



**UNIVERSIDAD
DE GRANADA**



ETSIIT

Escuela Técnica Superior
de Ingenierías Informática
y de Telecomunicación

Control y coordinación de drones Crazyflie

ALUMNO: ÁNGEL HURTADO FLORES

TUTOR: HÉCTOR GARCÍA DE MARINA PEINADO

ÍNDICE

1. Introducción

1. Motivación
2. Descripción general
3. Objetivos
4. Planificación

2. Preliminares

1. Pruebas con software oficial
2. Pruebas con Paparazzi
3. Desarrollo básico en Paparazzi

3. Control de un Crazyflie

1. Control básico
2. Algoritmos GVF
3. Implementación de GVF
4. Coordinación entre Crazyflies
 1. Algoritmos de coordinación
 2. Implementación de la coordinación
 3. Simulación de formaciones
 4. Resultados de las formaciones

5. Conclusiones

MOTIVACIÓN

- Explorar conceptos de robótica, control y coordinación
- Entender estos conceptos en el marco de los drones
- Aplicar, entender y diseñar algoritmos avanzados de control y coordinación
- En el futuro, aplicar la coordinación en escenarios reales prácticos

DESCRIPCIÓN GENERAL

Crazyflie 2.1

- UAV (popularmente llamado dron) de contenidas dimensiones y peso
- Se le puede cambiar el firmware por uno *open source*
- Queremos mover coordinadamente 2 o más drones



Posicionamiento

- Permite la localización en el espacio. Ejemplo:
- Loco Positioning Node como puntos de referencia o *anchors*
- Loco Positioning Deck para cada Crazyflie
- Existen más alternativas



Paparazzi UAV

- Proyecto que abarca software, firmware y hardware para UAVs
- Usaremos un firmware para Crazyflie 2.1 que expandiremos
- Usaremos el software Paparazzi Center para control del dron



OBJETIVOS

El objetivo final trata de conseguir que los Crazyflies puedan seguir trayectorias de forma coordinada utilizando firmware y software *open source* de Paparazzi UAV.

- **OBJ-1:** familiarización con el trabajo. Realización de la planificación, objetivos etc...
- **Preliminares (OBJ-2):** se montará el dron y se harán pruebas que verifiquen su correcto funcionamiento básico
- **Control de un Crazyflye (OBJ-3):** control de un solo Crazyflye haciendo hovering (PID) y posterior adición de algoritmos de control (GVF)
- **Coordinación entre Crazyflies (OBJ-4):** extender el objetivo 3 añadiendo coordinación entre 2 o más drones.
- **Conclusiones (OBJ-5):** entender los resultados, posibles aplicaciones y extraer conclusiones
- **OBJ-6:** Repaso general del trabajo. Modificaciones leves al resto de objetivos

PLANIFICACIÓN

Planificación General																			
Febrero				Marzo				Abril				Mayo				Junio			
W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
	OBJ-1																		
		OBJ-2																	
			OBJ-3																
								OBJ-4											
												OBJ-5							
																OBJ-6			

ÍNDICE

1. Introducción

1. Motivación
2. Descripción general
3. Objetivos
4. Planificación

2. Preliminares

1. Pruebas con software oficial
2. Pruebas con Paparazzi
3. Desarrollo básico en Paparazzi

3. Control de un Crazyflie

1. Control básico
2. Algoritmos GVF
3. Implementación de GVF
4. Coordinación entre Crazyflies
 1. Algoritmos de coordinación
 2. Implementación de la coordinación
 3. Simulación de formaciones
 4. Resultados de las formaciones
5. Conclusiones

PRUEBAS CON SOFTWARE OFICIAL

Crazyflie 2.1

- Montamos los drones
- Actualizamos firmware, no obligatorio, pero recomendable



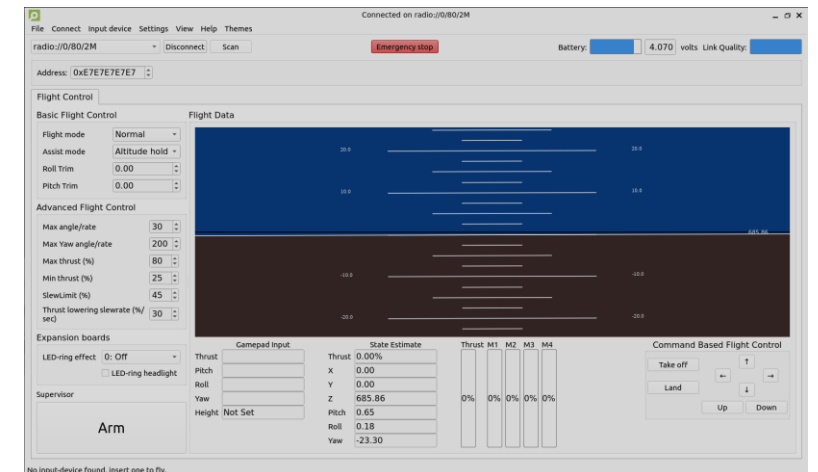
Crazyradio 2.0

- Permite conectarnos a los Crazyflie 2.1
- No funciona de fábrica
- Necesitamos cargar el firmware



Crazyflie Client

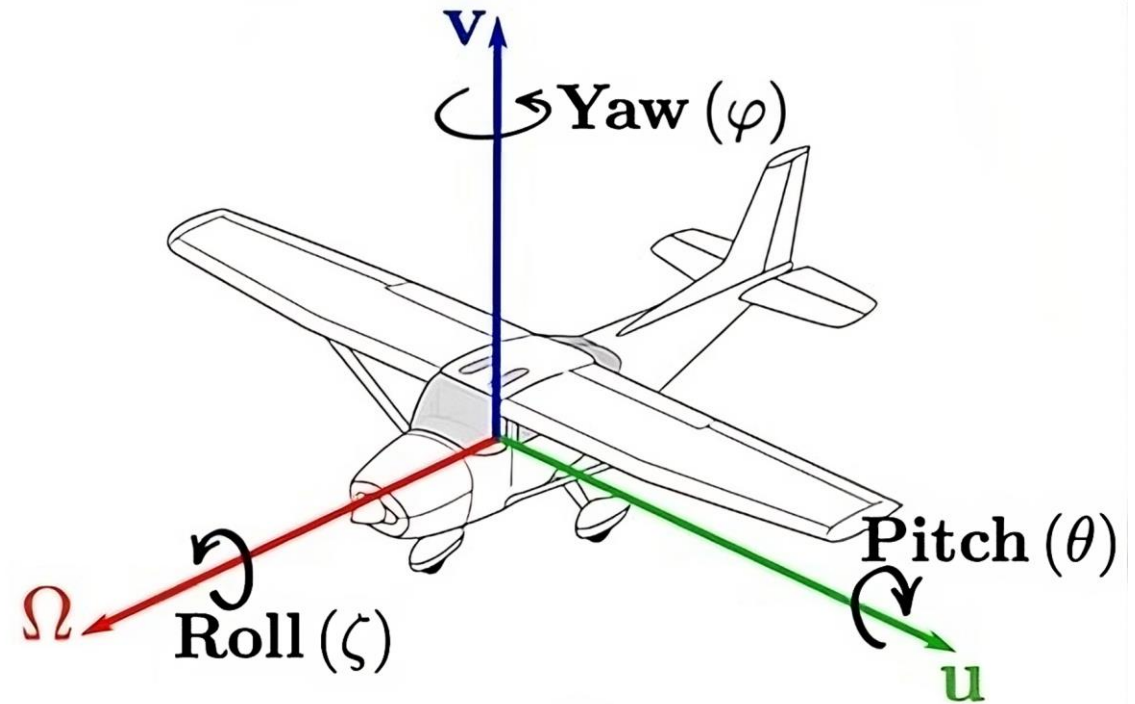
- Permite controlar los Crazyflie
- Puede actualizar el firmware de los Crazyflie
- Necesita Crazyradio 2.0



VARIABLES DE CONTROL DE UN ROTORCRAFT

Entre las variables más típicas de control en un dron tenemos las siguientes. En **negrita** las que Crazyflie Client nos muestra:

- **Roll, pitch y yaw:** Rotación sobre los ejes X,Y,Z, respectivamente
- **Posición en los ejes X,Y,Z** (posición relativa)
- Coordenadas GPS (posición absoluta)
- **Altura**, respecto a nivel del mar o suelo
- **Throttle/Thrust o uso/empuje** de los motores
- Nivel de batería
- Velocidades y aceleraciones en cada eje



PRIMERAS PRUEBAS



PRUEBAS CON PAPARAZZI

Paparazzi Center

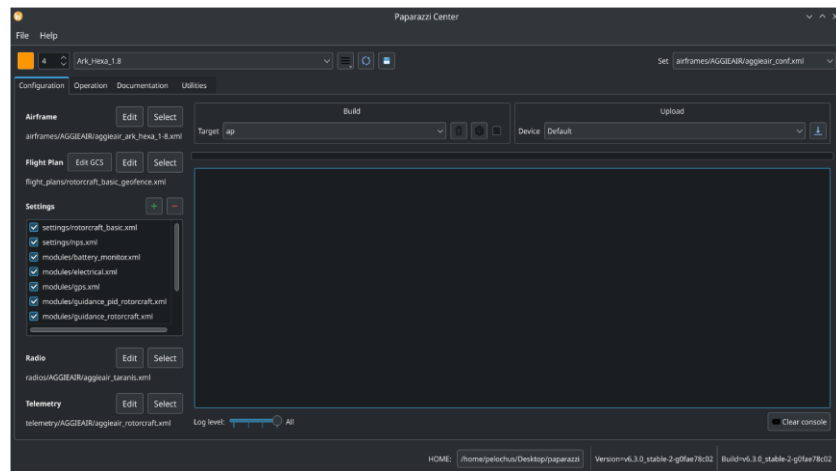
- Sustituye a Crazyflie Client
- Incluye otras herramientas como Paparazzi GCS para control de los drones

Simulaciones

- Podemos realizar simulaciones de vuelo usando Paparazzi Center + Paparazzi Ground Control Station

Firmware Crazyflie 2.1

- Instalamos dependencias diversas y ajustamos ciertos parámetros
- Cambiamos el firmware al firmware de Paparazzi siguiendo la guía oficial



CARGAR FIRMWARE DE PAPARAZZI

The screenshot displays the Paparazzi GCS interface with the following elements:

- Top Bar:** Includes 'File' and 'Help' menus, a dropdown menu set to 'CRAZYFLIE', and a 'Set' dropdown menu set to 'airframes/test_boards/conf_test_boards.xml'.
- Left Panel:**
 - Configuration:** Contains sections for 'Airframe' (set to 'airframes/UGR/crazyflie_2.1.xml'), 'Flight Plan' (set to 'flight_plans/ENAC/crazyflie_multi_ranger_test.xml'), 'Settings' (a list of checked modules including 'rotorcraft_basic.xml', 'ahrs_madgwick.xml', 'air_data.xml', 'electrical.xml', 'gps.xml', and 'gps_ublox.xml'), 'Radio' (set to 'radios/cockpitSX.xml'), and 'Telemetry' (set to 'telemetry/default_rotorcraft.xml').
- Build Section:** Features a 'Target' dropdown set to 'ap' and a 'Device' dropdown set to 'USB DFU-UTIL'.
- Upload Section:** Contains a large text area displaying the progress of the firmware upload, showing a series of 'Download' messages with increasing percentages from 54% to 100%, followed by 'Download done.' and 'File downloaded successfully'. The final status is 'Flash CRAZYFLIE Done'. A 'Clear console' button is located at the bottom right of this section.
- Bottom Bar:** Displays the 'HOME' path as '/home/pelochus/Desktop/paparazzi', the version as 'v6.3.0_stable-2-g0fae78c02-dirty', and the build as 'v6.3.0_stable-2-g0fae78c02-dirty'.

CONEXIÓN CON RADIO

The screenshot displays the Paparazzi GUI interface for configuring and operating a radio connection. The top bar shows the configuration file set to `airframes/test_boards/conf_test_boards.xml`. The main interface is divided into several sections:

- Configuration:** Includes tabs for Configuration, Operation (selected), Documentation, and Utilities.
- control panel:** Shows the selected control panel as `control_panel_example.xml` and the session as `Flight USB-serial@57600`.
- Programs:** A list of programs to be loaded. Two programs are listed:
 - `/home/pelochus/Desktop/paparazzi/sw/ground_segment/python/bitcraze/crazyradio2ivy.py` (highlighted with a red box)
 - `/home/pelochus/Desktop/paparazzi/sw/ground_segment/tmtc/messages` (highlighted with a green box)
- Messages:** A window displaying the received messages. The selected message is `telemetry:5`. The messages are listed in a table:

Message	Value
AHRS_REF_QUAT	int32 ins_x : 0 (0.000000m)
AIR_DATA	int32 ins_y : 0 (0.000000m)
ALIVE	int32 ins_z : -116 (-0.453131m)
AUTOPILOT_VERSION	int32 ins_xd : 0 (0.000000m/s)
DATALINK_REPORT	int32 ins_yd : 0 (0.000000m/s)
DL_VALUE	int32 ins_zd : -41644 (-0.079124m/s)
ENERGY	int32 ins_xdd : 30 (0.029298m/s ²)
GPS_INT	int32 ins_ydd : -12 (-0.011719m/s ²)
I2C_ERRORS	int32 ins_zdd : 5 (0.004883m/s ²)
INS	1
INS_REF	
ROTORCRAFT_FP	
ROTORCRAFT_NAV_STATUS	
ROTORCRAFT_STATUS	
RTOS_MON	

The status bar at the bottom indicates the Paparazzi version and build information: `Version=v7.0_unstable-none-g1598913ce Build=v7.0_unstable-none-g88f769a65`.

SIMULACIÓN



DESARROLLO BÁSICO CON PAPARAZZI

```
<!DOCTYPE module SYSTEM "module.dtd">
```

```
<module name="demo_module">
  <doc>
    <description>Demo module</description>
  </doc>
  <header>
    <file name="demo_module.h"/>
  </header>
  <init fun="init_demo()"/>
  <periodic fun="periodic_1Hz_demo()" freq="1." start="start_demo()" stop="stop_demo()" autorun="TRUE"/>
  <periodic fun="periodic_10Hz_demo()" period="0.1" start="start_demo()" stop="stop_demo()" autorun="FALSE"/>
  <makefile>
    <raw>
#Example of RAW makefile part
    </raw>
    <define name="DEMO_MODULE_LED" value="2"/>
    <file name="demo_module.c"/>
  </makefile>
  <makefile target="demo">
    <define name="SOME_FLAG"/>
    <configure name="SOME_DEFINE" value="bla"/>
  </makefile>
</module>
```

Lenguaje **C** y archivos **XML**

Paparazzi Center

File Help

CRAZYFLIE

ConfigurationOperationDocumentationUtilities

control panelSession

control_panel.xmlSimulation

Programs

+ Add tool

loggie

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Toggle

Log level: All

HOME: /home/pelochus/Desktop/paparazzi Version=v7.0_unstable-13-g1

pprzgcs

File Aircrafts Help Map

SRTM No SRTM WGS84 43.564238, 1.478820

CRAZYFLIE

demo

searching...

periodic_10Hz_demo 0

START STOP

periodic_1Hz_demo 1

START STOP

CRAZYFLIE

00:00:00

FPInit

4.20 V

-0m | -0.3 m

NAV

0.0 %

0.0 m/s

CRAZYFLIE

Blocks

Details

FPInit

Holding point

Start Engine

Takeoff

Standby

Avoid

Guided avoid

Land Here

Land

Land Target

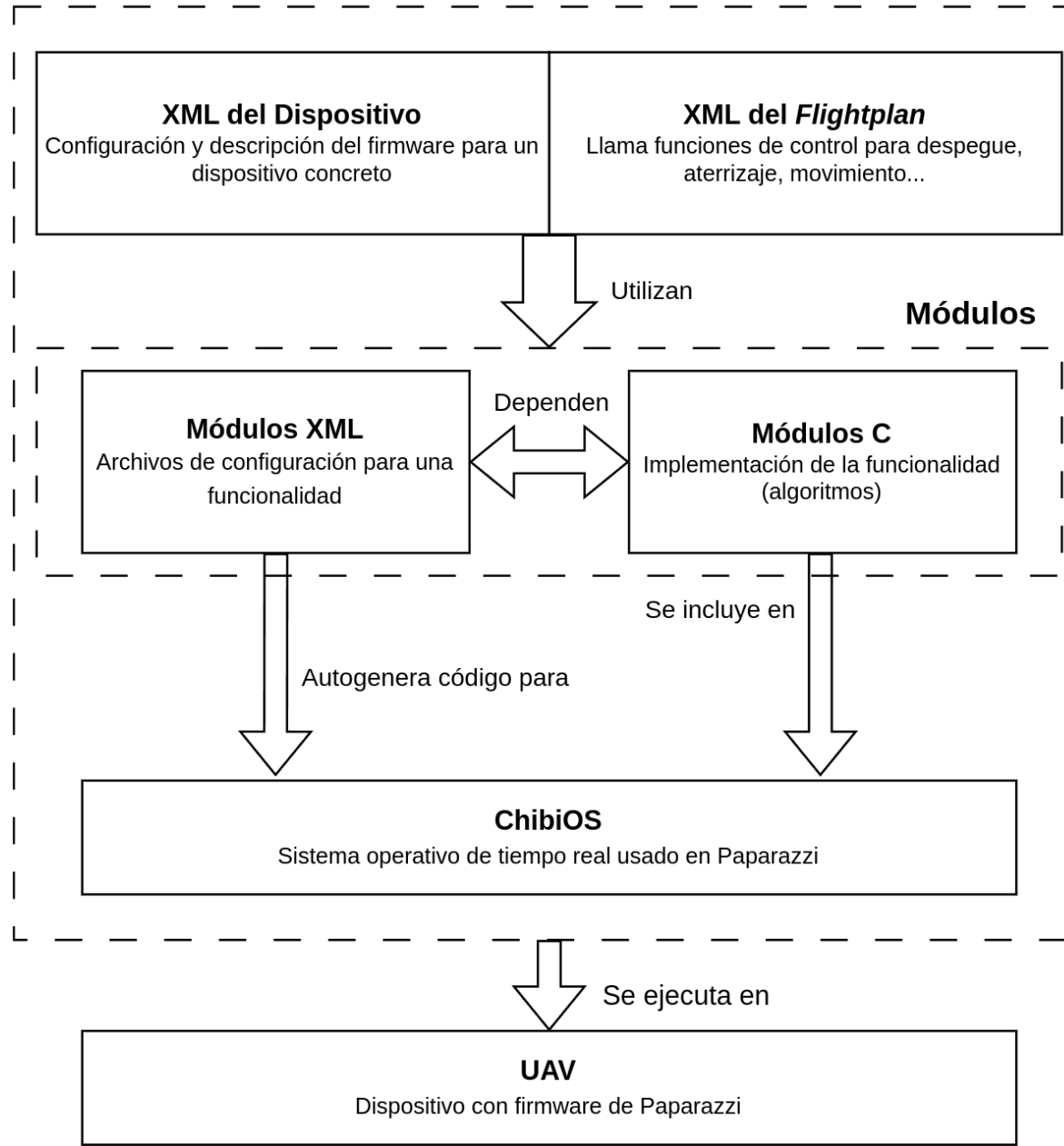
Ramp down

Kill Engine

HOME

server status:

Firmware



ÍNDICE

1. Introducción

1. Motivación
2. Descripción general
3. Objetivos
4. Planificación

2. Preliminares

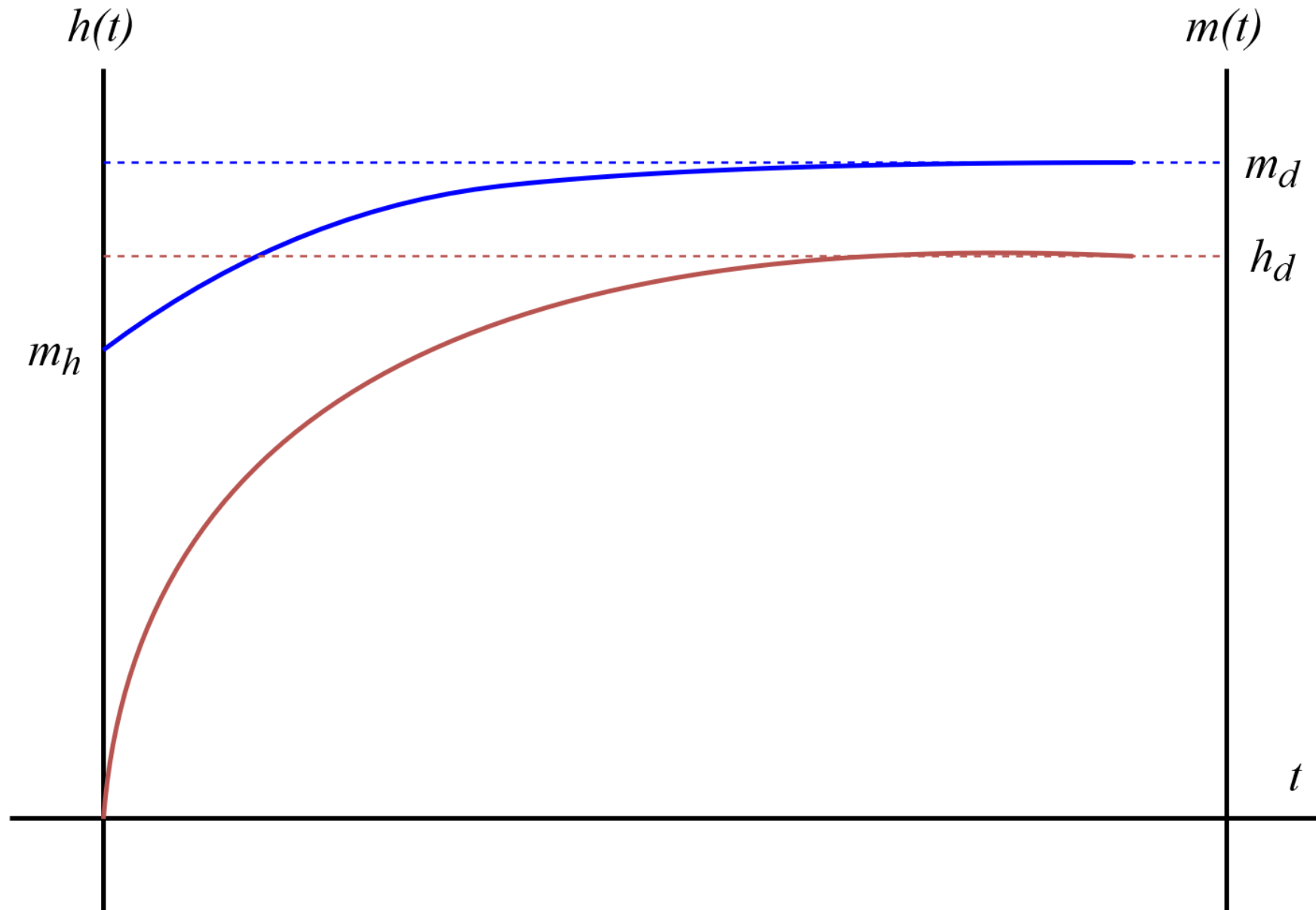
1. Pruebas con software oficial
2. Pruebas con Paparazzi
3. Desarrollo básico en Paparazzi

3. Control de un Crazyflie

1. Control básico
2. Algoritmos GVF
3. Implementación de GVF
4. Coordinación entre Crazyflies
 1. Algoritmos de coordinación
 2. Implementación de la coordinación
 3. Simulación de formaciones
 4. Resultados de las formaciones
5. Conclusiones

(DES) CONTROL BÁSICO



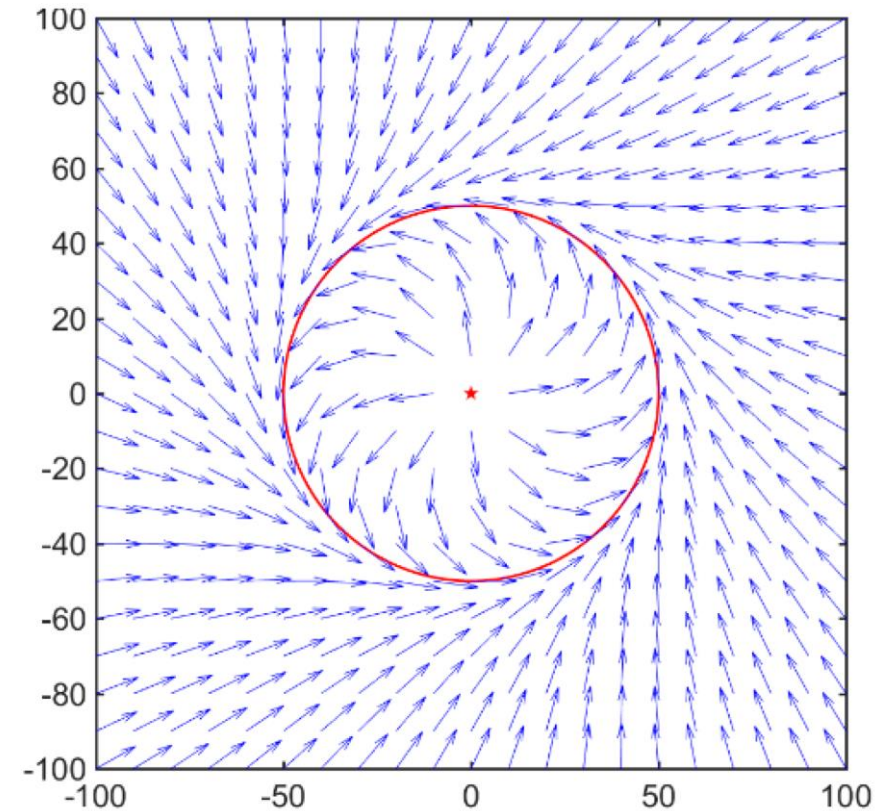
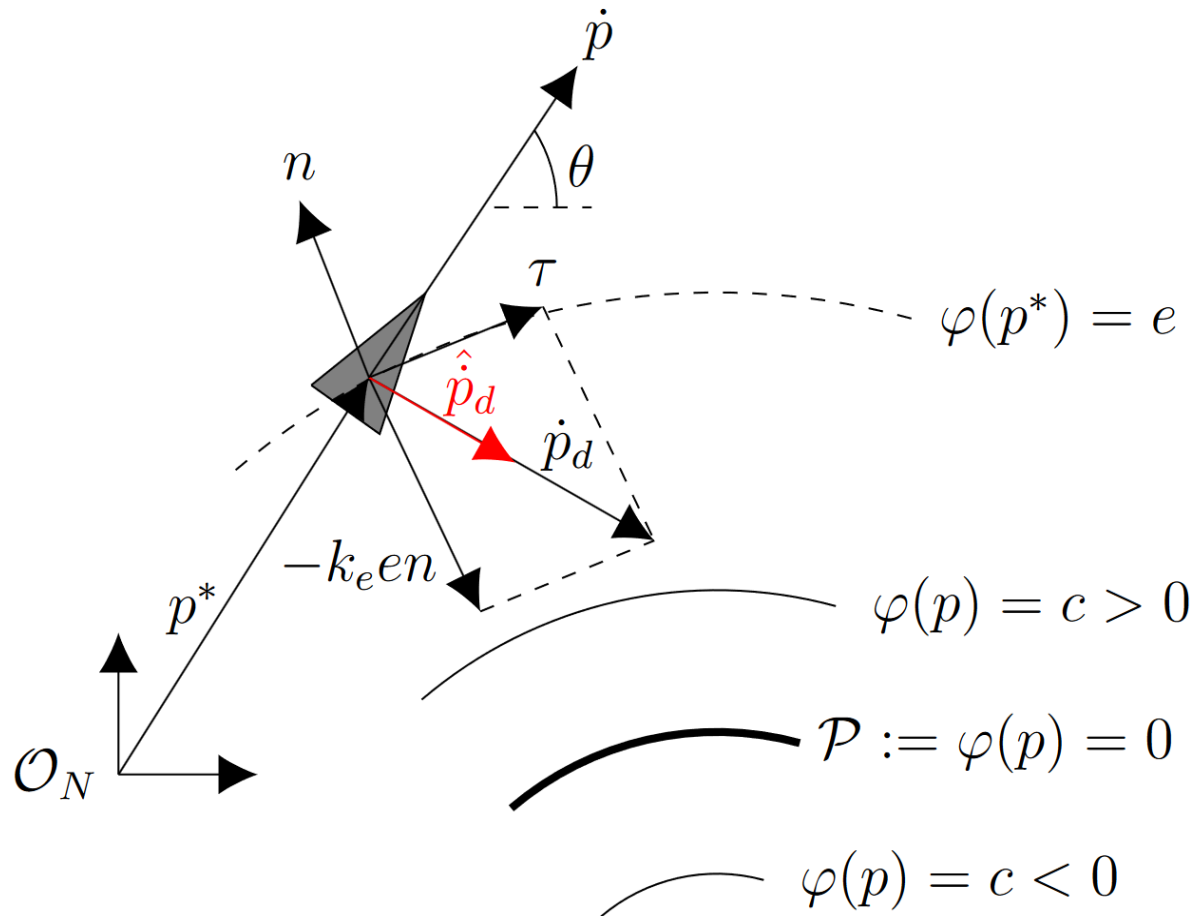


CONTROL BÁSICO: HOVERING



ALGORITMOS GVF

Campo vectorial de guiado, del inglés *Guiding Vector Field*



IMPLEMENTACIÓN DEL ALGORITMO

Control mínimo

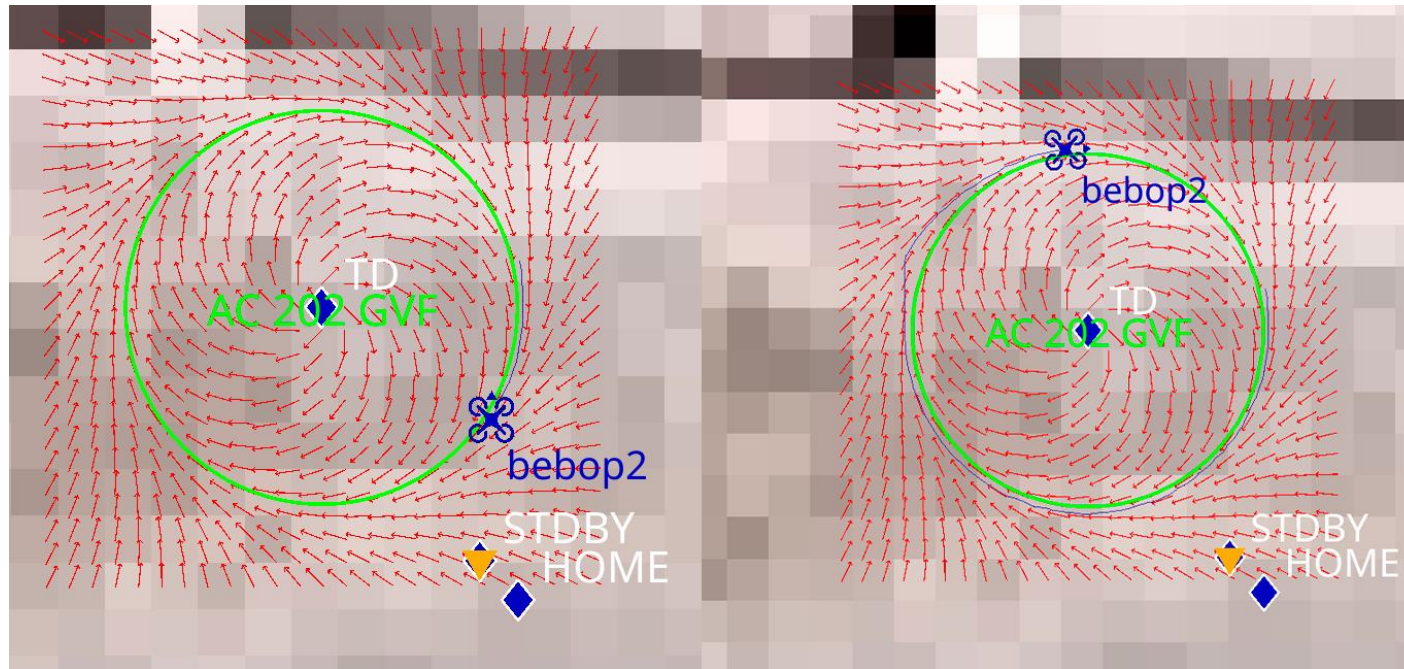
- Integrado parcialmente en Paparazzi
- Falta el control para rotorcrafts

Control de velocidad

- Aprovechando las variables de GVF, podemos añadir control de velocidad constante

Alineación con trayectoria

- Los rotorcrafts pueden alinearse o no con la trayectoria, al tener más grados de libertad
- Modificamos el **yaw** para que mire en la dirección del campo



IMPLEMENTACIÓN DEL ALGORITMO: PRUEBA

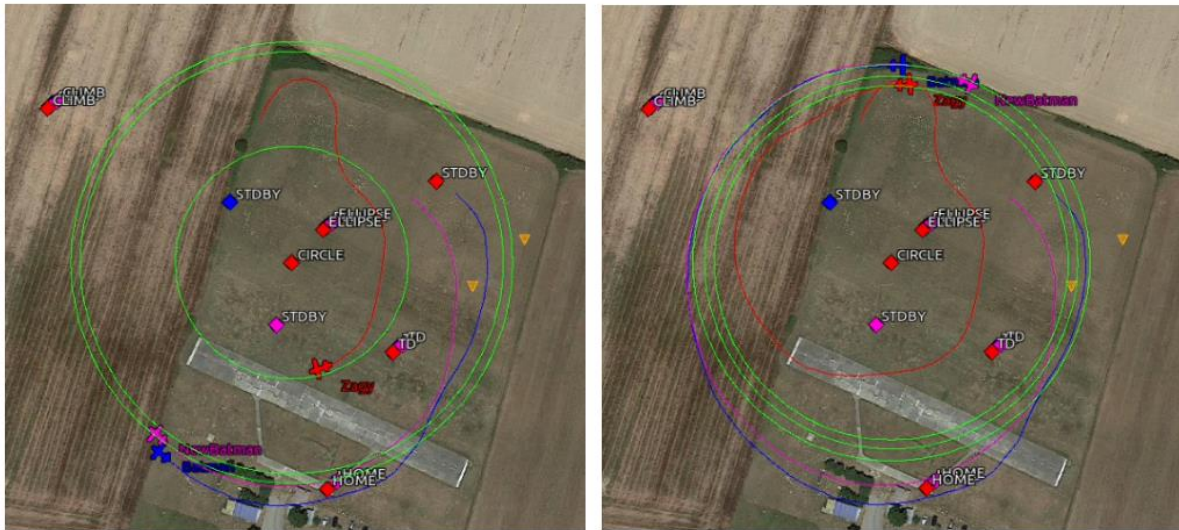
ÍNDICE

1. Introducción
 1. Motivación
 2. Descripción general
 3. Objetivos
 4. Planificación
2. Preliminares
 1. Pruebas con software oficial
 2. Pruebas con Paparazzi
 3. Desarrollo básico en Paparazzi
3. Control de un Crazyflie
 1. Control básico
 2. Algoritmos GVF
 3. Implementación de GVF
4. **Coordinación entre Crazyflies**
 1. **Algoritmos de coordinación**
 2. **Implementación de la coordinación**
 3. **Simulación de formaciones**
 4. **Resultados de las formaciones**
5. Conclusiones

ALGORITMOS DE COORDINACIÓN

Formaciones Circulares

- UAVs se coordinan para ir en círculos con cierto desfase deseado entre ellos **con velocidad constante**
- Integrado parcialmente en Paparazzi, falta soporte para rotorcrafts



Formaciones en segmentos paralelos

- Ida y vuelta en segmentos de misma longitud y paralelos, los UAVs deberán tener un desfase fijo
- Nueva implementación y nuevo algoritmo sólo para rotorcrafts

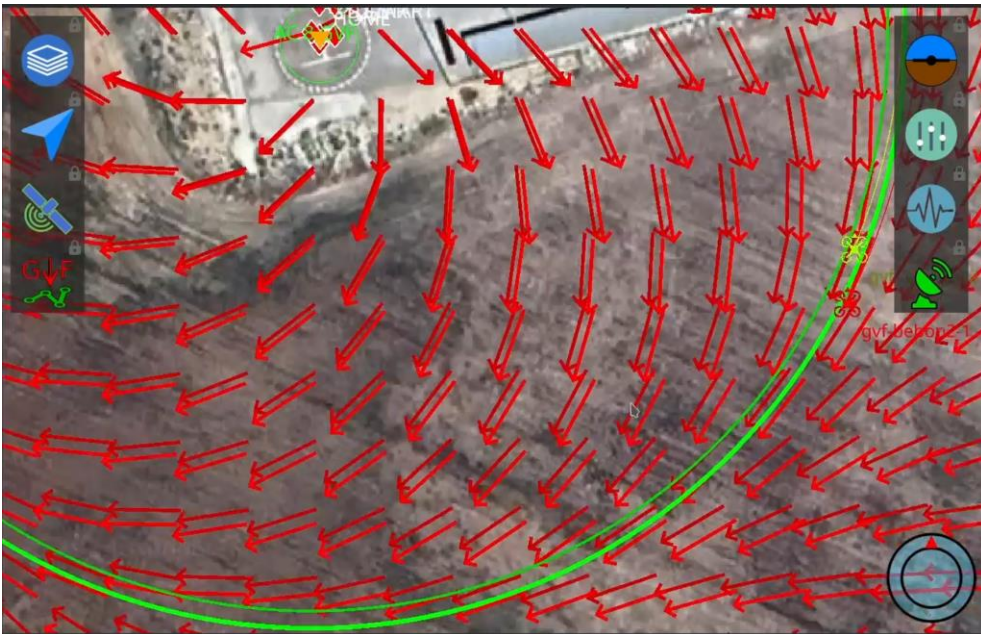


Ambos están pensados para aplicarse sobre un algoritmo de control, no necesariamente **GVF**

IMPLEMENTACIÓN DE LA COORDINACIÓN

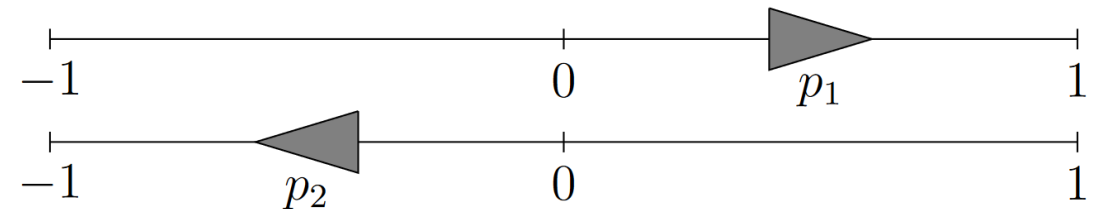
Formaciones Circulares

- Al estar ya integrado en Paparazzi, tan sólo se modifica un script de Python para que pueda interactuar con rotorcrafts (distintas variables de telemetría, control etc...). **Pensado para usarse con GVF**

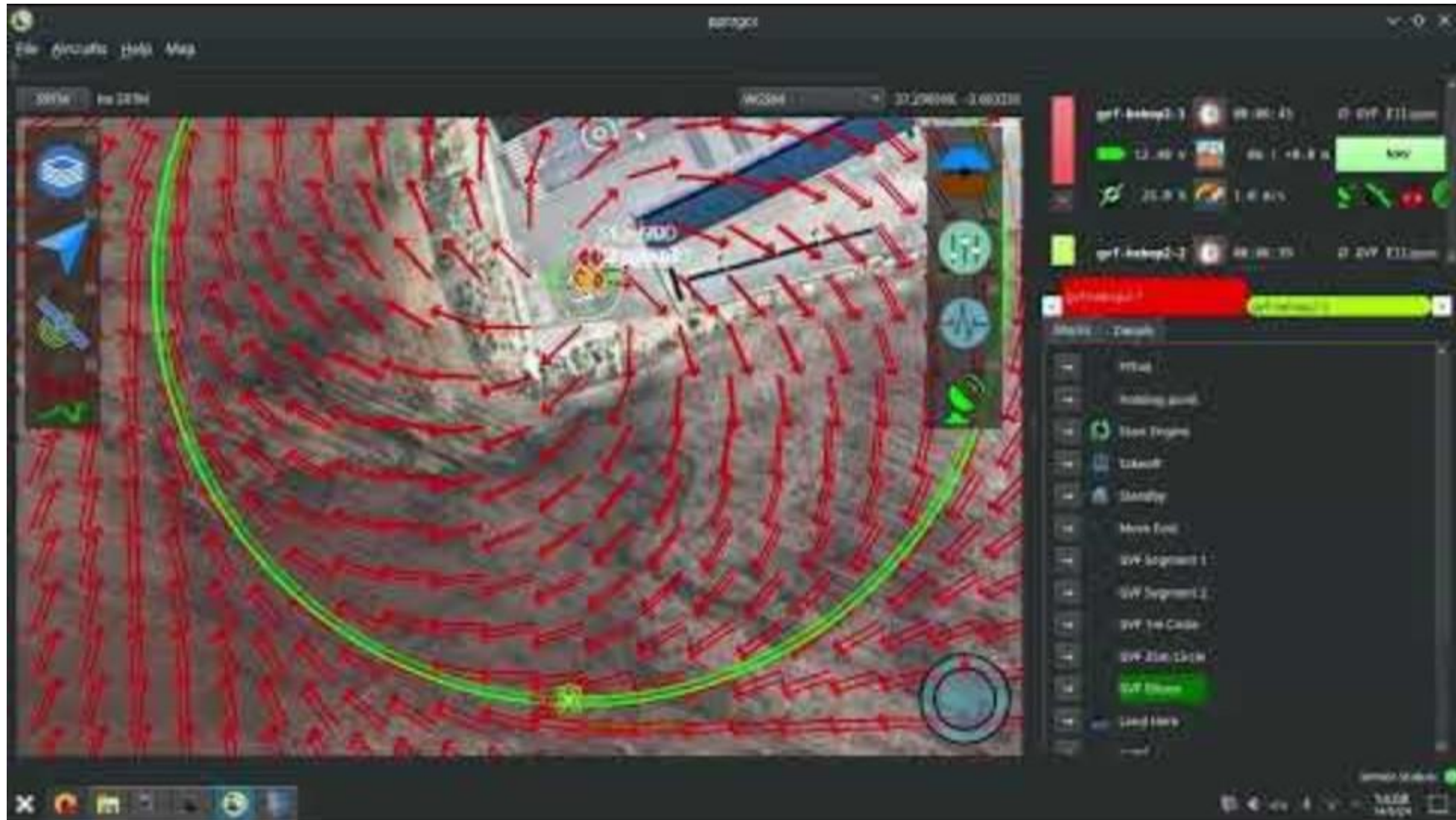


Formaciones en segmentos paralelos

- Basado en el ejemplo de formaciones circulares, modificamos la implementación para que funcione con segmentos (puede o no con GVF)
- Se añade normalización del segmento



SIMULACIÓN DE FORMACIONES: CÍRCULOS



SIMULACIÓN DE FORMACIONES: CÍRCULOS



SIMULACIÓN DE FORMACIONES: SEGMENTOS



SIMULACIÓN DE FORMACIONES: SEGMENTOS



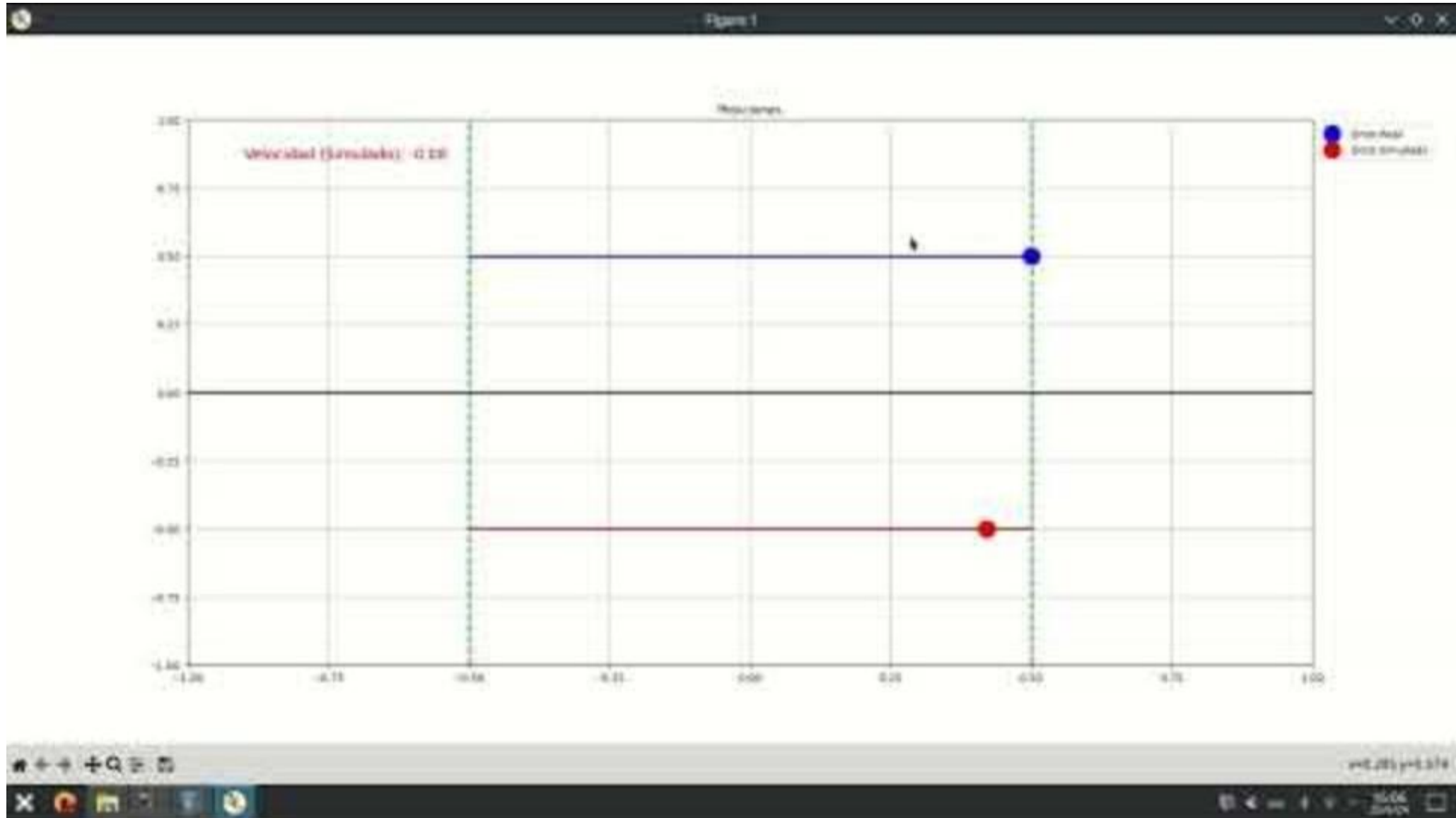
RESULTADOS DE LAS FORMACIONES: PAPARAZZI



RESULTADOS DE LAS FORMACIONES: BITCRAZE



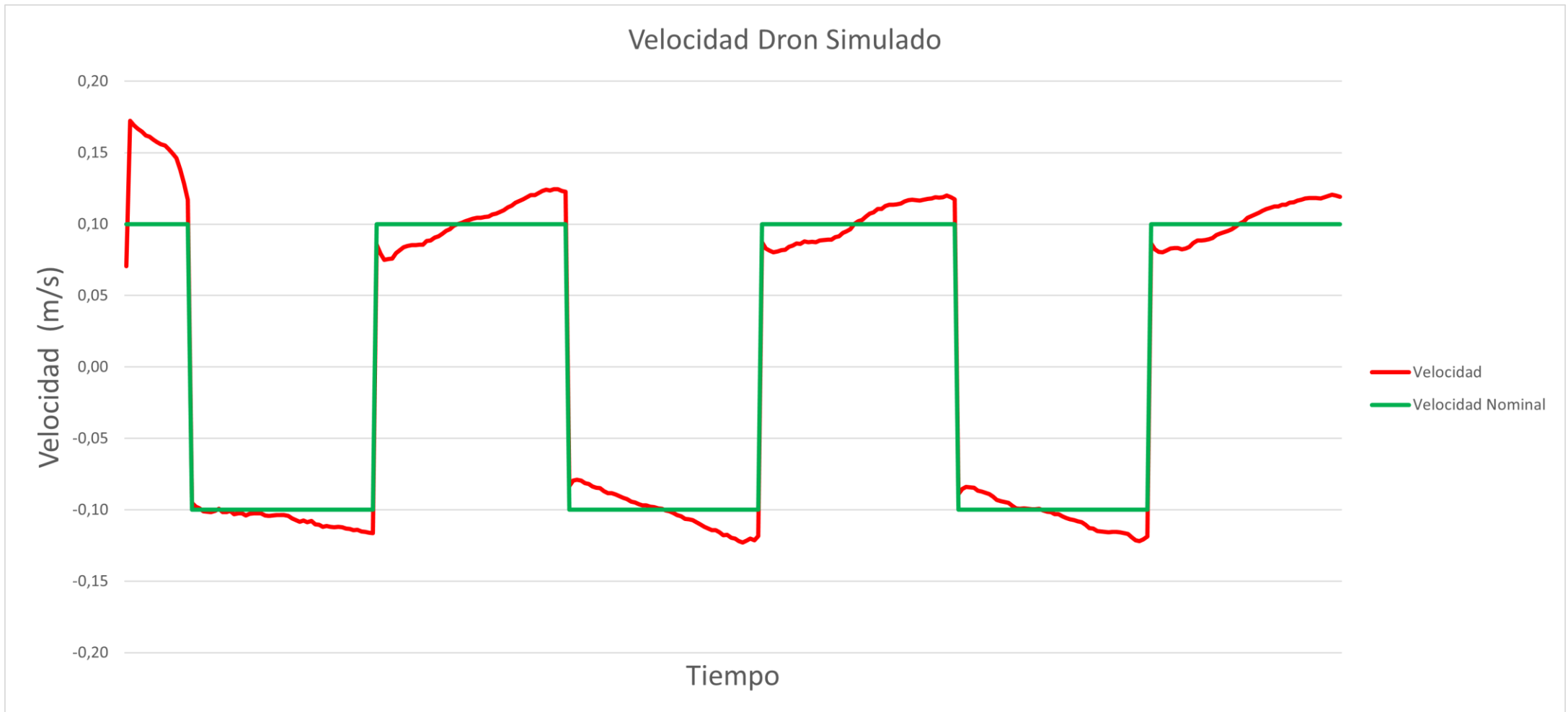
RESULTADOS DE LAS FORMACIONES



RESULTADOS DE LAS FORMACIONES



RESULTADOS DE LAS FORMACIONES



ÍNDICE

1. Introducción
 1. Motivación
 2. Descripción general
 3. Objetivos
 4. Planificación
2. Preliminares
 1. Pruebas con software oficial
 2. Pruebas con Paparazzi
 3. Desarrollo básico en Paparazzi
3. Control de un Crazyflie
 1. Control básico
 2. Algoritmos GVF
 3. Implementación de GVF
4. Coordinación entre Crazyflies
 1. Algoritmos de coordinación
 2. Implementación de la coordinación
 3. Simulación de formaciones
 4. Resultados de las formaciones
5. Conclusiones

CONCLUSIONES

Planificación (Resultado Real)																			
Febrero				Marzo				Abril				Mayo				Junio			
W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
	OBJ-1																		
		OBJ-2																	
								OBJ-3											
												OBJ-4							
																OBJ-5			
		OBJ-6																	

CONCLUSIONES

- Resultados prometedores, GVF permite buen control y la coordinación tiende a bajo error.
- Aplicaciones y usos futuros muy diversos
- Muchas ramificaciones futuras: mejor coordinación, coordinación descentralizada, uso para N drones...
- En general, buenos resultados y potencial a futuro

FIN

¿PREGUNTAS?