

---

# **Control en Red basado en Predictor-Observador No Uniforme (NUPO) para Sistemas Aéreos Pilotados de forma Remota (RPAS)**

---

*Autor:*

Javier ERRANDO HERRANZ  
*errando.javier@gmail.com*

*Tutor:*

Ángel Miguel CUENCA LACRUZ  
*acuenca@isa.upv.es*

*Co-tutor:*

Pedro José GARCÍA GIL  
*pggil@isa.upv.es*

DEPARTAMENTO DE ING. DE SISTEMAS Y AUTOMÁTICA  
UNIVERSITAT POLITÈCNICA DE VALÈNCIA



18 de julio de 2015

## **Resumen**

El objetivo del presente documento es exponer la Tesina de Fin de Máster desarrollada por el alumno. Ésta tiene como finalidad llevar a cabo el control de un Sistema Aéreo Pilotado de forma Remota (RPAS). El escenario que se presenta es un sistema de control en red o Networked Control System (NCS), en el que aparecen problemas debidos a retardos, desorden y pérdida de paquetes, tanto en el canal sensor-controlador como en el canal controlador-actuador. Para solventar dichos problemas, se aplicarán estrategias de control basadas en un Predictor - Observador No Uniforme (NUPO), además de un control basado en paquetes para el canal controlador-actuador. En primer lugar, se afrontará el caso de retardos fijos, en el cual no aparece el desorden de paquetes. En segundo lugar, se analizará el caso de retardos variantes, donde tenemos todos los problemas expuestos, incluyendo el desorden de datos. Por último, se aplicará lo expuesto en la plataforma real, compuesta por un dron del tipo quadrotor y un PC que envía y recibe datos de forma remota, a través de una red Wi-Fi.

## **Abstract**

The aim of this document is to expose the Final Master Thesis developed by the student. Its objective is to control a Remotely Piloted Aircraft System (RPAS). The scenario consists on a Networked Control System (NCS), which presents problems due to delays, packet loss and data disorder, in sensor-controller and controller-actuator channels. In order to solve these problems, Non-Uniform Predictor-Observer control strategies will be applied, as well as a packet-based system for the controller-actuator channel. Firstly, a constant delay case is going to be faced, in which the packet disorder problem does not appear. Secondly, a variable delay case will be analyzed, where every exposed problem appears, data disorder included. Finally, these strategies will be put into practice in the real platform, formed by a quadrotor (drone) and a PC which sends and receives data remotely through a Wi-Fi network.

# Índice

<b>Índice de figuras</b>	<b>3</b>
<b>1 Introducción y motivación</b>	<b>4</b>
1.1 Ventajas . . . . .	6
1.2 Desventajas . . . . .	7
<b>2 Descripción del problema</b>	<b>8</b>
<b>3 Solución adoptada</b>	<b>10</b>
3.1 Selector de medidas. Controlador . . . . .	11
3.2 Observador no uniforme . . . . .	12
3.3 Predictor . . . . .	14
3.4 Ley de control . . . . .	15
3.4.1 Control basado en paquetes . . . . .	15
3.5 Selector de medidas. Planta . . . . .	17
<b>4 Simulación</b>	<b>19</b>
4.1 Características del sistema . . . . .	19
4.2 Estudios previos. Canal sensor - controlador . . . . .	21
4.2.1 Retardo fijo . . . . .	21
4.2.2 Retardo variable . . . . .	24
4.3 Estrategia propuesta. Buffer de acciones de control . . . . .	26
4.3.1 Retardo fijo . . . . .	26
4.3.2 Retardo variable . . . . .	28
4.4 Comparación . . . . .	31
<b>5 Aplicación real</b>	<b>35</b>
5.1 Caracterización del funcionamiento . . . . .	36
5.2 Validación del predictor . . . . .	37
5.3 Validación del buffer . . . . .	40
<b>6 Conclusiones</b>	<b>44</b>
<b>Referencias</b>	<b>45</b>

# Índice de figuras

1	Esquema general NCS. . . . .	4
2	Problemática NCS. . . . .	8
3	Solución adoptada . . . . .	10
4	Selector de medidas. Controlador. Modo de operación . . . . .	11
5	Controlador completo . . . . .	16
6	Selector de medidas. Planta. Modo de operación . . . . .	18
7	Retardo fijo. Diferentes $d_k$ . . . . .	21
8	Retardo fijo. Diferentes $h$ . . . . .	23
9	Distribución retardos variables. Canal SC . . . . .	24
10	Retardo variable. Diferentes $h$ . . . . .	25
11	Comparación empleo del buffer. Retardo fijo . . . . .	26
12	Empleo del buffer. Diferentes $h$ . Retardo fijo . . . . .	27
13	Distribución retardos variables SC y CA . . . . .	28
14	Comparación empleo del buffer. Retardo variable . . . . .	29
15	Empleo del buffer. Diferentes $h$ . Retardo variable . . . . .	30
16	Comparación paper - buffer. Retardo fijo . . . . .	31
17	Comparación paper - buffer. Retardo variable . . . . .	32
18	Comparación paper - buffer. Retardo variable. Detalle . . . . .	33
19	Plataforma de laboratorio . . . . .	35
20	Funcionamiento normal ante distintos retardos . . . . .	36
21	Laboratorio. Retardo SC. Diferentes $h$ . . . . .	37
22	Laboratorio. Retardo CA. Diferentes $h$ . . . . .	38
23	Laboratorio. Retardos SC y CA. Diferentes $h$ . . . . .	38
24	Laboratorio. Empleo buffer. Retardo fijo SC . . . . .	40
25	Laboratorio. Empleo buffer. Retardo fijo SC y CA . . . . .	41
26	Laboratorio. Distribución retardos variables. . . . .	42
27	Laboratorio. Empleo buffer. Retardo variable . . . . .	43

# 1 Introducción y motivación

Los sistemas de control basados en red o *Networked Control Systems* (NCS) son un tipo especial de sistemas. En ellos, la información se transmite entre los distintos componentes que lo forman a través de una red compartida. Dicha red puede estar formada por buses de campo (CAN), *Ethernet*, RS-232 o redes inalámbricas, como por ejemplo *Bluetooth* o *WiFi*. Ésta última es la que se ha empleado para este TFM. Así, planta y controlador no comparten información de manera directa, sino a través de una red que, además, puede estar siendo empleada para otras tareas, relacionadas o no con el sistema a controlar. A este respecto, se pueden considerar dos perspectivas en un esquema NCS: el Control Directo (*Direct Control*) y el Control Jerárquico (*Hierarchical Control*). En la primera, el controlador se implementa íntegramente en la parte remota. En la segunda, se emplea un control remoto y otro local junto al proceso, como es el caso de numerosos tipos de control multifrecuencia.

En ocasiones, esta red se considera compartida solo en un sentido (sensor-controlador), mientras que para la inyección de acciones de control (controlador-actuador) se emplea un cableado o red dedicada. Para este trabajo consideraremos presencia de red en ambas, tal como se muestra en la Figura 1.

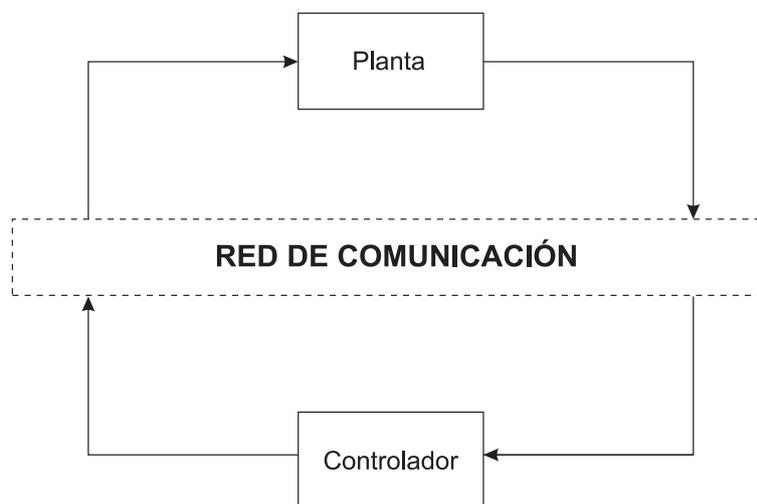


Figura 1: Esquema general NCS.

Las aplicaciones de los sistemas de control en red son múltiples. Por una parte, tienen utilidad en el ámbito propiamente industrial: manufactura, automoción, aeronáutica, etc. Otras aplicaciones son el diagnóstico remoto y la tele-operación,

la exploración espacial o el sector sanitario. Por otra parte, también son usados en el ámbito investigador y académico.

## 1.1 Ventajas

El hecho de realizar un control basado en red introduce una serie de ventajas que cabe remarcar, como por ejemplo las siguientes:

- Los NCS permiten eliminar cableado que en un sistema de control convencional sería necesario, ya que para uno o ambos canales se puede emplear un bus común por el que enviar/recibir las tramas. Esto repercute en una mayor facilidad de mantenimiento, además de una clara reducción de costes, lo cual siempre es ventajoso, tanto a nivel industrial como a nivel de investigación. Un menor volumen de cableado puede ser decisivo en instalaciones donde el espacio es limitado (aviónica) o donde es necesario cubrir largas distancias (plantas químicas).
- Otra ventaja radica en los cambios estructurales. Así, si se desea diseñar un control más complejo (numerosos filtros, redes neuronales, etc.), pero se dispone de hardware local anticuado, gracias a los NCS no es necesario reemplazarlo por uno más moderno y potente. Por el contrario, el hardware remoto será el que lleve a cabo todos los cálculos. Esto es muy ventajoso en los casos en que dispones de varios elementos a controlar, ya que no se requieren cambios en todos ellos, sino solamente en el dispositivo remoto.
- La tercera ventaja está más relacionada con los vehículos móviles en sí. La realización de misiones conjuntas por parte de varios UAVs (o incluso también junto a vehículos terrestres) es un ámbito que está muy en auge en los últimos tiempos, tanto para temas militares (bombardeos, patrullas) como civiles (topografía, extinción de incendios). Para estos casos, es necesaria la sincronización de las plataformas, por lo que conviene tener un controlador centralizado en remoto que capte información de todos los RPAS y envíe las acciones de control correspondientes en cada periodo de muestreo. Con varios controles en local en cada UAV no se tiene certeza de cuál es el estado actual de cada una de ellas, siendo muy posible que se pierda la sincronización y fracase la misión.

## 1.2 Desventajas

Por otra parte, el control en red presenta una serie de desventajas asociadas, las cuales se exponen a continuación.

- La primera, los retardos variantes debidos a la comunicación en red. Al no encontrarse controlador, sensor y actuador en el mismo dispositivo, aparecen una serie de retrasos que dificultan la consecución de los objetivos de control. Al tratarse de un medio de comunicación compartido y no encontrarse los dispositivos conectados físicamente, no se puede garantizar que la información llegue al destino de manera instantánea.
- Otra desventaja radica en la pérdida de paquetes, también debida al uso de la red. En ocasiones puede ocurrir que, a pesar de haber enviado determinada información al controlador (o actuador, dependiendo del caso), ésta se pierda y no llegue. Así, en este documento se propondrán estrategias para tratar de paliar tanto esta como la primera desventaja.
- El ancho de banda también es un problema que encontramos en los NCS. A este respecto, la planta y controlador a comunicar no pueden ocupar toda la banda disponible en la red, ya que ésta es empleada también, como ya hemos dicho, por diferentes dispositivos. Por ello, puede ocurrir que el periodo de muestreo esté limitado, y las prestaciones de control queden por ello también limitadas.
- Al tratarse de dispositivos diferentes, se puede producir el problema de que los muestreadores se desfasen uno con respecto al otro. Dependiendo del sistema, esto puede hacer que se requieran métodos de sincronización entre ambas partes.

## 2 Descripción del problema

Como ya ha sido puesto de manifiesto anteriormente, en un sistema en red se dan una serie de problemas. Sin embargo, en un sistema de control convencional, el cual podemos expresar en representación interna o espacio de estados por las ecuaciones

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \quad (1)$$

la acción de control se aplica instantáneamente tras su cálculo, si tomamos como despreciable el tiempo de cómputo. Del mismo modo, el tiempo que pasa desde que se toma una medida  $y(t)$  hasta que se procede al cálculo de la acción  $u(t)$  también es nulo o despreciable.

Por el contrario, en un sistema de control basado en red aparecen retardos entre los diferentes componentes que lo forman, además de posibles pérdidas de datos. Así, como planta, sensor y actuador están separados del controlador, se producen retardos en ambos canales, tanto en sensor-controlador (SC) como en controlador-actuador (CA). Así, y notando dichos retardos como  $d_k^{SC}$  y  $d_k^{CA}$ , respectivamente para sensor-controlador y controlador-actuador, tenemos lo mostrado en la Figura 2.

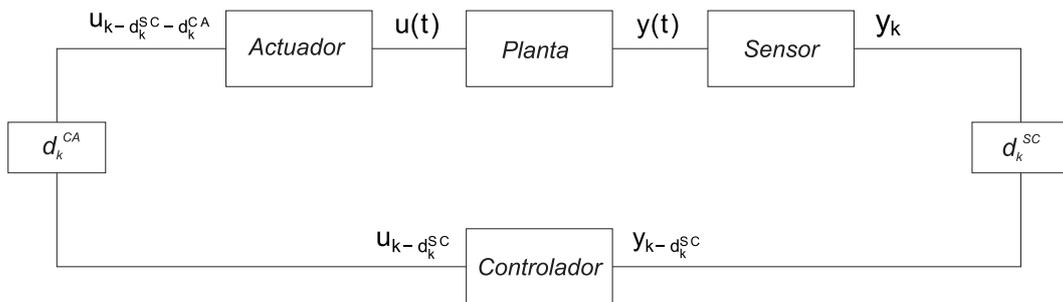


Figura 2: Problemática NCS.

Como se puede apreciar, el controlador recibe una señal retrasada y no uniforme. Éste computa una acción de control, la cual es enviada a través del canal CA, por lo que el actuador recibe la señal con un retardo añadido. Este fenómeno se repite en cada iteración.

Por otra parte, si los retardos son mayores que el periodo de muestreo del sistema a controlar, aparece el desorden de paquetes. Éste es un aspecto importante, ya que el controlador recibe, además de una señal retrasada, paquetes que pueden ser más antiguos que los recibidos en una iteración anterior. Esto puede llevar a errores en el tratamiento de los datos por parte del controlador, así como en la recepción de los mismos por parte de planta.

Por todo ello, se tendrá que actuar en consonancia para tratar de cubrir dichos retardos, desordenes y pérdidas. En las siguientes secciones se propondrán y analizarán las diferentes técnicas empleadas para ello: en primer lugar para corregir el desorden de paquetes. En segundo lugar, para compensar el retardo existente en *SC* y *CA*. Por último, también se aplicarán estrategias para corregir una o varias posibles pérdidas de datos consecutivas.

### 3 Solución adoptada

La solución de control empleada para solventar los problemas existentes pasa por diferentes etapas, cada una diseñada para cubrir un aspecto distinto. La parte del controlador está formada por: selector de medidas, observador, predictor y ley de control. La parte de la planta está conformada por, aparte de actuador, sensor y la propia planta en sí, un selector de medidas. Este selector tiene una estructura diferente al primero. Todas estas etapas serán explicadas y desarrolladas a lo largo de esta sección.

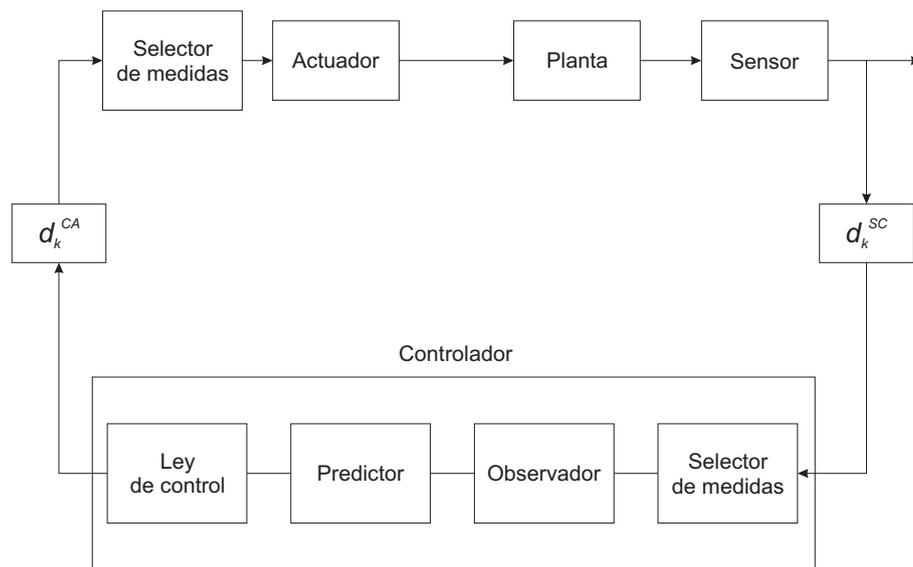


Figura 3: Solución adoptada

### 3.1 Selector de medidas. Controlador

En primer lugar, se expondrá la etapa del selector de medidas correspondiente a la parte del controlador. Ésta tiene un funcionamiento sencillo, basado en técnicas de sincronización y time-stamping, que facilita el tratamiento de muestras desordenadas. Nos basaremos en la expresión

$$\tilde{y}_k = \theta_k y_k \quad (2)$$

siendo  $y_k$  la muestra que llega al controlador en el instante  $k$ . Así, la muestra sale del sensor en dicho instante junto con una muestra del tiempo actual. Si ésta es más reciente que la muestra del instante  $k - 1$ , se toma como válida, haciendo  $\theta_k = 1$ . Sino, se descarta y se toma como un paquete perdido,  $\theta_k = 0$ . Si se da este caso, los estados en dicho instante deberán ser observados.

Dependiendo de la cantidad de medidas desordenadas/perdidas, las siguientes etapas tendrán un comportamiento diferente. Para ello, cada vez que se produce una pérdida se aumenta un contador  $N_k$ . Dicho contador será comunicado al observador no uniforme, el cual calculará los estados observados de acuerdo a  $N_k$ . El funcionamiento de dicha etapa se muestra en el siguiente apartado. En la Figura 4 se puede observar cómo se comporta el selector de medidas, a modo de ejemplo.

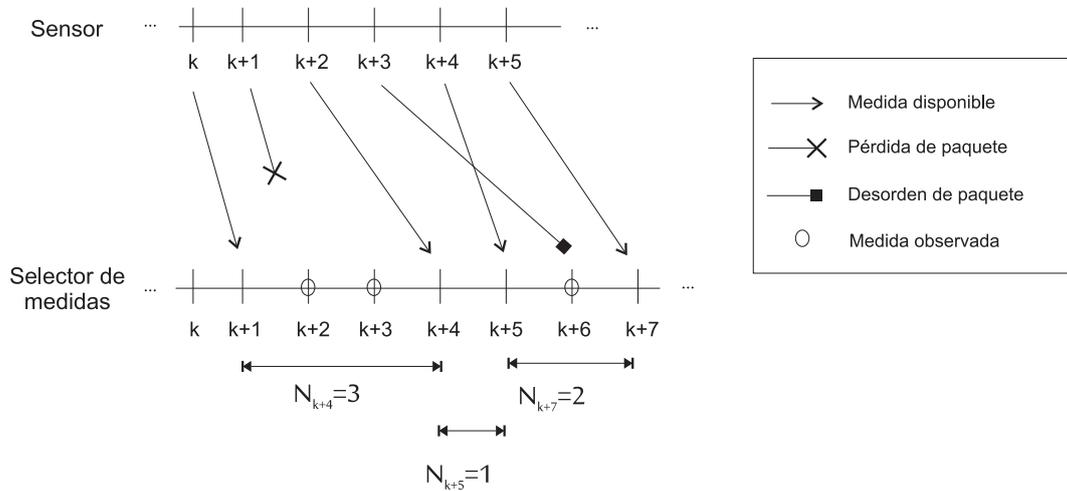


Figura 4: Selector de medidas. Controlador. Modo de operación

### 3.2 Observador no uniforme

Tras pasar por el selector de medidas, la siguiente etapa consiste en la inclusión de un observador. Esto se debe al hecho de que, con los actuadores disponibles, no es posible acceder a todos los estados del sistema. Cabe destacar que a mayor similitud entre el modelo y el proceso real, mayor precisión se tendrá en la observación. Además, al no tener siempre muestra disponible, éste no actuará de manera uniforme, dependiendo del número de pérdidas producidas de manera consecutiva. A continuación se expondrán ambos modos de funcionamiento. Para ambos casos será necesario ubicar los parámetros del sistema en base al retardo correspondiente a cada iteración  $d_k^{SC}$  en el canal sensor-controlador.

1. El **primer caso** posible es aquél en el cual **no existe muestra disponible** o ha llegado desordenada, es decir, cuando  $\tilde{y}_k = 0$  (o de manera más correcta  $\tilde{y}_{k-d_k^{SC}} = 0$ ). Por tanto, se realizará la observación solamente en base a modelo. De este modo, la ecuación a emplear es la siguiente

$$\tilde{x}_{k-d_k^{SC}+1} = A\tilde{x}_{k-d_k^{SC}} + Bu_{k-h} \quad (3)$$

siendo  $A$  y  $B$  las matrices discretas del sistema analizado. A pesar de que más adelante se detallará la forma de lidiar con las acciones de control, cabe remarcar que, tanto para este caso como para los siguientes, no se puede emplear la acción de control en el instante actual  $u_k$  para la ecuación del observador. De hacerlo, se incurriría en error. Por ello, se debería emplear la acción ubicada según el retardo existente. Tanto si se consideran retardos en un canal como en ambos, el controlador no tiene certeza de dicho retardo (o retardos) cuando se observan los estados, por lo que se debe calcular mediante una acción  $u_{k-h}$ . El índice  $h$  será detallado en la siguiente subsección, pero a grosso modo se trata de una constante empleada, en este caso, para estimar los retardos presentes en ambos canales, y poder emplear la acción más correcta posible dentro de las almacenadas en el controlador.

2. El **segundo caso** es aquél en el que **existe muestra para el controlador**, es decir cuando  $\tilde{y}_{k-d_k^{SC}} = 1$ . Al poseer mayor cantidad de información, es posible realizar una corrección mediante la medida obtenida y los estados observados en el instante previo. Si tenemos dos muestras válidas consecutivas, la nueva ecuación a emplear queda de la forma

$$\tilde{x}_{k-d_k^{SC}+1} = A\tilde{x}_{k-d_k^{SC}} + Bu_{k-h} + L(\tilde{y}_{k-d_k^{SC}} - C\tilde{x}_{k-d_k^{SC}}) \quad (4)$$

Al tener dicha corrección, el resultado de la observación se hace más preciso. La ganancia del observador  $L$  será elegida de modo que la matriz  $(A - LC)$  sea estable. Dependiendo del caso a estudiar, los polos del observador se posicionarán en unos determinados puntos u otros, siempre procurando que caigan dentro del círculo unidad para el caso discreto, o tengan parte real negativa para el caso continuo.

Por otra parte, si se producen varias pérdidas consecutivas, la expresión para calcular la ganancia del observador será diferente. El contador  $N_k$  influye en este cálculo. La nueva expresión del observador es

$$\tilde{x}_{k-d_k^{SC}+1} = A\tilde{x}_{k-d_k^{SC}} + Bu_{k-h} + \tilde{L}(\tilde{y}_{k-d_k^{SC}} - C\tilde{x}_{k-d_k^{SC}}) \quad (5)$$

y para este caso la ganancia  $\tilde{L}$  se computará tal que

$$A^{N_k} - \tilde{L}CA^{N_k-1} \quad (6)$$

sea estable. Se puede comprobar fácilmente que si tenemos dos medidas disponibles consecutivas,  $N_k = 1$  y la ecuación empleada es (4)

### 3.3 Predictor

El observador introducido en la etapa previa permitiría, en condiciones normales, calcular una acción de control válida para enviar al sistema gracias a un control simple por realimentación del estado, tal que

$$u_k = -Kx_k \quad (7)$$

Sin embargo, al tratarse de un control basado en red, los estados estimados en el instante  $k - d_k^{SC} - d_k^{CA}$  no se corresponden con los reales en el instante  $k$ . Es por ello que se deben determinar los estados  $\bar{x}_{k+h}$ , donde  $h$  se encargue de compensar los retardos existentes en ambos canales: sensor-controlador y controlador-actuador. Así, la predicción será más exacta cuanto más se aproxime  $h$  a la suma de los retardos medios en cada canal, tal como establece la siguiente ecuación

$$h \approx \bar{d}_k^{SC} + \bar{d}_k^{CA} \quad (8)$$

El predictor a utilizar es el empleado en el artículo [2] referenciado en la bibliografía, así como el que se emplea también en el artículo [1]. Dicho predictor trabaja con  $h$  como parámetro fijo. Así, y tal como ya ha sido comentado, si se cumple (8), se obtendrán mejores prestaciones. Por el contrario, si  $h$  difiere en gran medida de  $\bar{d}_k^{SC} + \bar{d}_k^{CA}$ , el comportamiento puede verse empeorado. La ecuación que determina la predicción del estado viene dada por

$$\bar{x}_{k+h} = A^h \tilde{x}_{k-d_k^{SC}} + A^{h-1} B u_{k-h} + \dots + B u_{k-1} = A^h \tilde{x}_{k-d_k^{SC}} + \sum_{i=1}^h A^{i-1} B u_{k-i} \quad (9)$$

Tras obtener la predicción, se hace posible calcular una acción de control mediante realimentación del estado. Dicha etapa será desarrollada en la siguiente subsección.

### 3.4 Ley de control

Una vez obtenida una predicción para los estados, el siguiente paso lógico es computar una acción de control válida. Tal como ya ha sido establecido, se empleará una ley simple de realimentación del estado, dada por

$$u_{k-d_k^{SC}-d_k^{CA}} = -K\bar{x}_{k+h} \quad (10)$$

La ganancia  $K$  vendrá dada por la posición deseada de los polos en bucle cerrado. Al igual que en la etapa del observador, dicha ganancia se seleccionará de modo que aquéllos se encuentren dentro del círculo unidad para el caso discreto, o tengan parte real negativa para el caso continuo.

Por otra parte, la ubicación de la acción mediante el índice  $k - d_k^{SC} - d_k^{CA}$  se debe a que, en el momento de su aplicación, están presentes los retardos de ambos canales. Es decir, se computa cuando solo existe  $d_k^{SC}$ , pero, al ser enviada por el canal controlador-actuador, se aplica cuando los dos retrasos han aparecido, incluyendo  $d_k^{CA}$ . Gracias al predictor, el modo de compensarlos ya está previsto. Sin embargo, éste no es capaz de evitar las pérdidas. Así, se propone una estrategia de control basada en paquetes, la cual será detallada a continuación. Por simplicidad, además de porque gracias a las estrategias aplicadas ya se han corregido ambos retardos, de ahora en adelante se notará como  $u_k$  a la acción computada en la ecuación (10), por lo que  $u_k = -K\bar{x}_{k+h}$

#### 3.4.1 Control basado en paquetes

En un sistema de control en red, una pérdida durante el envío de datos hacia el actuador implica no tener acción disponible para aplicar en un periodo de muestreo determinado. Este hecho puede dar lugar a que el sistema a controlar se desestabilice, más aun si se prolonga durante varios periodos al producirse varias pérdidas consecutivas. Es por ello que se propone una estrategia basada en paquetes, por la cual se envía desde el controlador la acción de control a aplicar junto con un paquete o *buffer* de acciones futuras. Dicho *buffer* se almacenará en memoria en la parte local (planta). Así, si se produce pérdida y/o desorden se podrá aplicar una acción, y evitar el funcionamiento del sistema en bucle abierto durante uno o varios periodos de muestreo.

La creación del paquete se realiza de manera análoga a como se crea  $u_k$ . En cada iteración se pasa a través las etapas de observador, predictor y generación de acción de control. Debido a la imposibilidad de tomar salidas del sistema  $y_k$

futuras, la observación se realizará siempre sin la corrección detallada en la ecuación (4), por lo que siempre se llevará a cabo mediante la ecuación (3). Una vez generada, se almacena y se pasa a generar la siguiente. La cantidad de acciones a calcular (longitud del *buffer*) depende del retardo medio y de la cota máxima de pérdidas esperados para la red empleada. Mayores valores para ambas variables producirán una mayor necesidad de generar un paquete con un número superior de elementos. De este modo, y tomando  $j$  como parámetro a iterar, las ecuaciones que definen la generación del *buffer* son las indicadas a continuación:

$$\begin{aligned}\tilde{x}_{k-d_k^{SC}+1+j} &= A\tilde{x}_{k-d_k^{SC}+j} + Bu_{k-h+j} \\ \bar{x}_{k+h+j} &= A^h\tilde{x}_{k-d_k^{SC}+j} + \sum_{i=1}^h A^{i-1}Bu_{k-i+j} \\ u_{k+j} &= -K\bar{x}_{k+h+j}\end{aligned}\quad (11)$$

El número de iteraciones que sean realizadas sobre  $j$  determinará la longitud del *buffer*. El empleo de esta estrategia será mostrado en el siguiente apartado: selector de medidas de la planta.

El funcionamiento general de la parte del controlador puede verse esquematizado en la Figura 5. En ella se puede apreciar la existencia de todas las etapas: selector de medidas, observador no uniforme, predictor y ley de control. La segunda y tercera forman lo que se puede definir como el predictor-observador no uniforme (*NUPO*). A su vez, la ley de control junto con el *NUPO*, ejecutadas de manera iterativa, da lugar a la estrategia anteriormente descrita: el control basado en paquetes. Éste crea el *buffer*, situado a la izquierda de la Figura 5, el cual será enviado al actuador.

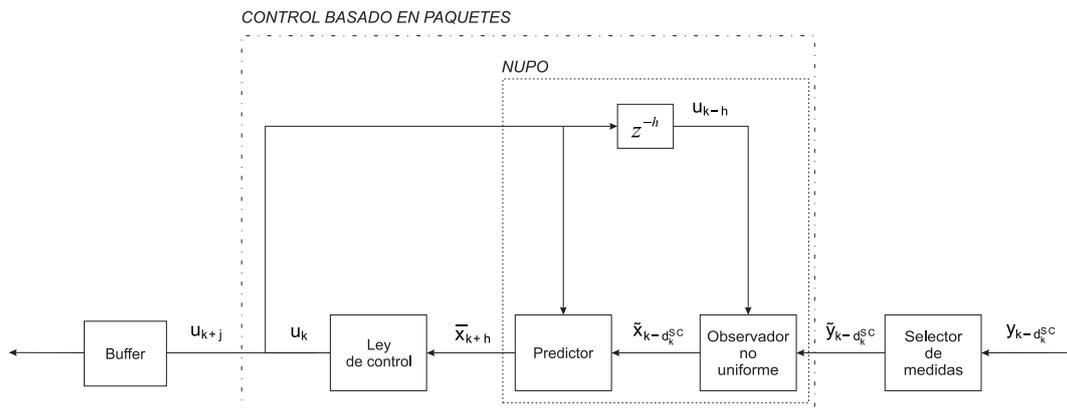


Figura 5: Controlador completo

### 3.5 Selector de medidas. Planta

El selector de medidas de la parte local (planta) tiene un funcionamiento similar al del controlador. Así, también está basado en técnicas de sincronización y time-stamping. La notación está basada en una expresión similar a la ecuación (2), dada por

$$\tilde{u}_{k-d_k^{SC}-d_k^{CA}} = \theta_k u_{k-d_k^{SC}-d_k^{CA}} \quad (12)$$

que también por simplicidad, tal como ha sido puesto de manifiesto en la sección anterior, será expresada de ahora en adelante por la siguiente ecuación:

$$\tilde{u}_k = \theta_k u_k \quad (13)$$

De este modo, si la muestra que llega al Selector de Medidas en el instante  $k$  es más reciente que la del instante  $k - 1$ , se toma como válida y  $\theta_k = 1$ . Por otra parte, si la muestra no es más reciente y, por tanto, ha llegado desordenada, se toma  $\theta_k = 0$ , tratándose como un paquete perdido. Dicha comparación se realizará solo para el primer elemento del *buffer* recibido. Cada muestra perdida/desordenada aumenta un contador  $M_k$  que influirá en el empleo del *buffer* enviado desde el controlador al Selector de Medidas.

Gracias a esta estrategia, si se producen  $M_k$  pérdidas (o llegadas de paquetes desordenados) consecutivas, el actuador no quedará sin acción que aplicar, pudiendo acceder al elemento correspondiente del *buffer* en cada iteración. La aplicación de las acciones correctas será más precisa si se cumplen dos condiciones.

- La primera, la similitud del modelo con el proceso real. A mayor similitud, mejor funcionamiento de las acciones del paquete, ya que para el cómputo de éstas ha pasado por varias fases en las que se ha empleado dicho modelo para la totalidad de los cálculos.
- La segunda, el número de pérdidas consecutivas en el canal CA. A mayor valor de  $M_k$ , menos precisas serán las predicciones. Esto se debe a que, tal como ha sido puesto de relieve anteriormente, la corrección con medida para el observador es inexistente, empleándose la ecuación (3) en vez de la ecuación (4), y perdiéndose de este modo precisión. Este hecho es inevitable, ya que no tenemos información sobre muestras de iteraciones futuras.

El funcionamiento del selector de medidas correspondiente a la planta se muestra esquematizado en la Figura 6. Como se puede apreciar, los paquetes desordenados se tratan como perdidos, y en ambos casos se pasa a emplear una acción del *buffer*. Por ejemplo, en el instante  $k + 2$  se accede al primer elemento ( $M_{k+2} = 1$ ) del *buffer* recibido en el instante  $k + 1$ . Como en el siguiente instante tampoco existe acción directa recibida, se debe emplear también una del *buffer*, pero en este caso el siguiente elemento,  $M_{k+3} = 2$ .

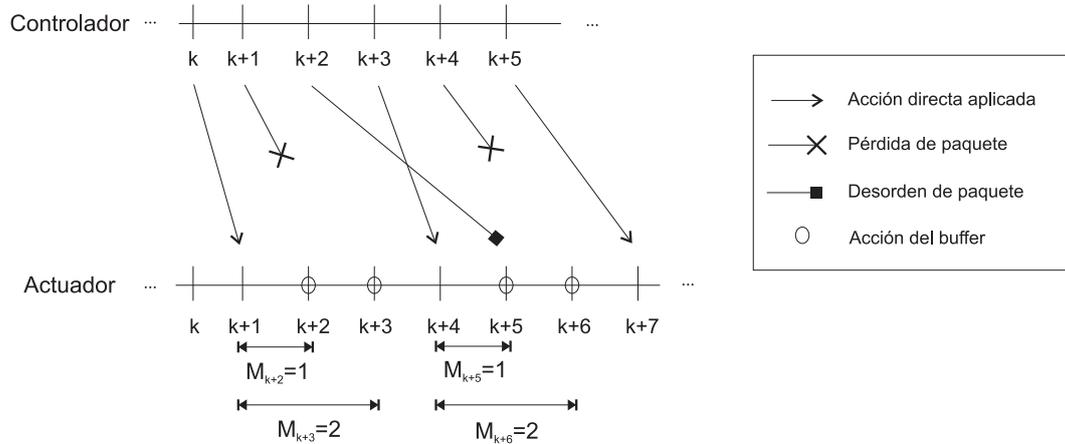


Figura 6: Selector de medidas. Planta. Modo de operación

## 4 Simulación

Una vez definida la estrategia a emplear, el siguiente paso consiste en llevar a cabo la simulación de la solución de control. En primer lugar se repasará el estudio previo llevado a cabo en el artículo [1], en el cual solamente se contempla el enlace *SC*. En segundo lugar, se añadirá el enlace *CA*, observándose los problemas que se dan cuando hay pérdidas/desorden y se inyecta la última acción disponible. Posteriormente, se añadirá la estrategia de control basada en paquetes con el fin de mejorar las prestaciones respecto al caso anterior. Por último, se comparará el estudio desarrollado en [1] con el propuesto en el presente documento.

### 4.1 Características del sistema

Para todos los casos anteriormente nombrados se estudiarán dos aproximaciones: cuando existe retardo fijo y cuando existe retardo variable, apareciendo en éste último el fenómeno del desorden de paquetes. Para los apartados donde existe retardo variable se asumirá que el sistema de control está implementado a través de una red IP. Para estos casos es común aproximar la distribución de retardos como una del tipo exponencial, dada por la expresión

$$P[d_k] = \begin{cases} \frac{1}{\phi} e^{-\frac{(d_k-\eta)}{\phi}} & d_k \geq \eta \\ 0 & d_k < \eta \end{cases} \quad (14)$$

Como se puede observar, cuando el retardo  $d_k$  (en cualquiera de los dos canales) supera un determinado umbral  $\eta$ , se aplica la función de probabilidad. Sino, la probabilidad de que se dé dicho retardo es nula. Esto indica la existencia de una cota mínima de retardo. La mayoría de retardos se distribuyen alrededor de un valor medio. La existencia de retardos más elevados se hace más improbable conforme más se alejan de dicho valor medio.

Para el estudio de simulación y posterior implementación en el laboratorio se podría elegir cualquier magnitud del vehículo RPAS. Por simplicidad, y también por limitaciones en la plataforma del laboratorio, no se realizará un estudio sobre la posición de la aeronave. Por el contrario, se llevará a cabo sobre uno de los ángulos de orientación. Gracias a los datos proporcionados se puede afirmar que la planta del vehículo puede ser representada como un doble integrador para cualquiera de los ángulos anteriormente nombrados. Sin embargo, existe una variación en la magnitud de la ganancia dependiendo de si escogemos *roll* (alabeo,

$\phi$ ), *pitch* (cabeceo,  $\theta$ ) o *yaw* (guiñada,  $\psi$ ). Dicho esto, la planta en continuo puede ser representada por la siguiente ecuación

$$G(s) = \frac{1}{s^2} \quad (15)$$

la cual en espacio de estados o representación interna viene dada por la forma continua

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \\ y(t) &= [1 \ 0] x(t) \end{aligned} \quad (16)$$

Sin embargo, para posterior tratamiento se debe obtener la forma discreta a partir de la ecuación (16). Ésta, respresentada para un periodo de muestreo constante  $T = 50 \text{ ms}$ , viene dada por

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & 0 \\ 0.05 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0.05 \\ 0.00125 \end{bmatrix} u_k \\ y_k &= [1 \ 0] x_k \end{aligned} \quad (17)$$

Con el fin de obtener unas determinadas prestaciones para el sistema, se controlará el sistema mediante realimentación del estado, tal como ya ha sido puesto de manifiesto. Tanto para simulación como para prueba experimental, se ha determinado el vector de ganancias  $K$  tal que los polos en bucle cerrado se posicionen en los puntos  $s = -2$  y  $s = -0.5$ . Éstos nos definen una ganancia aproximada de  $K = [2.374 \ 0.940]$ , la cual será aplicada en cada periodo de muestreo con el fin de calcular las acciones de control para el buffer.

Por otra parte, la ganancia del observador será, en general, diferente en cada iteración, tal como ya ha sido remarcado en anteriores apartados. Sin embargo, la posición de dichos polos sí será constante durante todo el tiempo: tomarán los valores  $z = 0.19$  y  $z = 0.15$ , en términos discretos. Estos valores han sido seleccionados de manera empírica, ya que incluso pequeños cambios en ellos producían grandes cambios en la respuesta del sistema al tratarse de un sistema tan complejo, en el que intervienen tantos factores.

Por último, las condiciones iniciales se fijarán en  $x_i = [0.5 \ -0.5]$  para todas las simulaciones de ahora en adelante.

## 4.2 Estudios previos. Canal sensor - controlador

El primer paso consiste en poner de manifiesto los resultados de partida, obtenidos en [1]. En dicho estudio solo se consideran problemas en el canal  $SC$ . Por tanto, se admite que en el canal  $CA$  no existen problemas de retardos ni pérdidas. Esta simplificación ayuda a facilitar el estudio y a mejorar las prestaciones respecto al caso menos favorable, en el que tenemos retardos y pérdidas en ambos canales. Para esta sección se emplearán indistintamente los términos  $d_k$  y  $d_k^{SC}$ , ya que hacen referencia a la misma variable.

### 4.2.1 Retardo fijo

El primer punto de partida es la simulación en la cual solo existe retardo fijo, esto es, considerando que cualquier muestra enviada del sensor al controlador se retrasa siempre en la misma medida.

Un caso interesante a estudiar es aquél en el que vamos aumentando el retardo  $d_k^{SC}$ . Conforme éste va aumentando, el comportamiento del sistema va empeorando. Además, fijamos  $h = 0.1$  s como parámetro para el predictor, y una cota de pérdidas fija  $N_k = 4$ . Cabe destacar que aunque el valor de  $h$  sea decimal, para los cálculos se adimensionalizará respecto al periodo de muestreo  $T$ . Con estos datos, se obtuvieron los resultados mostrados en la Figura 7.

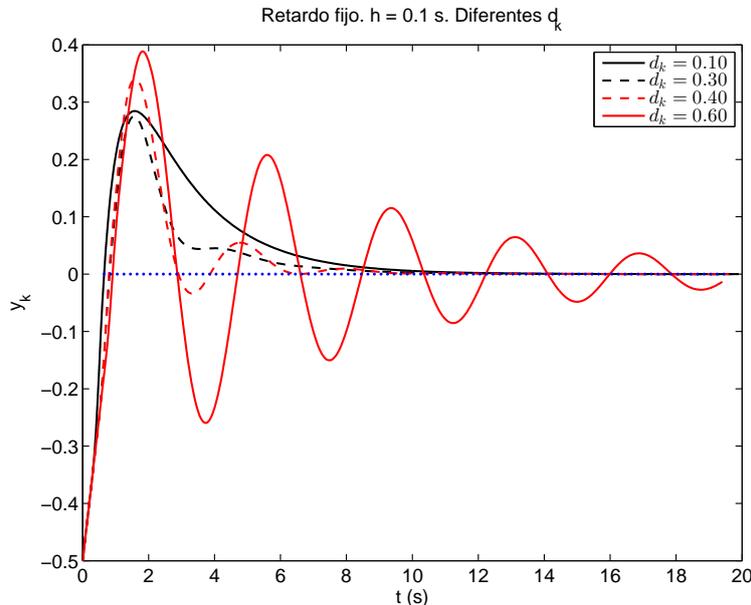


Figura 7: Retardo fijo. Diferentes  $d_k$

El patrón de puntos azules sobre el eje  $X$  indica cuándo se tiene muestra disponible. Como se puede apreciar, a mayor retardo peor comportamiento en general. Cabe destacar que el estudio de *LMI (Linear Matrix Inequalities)* llevado a cabo en [1] predice que para retardos comprendidos entre  $d_1 = 0.1$  s y  $d_2 = 0.3$  s la respuesta del sistema se va a mantener dentro del criterio de estabilidad deseado. Es por esto que las respuestas procedentes de ambos retardos tienen un comportamiento similar. A partir de  $d_2$ , se hace más difícil asegurar dicha estabilidad. Por otra parte, la respuesta óptima se da cuando el parámetro  $h$  del predictor coincide con el retardo  $d_k$  a cubrir, lo cual confirma la validez del predictor empleado. La tabla mostrada a continuación indica los diferentes parámetros procedentes de las repuestas del sistema.

$d_k$ [s]	$t_e(98\%)$ [s]	$\delta$ [%]
0.1	7.40	28.4
0.3	5.75	27.6
0.4	5.70	34.0
0.6	-	38.9

Desde otro punto de vista, es posible obtener y comparar las respuestas para un retardo fijo  $d_k^{SC} = 0.1$  s y observar qué ocurre mientras  $h$  se aleja de dicho valor. Este análisis lo tenemos representado en la Figura 8. En ella vemos cómo se deteriora la respuesta  $y_k$  conforme el parámetro del predictor va cubriendo un mayor rango del que debe para el valor de retardo que tenemos.

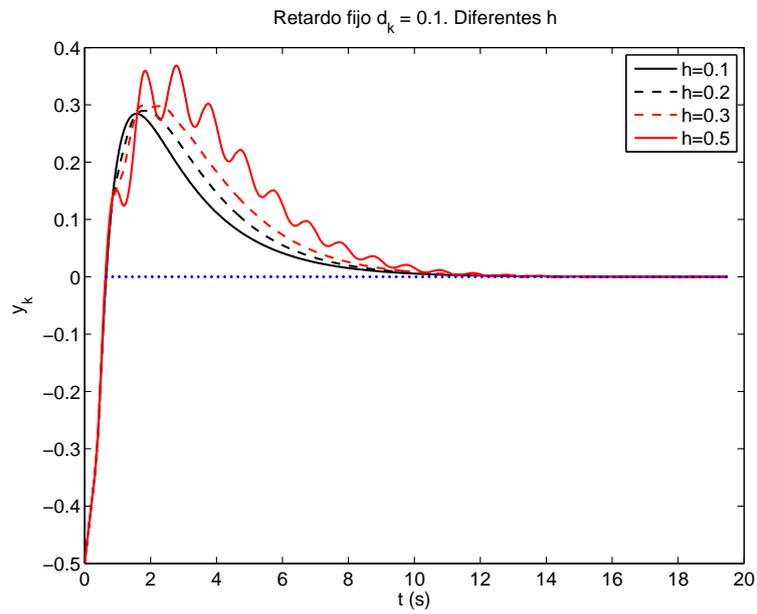


Figura 8: Retardo fijo. Diferentes  $h$

Como en el caso anterior, la siguiente tabla condensa los datos obtenidos en la Figura 8

$h$ [s]	$t_e[98\%]$ (s)	$\delta$ [%]
0.1	7.45	28.4
0.2	7.95	29.0
0.3	8.45	30.0
0.5	9.80	36.9

### 4.2.2 Retardo variable

El siguiente caso desarrollado por el artículo es el del retardo variable. En él aparecen tanto los problemas de pérdidas de paquetes como los de desorden. En primer lugar, y mediante la distribución (14), se definió la serie de retardos aleatorios mostrada a continuación, en la Figura 9

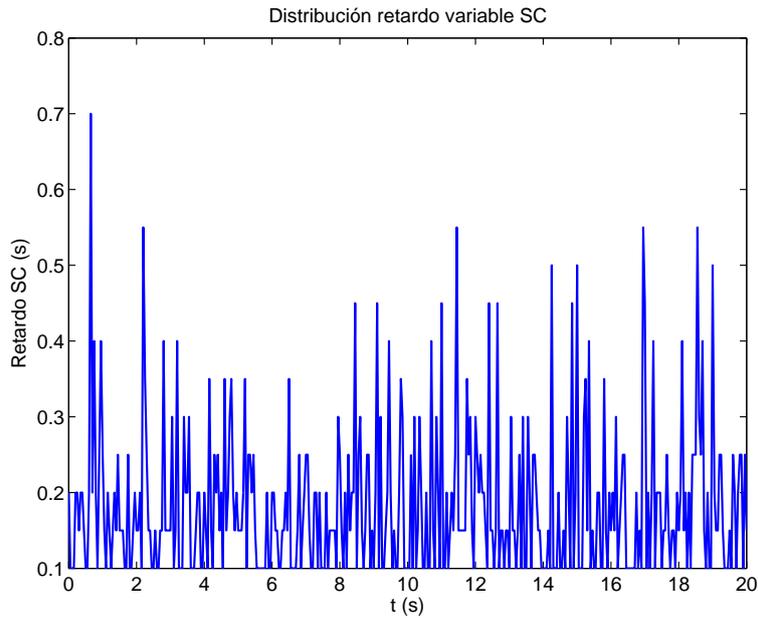


Figura 9: Distribución retardos variables. Canal SC

A partir de ellos, y siguiendo la estrategia ya conocida, se obtuvieron resultados mostrados en la Figura 10. Al igual que en el caso anterior, se ha fijado una cota de pérdidas  $N_k = 4$ , pero en este caso el patrón que se muestra sobre el eje es no-uniforme, ya que también interviene el problema del desorden.

Tal como podemos observar, el comportamiento empeora conforme aumenta  $h$ , hasta llegar a hacerse prácticamente inestable con  $h = 0.6 s$ . El sistema posee menores valores de sobreoscilación y tiempo de establecimiento para  $h = 0.1 s$ , es decir, cuando dicho parámetro es más cercano a la media de retardos  $\bar{d}_k^{SC} = 0.18 s$ . La tabla mostrada a continuación indica los resultados obtenidos

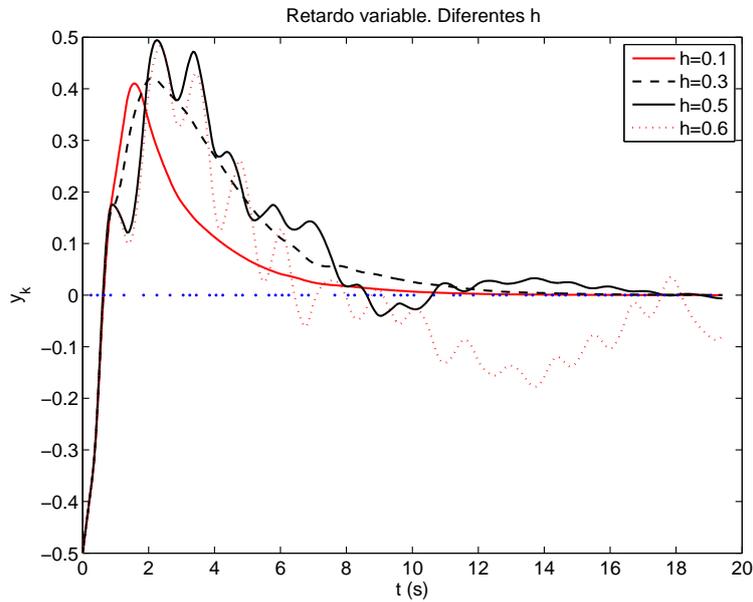


Figura 10: Retardo variable. Diferentes h

h [s]	$t_e(98\%)$ [s]	$\delta$ [%]
0.1	7.45	41.0
0.3	10.65	42.1
0.5	15.05	49.5
0.6	-	48.3

A partir de estos resultados, se pudo dar por válida la estrategia presentada en [1]. El siguiente paso consiste en aplicar la nueva propuesta desarrollada en el presente documento, la cual supone problemas en ambos canales y el *buffer* de acciones de control para solventarlos.

### 4.3 Estrategia propuesta. Buffer de acciones de control

Tal como ya ha sido comentado, en el caso que nos concierne aparecen problemas tanto en el canal  $SC$  como en el  $CA$ . En primer lugar se estudiará el caso del retardo fijo, para después pasar al de retardo variable. En ambos se analizará el uso del *buffer* frente a otras estrategias empleadas cuando no se tiene acción de control disponible. Además, se observará otros casos interesantes de estudio.

#### 4.3.1 Retardo fijo

El primer ejemplo a mostrar es el más notable en cuanto a este documento, es decir, la comparación entre emplear el *buffer* o no emplearlo. No emplearlo implica emplear una estrategia más simple: por ejemplo aplicar la acción de control más reciente almacenada en el actuador, esto es, la última que no se ha perdido. Por similitud con la subsección anterior, se decide conservar el patrón de pérdidas  $N_k = 4$ , y emplearlo también para el canal  $CA$ , por lo que  $M_k = 4$ . Asimismo, se simula un caso más desfavorable todavía, en el que el retardo total es  $d = 0.6 s$ , distribuido por igual en ambos canales:  $d_k^{SC} = d_k^{CA} = 0.3 s$ . Además, se establece el parámetro del predictor tal que, a priori, no cubra dichos retrasos,  $h = 0.2 s$

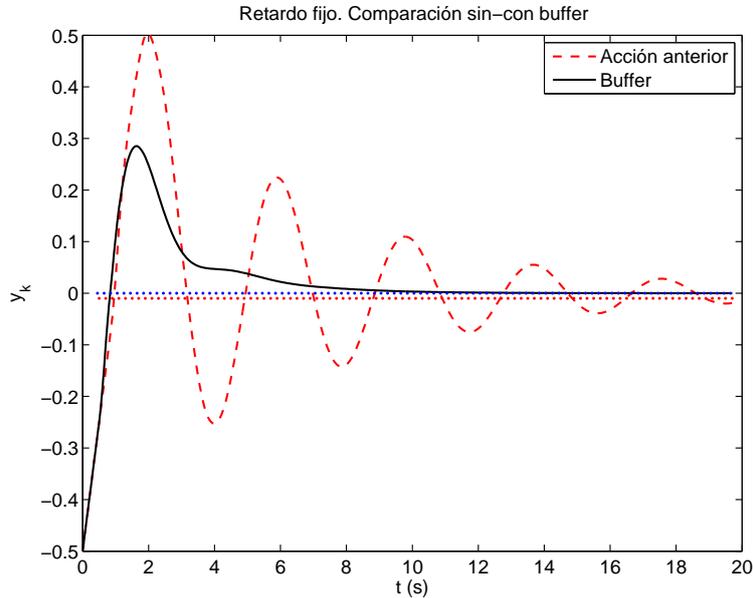


Figura 11: Comparación empleo del buffer. Retardo fijo

Cabe remarcar que los puntos rojos cercanos al eje  $X$  indican la presencia de acción de control directa disponible. Observamos una misma distribución de retardo que para el canal  $SC$ , pero en este caso los puntos de recepción se encuentran desplazados ligeramente en el tiempo debido al propio retardo en  $CA$ .

Como se puede observar en la Figura 11, la curva discontinua no llega siquiera a establecerse en el tiempo de simulación, mientras que empleando el paquete de acciones se establece alrededor de  $t = 5.9$  s. Además, la sobreoscilación para el primer caso alcanza el 50%, mientras que en el segundo se queda alrededor del 28%, lo que supone una mejora del 44%. La mejora respecto al empleo de la última acción almacenada es clara, por lo que se puede dar por válida la estrategia propuesta.

El siguiente paso consiste en observar qué ocurre al variar los diferentes parámetros del sistema. En la Figura 12 apreciamos la aplicación del paquete de acciones para un sistema  $NCS$  en el que los retardos existentes son  $d_k^{SC} = d_k^{CA} = 0.1$  s. Así, y tal como se observó en [1], el comportamiento del vehículo se ve empeorado conforme  $h$  se aleja del retardo total.

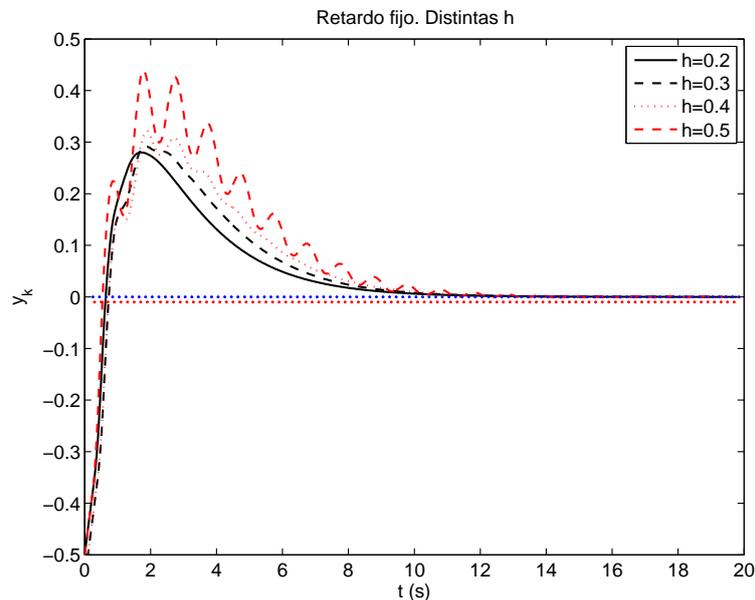


Figura 12: Empleo del buffer. Diferentes  $h$ . Retardo fijo

En la siguiente tabla vemos reflejados los resultados obtenidos

$h$ [s]	$t_e(98\%)$ [s]	$\delta$ [%]
0.2	7.70	28
0.3	8.30	29
0.4	8.75	32
0.5	9.95	44

Cuando  $h$  coincide con el retardo total, el sistema tiene los mejores valores de tiempo de establecimiento  $t_e$  y sobreoscilación  $\delta$ . Conforme se incrementa  $h$ , estos aumentan haciendo incluso que se produzcan mayor cantidad de oscilaciones y se comporte de una manera menos fluida, ya que el predictor es incapaz de actuar de manera correcta.

### 4.3.2 Retardo variable

Al igual que en apartados anteriores, ahora se procede a la simulación del caso de retardo variable, con los problemas que eso conlleva. El paquete de retardos variables también será definido mediante las ecuaciones dadas por (14), estableciendo unos límites no muy elevados. Así, las distribuciones aleatorias aplicadas son las mostradas en la Figura 13.

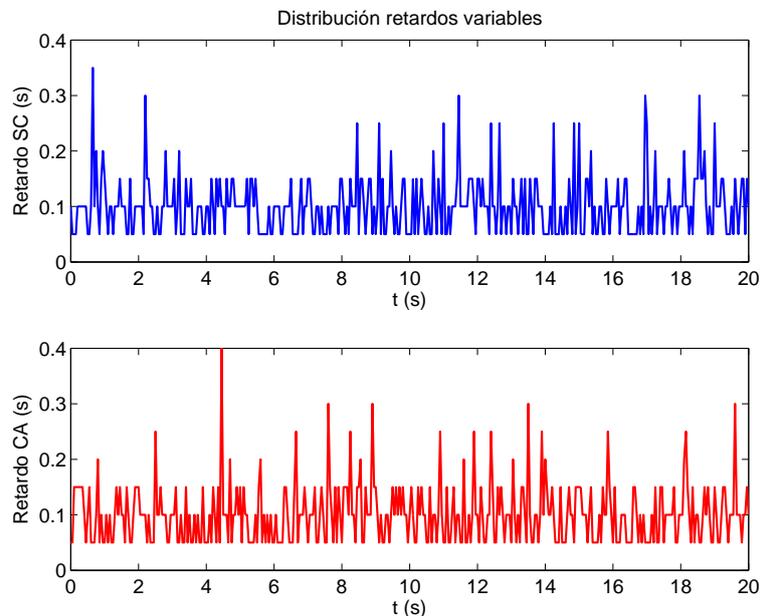


Figura 13: Distribución retardos variables SC y CA

La media de ambos es aproximadamente  $\bar{d}_k^{SC} = \bar{d}_k^{CA} \approx 0.1 s$ , por lo que el parámetro del predictor que en principio debe cubrir mejor ambos retardos es  $h = 0.2 s$ . Aplicándolo al sistema, así como unos patrones de pérdidas constantes  $N_k = M_k = 4$  obtenemos la Figura 14.

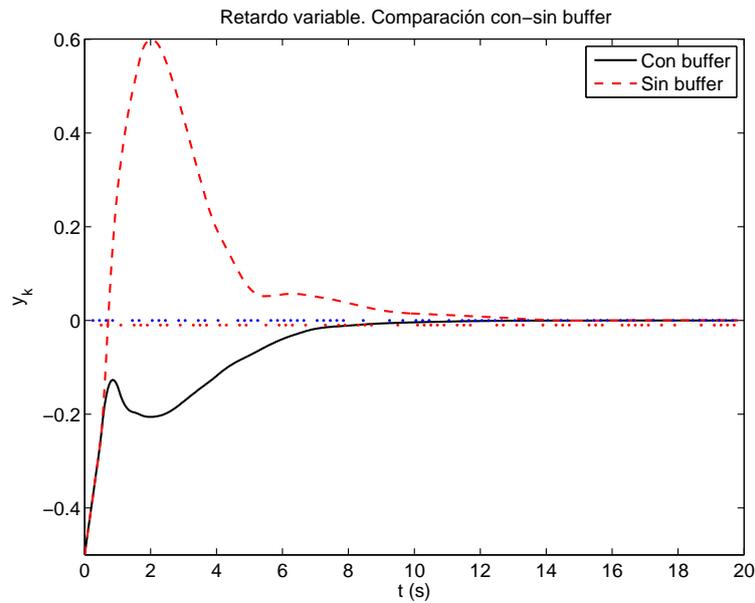


Figura 14: Comparación empleo del buffer. Retardo variable

Por una parte, los patrones sobre el eje de abscisas muestran la irregularidad en cuanto a recepción de muestras. Por otra parte, se puede apreciar una mejoría de prestaciones al emplear el *buffer* frente a no usarlo: los tiempos de establecimiento son  $t_e(98\%) = 6.85 s$  y  $t_e(98\%) = 9.15 s$ , respectivamente, lo cual supone una mejora de más del 25%. Además, en el primer caso se consigue amortiguar la respuesta, lo que puede ser crucial en un sistema como el vehículo *RPAS*; mientras que en el segundo se llega a una sobreoscilación  $\delta = 60\%$

Por otra parte, también es posible observar cómo se comporta el buffer con la variación de  $h$ , lo cual queda representado a continuación. Cabe destacar que el patrón de retardos es el mismo empleado en la Figura 14.

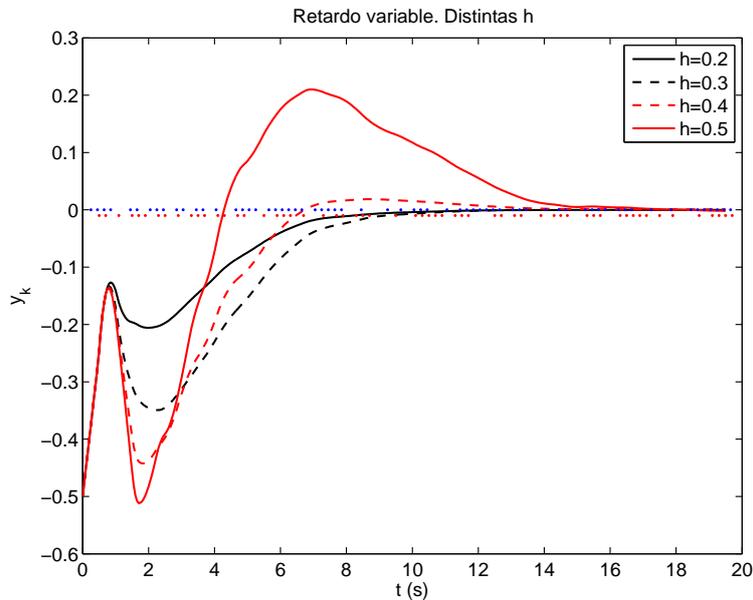


Figura 15: Empleo del buffer. Diferentes  $h$ . Retardo variable

En general, y tal como vimos para retardo fijo, a mayor  $h$  se tiene un tiempo de establecimiento y sobreoscilación superiores. Los resultados quedan reflejados en la siguiente tabla.

$h$ [s]	$t_e(98\%)$ [s]	$\delta$ [%]
0.2	6.85	-
0.3	8.20	-
0.4	6.20	2
0.5	13.35	21

## 4.4 Comparación

Como último apunte para esta sección, se procederá a comparar el estudio llevado a cabo en [1] con el propuesto en este documento. En general, se deberían obtener resultados similares cuando existen problemas en un canal solamente (*SC*) y cuando los hay en ambos (*SC* y *CA*), empleando la técnica de control basada en paquetes. Durante toda la comparación se mantendrá unos índices de pérdidas constantes e iguales a los apartados anteriores, esto es,  $N_k = M_k = 4$

En primer lugar se desea comparar el caso del **retardo fijo**. Así, se deben obtener resultados similares para el caso más favorable de la Figura 7 ( $d_k^{SC} = 0.1 s$  y  $h = 0.1 s$ ) frente a el empleo del *buffer* cuando el retardo total tiene la misma magnitud que en el caso anterior (repartidos de igual manera en ambos canales,  $d_k^{SC} = d_k^{CA} = 0.05 s$ ) y parámetro del predictor  $h = 0.1 s$ . Con estos supuestos, obtenemos lo representado en la Figura 16.

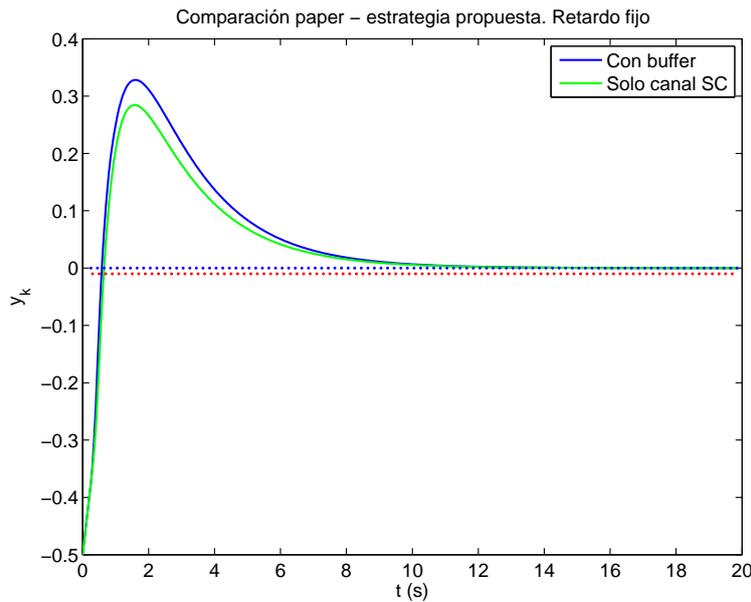


Figura 16: Comparación paper - buffer. Retardo fijo

Los resultados obtenidos son muy similares. Mientras que con solo un canal problemático se obtiene un  $t_e(98\%) = 7.45 s$  y  $\delta = 28\%$ , con la presencia de ambos canales y el sistema propuesto se tienen  $t_e(98\%) = 7.8 s$  y  $\delta = 32\%$ . Los tiempos de establecimiento son muy similares, así como los valores de sobreoscilación, los cuales solo difieren en un 4%

En segundo lugar, tenemos el caso del **retardo variable**. Éste es más complicado de comparar, ya que se puede ver desde dos puntos de vista diferentes:

1. Aplicar el retardo original (Figura 9) escalado a la mitad para ambos canales. Así, el retardo medio se mantendría igual que en la Figura 10, por lo que se podría aceptar como una comparación válida. Sin embargo, haciendo esto no se consigue un mismo patrón de recepción de muestras, debido al nuevo desorden de datos. Es decir, los puntos azules y rojos sobre el eje no coincidirían para ambas curvas. Esto se debe a que al escalar el patrón se obtienen retardos menores para cada iteración.
2. Aplicar el retardo original de la Figura 9 para ambos canales. Así, sí que se aplica el mismo patrón de recepción de muestras que el original, pero el retardo medio (y el presente en cada iteración) se duplica, por lo que el parámetro  $h$  también debe ser modificado, y el comportamiento final tampoco será el mismo.

Dicho esto, queda claro que no se puede realizar una comparación tan precisa como la llevada a cabo para retardo fijo. Por ello, realizaremos ambas comparaciones para ver si, dentro de unos límites, el sistema tiene un comportamiento similar. Dichas comparaciones quedan reflejadas en la Figura 17.

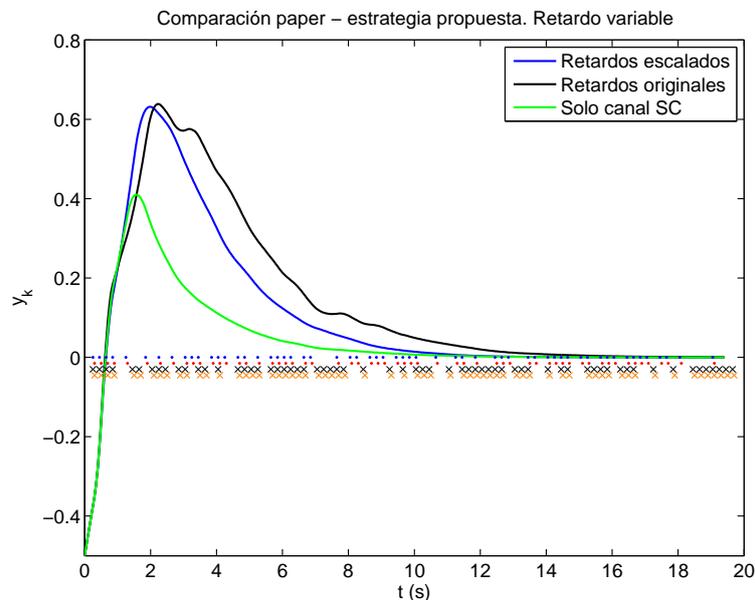


Figura 17: Comparación paper - buffer. Retardo variable

Los puntos azules y rojos representan los instantes de recepción de muestras en controlador y actuador, respectivamente, para el caso de los retardos originales. Los retardos escalados, a su vez, se encuentran representados por las cruces. Las que poseen color negro corresponden a los instantes de recepción en controlador, mientras que las naranjas corresponden al lado del actuador. La Figura 18 muestra dichas distribuciones con un mayor detalle.

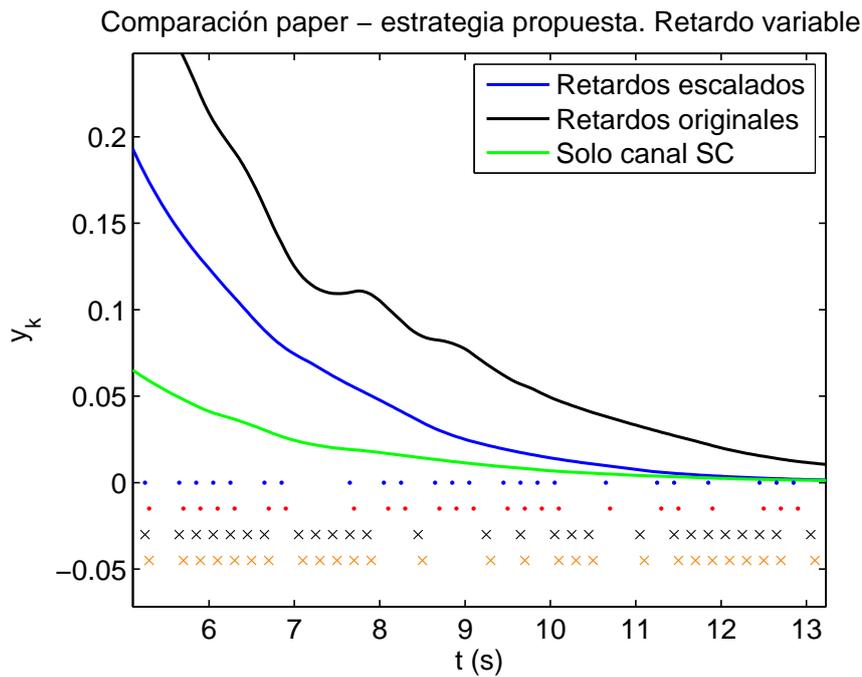


Figura 18: Comparación paper - buffer. Retardo variable. Detalle

Caso	$t_e(98\%)$ [s]	$\delta$ [%]
Escalados	9.40	63.2
Originales	12.00	63.9
Solo canal SC	7.45	41.0

Como se puede apreciar, los  $t_e$  difieren notablemente de un caso a otro. Existe una diferencia del 20.7% entre el caso de los retardos escalados y el caso del canal SC, mientras que éste difiere un 37.9% del  $t_e$  obtenido con el *buffer* y los retardos originales. Por otro lado, las sobreoscilaciones son similares con el empleo del *buffer*, pero hay una diferencia aproximada del 22% respecto del caso expuesto en [1].

Por todo esto, y tal como ya ha sido comentado, es complicado establecer una comparación válida para el caso del retardo variable. Las condiciones no pueden ser las mismas que en [1] en ningún caso, y al tratarse de un caso de estudio en el que intervienen tantos factores, es usual que un pequeño cambio afecte en gran medida al comportamiento. Sin embargo, se puede afirmar que la evolución del sistema es, dentro de unos límites, similar para los tres casos. Por ello, podemos aceptar la comparación como satisfactoria.

## 5 Aplicación real

La segunda parte de este trabajo consiste en la aplicación de las estrategias desarrolladas en la plataforma real. Para ello se ha empleado gran cantidad de *hardware* y *software* presente en uno de los laboratorios del *ai2*. El vehículo RPAS utilizado es del tipo quadrotor, desarrollado íntegramente por personal del laboratorio, el cual podemos ver en la Figura 19. Las órdenes se envían gracias a una interfaz creada mediante el software libre *Qt Creator*, mientras que el código que corre en el quadrotor se compila en *Code::Blocks*, también de software libre. Todo ello se ejecuta gracias al sistema operativo Ubuntu de Linux. El lenguaje de programación de alto nivel empleado es C++.

Tal como constatamos en la sección anterior, la magnitud a controlar es uno de los tres ángulos de orientación, el *yaw* (guiñada), elegido por ser el menos crítico y el más sencillo para una primera aproximación. Cabe destacar que los diferentes experimentos realizados se han llevado a cabo bajo las mismas condiciones: un escalón de 0 a 20 *deg*, desde condiciones iniciales estables.

La primera parte del trabajo de laboratorio consiste en medir los retardos presentes en la red. Para ello se han empleado diversas metodologías propias de librerías de C++. Sin embargo, una vez hecho esto, se constató que los retardos presentes en la red eran ínfimos en comparación con los vistos en simulación. Tanto que, en ocasiones, éstos eran inferiores al milisegundo. Por tanto, se decidió que la mejor estrategia era simular por código los retardos en la red, así como las pérdidas de paquetes.

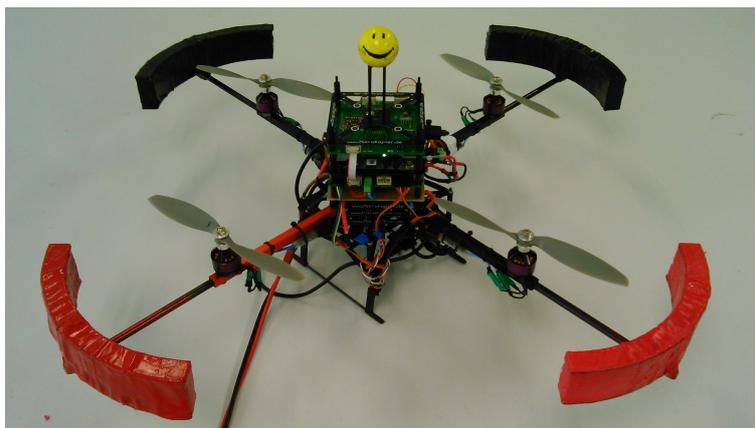


Figura 19: Plataforma de laboratorio

## 5.1 Caracterización del funcionamiento

El primer paso para llevar a cabo la aplicación requiere la caracterización del sistema con datos reales. Así, se realizaron varias pruebas en vuelo para ajustar determinados parámetros del observador, ganancias, etc. Una vez hecho esto, es posible implementar el controlador en la plataforma remota (esto es, en el Qt Creator del PC). Esto conlleva la implementación del observador, ya que solo es posible medir uno de los estados; y del controlador propiamente dicho. En este paso no se implementa todavía la solución de control completa, es decir, sin predictor, buffer, etc. La idea consiste en observar como evoluciona el sistema introduciendo retardos en ambos canales. Por similitud con apartados anteriores, introducimos retardos fijos e iguales en ambos canales, concretamente de  $d_k^{SC} = d_k^{CA} = 25 \text{ ms}$  para un caso, y de  $d_k^{SC} = d_k^{CA} = 50 \text{ ms}$  para el otro. Con ello, obtenemos lo representado en la Figura 20.

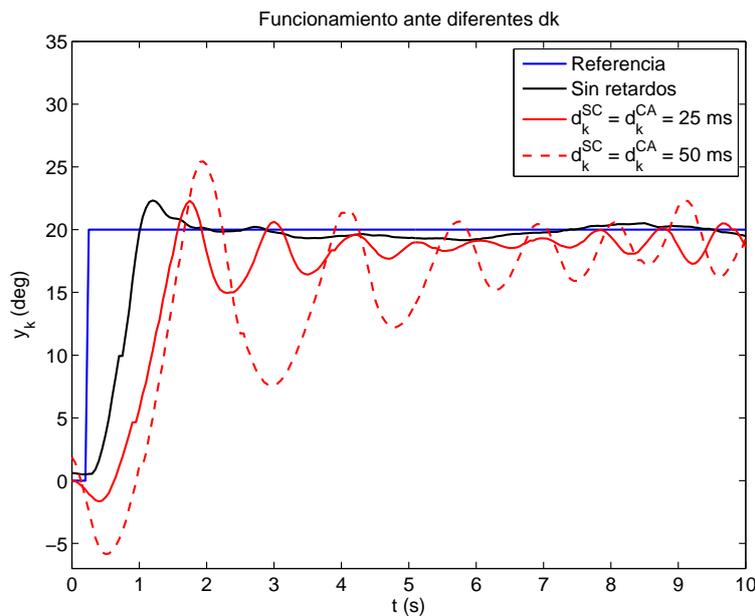


Figura 20: Funcionamiento normal ante distintos retardos

Tal como podemos apreciar, el comportamiento sin ningún tipo de retardo es claramente mejor al resto. La respuesta del sistema empeora conforme aumenta la magnitud del retardo, tendiendo hacia la inestabilidad cuando éste empieza a ser del orden de  $50 \text{ ms}$  en ambos canales. De la misma manera, ante un aumento en la cantidad de pérdidas, cabe esperar también una clara degradación del comportamiento del sistema. Estos problemas se intentarán solventar con las estrategias propuestas en secciones previas.

## 5.2 Validación del predictor

El siguiente caso consiste en validar el empleo del predictor. Para ello realizamos varias pruebas con diferentes aproximaciones. La idea es que el retardo total en ambos canales sea de un periodo de muestreo completo, siendo éste  $T = 50 \text{ ms}$ . Así, es posible introducir un parámetro  $h$  que lo pueda cubrir:  $h = 50 \text{ ms}$ , o en múltiplos enteros de  $T$ ,  $h = 1$ . Por otra parte, también es posible apreciar el comportamiento que tendrá el sistema para diferentes valores de  $h$ .

De este modo, para las distintas Figuras hemos representado, respectivamente,  $d_k^{SC} = 50 \text{ ms}$  para la Figura 21,  $d_k^{CA} = 50 \text{ ms}$  para la Figura 22, y  $d_k^{SC} = d_k^{CA} = 25 \text{ ms}$  para la Figura 23.

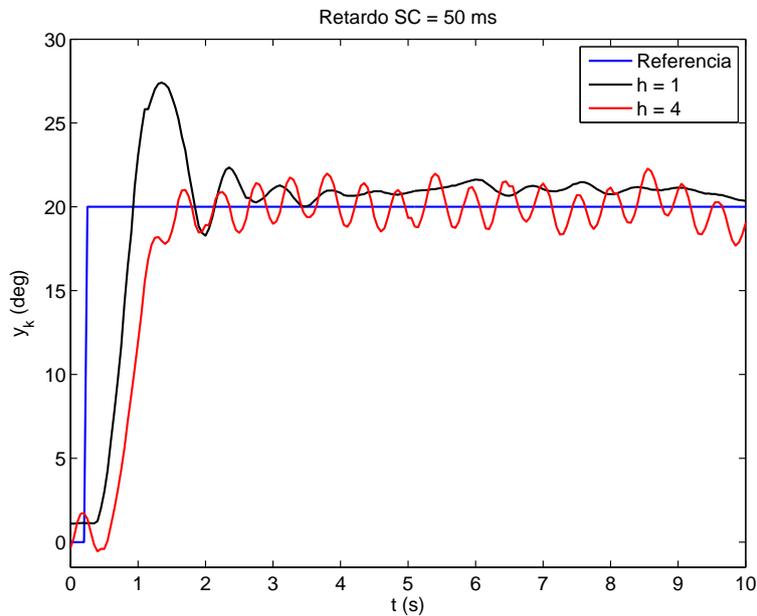


Figura 21: Laboratorio. Retardo SC. Diferentes h

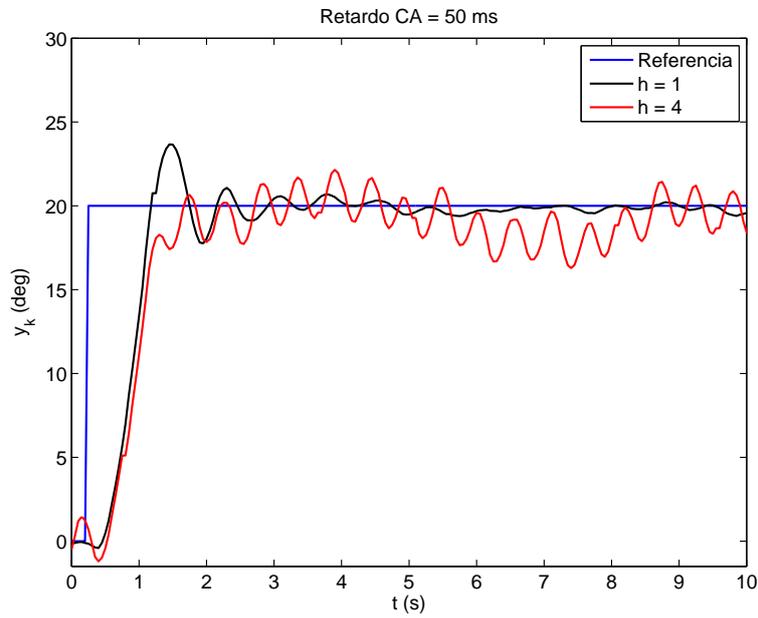


Figura 22: Laboratorio. Retardo CA. Diferentes h

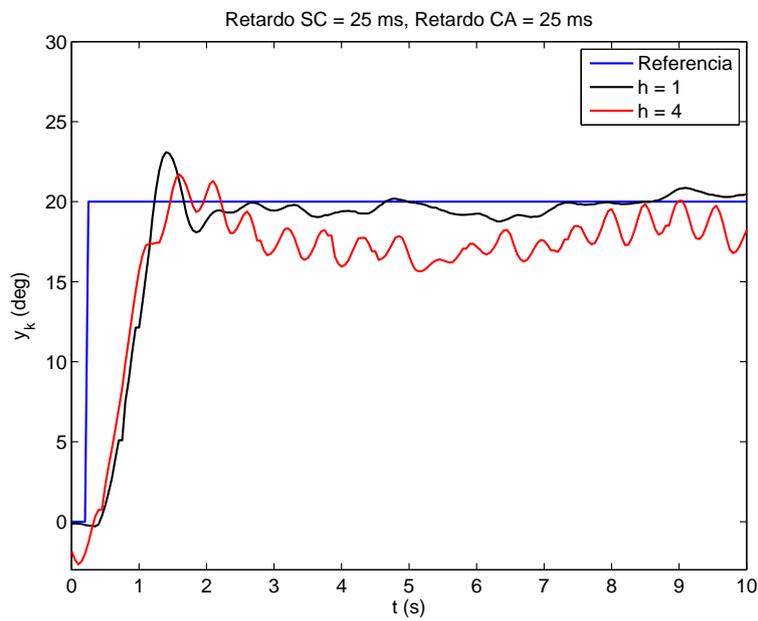


Figura 23: Laboratorio. Retardos SC y CA. Diferentes h

Como podemos apreciar, el sistema posee un mejor comportamiento para  $h = 1$ , parámetro que cubre completamente el retardo total del sistema para los

tres casos. Por el contrario, empleando  $h = 4$  las predicciones se realizan en base a un horizonte demasiado extenso. Esto genera un movimiento oscilatorio continuo que hace que el quadrotor no llegue nunca a estabilizarse. Estas tres Figuras pueden compararse con lo obtenido por simulación, de manera cualitativa más que cuantitativa. Así, la Figura 21 muestra un caso comparable a los estudios previos sobre el NUPO, por lo que podríamos equipararla a la Figura 8, en la cual tenemos también retardo fijo solamente en SC y vamos probando diferentes  $h$ . Como se puede apreciar, en ambos casos el comportamiento del sistema mejora empleando el parámetro adecuado. Por esto, podemos dar por válido el empleo del predictor.

El comportamiento para  $h = 1$  no es igual de preciso para los tres casos: en las Figuras 22 y 23 es parecido, mientras que se aprecia mayor sobreoscilación (y oscilaciones en general) en 21. Esto se debe a que, aunque teóricamente la distribución del retardo sea indiferente al canal en el que se introduce, a nivel de implementación no lo es. Las tareas llevadas a cabo en ambas partes son de distinta naturaleza e importancia, por lo que introducir todo el retardo íntegramente en uno de los canales genera comportamientos diferentes.

Por otra parte, cabe destacar que la evolución del sistema nunca podrá ser tan precisa como en la simulación, debido en parte a ruidos de medida, perturbaciones, etc. Por ello, un error alrededor de  $\pm 2 \text{ deg}$  es aceptable. Además, cada ejecución del algoritmo es diferente a la anterior, y su grado de éxito depende del estado de la red, carga de las CPUs, etc. Esto es válido tanto para éstas como para posteriores Figuras correspondientes a la aplicación real.

### 5.3 Validación del buffer

El siguiente paso consiste en validar la propuesta principal de este documento: el *buffer* de acciones de control. Éste permite que el sistema disponga de acción de control a utilizar cuando se pierde el paquete en el canal CA en una o varias iteraciones consecutivas. Así, introduciendo un índice de cota en CA de 3 (llegan una de cada tres muestras), y un retardo de  $d_k^{SC} = 50 \text{ ms}$ , obtenemos lo reflejado en la Figura 24.

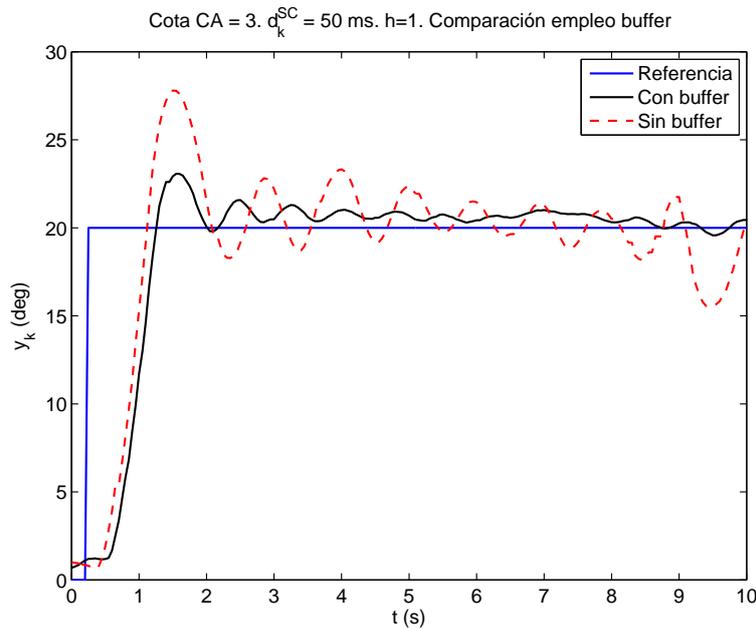


Figura 24: Laboratorio. Empleo buffer. Retardo fijo SC

Como se puede observar, el comportamiento sin el empleo del *buffer* se encuentra en el límite de la estabilidad, mientras que empleándolo se logra estabilizar el sistema. A este hecho contribuye también lo destacado en el apartado anterior: el uso del predictor con el correcto valor de  $h$ , en este caso  $h = 1$

Por otro lado, también es posible repartir el retardo total de la red en ambos canales, para simular un entorno más real. Este hecho queda representado en la Figura 25. En este caso se ha tenido que disminuir la cota CA, ya que el hecho de repartir los retardos en los canales daba más problemas. En este caso, también tenemos  $h = 1$  al ser el retardo total de  $50 \text{ ms}$

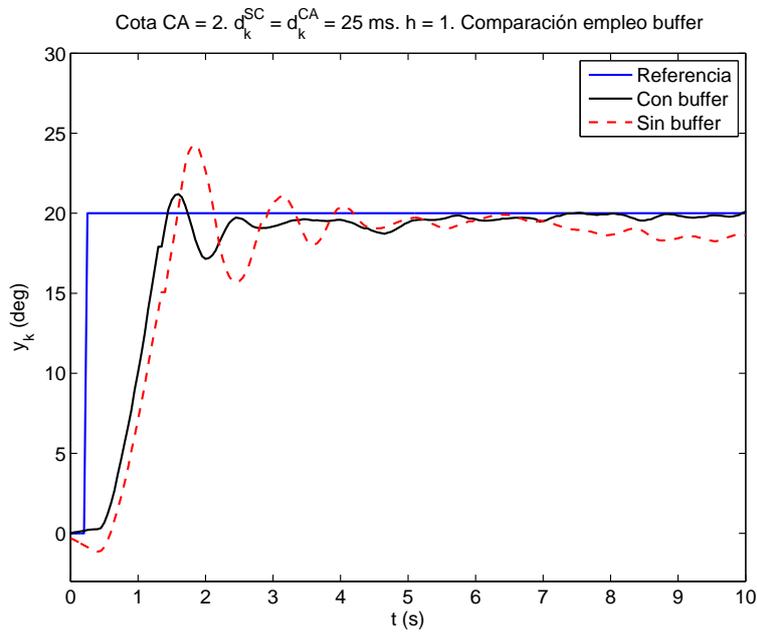


Figura 25: Laboratorio. Empleo buffer. Retardo fijo SC y CA

Las Figuras 24 y 25 (sobre todo ésta última) son equiparables, cualitativamente hablando, al estudio representado por la Figura 11, ya que comparamos el empleo del *buffer* para el caso del retardo fijo. No obstante, en las Figuras del laboratorio usamos la  $h$  correcta en base a los retardos presentes. Aun así, podemos observar que la Figura 25 tiene un comportamiento bastante similar a la 11, en el sentido de que el sistema sobreoscila en mayor medida cuando no empleamos el *buffer*. Por esto, es posible dar por válida la propuesta para retardo fijo.

Por último, queda comprobar el caso final, donde tenemos pérdidas y retardos, siendo éstos últimos variables. La idea es evaluar el caso más parecido a la situación real de una red menos optimizada que la que existe en el laboratorio. Para ello, empleamos condiciones parecidas a las de las anteriores pruebas: cota de pérdidas en CA de 3, y  $h = 1$ , para poder cubrir la media de retardos en ambos canales, que son los mostrados en la Figura 26. Como se puede comprobar, la media de los retardos para ambos canales por separado es de aproximadamente  $d_k^{SC} = d_k^{CA} \approx 25 \text{ ms}$ , de ahí la elección del parámetro del predictor  $h$ .

Con estas distribuciones, y junto al resto de condiciones descritas, obtenemos las curvas presentes en la Figura 27.

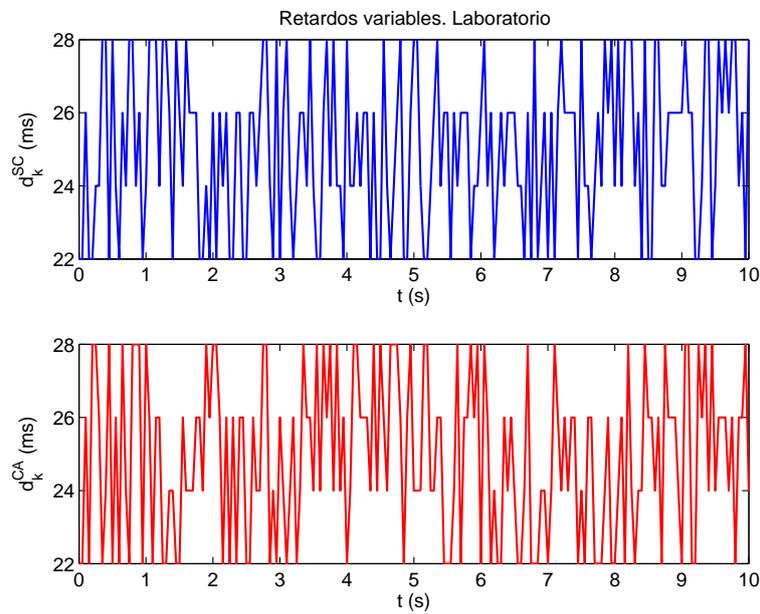


Figura 26: Laboratorio. Distribución retardos variables.

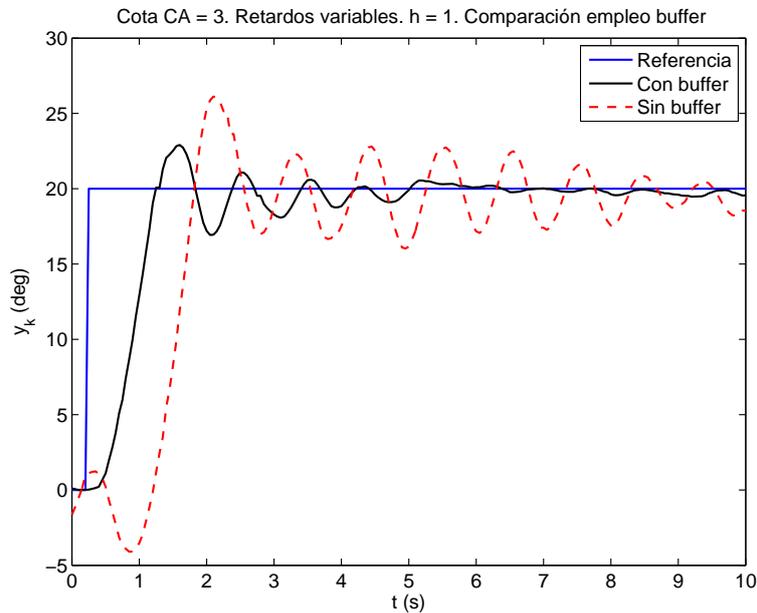


Figura 27: Laboratorio. Empleo buffer. Retardo variable

La respuesta empeora en gran medida sin el empleo del *buffer* de acciones de control. Esto se debe a que una de cada dos acciones no llegan al actuador, por lo que éste aplica la última acción disponible. Por el contrario, empleando el *buffer* se pueden aplicar acciones calculadas en iteraciones anteriores para instantes futuros.

Una vez más, es posible comparar el estudio de laboratorio con el llevado a cabo por simulación. En este caso, la Figura 27 es equiparable a los datos obtenidos y representados en la Figura 14. En ambos casos, el sistema sobreoscila en mayor medida (aunque de diferente forma), y se tiene un  $t_e$  bastante mayor, cuando no se emplea el *buffer* de acciones. Por el contrario, empleándolo se obtienen mejores resultados, alcanzándose antes la estabilidad. Por ello, podemos dar también por válida la estrategia de control propuesta para retardo variable.

## 6 Conclusiones

Tras la realización de esta tesina, es posible realizar varias conclusiones acerca del desarrollo llevado a cabo, así como de los resultados.

Los sistemas de control basados en red o NCS conforman un campo cada vez más extendido en el área del control automático, debido a su utilidad en un mundo cada vez más tecnológico e indiscutiblemente inmerso en las comunicaciones. Es por ello por lo que la investigación acerca de estos temas posee su importancia, y los primeros temas a solventar en esta investigación son los problemas que aparecen. Tal como hemos visto a lo largo del documento, los principales problemas encontrados son los mostrados a continuación:

- **Retardos variantes:** la información no viaja de manera instantánea a través de la red, por lo que se producen retardos desde que se envía una señal hasta que se recibe en la parte contraria. Así, una buena caracterización de la distribución de retardos ayudará al diseño del predictor, el cual trata de cubrir dichos retrasos.
- **Pérdidas:** por la misma razón expuesta en el bloque anterior, no todos los paquetes de datos enviados se reciben en la parte remota, en ocasiones algunos se pierden y no llegan al destino. Para ello, este trabajo propone dos métodos, dependiendo del canal. Para el canal SC, existe un observador no-uniforme, que tiene un comportamiento diferente dependiendo del número de pérdidas consecutivas  $N$ . Para el canal CA, se implementa un buffer, el cual posee información de acciones futuras a aplicar. Éste se envía completo al actuador para que, en caso de pérdida, acceda a la  $M$ -ésima acción de dicha lista.
- **Desorden de datos:** el hecho de tener retardos variantes puede dar lugar a que determinadas muestras lleguen a la parte remota antes que otras, siendo éstas últimas realmente anteriores en el tiempo. Esto se soluciona con un selector de medidas (tanto en la parte del controlador como en la del actuador), con el fin de que descarte medidas antiguas siempre que se posean más recientes.

Así, el presente documento trata de paliar los problemas expuestos exponiendo soluciones que, tanto en simulación como experimentalmente, queda demostrado que cumplen las expectativas. Por tanto, es posible dar por válidas dichas estrategias.

## Referencias

- [1] Cuenca Lacruz AM, García Gil PJ, Albertos Pérez P, Salt Llobregat J. *A Non-Uniform Predictor-Observer for a Networked Control System*. International Journal of Control, Automation, and Systems (2011).
- [2] García Gil PJ, Castillo P, Lozano R, Albertos Pérez P. *Robustness with respect to delay uncertainties of a predictor-observer based discrete-time controller*. Proceedings of the 45th IEEE Conference on Decision & Control (2006).
- [3] Rodríguez Ramírez D, Bordóns Alba C. *Análisis y Control de Sistemas en Espacio de Estado*. Depto. de Ingeniería de Sistemas y Automática - Universidad de Sevilla (2005).
- [4] Gupta RA, Chow M-Y. *Overview of Networked Control Systems*. North Carolina State University, Raleigh (2008).
- [5] Casanova Calvo V, Salt Llobregat J. *Sistemas de Control basados en red. Modelado y Diseño de estructuras de control* (Tesis doctoral). Depto. de Ingeniería de Sistemas y Automática - Universidad Politécnica de Valencia (2005).
- [6] Salt J, Casanova V, Cuenca A, Pizá R. *Sistemas de Control Basados en Red Modelado y Diseño de Estructuras de Control*. Revista Iberoamericana de Automática e Informática Industrial (2008).