



CEA
comité
español de
automática

Concurso en Ingeniería de Control 2020

Control de la orientación de un
multirrotor

Descripción del concurso

Organiza el Grupo Temático de
Ingeniería de Control de CEA

<https://www.ceautomatica.es/ingenieria-de-control/>



Organiza:

Grupo Temático en Ingeniería de Control de CEA



Javier Rico Azagra

Montserrat Gil Martínez

Silvano Nájera Canal

Carlos Elvira Izurategui

Ramón Rico Azagra

Grupo de Ingeniería de Control – Dpto. Ingeniería Eléctrica

Universidad de la Rioja

Patrocinan



Sección
Española



1. Introducción

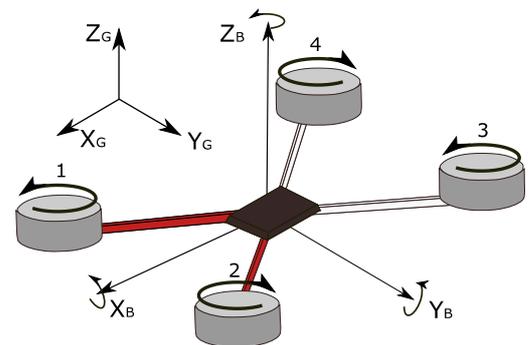
Este documento presenta la descripción técnica del concurso que el Grupo Temático de Ingeniería de Control de CEA propone a estudiantes de grado, máster y posgrado en la edición 2020 (en adelante, CIC2020).

Aprovechando la popularidad de los vehículos aéreos no tripulados (UAVs) de tipo multirrotor, esta propuesta pretende ser motivadora para el aprendizaje del control automático, a la vez que aproximarse a un planteamiento real. Tanto en la operación del vuelo desde una emisora radio control (RPAS), como desde una estación de tierra (por ejemplo, en un guiado autónomo por coordenadas), la controladora de vuelo abordo es el cerebro encargado de que el UAV alcance el posicionamiento deseado. Para ello, se implementan algoritmos que responden a estructuras jerárquicas de control, donde el éxito de las capas superiores está supeditado a un correcto funcionamiento de las capas inferiores, que en último término deben controlar los ángulos de orientación espacial del UAV. En este marco, el problema de control propuesto para el CIC2020 emula el control RPAS de la orientación de un multirrotor, que viene descrita por los ángulos de Euler¹: *roll*, *pitch* y *yaw*. Para ello, se emplea una plataforma (Figura 1a) desarrollada por el grupo de investigación en Ingeniería de Control de la Universidad de la Rioja.

La descripción del concurso en este documento se organiza como sigue. La Sección 2 describe el sistema físico, la Sección 3 define el problema de control, y la Sección 4 contiene la información específica de la fase clasificatoria (fase 1). De cara a la fase final (fase 2), podrá aportarse documentación complementaria si se considera necesario. Toda la información asociada al concurso estará accesible desde <https://www.unirioja.es/dptos/die/cic2020/>



(a) Estructura con cuatrirrotor



(b) Sistemas de referencia

Figura 1: Plataforma para el control de la orientación de un multirrotor

¹ Se utilizará la designación inglesa para ángulos de alabeo, cabeceo y guiñada debido a su uso generalizado.

2. Descripción del sistema físico

Un cuatrirrotor de 250 mm envergadura se encuentra confinado en una estructura que permite su rotación alrededor de sus tres ejes principales (X_B , Y_B , Z_B) en el sistema de referencia no inercial del cuerpo (SRB). Estas rotaciones, traducidas al sistema de referencia inercial terrestre (SRG) de ejes (X_G , Y_G , Z_G), definen la orientación según los ángulos de Euler: *roll* (ϕ), *pitch* (θ) y *yaw* (ψ). La estructura se muestra en la Figura 1a. Está formada por 2 aros concéntricos, que giran sobre ejes enfrentados 90° entre sí, y que permiten la orientación deseada en *yaw* y *pitch*. El aro exterior presenta un diámetro exterior de 580 mm y el aro interior un diámetro exterior de 470 mm. A 90° del eje de giro del aro interno, se acopla una varilla que permite la orientación deseada en *roll*. Sobre esta varilla se coloca el cuatrirrotor y su equipamiento tal que el centro de gravedad se sitúe por debajo del centro de giro. La disposición de los cuatro brazos es en modo aspa, quedando los sistemas de referencia SRG y SRB tal y como se indica en la Figura 1. El eje de giro del aro exterior está sujeto a una estructura metálica que conforma un cubo. Las variables dinámicas proporcionadas para el concurso son: los ángulos de orientación en SRG, $[\phi, \theta, \psi]^T$ (objeto de control), y las velocidades de rotación en SRB, $[p, q, r]^T$.

Las principales connotaciones de la plataforma son las siguientes:

- La estructura de sujeción que permite el giro sin desplazamiento del cuatrirrotor modifica ligeramente el modelo dinámico respecto a un vuelo libre.
- Opcionalmente, un sistema de bloqueo de ejes permite que el control de actitud (*roll* y *pitch*) no se vea afectado por el acoplamiento del giro en *yaw*. Esto permite plantear objetivos independientes de control de la rotación en sólo dos ejes, o en los tres.
- En los acoplamientos entre ejes de giro se ubican sensores de tipo *encoder* que miden de forma absoluta el ángulo rotado. Para un vuelo libre, esto permitiría validar los estimadores de ángulos que fusionan la información de sensores abordo y se programan en la controladora de vuelo.
- En los acoplamientos de giro se emplean anillos colectores (*slip-rings*), que permiten el intercambio cableado de datos con el exterior y la alimentación ininterrumpida de los motores mediante una fuente de tensión regulable que sustituye a las baterías.
- En un vuelo libre, tanto la variación en la carga útil (*payload*) como la descarga de las baterías, modifican la señal de control necesaria en los rotores para mantener la altura, y, por tanto, alteran el rango disponible para controlar la orientación. En la plataforma es posible emular ambos efectos de forma independiente: modificando la señal de control de inicialización de los motores y cambiando la tensión de alimentación.

El sistema cuatrirrotor integra los elementos detallados en la Figura 2, de entre los que resumiremos brevemente los imprescindibles para el concurso:

- El sistema de propulsión utiliza hélices tri-pala movidas por motores de corriente continua sin escobillas (BLDC). Su controlador electrónico de velocidad (ESC) responde a señales según el protocolo radio control de comunicaciones por modulación de anchura de pulso (PWM RC). Específicamente, la anchura del pulso de la señal de control se establece en el rango $1000 \mu s$ a $2000 \mu s$, que se corresponde con potencias del 0%

al 100%. Como no se sustenta carga y la alimentación es externa, el punto ideal de funcionamiento se corresponde con un *offset* de 1500 μs para $\phi = 0^\circ$, $\theta = 0^\circ$ y $\psi = 0^\circ$, con el fin maximizar el rango disponible para el control de la orientación según otros ángulos. Para simular la variación de la carga sustentada o la descarga de las baterías, se puede modificar *offset*. La descompensación de potencia entre rotores, propia de los dispositivos físicos, demandará una señal de control diferente de *offset* en cada motor para mantener la orientación inicial. Los ESC gestionan la actualización de la señal de control con una cadencia temporal de 0.02 ms.

- La controladora de vuelo es una ArduPilot Mega - APM 3.1 Pro-, que contiene un micro-controlador ATMEGA2560 de 8 bits y tiempo de muestreo de 0.01 ms.
- El hardware APM integra una unidad de medida inercial (IMU) de seis ejes y un barómetro. Opcionalmente, pueden conectarse de forma externa un magnetómetro de tres ejes y un GPS. Las medidas *raw* de estos sensores de bajo coste, que no serán accesibles a los concursantes, se fusionan convenientemente en el sistema de navegación. Este estima la velocidad de rotación, $[p, q, r]^T$ en SRB, y los ángulos, $[\phi, \theta, \psi]^T$ en SRG, que sí serán accesibles. Considerando la velocidad del micro-controlador, es posible actualizar esta información cada 0.01 ms.
- Una emisora radio control permite la operación en modo RPAS. En este concurso se empleará fundamentalmente para la desconexión de los motores, garantizando la seguridad ante hipotéticas desestabilizaciones del sistema.
- El sistema UAV se enlaza por puerto USB con la estación de tierra (PC) para la programación, para el envío de consignas, y para la recogida de datos en tiempo real del movimiento, de acuerdo a los experimentos propuestos en el concurso.
- El *firmware* de la controladora de vuelo gestiona el sistema radio control, el sistema de navegación, el sistema de control y actuación, y el sistema de comunicaciones. Ha sido íntegramente desarrollado en MATLAB-Simulink[®] para poder ser portado a plataformas Arduino[®].

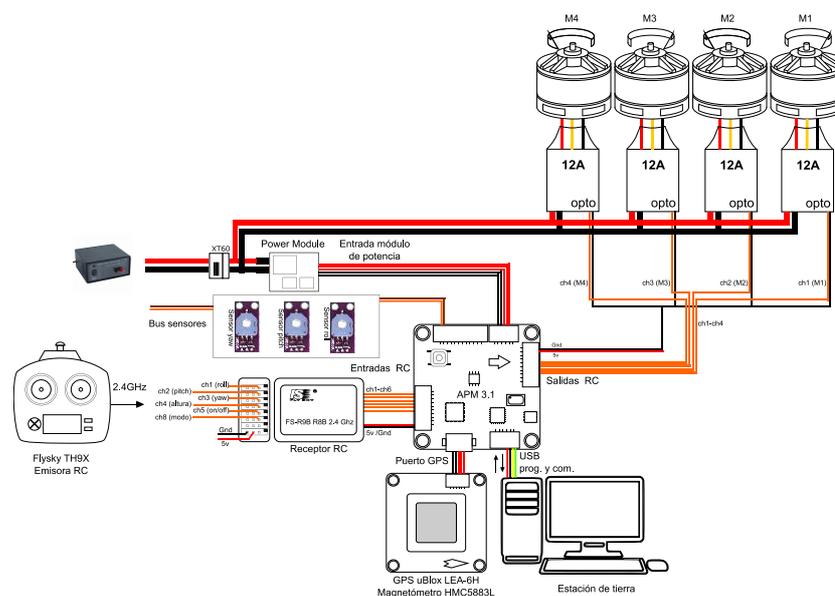


Figura 2: Sistema cuatrirrotor

3.- El problema de control

Se pretende controlar la orientación del cuatrirrotor según los ángulos de Euler $[\phi, \theta, \psi]^T$:

- En la categoría 1 se bloquea el giro en el eje Z_G ($\psi = 0^\circ$), siendo el objetivo del concurso alcanzar y mantener los ángulos ϕ y θ de acuerdo a las referencias que respectivamente se establezcan, r_ϕ y r_θ .
- En la categoría 2 se deberán controlar los tres ángulos de orientación, ϕ , θ y ψ , de acuerdo a sus respectivas referencias, r_ϕ , r_θ y r_ψ .

Para ello, los algoritmos propuestos por los concursantes deben generar las acciones de control pertinentes, u_ϕ , u_θ y u_ψ , que ejecuten el giro deseado en cada eje.

En la fase clasificatoria (fase 1) del concurso, el control se efectuará sobre un modelo desarrollado en Simulink[®] que es representativo del sistema real, y como señales externas al sistema se introducirán cambios en las referencias. En la fase final (fase 2) del concurso se abordará el control del sistema real, de acuerdo a las condiciones que se establezcan en su momento.

CIC2020. CONTROL DE LA ORIENTACIÓN DE UN UAV MULTIRROTOR

Categoría 2. Control de los angulos Roll, Pitch y Yaw.

Última modificación: 10/12/2019

Grupo de Investigación
de Ingeniería de Control
Universidad de La Rioja **ICON**

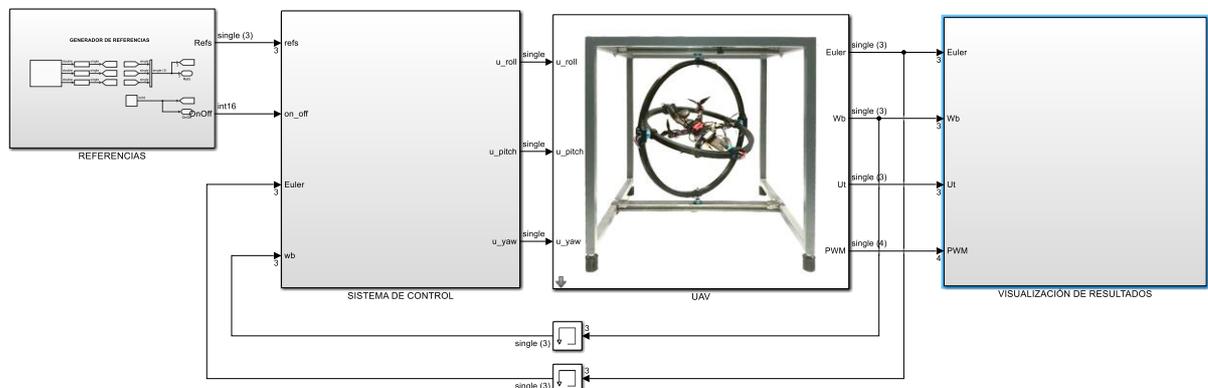


Figura 3: Entorno Simulink[®] para la categoría 2 (en la categoría 1 se elimina u_{yaw})

El entorno de trabajo en MATLAB-Simulink[®] que se facilita a los concursantes se muestra en el diagrama de bloques de la Figura 3. A grandes rasgos la funcionalidad de los bloques es la siguiente:

- El bloque REFERENCIAS genera las referencias de orientación deseadas para los ángulos de Euler, y una señal de desactivación/activación de los motores.
- El bloque SISTEMA DE CONTROL contiene los algoritmos/estructuras de control que, teniendo en cuenta las referencias y la estimación del estado del sistema, generan las acciones de control. Este es el bloque que deberán desarrollar los concursantes.
- El bloque UAV recibe las acciones de control calculadas por el sistema de control y entrega la información sobre el estado del sistema. A nivel de programación contiene

las funciones de envío de señales a los ESC, y las funciones que estiman ángulos y velocidades. En la fase 1 estas señales son enviadas/recibidas a/desde un modelo representativo y encriptado del sistema físico.

- El bloque `VISUALIZACIÓN DE RESULTADOS` permite el acceso a todas las variables de interés, que se canalizan a bloques de representación gráfica para la interpretación de los resultados.

3.1- El sistema de control

Su elemento principal es el bloque `SISTEMA DE CONTROL`, que se tomará como punto de partida para describir las variables Simulink (Tabla 1) y el resto de bloques antes mencionados.

La información de entrada al bloque `SISTEMA DE CONTROL` es la siguiente:

- Las referencias para los tres ángulos de Euler (r_ϕ, r_θ, r_ψ), que llegan desde el bloque `REFERENCIAS`, se reciben en cada instante de muestreo unificadas en la variable `Refs` 3x1 con datos de tipo *single*².
- Las variables medidas (estimadas), que provienen del bloque `UAV`, se generan para cada instante de muestreo como datos de tipo *single* y son: los ángulos de orientación en el SRG (ϕ, θ, ψ), unificados en la variable `Euler` 3x1; y las velocidades de rotación (p, q, r) en el SRB, unificadas en `Wb` 3x1.
- Una señal `on_off`, que informa sobre la desactivación (1) o activación (0) de los motores en cada instante de muestreo, se recibe como dato de tipo `int16` desde el bloque `REFERENCIAS`. En la fase 1 `on_off` se mantiene permanentemente en estado 0 (motores conectados). Su utilidad se remite a la fase 2 del concurso con el fin de evitar el posible *wind-up* de la acción de control calculada cuando ésta no está siendo aplicada en el sistema físico (motores desconectados).

Las salidas del bloque `SISTEMA DE CONTROL` son las acciones de control necesarias para el movimiento en cada eje (u_ϕ, u_θ, u_ψ), y que por tanto se canalizan al bloque `UAV`. En la categoría 1 se precisa generar `u_roll` y `u_pitch`. A estas acciones se añade `u_yaw` en la categoría 2. Cada acción de control se genera como dato 1x1 de tipo *single* en cada instante de muestreo.

Dentro del bloque `UAV`, las acciones de control son distribuidas convenientemente a los ESC de cada uno de los motores por la función `PID MIX`. Específicamente, para la configuración en aspa de cuatro rotores de la Figura 1, las señales en cada motor son:

$$M_1 = offset - u_\phi - u_\theta - u_\psi \quad (1)$$

$$M_2 = offset + u_\phi - u_\theta + u_\psi \quad (2)$$

² El tipo de datos manejado en MATLAB-Simulink® es el mismo que el utilizado en el *firmware* desarrollado para la controladora hardware APM 3.1 Pro.

$$M_3 = offset + u_\phi + u_\theta - u_\psi \quad (3)$$

$$M_4 = offset - u_\phi + u_\theta + u_\psi \quad (4)$$

siendo M_i una señal PWM RC entre 1000 μ s y 2000 μ s. Desde la máscara del bloque UAV es configurable la señal de `offset`, que representa la señal de control de inicialización de los motores, y permite emular diferentes cargas útiles soportadas por el UAV. La máscara del bloque UAV permite también configurar diferentes niveles de tensión de carga en las baterías en el rango [9, 12.6] V, que modifican las fuerzas de empuje y arrastre de los rotores para una misma señal de control M_i ; el valor seleccionado se almacena en `v_bat`. Por último, la máscara del bloque UAV permite activar (categoría 2) o desactivar (categoría 1) el control de `yaw` (estado que se almacena en `yaw_on`). El modelo de simulación del UAV en la fase 1 también incorpora desequilibrios reales en las potencias desarrolladas por cada rotor. Así, por ejemplo, podrá comprobarse que para lograr $\phi=\theta=\psi=0^\circ$, se necesitan valores $M_{i=1,\dots,4}$ distintos a `offset` y diferentes entre sí, que serán generadas por señales u_ϕ, u_θ, u_ψ , distintas de cero.

Por otro lado, el bloque UAV contiene la estimación del estado y entrega la evolución de las variables más significativas tanto para realimentación como para visualización o almacenamiento. En cada instante de muestreo se entregan las siguientes variables con datos de tipo *single*: $Euler=[\phi, \theta, \psi]^T$, $Wb=[p, q, r]^T$, $Ut=[u_\phi, u_\theta, u_\psi]^T$, $PWM=[M_1, M_2, M_3, M_4]^T$; y las unidades físicas son: grados x 10 para los ángulos, grados x 10 /segundo para las velocidades de rotación, y micro-segundos para las acciones de control y para las señales en los ESC de los motores. Por consiguiente, las unidades físicas de $Refs=[r_\phi, r_\theta, r_\psi]^T$ son grados x 10.

La estructura de control, algoritmos o funciones que el concursante puede implementar dentro del bloque SISTEMA DE CONTROL son libres, teniendo en cuenta las siguientes consideraciones:

- En la fase clasificatoria (fase 1), el bloque deberá poderse simular desde un equipo PC con una instalación básica de MATLAB®, Simulink®, y Control System Toolbox™.
- En la fase sobre la maqueta real, el bloque se compilará y volcará en la controladora hardware APM 3.1 Pro. Por tanto, deberá incluir todas las funciones que permitan su compilación. Además, dicha controladora tiene una capacidad de almacenamiento limitada, y su principal restricción es el micro-controlador de 8 bits. Las operaciones planteadas por los algoritmos deberán poder ser ejecutadas de acuerdo a los tiempos indicados a continuación, para asegurar el control en tiempo real y la estabilidad del sistema.
- En la fase de simulación (fase 1), el *solver* de ejecución es de paso fijo, con un tiempo de muestreo $T_{s3}=0.01$ segundos, que coincide con la velocidad a la que trabaja el micro de la controladora de vuelo (fase 2).
- Sin embargo, el protocolo de control de los accionamientos sólo permite actualizar la señal de mando cada $T_s=0.02$ segundos, que es el tiempo con el que las señales PWM son enviadas a los motores.

Tabla 1: Variables en el diagrama de bloques MATLAB-Simulink®

Etiqueta ³	Significado
r_roll	Referencia para el ángulo de <i>roll</i>
r_pitch	Referencia para el ángulo de <i>pitch</i>
r_yaw	Referencia para el ángulo de <i>yaw</i>
Refs	Referencias para los tres ángulos de <i>Euler</i>
roll	Ángulo de <i>roll</i>
pitch	Ángulo de <i>pitch</i>
yaw	Ángulo de <i>yaw</i>
Euler	Ángulos de Euler
roll_o	Ángulo de <i>roll</i> del sistema de control de referencia
pitch_o	Ángulo de <i>pitch</i> del sistema de control de referencia
yaw_o	Ángulo de <i>yaw</i> del sistema de control de referencia
p	Velocidad de rotación en eje X del cuerpo
q	Velocidad de rotación en eje Y del cuerpo
r	Velocidad de rotación en eje Z del cuerpo
Wb	Velocidades de rotación en los tres ejes del sistema de referencia del cuerpo
p_o	Velocidad de rotación en eje X del cuerpo del sistema de control de referencia
q_o	Velocidad de rotación en eje Y del cuerpo del sistema de control de referencia
r_o	Velocidad de rotación en eje Z del cuerpo del sistema de control de referencia
u_roll	Acción de control en <i>roll</i>
u_pitch	Acción de control en <i>pitch</i>
u_yaw	Acción de control en <i>yaw</i>
Ut	Acciones de control en los tres ejes
u_roll_o	Acción de control en <i>roll</i> del sistema de control de referencia
u_pitch_o	Acción de control en <i>pitch</i> del sistema de control de referencia
u_yaw_o	Acción de control en <i>yaw</i> del sistema de control de referencia
PWM	Señales de control en los ESC de los cuatro motores
PWM_o	Señales de control en los ESC de los cuatro motores del sistema de control de referencia
offset	Señal PWM RC inicial en los rotores
vbat	Tensión de la batería
yaw_on	Control de <i>yaw</i> activo o no activo
on_off	Desactivación / activación de motores
Ts	Tiempo de actualización de la señal en los ESC de los motores
Ts3	Tiempo de muestreo del sistema

³ Están condicionadas por los símbolos permitidos en Simulink®.

3.2.- Evaluación del comportamiento

En función de los objetivos y experimentos propios de cada fase y categoría, se configura el bloque GENERADOR DE REFERENCIAS. Para éstas, se archiva el comportamiento de un sistema de control de referencia, de estructura y valores no desvelados, que se utilizará como patrón para establecer comparativas con el sistema de control actual.

Durante una simulación el bloque VISUALIZACIÓN DE RESULTADOS canaliza la información a bloques gráficos, como ilustra la Figura 4; su cadencia de actualización y almacenamiento es de $T_s=0.02$ segundos. Dichas gráficas muestran el desempeño en cada eje (*scopes* Roll, Pitch, Yaw), en el conjunto de ejes (*scope* Actitud y Orientación), y las señales en los motores (*scope* Rotores), para el sistema de control actual y para el de referencia.

VISUALIZACIÓN DE RESULTADOS

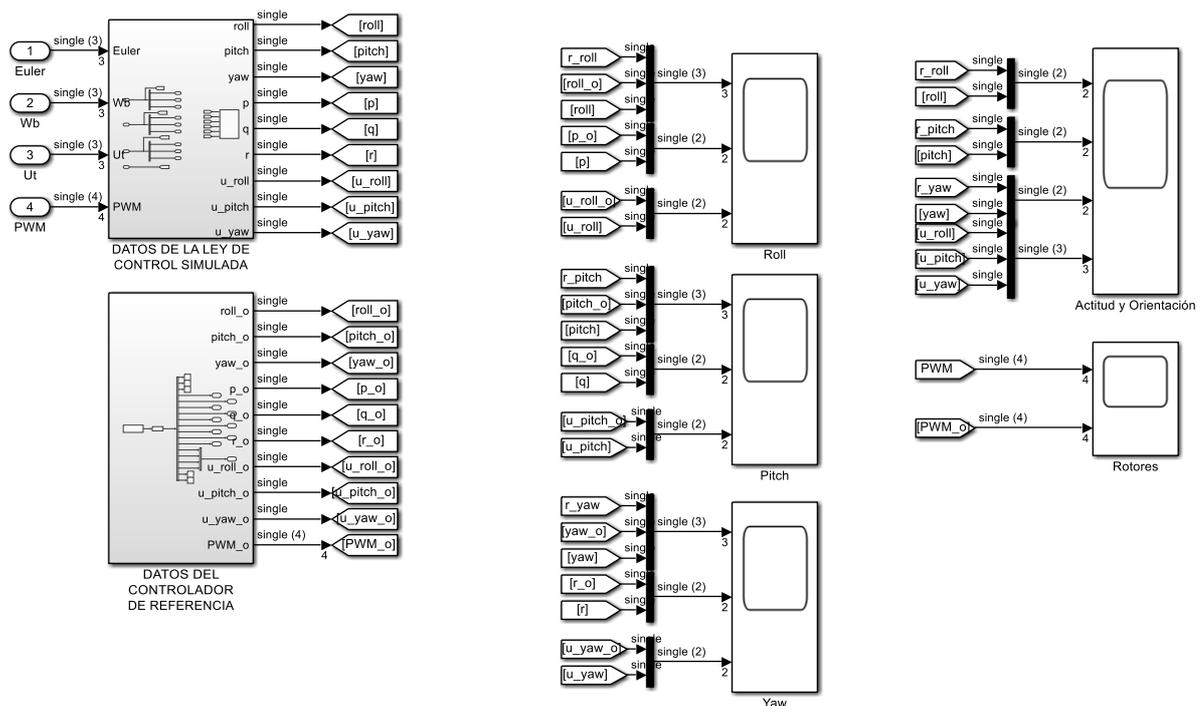


Figura 4: Bloque VISUALIZACIÓN DE RESULTADOS de la categoría 2. En el de la categoría 1 se excluyen las variables relacionadas con el ángulo de yaw

Al finalizar la simulación, se genera en el espacio de trabajo (*workspace*) de MATLAB® DatosC# (siendo # igual a 1 o 2 según la categoría) con los resultados del comportamiento del sistema de control actual. Se trata de una variable de tipo estructura (*struct*) con tiempo, y los valores de señal se organizan en 18 columnas que registran todas las muestras de las variables según

el siguiente orden: Refs, Euler, Wb, Ut, PWM, on_off, tiempo⁴. Para el caso del sistema de control de referencia, los resultados de comportamiento se guardan en DatosCrefC#.

Para evaluar cuantitativamente el desempeño del sistema de control, se emplearán los siguientes índices:

- IAE - Integral del Error Absoluto-, que particularizada para cada una de las tres variables controladas responde a:

$$IAE_{\phi} = \int_{t_0}^{t_f} |r_{\phi}(t) - \phi(t)| dt \quad (5)$$

$$IAE_{\theta} = \int_{t_0}^{t_f} |r_{\theta}(t) - \theta(t)| dt \quad (6)$$

$$IAE_{\psi} = \int_{t_0}^{t_f} |r_{\psi}(t) - \psi(t)| dt \quad (7)$$

Donde t_0 y t_f son los tiempos inicial y final del experimento.

- ITAE - Integral del Error Absoluto ponderado a lo largo del Tiempo-, que particularizada para cada una de las tres variables controladas responde a:

$$ITAE_{\phi} = \sum_{i_{\phi}=0}^{nT_{\phi}-1} \left(\int_{t_{0_{\phi}} + i_{\phi}\Delta t_{\phi}}^{(i_{\phi}+1)\Delta t_{\phi}} |r_{\phi}(t) - \phi(t)| (t - t_{0_{\phi}} - i_{\phi}\Delta t_{\phi}) dt \right) \quad (8)$$

$$ITAE_{\theta} = \sum_{i_{\theta}=0}^{nT_{\theta}-1} \left(\int_{t_{0_{\theta}} + i_{\theta}\Delta t_{\theta}}^{(i_{\theta}+1)\Delta t_{\theta}} |r_{\theta}(t) - \theta(t)| (t - t_{0_{\theta}} - i_{\theta}\Delta t_{\theta}) dt \right) \quad (9)$$

$$ITAE_{\psi} = \sum_{i_{\psi}=0}^{nT_{\psi}-1} \left(\int_{t_{0_{\psi}} + i_{\psi}\Delta t_{\psi}}^{(i_{\psi}+1)\Delta t_{\psi}} |r_{\psi}(t) - \psi(t)| (t - t_{0_{\psi}} - i_{\psi}\Delta t_{\psi}) dt \right) \quad (10)$$

En cada ángulo # se evalúa un cierto número de tramos $nT_{\#}$, que son recorridos a través del índice $i_{\#}$. El intervalo temporal evaluado en cada tramo $\Delta t_{\#}$ arranca en el instante de cambio de la señal externa $t_{0_{\#}} + i_{\#}\Delta t_{\#}$.

- IAVU - Integral de la Variación de la Acción de control Absoluta -, que particularizada para cada una de las tres variables de control responde a:

⁴ La columna tiempo se genera con un intervalo entre muestras de $T_s = 0.02$ segundos.

$$IAVU_{\phi} = \int_{t_0}^{t_f} \left| \frac{d u_{\phi}(t)}{dt} \right| dt \quad (11)$$

$$IAVU_{\theta} = \int_{t_0}^{t_f} \left| \frac{d u_{\theta}(t)}{dt} \right| dt \quad (12)$$

$$IAVU_{\psi} = \int_{t_0}^{t_f} \left| \frac{d u_{\psi}(t)}{dt} \right| dt \quad (13)$$

Donde t_0 y t_f son los tiempos inicial y final del experimento.

Para cada uno de los índices anteriores (5)-(13), se calcula la ratio entre el valor del índice para el sistema de control actual (SC) y el valor del índice para el sistema de control de referencia (SC_0). Estos valores ratio se designarán: $RIAE_{\phi}$, $RIAE_{\theta}$, $RIAE_{\psi}$, $RITAE_{\phi}$, $RITAE_{\theta}$, $RITAE_{\psi}$, $RIAVU_{\phi}$, $RIAVU_{\theta}$ y $RIAVU_{\psi}$, respectivamente. Cada uno de estos índices-ratio alcanza: el valor uno, cuando $SC=SC_0$; un valor menor que uno, cuando SC mejora a SC_0 ; y un valor mayor que uno, cuando SC empeora respecto a SC_0 .

Finalmente, un índice de evaluación global combina ponderadamente los índices anteriores:

$$I = w_1 RIAE_{\phi} + w_2 RIAE_{\theta} + w_3 RIAE_{\psi} + w_4 RITAE_{\phi} + w_5 RITAE_{\theta} + w_6 RITAE_{\psi} + w_7 RIAVU_{\phi} + w_8 RIAVU_{\theta} + w_9 RIAVU_{\psi} \quad (14)$$

Los factores de ponderación $w_{i=1,\dots,9}$ no son accesibles. Para la categoría 1 se cumple $w_3 = w_6 = w_9 = 0$ y $\sum_{i \neq 3,6,9} w_i = 1$. Para la categoría 2 se cumple $w_i \neq 0$ y $\sum w_i = 1$.

Se facilita la función de evaluación que calcula estos índices para guiar al concursante en el diseño, y para la clasificación de los equipos participantes.

4.- Fase clasificatoria (fase 1)

4.1- Experimento y evaluación

Categoría 1

En la Figura 5 se muestran las señales de referencia de actitud (r_{ϕ} , r_{θ}) con que se evaluará la fase 1 de la categoría 1. Se trata de cambios de tipo escalón positivos y negativos en cada eje, no simultáneos. También se ilustra la respuesta de las variables controladas (ϕ , θ), las acciones de control (u_{ϕ} , u_{θ}), y las señales en los ESC de los motores (M_1 , M_2 , M_3 , M_4), cuando se utiliza el sistema de control de referencia. El *offset* inicial en los ESC es de 1500 μ s y la tensión de la batería 11 V.

Para calcular IAE (5) (6), e IAV (11) (12), se considera $t_0=0$ y $t_f=45$. Para calcular ITAE (8) (9) se consideran 4 tramos en cada ángulo, y para cada ángulo, cada tramo comienza en el instante de tiempo en que cambia su señal de referencia y termina 5 segundos después. La función MATLAB que calcula (14) y cada una de sus componentes es `CIC2020_EvaluaC1_F1.p`.

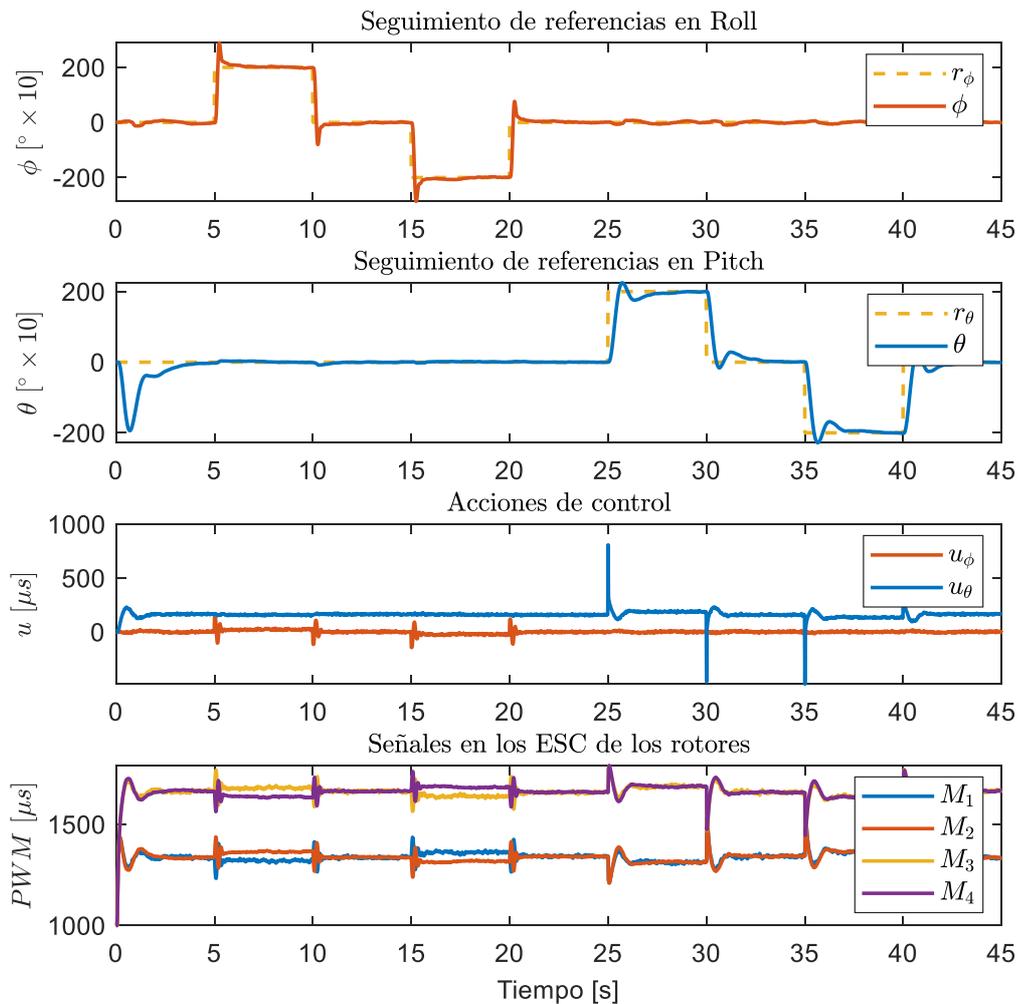


Figura 5: Experimento de evaluación de la categoría 1 en la fase 1

Categoría 2

En la Figura 6 se muestran las señales de referencia de orientación (r_ϕ , r_θ , r_ψ) con que se evaluará la fase 1 de la categoría 2. Se trata de cambios no simultáneos en cada eje: de tipo escalón en r_ϕ y r_θ , y de tipo rampa en r_ψ . También se ilustran: las respuestas de las variables controladas (ϕ , θ , ψ), las acciones de control (u_ϕ , u_θ , u_ψ), y las señales en los ESC de los motores (M_1 , M_2 , M_3 , M_4), cuando se utiliza el sistema de control de referencia. El *offset* inicial en los ESC es de 1500 μs y la tensión de la batería 11 V.

Para calcular IAE (5)-(7), e IAV (11)-(13), se considera $t_0=0$ y $t_f=65$. Para calcular ITAE (8)-(10) se consideran 4 tramos en cada ángulo, y para cada ángulo, cada tramo comienza en el instante de tiempo en que cambia su señal de referencia y termina 5 segundos después. La función MATLAB que calcula (14) y cada una de sus componentes es `CIC2020_EvaluaC2_F1.p`.

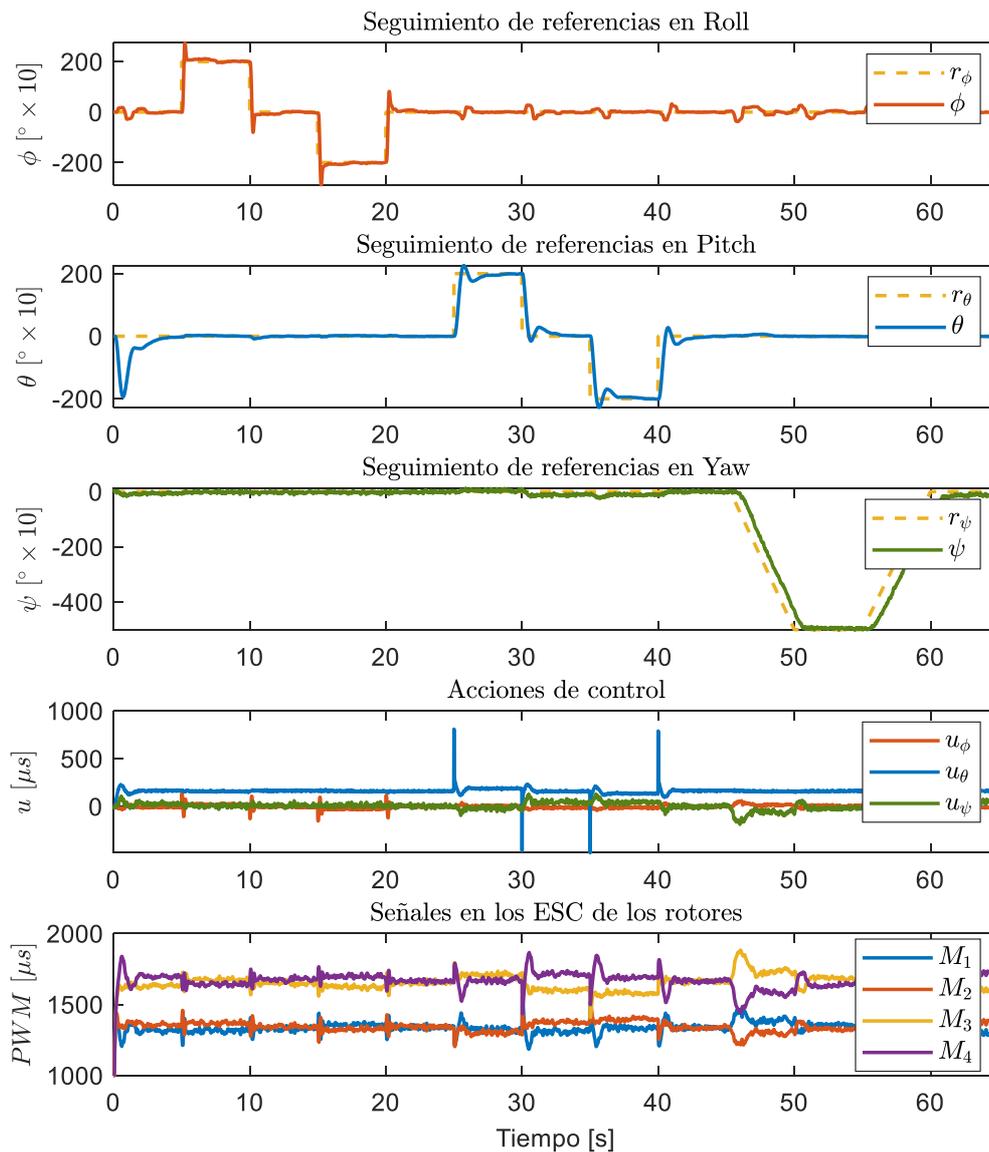


Figura 6: Experimento de evaluación de la categoría 2 en la fase 1

4.2. Entregables y clasificación

Cada equipo participante, que tendrá asignado un identificador ###, hará entrega en fecha y forma de acuerdo a las Bases del Concurso de un fichero comprimido `CIC2020_###.zip` con los siguientes ficheros:

- `CIC2020_F1_###.slx`. Fichero Simulink v9.2 (MATLAB R2018b) que debe contener sólo el bloque `SISTEMA DE CONTROL` sometido a concurso. Dicho bloque debe mantener el formato de entradas-salidas del bloque tipo facilitado.
- `CIC2020_F1_###.pdf`. Documento justificativo de la solución de control implementada, siguiendo el formato de los trabajos presentados a las Jornadas de Automática.

La evaluación de la fase 1 valorará: en un 70%, el resultado obtenido por el índice de desempeño total del sistema de control presentado, y en un 30%, la memoria justificativa de la estrategia seguida. Conforme a esta nota se clasificará a los concursantes.

4.3. Documentación

En este apartado se relacionan y comentan los archivos que los concursantes podrán descargar desde: <https://www.unirioja.es/dptos/die/cic2020/>.

`CIC2020_Descripcion.pdf`. Es la versión electrónica de este documento.

Categoría 1

Comprimidos en el fichero `CIC2020_F1C1.zip` se encuentran:

- `Simulador_C1.slx`. Fichero Simulink v9.2 que contiene el entorno de la Figura 3. El concursante debe incluir sus algoritmos/estructura de control dentro del bloque `SISTEMA DE CONTROL`. Inicialmente contiene una estructura de control para poder ejecutar una simulación.

Los cuatro siguientes ficheros son necesarios para la correcta ejecución del anterior.

- `UAV_sf.mexw64`. Fichero compilado que contiene el modelo representativo de la orientación del cuatrirrotor fijado a la plataforma.
- `UAV_sf.c` y `UAV_sf.h`. Ficheros C para la *S-function* del modelo.
- `UAV.png`. Imagen del cuatrirrotor que se visualiza sobre el bloque `UAV`.
- `DatosCrefC1.mat`. Fichero de datos MATLAB que contiene el comportamiento del sistema de control de referencia.
- `CIC2020_EvaluaC1_F1.p`. Función de evaluación que de forma encriptada implementa la ecuación (14).
Su argumento de entrada son los datos a evaluar. Ejemplo de llamada desde la línea de comandos de MATLAB: `>> CIC2020_EvaluaC1_F1(DatosC1)`, siendo `DatosC1` la variable *struct* que se genera en el *workspace* con todas las variables de interés tras la ejecución de `Simulador_C1.slx`.

La función devuelve en la línea de comandos el valor de I (14), y de cada uno de los índices ratio que lo componen. Por ejemplo, para cierto sistema de control, arroja:

Resultados de la evaluación				
	IAE	ITAE	IAVU	Total
Roll	1.3253	2.1465	0.27728	1.4442
Pitch	0.92204	0.76392	0.51564	0.77751
Resultado total de la ley de control: 1.1108				

Además, la función de evaluación genera tres figuras: `Roll`, con la evolución temporal del comportamiento en *roll* (r_ϕ, ϕ, p, u_ϕ , para el sistema de control actual y el de referencia); `Pitch`, con la evolución temporal del comportamiento en *pitch* ($r_\theta, \theta, q, u_\theta$, para el sistema de control actual y el de referencia); y `Actitud`, que resume el comportamiento del controlador actual ($r_\phi, \phi, r_\theta, \theta, u_\phi, u_\theta, M_1, M_2, M_3, M_4$).

Categoría 2

Comprimidos en el fichero `CIC2020_F1C2.zip` se encuentran:

- `Simulador_C2.slx`. Fichero Simulink v9.2 que contiene el entorno de la Figura 3. El concursante debe incluir sus algoritmos/estructura de control dentro del bloque `SISTEMA DE CONTROL`. Inicialmente contiene una estructura de control para poder ejecutar una simulación.

Los cuatro ficheros siguientes son necesarios para la correcta ejecución del anterior.

- `UAV_sf.mexw64`. Fichero compilado que contiene el modelo representativo de la orientación del cuatrirrotor fijado a la plataforma.
- `UAV_sf.c` y `UAV_sf.h`. Ficheros C para la *S-function* del modelo.
- `UAV.png`. Imagen del cuatrirrotor que se visualiza sobre el bloque `UAV`.
- `DatosCrefC2.mat`. Fichero de datos MATLAB que contiene el comportamiento del sistema de control de referencia.
- `CIC2020_EvaluaC2_F1.p`. Función de evaluación que de forma encriptada implementa la ecuación (14).

Su argumento de entrada son los datos a evaluar. Ejemplo de llamada desde la línea de comandos de MATLAB: `>> CIC2020_EvaluaC2_F1(DatosC2)`, siendo `DatosC2` la variable *struct* que se genera en el *wokspace* con todas las variables de interés tras la

ejecución de `Simulador_C2.slx`.

La función devuelve en la línea de comandos el valor de I (14), y de cada uno de los índices ratio que lo componen. Por ejemplo, para cierto sistema de control, arroja:

Resultados de la evaluación				
	IAE	ITAE	IAVU	Total
	_____	_____	_____	_____
Roll	1.1533	1.0885	0.96296	1.0893
Pitch	1.0913	1.1503	0.99413	1.0955
Yaw	0.86156	0.8473	1.3368	0.9509
Resultado total de la ley de control:				
	1.0452			

Además, la función de evaluación genera cuatro figuras: *Roll*, con la evolución temporal del comportamiento en *roll* (r_ϕ, ϕ, p, u_ϕ , para el sistema de control actual y el de referencia); *Pitch*, con la evolución temporal del comportamiento en *pitch* ($r_\theta, \theta, q, u_\theta$, para el sistema de control actual y el de referencia); *Yaw*, con la evolución temporal del comportamiento en *yaw* (r_ψ, ψ, r, u_ψ , para el sistema de control actual y el de referencia); y *Orientación*, que resume el comportamiento del controlador actual ($r_\phi, \phi, r_\theta, \theta, r_\psi, \psi, u_\phi, u_\theta, u_\psi, M_1, M_2, M_3, M_4$).

Los archivos MATLAB-Simulink han sido probados con las siguientes versiones:

Versión	Release	Año
MATLAB 9.5	R2018b	2018
Simulink 9.2	R2018b	2018