; % jitter actuador
99 noise.q_u = 1.0; % cuantizacion u
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
837
838
839
839
840
841
842
843
843
844
845
845
846
847
847
848
848
849
849
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
```

```

93 noise.sigma_w = 0.0; % ruido de proceso
94
95 noise.vy = noise.sigma_y * randn(1,N);
96 noise.vu = noise.sigma_u * randn(1,N);
97 noise.wx = noise.sigma_w * randn(n,N);
98
99 noise_off = noise;
100 noise_off.enable = false;
101 noise_off.vy = zeros(1,N);
102 noise_off.vu = zeros(1,N);
103 noise_off.wx = zeros(n,N);
104
105 % --- IDEAL (sin sat, sin ruido) ---
106 out_pred_ideal = sim_obs_loop(A,B,C,D,K,Nbar,
107 Ke_pred,Ke_act,r,Ts,ulim_inf,false,false,noise_off);
108 out_act_ideal = sim_obs_loop(A,B,C,D,K,Nbar,
109 Ke_pred,Ke_act,r,Ts,ulim_inf,true ,false,noise_off);
110
111 % --- REAL (sat, sin ruido) ---
112 out_pred_sat_clean = sim_obs_loop(A,B,C,D,K,
113 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,false,false,
114 noise_off);
115 out_act_sat_clean = sim_obs_loop(A,B,C,D,K,
116 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,true ,false,
117 noise_off);
118
119 % --- REAL (sat, con ruido) ---
120 out_pred_sat_noise = sim_obs_loop(A,B,C,D,K,
121 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,false,noise);
122 out_act_sat_noise = sim_obs_loop(A,B,C,D,K,
123 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,true ,noise);
124
125 % --- REAL (sat, con ruido) en FLOAT32 ---
126 out_pred_sat_noise_f = sim_obs_loop(A,B,C,D,K,
127 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,false,true,noise);
128 out_act_sat_noise_f = sim_obs_loop(A,B,C,D,K,
129 Nbar,Ke_pred,Ke_act,r,Ts,ulim_sat,true ,true,noise);
130
131 t = (0:N-1)*Ts;
132
133 % % =====
134 % 5) DIAGNOSTICOS
135 % % =====
136
137 eyp = out_pred_sat_noise.y_meas -
138 out_pred_sat_noise.y_true;
139 eya = out_act_sat_noise.y_meas -
140 out_act_sat_noise.y_true;
141
142 fprintf('\n----_CHECK_RUIDO_---\n');
143 fprintf('sigma_y=% .3g_\n', noise.sigma_y, rms(eyp),
144 rms(eya));
145 fprintf('sigma_u=% .3g_\n', noise.sigma_u, noise.q_u);
146
147 fprintf('\n----_SATURACION_---\n');
148
149 fprintf('Pred_clean:_maxlul= %.2f_\nsat= %d\n', max(
150 abs(out_pred_sat_clean.u)), sum(abs(
151 out_pred_sat_clean.u)) >= ulim_sat-1e-9));
152 fprintf('Pred_noise:_maxlul= %.2f_\nsat= %d\n', max(
153 abs(out_pred_sat_noise.u)), sum(abs(
154 out_pred_sat_noise.u)) >= ulim_sat-1e-9));
155 fprintf('Act_clean:_maxlul= %.2f_\nsat= %d\n', max(
156 abs(out_act_sat_clean.u)), sum(abs(out_act_sat_clean.
157 u)) >= ulim_sat-1e-9));
158 fprintf('Act_noise:_maxlul= %.2f_\nsat= %d\n', max(
159 abs(out_act_sat_noise.u)), sum(abs(out_act_sat_noise.
160 u)) >= ulim_sat-1e-9));
161
162 % % =====
163 % 6) PLOTS
164 % % =====
165 figure('Name','Predictor: y_(clean_vs_noise)_[sat]')
166 ;
167 plot(t, out_pred_sat_clean.y_true, 'LineWidth',1.6);
168 hold on;
169 plot(t, out_pred_sat_noise.y_meas, '-');
170 plot(t, r, 'k--','LineWidth',1.2);
171 grid on; xlabel('t_[s]'); ylabel('y');
172 legend('y_true_(clean)','y_meas_(noise)',r,'
173 Location','best');
174
175 figure('Name','Predictor: u_(clean_vs_noise)_[sat]')
176 ;
177 plot(t, out_pred_sat_clean.u, 'LineWidth',1.6); hold
178 on;
179 plot(t, out_pred_sat_noise.u, '-');
180 yline(+ulim_sat,k--'); yline(-ulim_sat,k--');
181 grid on; xlabel('t_[s]'); ylabel('u');
182
183 figure('Name','Actual: y_(clean_vs_noise)_[sat]');
184 plot(t, out_act_sat_clean.y_true, 'LineWidth',1.6);
185 hold on;
186 plot(t, out_act_sat_noise.y_meas, '-');
187 plot(t, r, 'k--','LineWidth',1.2);
188 grid on; xlabel('t_[s]'); ylabel('y');
189
190 figure('Name','Actual: u_(clean_vs_noise)_[sat]');
191 plot(t, out_act_sat_clean.u, 'LineWidth',1.6); hold
192 on;
193 plot(t, out_act_sat_noise.u, '-');
194 yline(+ulim_sat,k--'); yline(-ulim_sat,k--');
195 grid on; xlabel('t_[s]'); ylabel('u');
196
197 figure('Name','Ruido_de_medicion_(y_meas_-_
198 y_true)');
199 plot(t, eyp, t, eya);
200 grid on; xlabel('t_[s]'); ylabel('error_de_medicion');
201 legend('Pred','Act','Location','best');
202
203 figure('Name','Float32_-_Double_(y_meas)_[sat+_
204 noise]');

```

```

173 plot(t, out_pred_sat_noise.y_meas -  

174 out_pred_sat_noise.f.y_meas, ...  

175 t, out_act_sat_noise.y_meas - out_act_sat_noise.f.  

176 y_meas);  

177 grid on; xlabel('t,[s]'); ylabel('double_-_-float32');  

178 legend('Pred','Act','Location','best');  

179  

180 % % ======  

181 % 7) Z-PLANE (polos planta, CL, obs)  

182 % % ======  

183 p_ol = eig(A);  

184 p_cl = eig(A - B*K);  

185 p_op = eig(A - Ke_pred*C);  

186 p_oa = eig(A - Ke_act*C*A);  

187  

188 try  

189 z_plant = tzero(ss(A,B,C,D,Ts));  

190 catch  

191 z_plant = [];  

192 end  

193  

194 plot_zplane('Z-plane:_Predictor', p_ol, p_cl, p_op,  

195 z_plant);  

196 plot_zplane('Z-plane:_Actual' , p_ol, p_cl, p_oa,  

197 z_plant);  

198  

199 % %  

200  

201 % ====== FUNCIONES  

202 % LOCALES ======  

203  

204 function out = sim_obs_loop(A,B,C,D,K,Nbar,  

205 Ke_pred,Ke_act,r,Ts,ulim,use_actual,use_single,noise)  

206 if nargin < 15 || isempty(noise)  

207 noise.enable=false; noise.vy=0; noise.vu=0; noise.  

208 wx=0; noise.q_u=Inf;  

209 end  

210  

211 if use_single  

212 A=single(A); B=single(B); C=single(C); D=single(  

213 D);  

214 K=single(K); Nbar=single(Nbar);  

215 Ke_pred=single(Ke_pred); Ke_act=single(Ke_act);  

216 r=single(r); Ts=single(Ts); ulim=single(ulim);  

217 end  

218  

219 n = size(A,1); N = numel(r);  

220  

221 x = zeros(n,N,'like',A);  

222 xh = zeros(n,N,'like',A);  

223  

224 y_true = zeros(1,N,'like',A);  

225 y_meas = zeros(1,N,'like',A);  

226  

227 u_app = zeros(1,N,'like',A);  

228 vy = zeros(1,N,'like',A);  

229 vu = zeros(1,N,'like',A);  

230 wx = zeros(n,N,'like',A);  

231  

232 if isfield(noise,'enable') && noise.enable  

233 vy = cast(noise.vy,'like',A);  

234 vu = cast(noise.vu,'like',A);  

235 wx = cast(noise.wx,'like',A);  

236 end  

237  

238 q_u = Inf;  

239 if isfield(noise,'q_u'), q_u = noise.q_u; end  

240 q_u = cast(q_u,'like',A);  

241  

242 y_true(1) = C*x(:,1) + D*0;  

243 y_meas(1) = y_true(1) + vy(1);  

244  

245 for k=1:N-1  

246 u_unsat = Nbar*r(k) - K*xh(:,k);  

247 u_cmd(k) = sat(u_unsat, ulim);  

248  

249 u_app(k) = u_cmd(k) + vu(k);  

250 if isnan(double(q_u))  

251 u_app(k) = round(u_app(k)/q_u)*q_u;  

252 end  

253  

254 u_app(k) = sat(u_app(k), ulim);  

255 x(:,k+1) = A*x(:,k) + B*u_app(k) + wx(:,k);  

256  

257 y_true(k) = C*x(:,k) + D*u_app(k);  

258 y_true(k+1) = C*x(:,k+1) + D*u_app(k);  

259  

260 y_meas(k) = y_true(k) + vy(k);  

261 y_meas(k+1) = y_true(k+1) + vy(k+1);  

262  

263 if ~use_actual  

264 yhat_k = C*xh(:,k) + D*u_app(k);  

265 xh(:,k+1) = A*xh(:,k) + B*u_app(k) + Ke_pred*(  

266 y_meas(k) - yhat_k );  

267 else  

268 z = A*xh(:,k) + B*u_app(k);  

269 yzh = C*z + D*u_app(k);  

270 xh(:,k+1) = z + Ke_act*( y_meas(k+1) - yzh );  

271 end  

272  

273 u_cmd(N) = sat(Nbar*r(N) - K*xh(:,N), ulim);  

274 u_app(N) = u_cmd(N) + vu(N);  

275 if isnan(double(q_u))  

276 u_app(N) = round(u_app(N)/q_u)*q_u;  

277 end  

278 u_app(N) = sat(u_app(N), ulim);  

279  

280 y_true(N) = C*x(:,N) + D*u_app(N);

```

<pre> 274 y_meas(N) = y_true(N) + vy(N); 275 276 out.x = x; out.xh = xh; 277 out.y_true = y_true; 278 out.y_meas = y_meas; 279 out.u_cmd = u_cmd; 280 out.u_app = u_app; 281 282 out.y = y_meas; 283 out.u = u_app; 284 <b>end</b> 285 286 <b>function</b> y = sat(u,lim) 287 <b>if</b> isnan(lim) 288 y = u; 289 <b>else</b> 290 y = min(max(u, -lim), lim); 291 <b>end</b> 292 <b>end</b> 293 294 <b>function</b> plot_zplane(figName, p_ol, p_cl, p_obs, 295 z_plant) 296 figure('Name',figName,'NumberTitle','off'); 297 hold on; grid on; grid minor; axis equal; 298 title('Z-plane'); 299 xlabel('Re{z}'); ylabel('Im{z}'); 300 301 th = linspace(0,2*pi,400); 302 plot(cos(th), sin(th), 'k:'); % unit circle 303 304 plot(real(p_ol), imag(p_ol), 'o', 'LineWidth', 1.5); 305 plot(real(p_cl), imag(p_cl), 'x', 'LineWidth', 1.8); 306 plot(real(p_obs), imag(p_obs), '^', 'LineWidth', 1.8); 307 308 <b>if</b> ~isempty(z_plant) 309 plot(real(z_plant), imag(z_plant), 's', 'LineWidth', 310 1.5); 311 legend('unit_circle','poles_plant_(A)','poles_CL_(A 312 -BK)','poles_obs','zeros_plant','Location','bestoutside'); 313 <b>else</b> 314 legend('unit_circle','poles_plant_(A)','poles_CL_(A 315 -BK)','poles_obs','Location','bestoutside'); 316 <b>end</b> 317 318 xlim([-1.2 1.2]); ylim([-1.2 1.2]); 319 <b>end</b> 320 321 <b>function</b> [Nx,Nu,Nbar] = refi(phi,gam,Hr,K) 322 I=eye(size(phi)); 323 [m,n]=size(Hr); 324 np=inv([phi-I gam;Hr zeros(m)])*([zeros(n,m);eye( 325 m)]); 326 Nx=np(1:n,:); 327 Nu=np(n+1:n+m,:); 328 Nbar=Nu+K*Nx; 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 </pre>	<pre> 324 <b>end</b> 325 326 <b>function</b> [P, K, e_cl, info] = dlqr_iter_nolic(A,B,Q, 327 R) 328 maxit = 5000; 329 tol = 1e-10; 330 P = Q; 331 err = Inf; 332 333 <b>for</b> it = 1:maxit 334 G = R + B'*P*B; 335 Ktmp = G \ (B'*P*A); 336 Pn = A'*P*A - A'*P*B*Ktmp + Q; 337 338 err = norm(Pn - P, 'fro'); 339 P = Pn; 340 341 <b>if</b> err &lt; tol 342 <b>break</b>; 343 <b>end</b> 344 <b>end</b> 345 346 K = (R + B'*P*B) \ (B'*P*A); 347 e_cl = eig(A - B*K); 348 349 info.iters = it; 350 info.err = err; 351 info.converged = (err &lt; tol); 352 353 <b>if</b> ~info.converged 354 warning('Riccati_iterativa_NO_convergio_(err=%. 3 e)', err); 355 <b>end</b> 356 <b>end</b> </pre>
--	--

Listing 2. LQR + Observador (predictor/actual) + Nbar + Simulacion con saturacion y ruido

En este apéndice se presenta el script utilizado para sintonizar la covarianza del ruido de proceso del filtro de Kalman manteniendo fija la covarianza de medición.

Se parametriza la matriz de proceso como  $Q = qI_n$  y se fija  $R = \sigma_v^2$ , donde  $\sigma_v$  se obtuvo a partir de mediciones empíricas del sensor (planta en reposo). El escalar  $q$  se selecciona mediante un criterio de consistencia estadística sobre la innovación normalizada.

La innovación se define como  $\nu_k = y_k - \hat{y}_{k|k-1}$  y su varianza teórica estacionaria como  $S = CPC^T + R$ , donde  $P$  es la solución estacionaria de Riccati. Se utiliza:

$$\eta_k = \frac{\nu_k}{\sqrt{S}}$$

y se escoge el valor  $q$  que aproxima  $\text{var}(\eta_k) \approx 1$ .

#### A. Script de sintonización (RQ\_tuning\_fixedR.m)

A continuación se incluye el código completo empleado para el barrido de  $q$  en escala logarítmica, el cálculo de

la ganancia estacionaria con `dlqe` y el guardado de los parámetros resultantes en `RQ_tuning_fixedR.mat`.

```

1 % %
2 % Tuning de Q = q*I para Kalman (dlqe) con R
3 % FIJO (sigma_v ~ 2..3 cm)
4 % usando datos reales (dato1,dato2) desde
5 % iddata.
6 %
7 % - Planta: BJ continuo en planta (1).mat
8 % -> G = B/F
9 % - Datos : datos.mat con dato1,dato2 (
10 % iddata) con y en cm, u en du_us
11 % - Ts_target: 0.01 s (decimación exacta si
12 % Ts original es divisor)
13 %
14 % Objetivo: var(eta) ~ 1 donde eta = nu /
15 % sqrt(S), S = C P C' + R
16 % (criterio consistente para elegir q cuando
17 % el modelo no es perfecto)
18 % %
19 close all; clear; clc
20 %
21 % % ===== rutas =====
22 mat_planta = 'planta_(1).mat';
23 mat_datos = 'D:\GitHub\auto\TPF-
24 % HelicopteroVertical\Matlab\
25 % PruebasEmpiricas\datos.mat';
26 %
27 Ts_target = 0.01; % 100 Hz
28 %
29 % % =====
30 % 1) CARGAR PLANTA Y DISCRETIZAR
31 % %
32 S = load(mat_planta);
33 %
34 if isfield(S,'plantaC')
35 plantaC = S.plantaC;
36 elseif isfield(S,'sysC')
37 plantaC = S.sysC;
38 else
39 error('No_encuentro_"plantaC"_ni_"sysC"_'
40 'dentro_de_%s', mat_planta);
41 end
42 %
43 % Planta determinista desde BJ: G = B/F
44 G = tf(plantaC.B, plantaC.F);
45 Gd = c2d(ss(G), Ts_target, 'zoh');
46 [A,B,C,D] = ssdata(Gd);
47 %
48 n = size(A,1);
49 if size(B,2) ~= 1
50 error('SISO_requerido._size(B,2)=%d', size(B
51 ,2));
52 end
53 if size(C,1) ~= 1
54 C = C(1,:);
55 D = D(1,:);
56 end
57 if isempty(D), D = 0; end
58 D = double(D);
59 %
60 fprintf('Planta_discretizada:_n=%d,_
61 Ts_target=%.6g\n', n, Ts_target);
62 %
63 % % =====
64 % 2) CARGAR DATOS (iddata) Y ARMAR u,y (

```

```

concatenados)
% =====
54 DD = load(mat_datos);
55 use_names = {'dato1','dato2'};
56
57 u_all = [];
58 y_all = [];
59
60 for i=1:numel(use_names)
61 nm = use_names{i};
62 if ~isfield(DD,nm)
63 error('No existe %s en %s', nm, mat_datos);
64 end
65
66 zi = DD.(nm);
67 if ~isa(zi,'iddata')
68 error('%s no es iddata.', nm);
69 end
70
71 Ts_i = zi.Ts;
72 if isempty(Ts_i) || Ts_i <= 0
73 error('%s: iddata sin Ts válido.', nm);
74 end
75
76 % señales crudas
77 yraw = zi.OutputData(:);
78 uraw = zi.InputData(:);
79
80 % limpiar NaN/Inf (sin isfinite por compat)
81 m = ~isnan(yraw) & ~isinf(yraw) & ~isnan(uraw)
82 yraw = yraw(m);
83 uraw = uraw(m);
84
85 % decimación exacta a Ts_target
86 if abs(Ts_i - Ts_target) > 1e-12
87 ratio = Ts_target / Ts_i;
88 if abs(ratio - round(ratio)) > 1e-12
89 error('%s: Ts_target/Ts_i no entero.%s_i
90 =%.17g,Ts_target=%.17g', nm, Ts_i,
91 Ts_target);
92 end
93 M = round(ratio);
94 fprintf('%s: decimación M=%d (Ts %.17g->%
95 %.17g)\n', nm, M, Ts_i, Ts_target);
96 yraw = yraw(1:M:end);
97 uraw = uraw(1:M:end);
98 else
99 fprintf('%s: ya coincide (%.17g)\n', nm,
100 Ts_i);
101 end
102
103 % (opcional) detrend: sacá media para no
104 % pelear con offsets
105 yraw = yraw - mean(yraw);
106 uraw = uraw - mean(uraw);
107
108 fprintf('%s: N=%d\n', nm, numel(yraw));
109
110 % concatenar
111 y_all = [y_all; yraw(:)]; %#ok<AGROW>
112 u_all = [u_all; uraw(:)]; %#ok<AGROW>
113 end
114
115 y = y_all;
116 u = u_all;
117
```

```

113 fprintf('TOTAL: N=%d muestras\n', numel(y));
114 % =====
115 % 3) FIJAR R (ruido de medición) y TUNEAR q
116 % en Q=q*I
117 % =====
118 sigma_v = 2.043227851; % cm (poné 2..3
119 % según tu medición)
120 R = sigma_v^2;
121 q_grid = logspace(-12, 2, 120);
122
123 best_q = NaN; best_err = Inf;
124 stats_q = zeros(numel(q_grid),1);
125 stats_var_eta = zeros(numel(q_grid),1);
126 stats_var_nu = zeros(numel(q_grid),1);
127
128 for i=1:numel(q_grid)
129 q = q_grid(i);
130 Q = q*eye(n);
131
132 % Kalman (w entra a estados)
133 Gk = eye(n);
134 [L,P,~] = dlqe(A,Gk,C,Q,R);
135
136 % Sinnov estacionario
137 Sinnov = C*P*C' + R;
138 if Sinnov < 1e-12, Sinnov = 1e-12; end
139
140 xhat = zeros(n,1);
141 nu = zeros(numel(y),1);
142
143 for k=1:numel(y)
144 ypred = C*xhat + D*u(k);
145 nu(k) = y(k) - ypred;
146
147 % delayed estimator estilo firmware
148 xhat = xhat + L*nu(k);
149 xhat = A*xhat + B*u(k);
150 end
151
152 eta = nu / sqrt(Sinnov);
153
154 var_eta = var(eta,1);
155 var_nu = var(nu,1);
156
157 stats_q(i) = q;
158 stats_var_eta(i) = var_eta;
159 stats_var_nu(i) = var_nu;
160
161 err = abs(log(var_eta)); % objetivo
162 % var_eta ~ 1
163 if err < best_err
164 best_err = err;
165 best_q = q;
166 best_L = L;
167 best_P = P;
168 best_S = Sinnov;
169 end
170 end
171
172 fprintf('\n====_Resultado_tuning_Q_con_R_fijo_'
173 '==\n');
174 fprintf('sigma_v=% .3f_cm=>_R= %.3f_cm^2\n',
175 % .3f_cm\n', best_S, sqrt(best_S));
176 fprintf('L_best=%n'); disp(best_L);
177 % =====
178 % 4) PLOTS
179 % =====
180 figure('Name','Tuning_Q:var(eta)');
181 semilogx(stats_q, stats_var_eta,'LineWidth'
182 ,1.2); grid on; grid minor;
183 xlabel('q'); ylabel('var(\eta)'); title('
184 Objetivo: var(\eta)_approx_1');
185
186 figure('Name','Tuning_Q:sigma(nu)');
187 semilogx(stats_q, sqrt(stats_var_nu),''
188 'LineWidth',1.2); grid on; grid minor;
189 xlabel('q'); ylabel('sigma(\nu)_[cm]'); title(
190 'Innovación_(predicción)_vs_q');
191
192 % =====
193 % 5) Guardar
194 % %
195 Q_best = best_q*eye(n);
196 L_best = best_L;
197 save('RQ_tuning_fixedR.mat','sigma_v','R',
198 'best_q','Q_best','L_best','best_P',
199 'best_S','A','B','C','D','Ts_target');
200
201 disp('Guardado:_RQ_tuning_fixedR.mat');

```

### B. Parámetros exportados para el diseño LQG

El script guarda el archivo `RQ_tuning_fixedR.mat` con los parámetros utilizados posteriormente en el diseño del estimador y controlador:

- `sigma_v`: desviación estándar empírica del ruido de medición.
- `R`: covarianza del ruido de medición ( $R = \sigma_v^2$ ).
- `best_q`: escalar óptimo de la parametrización del ruido de proceso.
- `Q_best`: matriz final de ruido de proceso ( $Q = qI_n$ ).
- `L_best`: ganancia estacionaria calculada con `dlqe`.
- `best_P` y `best_S`: covarianza estacionaria y varianza de innovación.
- `A, B, C, D, Ts_target`: modelo discretizado usado en la sintonización.

En este apéndice se incluye el script utilizado para obtener los parámetros numéricos del esquema LQG/LQI empleado en el trabajo: ganancia del filtro de Kalman en régimen permanente  $L$ , ganancias de realimentación de estados  $K_x$  y del integrador  $K_i$ , junto con la verificación de polos del observador  $\lambda(A-LC)$  y la simulación en presencia de ruido de proceso y medición.

El script fue construido con el objetivo de ser *firmware friendly*, es decir, replicar la lógica de estimación y control que luego se implementa en el microcontrolador:

- **Estimador tipo `current`**: se calcula  $L$  con `kalman(...,'current')`.
- **Ruido de proceso y medición**: se inyecta  $w_k$  en la dinámica del estado y  $v_k$  en la medición.
- **Control con integrador**: se emplea una acción integral  $\xi_k$  para asegurar error estacionario nulo.

### C. Estructura del diseño

*C0a. Planta discretizada:* A partir del modelo continuo identificado (planta (1).mat) se discretiza con ZOH a  $T_s = 0.01$  s (100 Hz), obteniendo el sistema discreto:

$$x_{k+1} = Ax_k + Bu_k, \quad y_k = Cx_k + Du_k$$

*C0b. Covarianzas de ruido:* Las covarianzas utilizadas por el estimador se toman del archivo RQ\_tuning\_fixedR.mat (Apéndice A):

$$Q_n = Q_{\text{best}}, \quad R_n = R$$

Adicionalmente, para la simulación se utilizan las desviaciones estándar:  $\sigma_v$  (medición) y  $\sigma_w$  (proceso), e inyección de ruido blanco gaussiano.

*C0c. Filtro de Kalman (current estimator):* Se construye un sistema extendido para el cálculo de Kalman, incorporando explícitamente el canal de ruido de proceso  $w$ :

$$x_{k+1} = Ax_k + Bu_k + Gw_k, \quad y_k = Cx_k + Du_k + Hw_k + v_k$$

En este trabajo se asumió:

$$G = I_n, \quad H = 0$$

y se obtiene la ganancia  $L$  en régimen permanente (y sus polos):

$$L = L_k, \quad \lambda(A - LCA)$$

*C0d. Control LQI con integrador:* Se incorpora un integrador escalar  $\xi_k$  sobre el error:

$$\xi_{k+1} = \xi_k + (r_k - y_k)$$

y se adopta la ley de control:

$$u_k = -K_x \hat{x}_k + K_i \xi_k$$

Las ganancias se obtienen mediante dlqr aplicado al sistema aumentado, usando pesos heurísticos basados en escalas prácticas: paso deseado  $\Delta y \approx 20$  y esfuerzo máximo  $|u| \lesssim 300$ .

### D. Script completo (LQG\_servo\_firmware\_friendly.m)

A continuación se incluye el script completo utilizado para calcular  $L$ ,  $K_x$ ,  $K_i$ , reportar polos y simular en presencia de ruido.

```
% % =====
% LQG SERVO "firmware friendly"
% (kalman + lqi + lqgtrack)
% + estimador CURRENT (default en
discreto)
% + ruido de proceso y medición (
sigma_w, sigma_v)
% % =====
close all; clear; clc
% % =====
```

```
%
% 1) CARGA + DISCRETIZACIÓN
% =====
S = load('planta_(1).mat');

if isfield(S,'plantaC')
    plantaC = S.plantaC;
elseif isfield(S,'sysC')
    plantaC = S.sysC;
else
    error('No_encuentro_plantaC_ni_
sysC_dentro_de_planta_(1).mat');
end

Ts = 1/100; % 100
Hz

sysD = c2d(plantaC, Ts, 'zoh');
[A,B,C,D] = ssdata(ss(sysD));

n = size(A,1);
if size(C,1) ~= 1
    C = C(1,:);
    D = D(1,:);
end
if size(B,2) ~= 1
    error('Este_script_asume_SISO_(1_
entrada).size(B,2)=%d', size(B,2));
end

fprintf('Ts=% .9f | n=%d\n', Ts, n);

%
% =====
% 2) RUIDOS (sigma -> covarianzas)
% =====
S = load("RQ_tuning_fixedR.mat");

Qn = S.Q_best; % cov(w) : nxn
Rn = S.R; % cov(v) : 1x1
sigma_v = S.sigma_v;
sigma_w = S.best_q;
Nn = zeros(n,1); % cov(wv')
asumimos 0

%
% =====
% 3) KALMAN: sys debe incluir el
canal de ruido w
% =====
G = eye(n); % w entra a todos
los estados
H = zeros(1,n); % y NO depende
directamente de w (H=0)

sysK = ss(A, [B G], C, [D H], Ts);
% inputs: [u ; w]

% Para tu MATLAB: TYPE va como ÚLTIMO argumento.
% 'current' es el default, pero lo
dejamos explícito.
[kest, Lk, Pk, Mx, Z, My] = kalman(
    sysK, Qn, Rn, Nn, 'current');

fprintf('\nLk_(kalman)_=%'), disp(Lk)
;
fprintf('poles(A-LC)_=%'), disp(eig(A -
    Lk*C).');
```

```

% % =====
% 4) LQI: K = [Kx Ki] para u = -Kx*xhat - Ki*x_i
% % =====
m = 1; %%
integrador escalar
Ahat = [A B; zeros(m, n+m) ];
Bhat = [zeros(n, m); eye(m) ];
Chat = [C zeros(1, m) ];

% --- interpretación física: "step ~20" y "u no pase ~300" ---
y_step = 20;
u_max = 300;

% knobs
wy = 20; % subí
=> seguís más (más agresivo)
wu = 50; % subí
=> penalizás más u (más tímido)
wv = 0.0000000000000001; % subí
=> penalizás integrador

% OJO: acá tu "escala física" es discutible. Esto es heurístico.
qy = wy*(1/(y_step));
ru = wu*(1/(u_max));
qv = wv*(1/(y_step));

Qa = blkdiag(qy*(C'*C) + 1e-8*eye(n),
qv);
R = ru;

[Khat, Pa, ecl] = dlqr(Ahat, Bhat, Qa,
R);

Aux = [A-eye(size(A)) B;
C*A C*B];

K2K1 = (Khat + [zeros(1, n) eye(m)]) /
Aux;

Kx = K2K1(1, 1:n);
Ki = K2K1(1, n+1);

fprintf('\nKx_(lqi)_='); disp(Kx);
fprintf('Ki_(lqi)_='); disp(Ki);

% % =====
% 6) SIMULACIÓN (tu estilo) con ruido de proceso+medición
% % =====
N = 1200;
t = 0:Ts:(N-1)*Ts;

r = ones(1, N)*25;
r(1:30) = 0;

umin = -u_max; umax = u_max;

X = zeros(n, N); % estados reales
Xh = zeros(n, N); % estados estimados (los que usás en tu control)
V = zeros(1, N); % integrador
U = zeros(1, N); % control

Y_true = zeros(1, N);
Y_meas = zeros(1, N);

for kidx = 1:N-1
rk = r(kidx);

% medición y[k]
y_true_k = C*X(:, kidx);
v_k = sigma_v*randn;
y_meas_k = y_true_k + v_k;

Y_true(kidx) = y_true_k;
Y_meas(kidx) = y_meas_k;

% integrador xi[k+1] = xi[k] + (r-y)
V(kidx+1) = V(kidx) + (rk - y_meas_k)
;

% control: u = -Kx*xhat - Ki*x_i
u_unsat = -Kx*Xh(:, kidx) + Ki*V(kidx+1);
u_k = min(max(u_unsat, umin), umax);
U(kidx) = u_k;

% planta con ruido de proceso: x[k+1] = A x + B u + w
w_k = sigma_w*randn(n, 1);
X(:, kidx+1) = A*X(:, kidx) + B*u_k +
w_k;

% estimador (usa Lk)
Xh(:, kidx+1) = A*Xh(:, kidx) + B*u_k +
...
Lk*(y_meas_k - (C*Xh(:, kidx) + D*u_k))
);
end

% último sample para plots
Y_true(N) = C*X(:, N);
Y_meas(N) = Y_true(N) + sigma_v*randn
;
U(N) = U(N-1);

% % =====
% 7) PLOTS
% % =====
figure('Name', 'y(t)--LQG+integrador+ruido');
plot(t, Y_true, 'LineWidth', 1.2);
hold on;
plot(t, Y_meas, '.', 'LineWidth', 1.0);
plot(t, r, 'k--', 'LineWidth', 1.1);
grid on; grid minor;
xlabel('t[s]'); ylabel('y');
legend('y_true', 'y_meas', 'r', 'Location', 'best');

figure('Name', 'u(t)--saturado');
plot(t, U, 'LineWidth', 1.4);
grid on; grid minor;
xlabel('t[s]'); ylabel('u');
ylim([umin umax]);

figure('Name', 'xi(t)--integrador');
plot(t, V, 'LineWidth', 1.4);

```

```

172 grid on; grid minor;
173 xlabel('t[s]'); ylabel('\xi');
174
175 Aaug = [A,B;...
176 Kx-Kx*A-Ki*C*A, 1-Kx*B-Ki*C*B];
177 Baug = [0;0;0;Ki];
178 Caug = [C,0];
179 D = 0;
180 sysDaug = ss(Aaug,Baug,Caug,D,Ts);
181
182 figure; pzmap(sysDaug); grid on;
zgrid;

```

Durante el desarrollo del trabajo práctico, la estructura física de la planta atravesó distintas etapas de diseño, las cuales permitieron identificar limitaciones mecánicas y realizar mejoras progresivas hasta alcanzar la configuración final utilizada en las prácticas experimentales. En este apéndice se describe la primera etapa de diseño de la estructura y se destacan las principales diferencias respecto de la versión final.

#### E. Primera etapa de diseño

La primera versión de la estructura fue concebida con una altura total aproximada de 80 cm, utilizando la misma base y el mismo techo de madera que se mantienen en el diseño final. Debido a las dimensiones de estos elementos, la altura útil de movimiento del cuerpo móvil en esta etapa era de aproximadamente 72 cm.

En esta configuración inicial, el diseño mecánico del cuerpo móvil era diferente al actual, presentando dimensiones ligeramente mayores. El sistema no contaba con elementos de seguridad adicionales, tales como topes mecánicos, amortiguación ante caídas ni cuerda de seguridad, dado que el recorrido vertical era considerablemente menor y el riesgo asociado a caídas desde grandes alturas resultaba limitado.

El guiado del cuerpo móvil se realizaba mediante vigas metálicas rectas y rígidas, las cuales no presentaban deformaciones apreciables. Debido a esta rigidez estructural, no fue necesario incorporar articulaciones pasivas tipo “muñeca” en las abrazaderas, ni estructuras auxiliares de madera para limitar deformaciones. En esta etapa, el contacto entre el cuerpo móvil y los rieles generaba fricción apreciable, la cual se manifestaba de forma consistente durante el movimiento vertical.

Cabe destacar que este comportamiento friccional, observable en la primera versión de la estructura, no se presenta de la misma manera en el diseño final. La incorporación de vigas metálicas de mayor longitud, junto con las deformaciones inherentes a las mismas y la inclusión de articulaciones pasivas en las abrazaderas, redujo significativamente la fricción directa entre el cuerpo móvil y los rieles, modificando así las características mecánicas del sistema.

En las figuras siguientes se presentan imágenes correspondientes a las primeras versiones de las piezas impresas en 3D utilizadas en esta etapa inicial del diseño, las cuales difieren de las empleadas en la configuración final de la planta.

En la primera etapa de diseño, el soporte superior del cuerpo móvil presentaba una altura aproximada de 3 cm y un diámetro de 4 cm. Las secciones sobresalientes destinadas al acople de

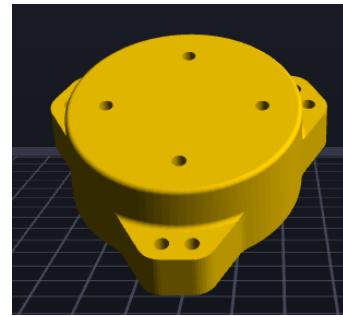


Figura 35. Primer diseño del soporte superior del motor.

los brazos contaban con una altura de aproximadamente 1,5 cm. Dicho soporte incluía orificios dimensionados específicamente para el montaje del motor brushless, con un diámetro de 3 mm, acorde al patrón de fijación del mismo.

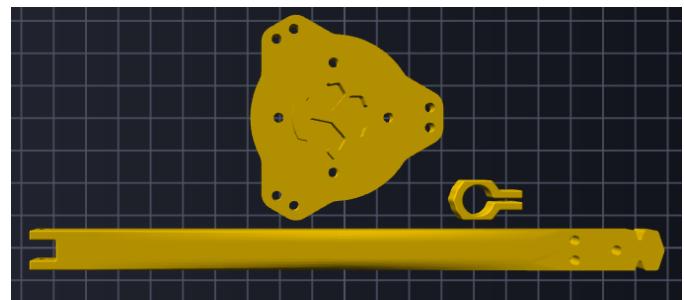


Figura 36. Primeros diseños del soporte inferior y de los brazos estructurales.

El soporte inferior del cuerpo móvil correspondía a una geometría espejo del soporte superior. En este componente se realizaba el encastre de los brazos estructurales, los cuales, en esta etapa inicial, presentaban dimensiones menores en comparación con el diseño final. Cada brazo tenía dimensiones aproximadas de 1 cm × 1 cm × 20 cm.

El sistema de agarre de los brazos difería del implementado en la versión final de la planta. Inicialmente, el agarre no contaba con movilidad angular, aunque permitía un ajuste manual respecto a la posición de la viga metálica, lo que condicionaba el guiado del cuerpo móvil y su interacción con los rieles.

#### F. Segunda etapa de diseño

En la segunda etapa de diseño, el componente que presentó mayores modificaciones fue el brazo estructural. A partir de la experiencia obtenida en la etapa inicial, se introdujeron variaciones geométricas en el diseño del brazo, incorporando curvaturas con el objetivo de mejorar el encastre y la interacción con la estructura de guiado.

En esta versión, el brazo y el sistema de agarre fueron integrados en una única pieza impresa en 3D, eliminando la separación entre ambos componentes. Dado que en esta etapa las vigas metálicas utilizadas como rieles presentaban una geometría recta y una rigidez suficiente, no fue necesaria la incorporación de articulaciones pasivas tipo “muñeca”. En

consecuencia, el guiado del cuerpo móvil se realizaba mediante un agarre rígido, sin movilidad angular.

Cabe destacar que, durante esta etapa, el diseño del cuerpo móvil se mantuvo sin modificaciones significativas respecto a la versión anterior. Las mejoras se concentraron exclusivamente en el diseño de los brazos y del sistema de agarre, manteniendo constante la geometría general del conjunto móvil.



Figura 37. Segundo diseño de brazo.

#### G. Tercera etapa de diseño

En la tercera etapa de diseño se introdujo una modificación significativa en la estructura general de la planta, extendiendo su altura máxima hasta aproximadamente 160 cm. Esta ampliación respondió a la necesidad de disponer de un mayor recorrido vertical para la realización de las prácticas de control, lo cual implicó nuevas exigencias mecánicas sobre el conjunto estructural y el cuerpo móvil.

Como consecuencia del aumento de altura de la estructura, el diseño de los brazos del cuerpo móvil volvió a ser modificado. En esta etapa, los brazos fueron rediseñados con mayor grosor y mayor altura, con el objetivo de incrementar su rigidez y capacidad de carga. Asimismo, se incorporaron aberturas longitudinales en los brazos, destinadas a permitir la inserción de elementos metálicos, con el fin de reforzar la estructura y mejorar su resistencia mecánica frente a esfuerzos y vibraciones.

El cuerpo móvil mantuvo su configuración general respecto a las etapas anteriores; sin embargo, el rediseño de los brazos resultó fundamental para adaptar el conjunto a las nuevas condiciones estructurales impuestas por la mayor altura de la planta.

#### H. Cuarta etapa de diseño: configuración final

La cuarta etapa de diseño corresponde a la configuración final de la estructura y del cuerpo móvil utilizada en las prácticas experimentales del trabajo. En esta etapa se introdujeron modificaciones orientadas principalmente a mejorar el guiado mecánico del cuerpo móvil y a reducir la masa total del conjunto.

Debido a que las vigas metálicas empleadas como rieles presentan deformaciones asociadas a su longitud, se incorporaron articulaciones pasivas tipo “muñeca” en el sistema de guiado. Estas articulaciones permiten un movimiento angular relativo entre el cuerpo móvil y los rieles, mejorando el desplazamiento vertical y evitando atascamientos o esfuerzos indeseados durante el recorrido.

Con el objetivo de reducir la masa del cuerpo móvil, se redimensionaron los soportes principales. El soporte superior, que en versiones anteriores presentaba una altura de 3 cm, fue



Figura 38. Tercer diseño de brazos.

reducido a aproximadamente 1,5 cm, mientras que el soporte inferior pasó de 2 cm a 0,5 cm. A pesar de esta reducción dimensional, se conservó el sistema de encastre tanto en el soporte superior como en el inferior, asegurando la rigidez estructural del conjunto.

Adicionalmente, se incorporó un soporte específico para la batería, integrado al cuerpo móvil. En dicho soporte se colocaron almohadillas internas con el fin de proteger la batería frente a vibraciones e impactos durante el funcionamiento del sistema.

Estas modificaciones permitieron obtener un diseño final más liviano, adaptable a las deformaciones estructurales de los rieles y adecuado para la implementación de las distintas estrategias de control desarrolladas en el presente trabajo.

Durante el desarrollo experimental del trabajo se presentaron fallas en los controladores electrónicos de velocidad (ESC) utilizados en las primeras etapas de prueba del sistema. En este apéndice se describen los incidentes observados, junto con el análisis de las posibles causas y las medidas adoptadas posteriormente.

#### I. Primer incidente: ESC de 30A con alimentación externa

En una primera instancia, se utilizó un ESC de 30 A alimentado mediante una batería para automóviles, con el objetivo de verificar el funcionamiento básico del sistema de propulsión. La conexión entre la fuente de alimentación y el ESC se realizó utilizando un cable unifilar de cobre de considerable longitud.



Figura 39. Tercer diseño de brazos.

Durante las pruebas iniciales, el sistema logró generar empuje y el cuerpo móvil llegó a elevarse. Sin embargo, tras un período de funcionamiento, el ESC comenzó a emitir una secuencia de señales acústicas consistente en cuatro pitidos cortos seguidos de un pitido largo. En ese momento no se contaba con una interpretación clara del significado de dicha señalización.

Posteriormente, mediante la consulta de documentación y experiencias previas, se determinó que dicha secuencia de pitidos está asociada a condiciones de protección del ESC, tales como sobrecorriente o sobretemperatura. Esta hipótesis se vio reforzada por el hecho de que los cables unifilares utilizados para la alimentación se calentaron excesivamente y llegaron a derretirse, indicando una circulación de corriente elevada y pérdidas resistivas significativas.

#### *J. Segundo incidente: reinicios y falla del ESC de 30 A*

En una segunda etapa de pruebas con el mismo ESC de 30 A, se reemplazaron los cables de alimentación por conductores adecuados para altas corrientes, conectando el ESC directamente a la batería utilizada en la planta. En esta configuración, el sistema no lograba elevarse de forma sostenida y el ESC emitía una secuencia de sonidos correspondiente a un reinicio del controlador.

Con el fin de descartar un problema en la señal de control, se analizó la señal PWM generada por el PSoC mediante un osciloscopio, verificándose que la misma presentaba una forma adecuada y estable, sin perturbaciones significativas. En consecuencia, se descartó que la falla estuviera asociada a errores en la generación de la señal de control.

Ante la hipótesis de una posible caída de tensión en la alimentación del ESC durante los transitorios de corriente, se incorporaron capacitores de desacople en la línea de alimentación. Tras esta modificación, el sistema logró generar empuje y elevarse durante breves instantes. No obstante, luego de un corto período de funcionamiento, se produjo la falla definitiva del ESC, observándose la quema de un MOSFET correspondiente a una de las fases del motor.

#### *K. Análisis y consideraciones*

A partir de los incidentes descritos, se identificaron como causas probables la combinación de sobrecorriente, exigencias térmicas elevadas y condiciones de alimentación no ideales durante las primeras pruebas. La utilización de una fuente de alimentación inadecuada, conductores con alta resistencia y la ausencia inicial de medidas de protección contribuyeron a someter al ESC a esfuerzos superiores a sus límites operativos.

Estos eventos pusieron de manifiesto la importancia de considerar cuidadosamente los aspectos de potencia, dissipación térmica y protección eléctrica en sistemas de propulsión basados en motores brushless, incluso en etapas preliminares de prueba.

Las lecciones aprendidas a partir de estas fallas motivaron la adopción de controladores de mayor capacidad de corriente, mejoras en el cableado de alimentación y la implementación de estrategias de operación más conservadoras, las cuales permitieron continuar con el desarrollo experimental del trabajo de manera segura y confiable.

#### *L. Tercer incidente: falla del ESC de 30 A durante operación con batería LiPo*

En un tercer incidente, se utilizó un ESC de 30 A alimentado mediante una batería LiPo para drones. Durante esta prueba, se incrementó la señal PWM hasta aproximadamente  $1500 \mu\text{s}$ , logrando que el sistema generara empuje suficiente para elevar el cuerpo móvil hasta la parte superior de la estructura.

Al intentar detener el movimiento, se adoptó un procedimiento no óptimo, consistente en bloquear mecánicamente la hélice con el fin de evitar una colisión con el techo de la estructura. Esta acción provocó el trabado de la hélice durante el funcionamiento del motor, lo cual generó un incremento abrupto de la corriente demandada. Como consecuencia, el ESC sufrió una falla catastrófica, produciéndose la quema de múltiples componentes internos y la pérdida total del controlador.

Este incidente permitió identificar el riesgo asociado al bloqueo mecánico del rotor en sistemas de propulsión brushless, dado que dicha condición conduce a corrientes elevadas que superan rápidamente la capacidad de los dispositivos de conmutación del ESC.

#### *M. Cuarto incidente: falla del ESC de 30 A por sobrecorriente*

En un cuarto incidente, se realizaron pruebas controladas con un nuevo ESC de 30 A, con el objetivo de determinar el valor máximo de PWM que el sistema podía soportar de manera segura. Durante esta prueba, el valor de PWM se incrementó progresivamente hasta alcanzar aproximadamente  $1600 \mu\text{s}$ .

En estas condiciones, el ESC volvió a presentar una falla similar a la observada en el segundo incidente, registrándose

la quema de un MOSFET correspondiente a una de las fases del motor. Este comportamiento reforzó la hipótesis de que el controlador se encontraba operando cerca de sus límites de corriente, incluso sin que se produjera un bloqueo mecánico del rotor.

#### N. Medidas adoptadas

La repetición de fallas en controladores de 30 A, tanto bajo condiciones transitorias como en operación sostenida, llevó a concluir que dicho margen de corriente resultaba insuficiente para el motor utilizado y las exigencias mecánicas de la planta. Asimismo, se consideró la posible influencia de algoritmos internos del ESC y de su calidad de construcción, los cuales podrían limitar su capacidad de manejo de sobrecorrientes.

En función de estas observaciones, se decidió sobredimensionar el sistema de actuación mediante la adquisición de un ESC de 40 A. Esta decisión permitió operar el motor con un mayor margen de seguridad, evitando la necesidad de reducir aún más la masa del cuerpo móvil y mejorando la confiabilidad del sistema durante las prácticas experimentales.

#### REFERENCIAS

- [1] Dongguan E-S Motor Co., Ltd., “E-s motor official website,” <https://cdn.robotshop.com/rbm/a00a7635-653b-4220-aac9-b0c23c5c5e2c/5f520795f9-301f-4e66-a0a3-019811d1f78b/222c434ca2212-brushless-motor.pdf>, 2026, accessed: Feb. 2026.
- [2] NEX Robotics Pvt, Ltd, “40a bldc esc,” <https://www.scribd.com/document/280924512/ESC-Datasheet>, 2015, accessed: Feb. 2026.
- [3] AMAZON, “Lipo 3s ovonic air,” <https://www.amazon.com/-/es/OVONIC-3s-Conector-Helic%C3%BC-B3ptero-Quadcopter/dp/B07MS8QF3K>, 1996-2026, accessed: Feb. 2026.
- [4] Benewake Co., Ltd., *TFmini Plus Single-Point LiDAR Range Sensor Datasheet*, 2019, rev. A02. [Online]. Available: [https://cdn.sparkfun.com/assets/2/b/0/3/8/TFmini\\_Plus-01-A02-Datasheet\\_EN.pdf](https://cdn.sparkfun.com/assets/2/b/0/3/8/TFmini_Plus-01-A02-Datasheet_EN.pdf)
- [5] Lic. Piero Codas Morselli, *Evaluación de Algoritmos de Control Digital Industrial*, 2005. [Online]. Available: <https://drive.google.com/file/d/1T1a3A4172OXRvjMa7e-VbKXvCpIqimyW/view>