**Facultad de Ingeniería**

**Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica**

**Sistemas de Control 1, 2025**

**Miguel Zea, Pablo Caal**

**Laboratorio:** Análisis del control de altura del

dron Crazyflie 2.1

**Objetivos**

* Comprender el comportamiento dinámico del dron Crazyflie 2.1 al aplicar un controlador PID en la altura de vuelo.
* Analizar el efecto de los parámetros KP, KI y KD en la respuesta del sistema, tanto en simulaciones como en pruebas experimentales.
* Comparar los resultados entre el modelo simplificado y el comportamiento real del dron, identificando las diferencias y limitaciones del modelo.
* Desarrollar habilidades en la implementación y ajuste de controladores PID en entornos de simulación y experimentación práctica.

Una bicicleta estacionada

Descripción generada automáticamente con confianza baja

**Figura 1.** Dron Crazyflie 2.1 ensamblado.

**Material y equipo necesario**

A continuación, se enlistan los materiales y el equipo necesario para desarrollar este laboratorio:

* Dron Crazyflie 2.1 con la placa de expansión Flow Deck integrada
* Dispositivo Crazyradio PA
* Ordenador con Windows 10/11 con Matlab y Python instalados
* Paquete/carpeta de herramientas de *software* descargado
* Sistema de Captura de Movimiento del ecosistema Robotat del laboratorio CIT-116

**Descripción**

En este laboratorio se analizará el comportamiento de vuelo del dron Crazyflie 2.1 utilizando un controlador PID de altura. Los estudiantes aprenderán a ajustar y evaluar los efectos de las ganancias KP, KI y KD de un controlador PID tanto en simulaciones como en experimentos reales con el dron.

Se espera que usted ya tenga instaladas en su computadora la librería de Python cflib y la carpeta con las funciones de control básico. Esto se debe a que el tiempo de laboratorio está dado exclusivamente para realizar experimentos con el controlador de posición del dron.

**Prelaboratorio: Manual de usuario**

Previo a asistir a la sesión de laboratorio designada, deberá leer el [manual de usuario del Crazyflie 2.1 con la placa de expansión Flow Deck incorporada](https://drive.google.com/file/d/157fDCxXf_BLKTRiDAqIZCCapgSy8fdp2/view?usp=sharing), procurando instalar en su ordenador todas las dependencias indicadas para su uso.

**Primera parte: Simulación del modelo simplificado en condiciones ideales**

En esta sección se realizará una simulación en Matlab que modelará el comportamiento del dron bajo la acción de un controlador PID para el control de altura. El objetivo es visualizar cómo las diferentes combinaciones de parámetros PID afectan el despegue y la estabilidad del dron.

1. Modelo simplificado del dron

Se considerará un modelo simplificado que describe la dinámica vertical de vuelo del dron. Asumimos una dinámica de segundo orden con las siguientes ecuaciones:

Donde

* *h* es la altura del dron
* *u* es la entrada de control (fuerza generada por los motores)
* *g* es la gravedad
* *m* es la masa del dron

1. Controlador PID

Implementaremos un controlador PID para calcular el valor de entrada de control en función del error de altura.

Donde KP, KI y KD son las ganancias del controlador.

1. Simulación

En Matlab, simulará el despegue del dron desde una altura inicial de cero metros (h = 0) hasta una altura deseada de 1 metro (hf = 1). La simulación se ejecutará durante un tiempo determinado y se registrarán las respuestas de altura del dron y su respectivo error.

Dentro de la carpeta del laboratorio encontrará el archivo **simulación\_crazyflie** que posee la estructura general para la implementación del simulador descrito. Complete las secciones correspondientes para realizar las simulaciones para el siguiente listado de combinaciones de parámetros del controlador PID:

|  |  |  |  |
| --- | --- | --- | --- |
| Prueba | KP | KI | KD |
| 1 | 2.50 | 0.50 | 0.00 |
| 2 | 4.50 | 0.50 | 0.00 |
| 3 | 2.50 | 1.00 | 0.00 |
| 4 | 4.50 | 1.00 | 0.00 |
| 5 | 4.50 | 1.00 | 0.05 |

**Cuadro 1.** Combinación de parámetros PID para los experimentos.

1. Preguntas para discusión

* ¿Cómo afectan los parámetros KP, KI y KD a la respuesta del sistema? Comente sobre la velocidad de respuesta, sobreimpulso y estabilidad.
* ¿Qué parámetros del controlador producen una respuesta más suave y cuál tiende a generar oscilaciones?
* ¿Cómo influye la integral del error en el comportamiento del dron durante el despegue?

**Segunda parte: Simulación del modelo simplificado en condiciones no ideales**

En esta sección realizará nuevamente una simulación en Matlab, pero ahora el modelo incluirá efectos no ideales en el modelo, como la resistencia del aire y la dinámica de los motores. Esto permitirá observar como las condiciones afectan el desempeño del dron.

La ecuación que incluye dichos efectos en el modelo resulta como:

Donde

* *a* es la aceleración del dron
* *v* es la velocidad del dron
* *u* es la entrada de control (fuerza generada por los motores)
* *g* es la gravedad
* *m* es la masa del dron
* *cdrag* es el coeficiente de resistencia de aire

Nuevamente realice la simulación con el listado de combinaciones de parámetros PID de la parte anterior y compare los resultados respondiendo las siguientes preguntas:

* ¿Cómo afectan la resistencia del aire y la dinámica de los motores al desempeño del dron?
* ¿Qué diferencia observas al comparar las condiciones ideales con las no ideales?

**Tercera parte: Experimentación física con el dron Crazyflie**

En esta sección, se llevará a cabo una serie de pruebas experimentales en las que los estudiantes deberán ajustar el controlador PID en el Crazyflie 2.1 y avaluar su desempeño real.

**Procedimiento Experimental**

1. Configuración del entorno

En esta parte estarán utilizando el ecosistema Robotat del laboratorio CIT-116 para registrar el vuelo del dron Crazyflie por medio del sistema de captura de movimiento. El ecosistema Robotat emplea un sistema de captura de movimiento basado en cámaras y marcadores reflectivos pasivos, los cuales se colocan en objetos para obtener su pose (posición y orientación).

Para interactuar con el sistema de captura de movimiento, utilizaremos las siguientes funciones en Matlab. Estas se encuentran presentes en la carpeta Robotat de los archivos extraídos.

* robotat\_connect(): Conecta el sistema con el servidor de captura de movimiento.
* robotat\_get\_pose(): Obtiene la pose (posición y orientación) de un marcador reflectivo.
* robotat\_disconnect(): Desconecta el sistema del servidor.

Para interactuar con el dron Crazyflie, utilizaremos las siguientes funciones de control básico en Matlab. Estas se encuentran presentes en la carpeta Crazyflie de los archivos extraídos.

* crazyflie\_connect(): Conecta el sistema con el dron Crazyflie correspondiente.
* crazyflie\_takeoff(): Ejecuta la secuencia de despegue en el dron Crazyflie.
* crazyflie\_land(): Ejecuta la secuencia de aterrizaje en el dron Crazyflie.
* crazyflie\_set\_pid\_z(): Modifica los parámetros PID del controlador de altura.
* crazyflie\_disconnect(): Desconecta el sistema del dron Crazyflie.

En la carpeta del laboratorio encontrará el archivo **experimentacion\_crazyflie** en donde se presenta una explicación detallada de cómo utilizar dichas funciones y una estructura de código que deberá completar para realizar los experimentos con el dron.

1. Pruebas de vuelo

Usted deberá realizar varios experimentos con distintas combinaciones de parámetros PID y registrar el resultado en Matlab mediante el sistema de captura de movimiento. Para ello, modifique el código mencionado para realizar con el conjunto de combinación de parámetros PID.

Registre correctamente los datos de vuelo para cada experimento y realice una gráfica de altura contra tiempo para cada combinación.

1. Comparación con simulación

Responda las siguientes preguntas:

* ¿Cómo se comporta el dron con cada combinación de parámetros PID en comparación con la simulación?
* ¿Qué diferencias se observan entre el comportamiento real y el modelo simplificado? ¿A qué podrían deberse estas diferencias?
* ¿Cómo afectan las condiciones del entorno experimental (como corrientes de aire o fricción) al rendimiento del controlador?
* ¿Qué recomendaciones harían para mejorar el modelo de simulación basado en los resultados experimentales?