



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCUELA TÉCNICA  
SUPERIOR INGENIEROS  
INDUSTRIALES VALENCIA

Curso Académico:



## AGRADECIMIENTOS

Quiero aprovechar la ocasión para agradecer al profesor Ángel Miguel Cuenca Lacruz la oportunidad de trabajar con él en este trabajo. Un profesor del cual guardaré siempre un especial recuerdo por su forma de involucrarse con los alumnos, con una metodología de enseñanza proactiva y con una gran devoción en su trabajo. Cualidades que valoro mucho y que hicieron muy sencilla mi decisión a la hora de escoger a un profesor como tutor para mi Trabajo Final de Máster. Muchas gracias por todo, Ángel.

En lo personal, siempre agradeceré a mis padres y hermano por la educación y valores que me han inculcado y por proporcionarme todos los medios necesarios para mi formación profesional. Además, quiero agradecerles todo el apoyo del que he gozado en los momentos difíciles.

Por último, quiero reconocer todo el apoyo y ánimo recibido de mi pareja y amigos que ha sido muy importante para mí. Asimismo no me puedo olvidar en este momento de todos mis compañeros a lo largo de estos años, quienes han compartido conmigo muchos momentos difíciles pero que al final han dado sus frutos.

A todos vosotros, Gracias.



## RESUMEN

El Trabajo Final de Máster está basado en el estudio de mejora del sistema de control del posicionamiento de los cabezales de lectura/escritura de un disco duro (Voice Coil Motor, VCM) para la obtención de un ahorro computacional y energético.

Para poder conseguir el objetivo del presente trabajo se va a hacer uso de la combinación de dos técnicas de control avanzadas (técnicas de control no convencionales):

- Sistemas multifrecuencia con entrelazado. Se trata de un sistema de control avanzado que ha tenido especial atención en los últimos años dentro del campo de la automática debido a las grandes ventajas que ésta puede aportar. Tiene un gran interés en los sistemas que tiene varias variables a controlar o que trabajan a diferentes períodos de muestreo.
- Muestre disparado por eventos periódicos. En esta forma de control va a ser la ocurrencia de una determinada condición la que marque la actualización de los parámetros, al contrario que en el control convencional en el cual esta actualización es llevada a cabo con una determinada frecuencia (período de muestreo).

Con el objetivo de poder observar los resultados de la combinación de estas técnicas avanzadas de control, se hará uso de la herramienta informática Truetime, de Matlab. Ésta permitirá la evaluación del control a tiempo real.

Para poder evaluar los resultados obtenidos de las diferentes simulaciones, se introducirán una serie de indicadores mediante los cuales se obtendrá la solución más idónea para la aplicación.

Finalmente se evaluará el coste económico que ha requerido el presente estudio de mejora del sistema de control.



# ÍNDICE

## Índice de la memoria

1.	Objeto del proyecto .....	5
2.	Introducción y motivación .....	5
3.	Desarrollo .....	7
3.1	Escenario de trabajo .....	7
3.2	Descripción de las técnicas avanzadas utilizadas .....	11
	Sistemas multifrecuencia con entrelazado .....	11
	Muestreo disparado por eventos periódicos .....	16
3.3	Descripción de la aplicación .....	19
3.4	Simulaciones .....	25
4.	Conclusiones .....	40

## Índice del presupuesto

5.	Presupuesto .....	47
5.1	Coste de redacción del proyecto .....	48
5.2	Coste de servicios formativos .....	48
5.3	Coste material .....	49
5.4	Resumen del presupuesto .....	52

## Anexos y bibliografía

6.	Anexos .....	55
6.1	Código para truetime .....	55
	Código para el sensor .....	55
	CÓDIGO PARA EL ACTUADOR .....	63
	function actuator_init .....	63
7.	Bibliografía .....	66





---

# MEMORIA





## 1. OBJETO DEL PROYECTO

El objeto del presente documento es el de realizar el estudio de mejora en el sistema de control del posicionamiento de los cabezales de lectura/escritura de un disco duro (Voice Coil Motor, VCM) con el fin de reducir la utilización de sus elementos, consiguiendo así un ahorro computacional y energético y aumentando su velocidad. Todo ello sin que estas mejoras ocasionen una pérdida de rendimiento significativa en el sistema.

Para ello se hará uso de nuevas técnicas de control avanzadas que actualmente están teniendo un gran interés de investigación en el campo de la automática.

## 2. INTRODUCCIÓN Y MOTIVACIÓN

El diseño de controladores es tradicionalmente un campo de gran interés dada la importancia de conseguir optimizar los sistemas de control de procesos. Numerosos estudios acerca de los sistemas de control son realizados para conseguir mejoras en el tránsito de información entre sus elementos (sensor, controlador y actuador), ya sea para control en red o Networked Control System (NCS) (desarrollado en [9]), en el cual el intercambio de información se realiza de forma indirecta a través del controlador y la planta por medio de paquetes, o, el caso convencional, que es el caso de estudio del presente trabajo, en el cual todos los elementos del sistemas están localizados en la misma ubicación, por tanto el intercambio de información se realiza de forma continua y directa sin necesidad de red. Para este último caso se han desarrollados estudios como [8].

Una de las vías para conseguir una reducción en el cómputo de información en el controlador es la basada en el control multifrecuencia. En este tipo de funcionamiento, las diferentes variables involucradas en el sistema de control trabajan a diferentes frecuencias y puede resultar muy interesante en aplicaciones industriales donde las acciones de control sean más rápidas que las medidas del sensor, como en lo estudiado en [1] o dónde tengamos varias acciones de control trabajando a diferente frecuencia, como en el estudio [2]. Esta técnica forma parte de un campo de estudio al cual se le presta ya desde hace años mucha atención por los buenos rendimientos de control que es capaz de conseguir. Además, el control multifrecuencia combinado con la técnica del entrelazado de acciones de control favorece la consecución de un mayor ahorro computacional. Un caso de aplicación para esta técnica de entrelazado se puede encontrar en el trabajo [3]



Otra forma de mejora en los sistemas de control es la debida a la consideración del muestreo basados en eventos, en la cual, es la ocurrencia de un evento la que va a determinar el muestreo y el control del sistema en lugar de un periodo de tiempo, que es el modo de funcionamiento que ha dominado hasta el momento en la ingeniería de control. Ésta técnica permite una reducción computacional y energética en el sistema, y tiene una gran influencia en el régimen permanente de la señal de respuesta. Algunos estudios han sido llevados a cabo acerca de esta técnica avanzada de control como los publicados por [4] o [5].

En el presente proyecto, para realizar la mejora en el sistema de control se hará un estudio de simulación donde el sistema de control incluirá las técnicas de control avanzadas descritas anteriormente. Cabe destacar que éstas técnicas de reducción de cómputo de información y energía están inversamente relacionadas con el rendimiento del sistema, pues una mayor reducción en éstos aspectos va a producir una decremento en el rendimiento del sistema, por tanto, debe haber un compromiso entre estas dos variables.

Con la consecución del proyecto, se va a obtener un sistema de control con una menor utilización de cada uno de sus elementos, lo que se traduce en un sistema de mayor velocidad debido a la reducción en el cómputo de información, que puede resultar muy interesante en el campo de la informática, y además con un consumo energético inferior ocasionado por una menor utilización del actuador. Ambos aspectos son de gran interés en la sociedad industrial actual, donde los conceptos de optimización y mejora de procesos están muy presentes.

### 3. DESARROLLO

#### 3.1 ESCENARIO DE TRABAJO

Como ya se ha expuesto anteriormente, el escenario de trabajo es el de un sistema de control convencional, en el cual no hay red de intercambio de información, por tanto, no se van a producir retardos y tampoco se correrá el riesgo de sufrir pérdidas de paquetes de datos en su traspaso. Con ésta forma de trabajo se puede considerar que el tiempo transcurrido entre la medida  $y(t)$  y la aplicación de la acción de control  $u(t)$  es nulo, siempre y cuando el sistema sea lo suficientemente rápido (sin retardos inherentes al propio sistema ni retardos de computación).

Para determinar el comportamiento dinámico del motor de posicionamiento de los cabezales de lectura/escritura del disco duro (Voice Coil Motor, VCM) la función de transferencia utilizada (fdt) será:

$$G(s) = \frac{7.153 * 10^6}{s^2 + 502.7s + 1.011 * 10^6} \quad (1)$$

la cual es una aproximación obtenida a partir de la ecuación 2 y los parámetros obtenidos a partir de la tesis [7]:

$$G'(s) = \frac{K_0 \omega_0}{s^2 + 2\xi_0 \omega_0 s + \omega_0^2} + \sum_{n=1}^3 \frac{K_n \omega_n}{s^2 + 2\xi_n \omega_n s + \omega_n^2} \quad (2)$$

Parámetros	Modo 1	Modo 2	Modo 3
Frecuencia natural $\omega_n$ (Hz)	5900	8400	11500
Factor de amortiguamiento $\xi_n$	0.022	0.023	0.035
Ganancia $K_n$	-0.04	0.0098	0.0048

$$K_0 = 3.6 * 10^7 \quad \xi_0 = 0.25 \quad \omega_0 = 160\text{Hz}$$

Tabla 1: Parámetros del VCM

La aproximación está basada en el hecho de que los modos de resonancia de alta frecuencia podrían ser reducidos con filtros Notch. A continuación podemos ver los resultados del diagrama de Bode para  $G(s)$ :

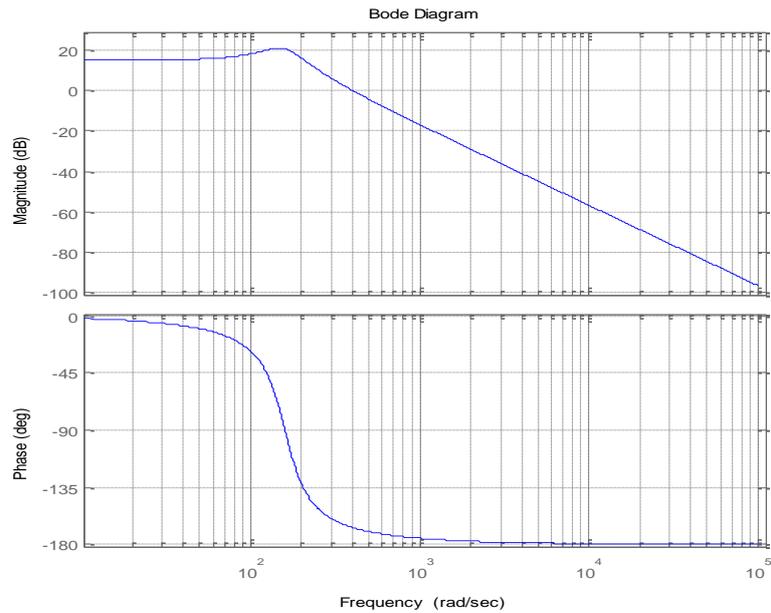


Figura 1: Diagrama de Bode

Estos resultados son aproximadamente los obtenidos en la tesis [7] para  $G'(s)$  con la salvedad de que no tenemos los modos resonantes eliminados por la introducción de los filtros Notch.

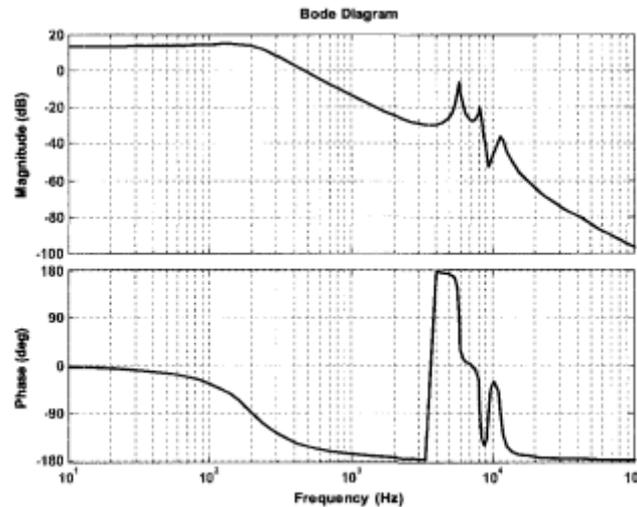


Figura 2: Respuesta en frecuencia del modelo de planta  $G(s)$

El regulador discreto utilizado para el sistema de control será el obtenido en la tesis [7] para el cual, el período de muestreo será de  $T=20\mu\text{sec}$

$$C_D(z) = 1.0882 * \frac{(z-0.9813)(z-0.9681)(z+1)}{(z-1)(z-0.8496)(z-0.6137)} \quad (3)$$

Finalmente el sistema inicial de estudio será el representado mediante el siguiente diagrama de bloques, el cual muestra un sistema de control discreto convencional (monofrecuencia):

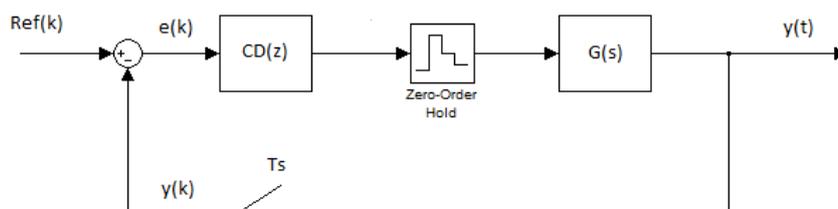
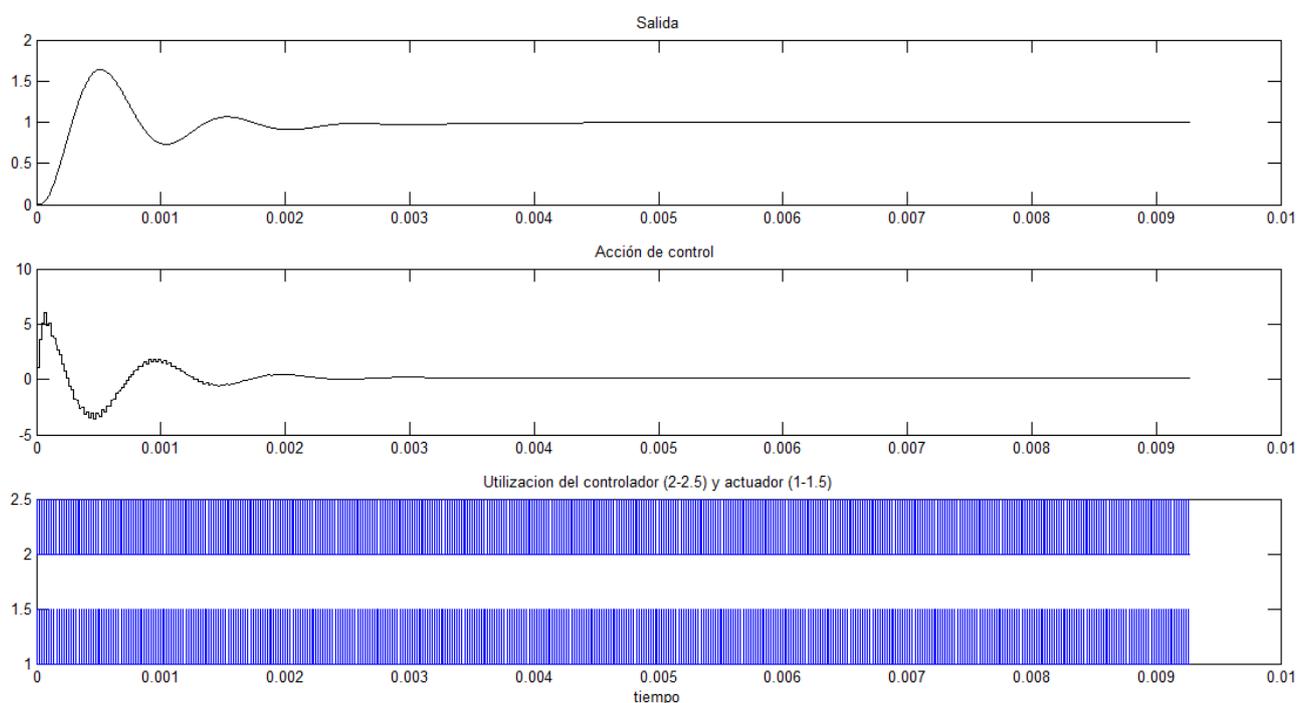


Figura 3: Diagrama de bloques inicial

Como se puede apreciar, se hace necesario en el diagrama de bloques de la *figura 3* un retenedor de orden cero (Zero-Order Hold), necesario para convertir la señal de salida del proceso a una señal continua. Puesto que la entrada al proceso debe ser mediante una señal discreta, se ha colocado un muestreador en la realimentación a período  $T_s=20\mu\text{sec}$

A continuación se muestra la señal de salida del sistema ante un escalón unitario. Esta señal será la que sirva como referencia para poder realizar el análisis de los resultados que se vayan obteniendo tras la introducción de las técnicas avanzadas de control.



*Gráfica 1: Respuesta del sistema para estrategia basada en tiempos*

Cabe destacar que esta señal de salida es la obtenida para un sistema dinámico **basado en tiempo**, para el cual su período de muestreo es de  $T=20\mu\text{sec}$ , igual a la frecuencia del regulador.

Como se puede apreciar en la *Gráfica 1*, la utilización del sensor y actuador es del 100%. Además, por tratarse de un sistema monofrecuencia, la utilización del controlador también es total.



## 3.2 DESCRIPCIÓN DE LAS TÉCNICAS AVANZADAS UTILIZADAS

En los apartados siguientes se van a describir las técnicas avanzadas de control que se van a introducir para conseguir el ahorro computacional y energético. Además, realizando la comparativa entre los resultados que se obtienen y el sistema inicial monofrecuencia basado en tiempo, se podrá comprobar el ahorro conseguido y la pérdida del rendimiento que estos cambios producen para poder encontrar una solución adecuada que permita un ahorro computacional y energético sin que tenga una pérdida de rendimiento significativa.

### SISTEMAS MULTIFRECUENCIA CON ENTRELAZADO

El empleo de sistemas multifrecuencia es una técnica avanzada de control que es utilizada en sistemas que realizan el control de diferentes variables, por ejemplo, una planta industrial. Cada una de estas variables puede tener un periodo de muestreo diferente debido a la complejidad de su medición, retardos inherentes, naturaleza de la variable, etc.

En los sistemas monofrecuencia utilizados en una planta con diferentes variables, el control sobre éstas se realiza para todas a la misma frecuencia lenta, correspondiente al período de muestreo más grande de las variables controladas. Esta condición no es deseable ya que las variables que deban ser medidas a un periodo de muestreo menor no serán controladas adecuadamente, y su respuesta no será la esperada.

Con el control multifrecuencia se pretende que el sistema sea capaz de trabajar con variables que tengan diferentes periodos de muestreo, sin que las variables más lentas en su medición condicionen al resto y así conseguir un control adecuado de todas las variables de la planta.

Además, este método de control es también utilizado para los casos en que el tiempo de medición de una variable es superior al tiempo de cálculo de la acción de control. Un ejemplo de este último caso es el desarrollado en [1].

También resulta una técnica muy interesante para los casos en que no es necesario realizar muchas mediciones de una variable, y por tanto, podríamos reducir la frecuencia de muestreo para dicha variable realizando menos medidas y así obtener un ahorro en los recursos del sistema o utilizarlos para otro fin.

En el presenta trabajo se va hacer uso de esta técnica de control mediante el empleo de un sistema **multifrecuencia**.

Esta forma de trabajo viene motivada por el deseo de conseguir un ahorro computacional y energético en el regulador encargado del control del proceso. La principal desventaja de establecer un control multifrecuencia radica en la dificultad añadida en la etapa de modelado por la utilización de esta técnica de control avanzada en lugar de la técnica de control convencional, la cual está diseñada sobre la base de la utilización de un mismo período de muestreo constante tanto en el sensor como en el controlador.

Para la obtención de un sistema multifrecuencia, el **regulador** se va a descomponer en paralelo, dando como resultado dos modos lentos ( $u_1$  y  $u_2$ ), que trabajarán con un periodo de muestreo de  $T=40\mu\text{sec}$  y uno rápido ( $u_3$ ) con  $T=20\mu\text{sec}$ .

A continuación se detallará el proceso de obtención de las fdt para los modos lentos y el rápido:

- 1) Se va a realizar la descomposición del regulador en paralelo quedando la ecuación 3 inicial del regulador definida de la siguiente forma:

$$C_D(z) = C_{Ds1}(z) + C_{Ds2}(z) + C_{Df}(z) + K_D = \\ = \frac{0.02242}{z-1} + \frac{-0.8848}{z-0.8496} + \frac{2.51}{z-0.6137} + 1.0882 \quad (4)$$

En la *ecuación 4* podemos ver que los polos más lentos del sistema, son los correspondientes para las posiciones  $z=1$  y  $z=0.8496$ , que serán los correspondientes a los modos lentos.

- 2) Para la obtención de los modos lentos, se va a proceder a aplicar la siguiente relación de la tesis del trabajo [7]:

Dado un bloque de tiempo discreto de primer orden  $\frac{1}{z-a}$ , si su entrada se mantiene constante durante  $N$  intervalos de tiempo, la función de transferencia a una tasa de muestreo más lento con un factor  $N$ :

$$\frac{a^{n-1} + a^{n-2} + \dots + a + 1}{z^n + a^n} \quad (5)$$

Y con la aplicación de esta relación se obtienen las siguientes funciones de transferencias (fdt) para los modos lentos:

$$G_{Ds1}(z^2) = \frac{0.04484}{z^2-1} \quad (6)$$

$$G_{Ds2}(z^2) = \frac{-1.6365}{z^2-0.7218} \quad (7)$$

3) Finalmente el modo rápido aglutina la constante y la parte del regulador correspondiente al polo más rápido:

$$G_{Df}(z) = \frac{2.51}{z-0.6137} + \frac{1.0882*(z-0.6137)}{(z-0.6137)} = \frac{1.0882z+1.84}{z-0.6137} \quad (8)$$

A fin de comprender mejor el funcionamiento del sistema multifrecuencia, se van a disponer las acciones de control en un diagrama temporal. Una primera aproximación podría ser la mostrada en la figura siguiente:

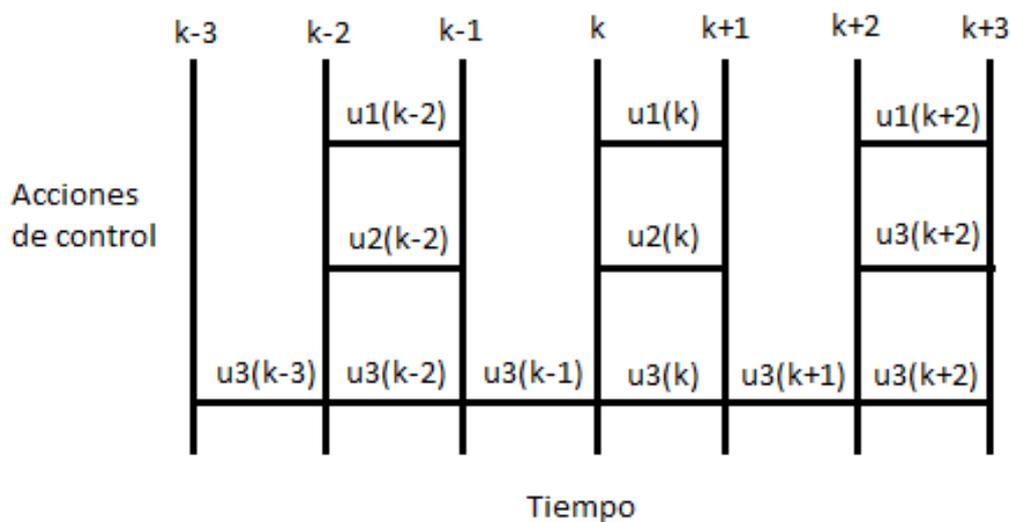


Figura 4: Diagrama temporal

Esta primera solución mostrada en la figura 4 se va a considerar como no válida, ya que el controlador para este caso no tiene la misma carga computacional en cada iteración del

proceso, ya que en las iteraciones pares el controlador tendrá una utilización del 100%, y en las impares del 33%. Esta forma de trabajo puede hacer rebasar ciertos límites de tiempos en las iteraciones pares, los cuales pueden ser muy restrictivos en aplicaciones como el posicionamiento del cabezal del disco duro, que implica altas velocidades.

Una segunda solución al problema de multifrecuencia consistirá en **intercalar o entrelazar** la aplicación de los modos lentos. El diagrama temporal obtenido mediante la técnica de multifrecuencia con entrelazado es el siguiente:

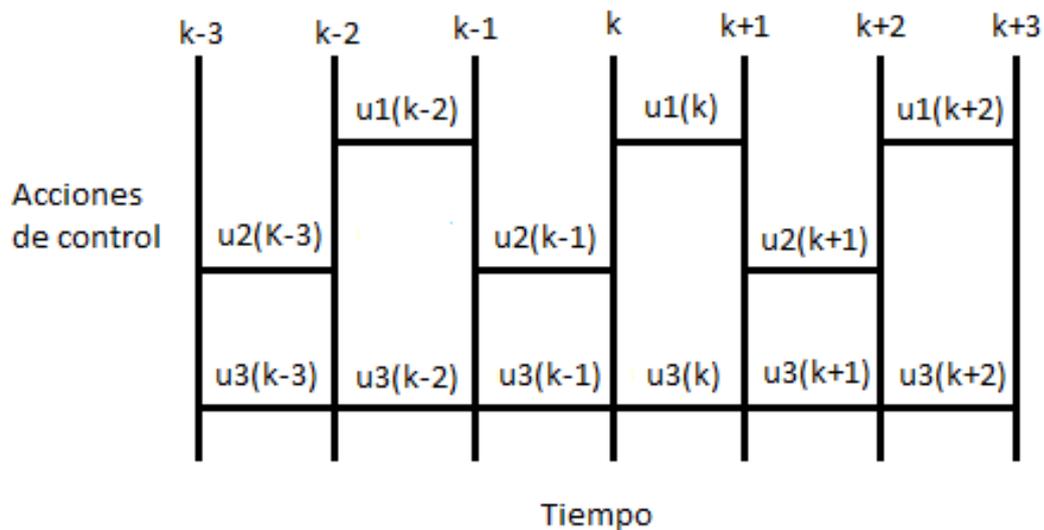


Figura 5: Diagrama temporal con entrelazado

Esta última solución es correcta, ya que la carga computacional del controlador ya se ha homogeneizado para todas las iteraciones, y además se ha conseguido reducir su utilización en un 33% respecto al sistema monofrecuencia, ya que para todas las iteraciones hay un modo lento que no realiza cálculo.

En definitiva, con ésta técnica de control multifrecuencia con entrelazado, se ha conseguido distribuir la utilización del controlador sin que ello produzca una pérdida de rendimiento en la señal de respuesta del sistema.

El diagrama de bloques correspondiente a éste sistema multifrecuencia con entrelazado es el siguiente:

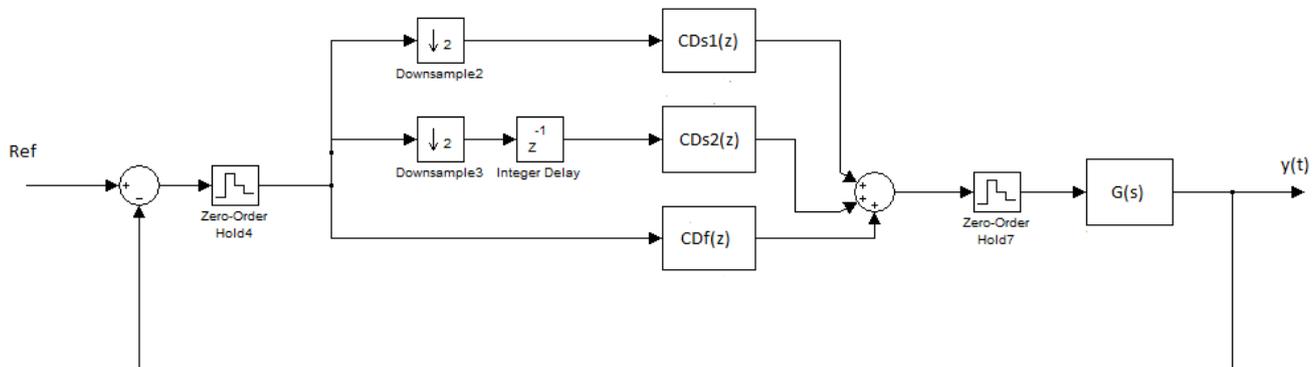
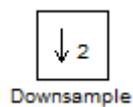
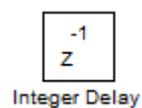


Figura 6: Diagrama de bloques para sistema multifrecuencia con entrelazado

en dónde:



Este bloque reduce la velocidad de muestreo suprimiendo muestras. Para el caso de estudio, éste bloque va a permitir la medición cada dos muestras. En el sistema se hace necesario ya que en la implementación en el programa de simulación sólo vamos a tener una frecuencia de muestreo e introduciendo este bloque se podrá trabajar con diferentes frecuencias.



Este bloque se hace necesario para poder realizar el entrelazado. Su función es la de introducir un retardo a una de las acciones de control para que los modos lentos trabajen de forma intercalada.

Este bloque se ha utilizado por dos motivos:



- Utilizado como si fuera un muestreador antes del bloque downsample, ya que éste necesita que a su entrada la señal sea discreta.
- Antes del modelo del sistema ( $G(s)$ ) para realizar la conversión de D/A que aparece en un sistema de control por computador

## MUESTREO DISPARADO POR EVENTOS PERIÓDICOS

En este apartado se va a describir una nueva técnica de control avanzada, la cual consiste en la introducción de la condición de eventos para la actualización de los parámetros del sistema de control y que está teniendo cada vez mayor interés en la ingeniería de control.

Los sistemas de control basados en tiempo, en los cuales un período de muestreo fijo marca la condición de la actualización de las variables, han dominado la forma de trabajo en la ingeniería de control. Esto es debido a la extensa teoría de control que se ha desarrollado en los últimos años.

Sin embargo, existen numerosas aplicaciones en la industrial en las cuales el método de trabajo basado en tiempo no es la mejor forma de llevar a cabo el control del sistema.

En éste contexto, surge la necesidad de estudiar una nueva forma de control, como es la introducción de la condición por eventos. En este caso, el sistema de control actualiza sus variables dependiendo de si se cumple cierta condición, en vez de realizar su actualización de manera periódica.

Algunos estudios acerca de esta nueva técnica avanzada de control han sido llevados a cabo, por ejemplo en [4], [5] y [6]

Una forma de entender bien la diferencia entre del control basado en tiempo y en eventos es a través del siguiente ejemplo cotidiano.

Un repartidor que tiene que entregar unos paquetes a sus clientes. En la situación del muestreo basado en tiempo cada cliente dispone de un reloj para ir a comprobar cada 10 minutos si ha llegado un nuevo paquete a su puerta. Si no ha llegado puede continuar su trabajo. Sin embargo, en el caso del muestreo basado en eventos, el repartidor llama a la puerta del cliente específico a quien tiene que entregar el paquete quien abre la puerta y lo recoge. Los otros clientes pueden continuar realizando sus tareas sin necesidad de ser interrumpidos ([6], pág 6)

Como se puede intuir en este ejemplo, con este método de trabajo basado en eventos se pueden obtener ventajas muy significativas en relación a la utilización de los elementos del sistema.

Esta técnica resulta muy interesante para aquellas aplicaciones industriales que tengan limitaciones en alguna de sus capacidades, por ejemplo:

Algunas aplicaciones tienen baja capacidad de cómputo o requieren de un bajo uso del procesador para destinar su utilización a otras acciones.

Otras aplicaciones industriales pueden tener limitaciones en la comunicación. Con la utilización del sistema basado en eventos se conseguirá una reducción del uso de bus de comunicación debido a la consecución de un menor número de muestras.

Además, esta forma de trabajo tiene la principal ventaja de que al reducir la cantidad de actualización de las variables y de los envíos de información, se está consiguiendo un aumento en la velocidad de respuesta del sistema.

Aunque parezca una alternativa muy ventajosa e interesante para el campo de la ingeniería de control, la técnica basada en eventos no está todavía muy extendida en la industria, pues presenta dificultades asociadas al pobre desarrollo de una teoría formal. Asimismo, ha sufrido numerosas críticas, ya que se está hablando de una técnica de control que sustituirá a la basada en tiempo, pero tiene la dificultad de que se debe comprobar periódicamente si se cumple la condición del evento para actualizar las variables del sistema. En el presente trabajo, será el sensor el que realice en todas las iteraciones la comprobación del cumplimiento del evento.

Dentro de este método de control, existen diferentes formas de implementar la condición del evento como se puede ver en el trabajo desarrollado en [6].

Codificación de una condición de evento absoluta.

$$t_{k+1} = \inf\{t > t_k \mid |e(t)| \geq \delta\} \quad (8)$$

donde  $\delta > 0$  y es un límite absoluto.

Codificación de una condición de evento mixta

$$t_{k+1} = \inf\{t > t_k \mid |e(t)| \geq \delta + \sigma \|x(t)\|\} \quad (9)$$

donde se ha combinado un límite absoluto y uno relativo.

En el **presente proyecto**, se va a hacer uso de esta técnica de control avanzada para conseguir un ahorro computacional y energético. Para ello, se va a introducir la condición de eventos en el sensor y en el controlador consiguiendo así reducir el número de envíos entre el sensor-controlador y controlador-actuador.

La condición de evento utilizada será una condición mixta en la cual se utiliza el cuadrado de los términos ya que permite un análisis de estabilidad del sistema de control más sencillo, como puede comprobarse en el documento [10].

- Dentro del sensor, se va a introducir la siguiente condición:

$$(y(k) - y(k - 1))^2 > \delta_s + \sigma_s * y(k)^2 \quad (10)$$

Siendo  $y(k)$  la respuesta del sistema para el instante actual,  $y(k-1)$  correspondiendo a la última señal de la salida que se envió entre el sensor y el controlador y  $\delta_s, \sigma_s$  dos constantes mayores que 0.

Gracias a esta condición introducida en el sensor, se va a reducir el número de envíos entre el sensor y controlador, lo que va a permitir una reducción en el número de cálculos de acciones de control (ahorro computacional).

- Dentro del controlador, se va a introducir la siguiente condición:

$$(u(k) - u(k - 1))^2 > \delta_c + \sigma_c * u(k)^2 \quad (11)$$

Donde  $u(k)$  es la acción de control para este instante, y  $u(k-1)$  corresponde a la última acción de control que se envió desde el controlador al actuador.

Con la introducción de esta condición se va a conseguir reducir el número de envíos entre el controlador y actuador, lo que va a reducir la utilización del actuador. Esto se traduce en un aumento de ahorro energético.



### 3.3 DESCRIPCIÓN DE LA APLICACIÓN

Para llevar a cabo el presente proyecto se ha hecho uso de la herramienta informática Truetime, que es un simulador basado en Simulink de Matlab para sistemas de control a tiempo real y en red.

Truetime utiliza dos tipos de bloques en la simulación que son Truetime Kernel y Truetime Network. Debido a la naturaleza de la aplicación, el presente trabajo se centrará en la utilización de los bloques del Kernel, y solo se modificarán algunos parámetros relacionados con la red.

Esta aplicación es de gran interés para el presente proyecto gracias a las facilidades que ésta presenta a la introducción y manejo de eventos en el sistema de control, ya que permite la co-simulación de las tareas de ejecución del controlador, las transmisiones de la red y la dinámica de plantas continuas.

Por tanto, va a permitir disponer de información de los elementos del sistema en cada instante de tiempo, así se podrá ver en un diagrama temporal la utilización de cada uno de los elementos del sistema, la señal de respuesta y de las acciones de control.

Además, permite la utilización del código C++ para realizar la programación ya sea para los comandos de inicialización o para las tareas de ejecución, lo que facilita su uso por ser un lenguaje de programación muy utilizado. También podrá ser utilizados archivos m. de Matlab para los mismos fines, siendo más aconsejables la utilización de éste último método.

Todos los detalles y la metodología para la implementación del código, inicialización, uso del simulador, compilación, etc. estarán expuestos en el documento [12].

A continuación se muestra el diagrama de bloques para el caso de estudio del proyecto y se pasará a describir cada uno de los puntos importantes de éste.

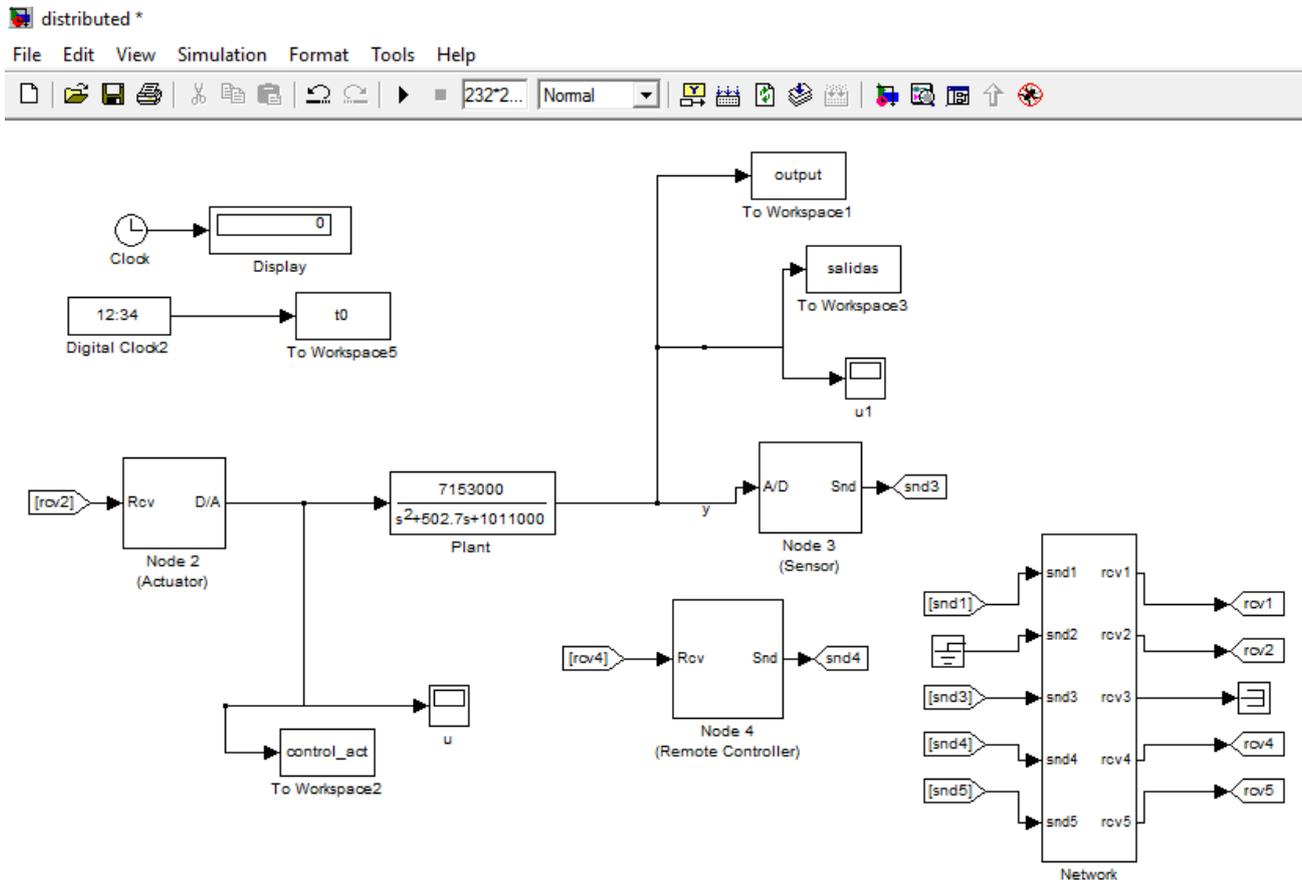


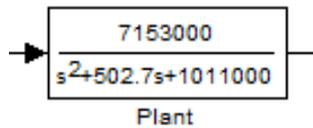
Figura 7: Diagrama de bloques para el sistema en Truetime

Como se puede comprobar en la anterior figura, Truetime utiliza el mismo entorno de trabajo al utilizado en Simulink, ya que está basado en este último.

En la parte superior de la imagen, se encuentra la barra de herramientas de la aplicación con diferentes funciones. En el apartado de tiempo de simulación se ha introducido una duración total de 9.28 milisegundos, tiempo suficiente para poder observar la dinámica de la respuesta del sistema.

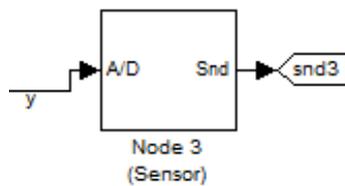
Los elementos más importantes que componen el sistema son los siguientes:

- **Modelo de la planta**



Este bloque introduce el modelo matemático que representa la dinámica que sigue el cabezal de posicionamiento del disco duro (Voice Coil Motor, VCM)

- **Sensor**

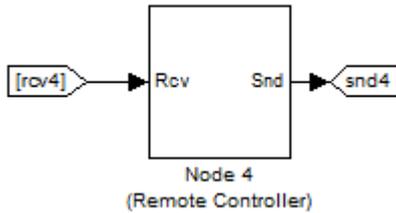


Se trata del elemento que mide la señal de respuesta del sistema. Como se puede ver en su bloque, éste lleva un convertidor analógico/digital para que el resto de elementos del sistema (controlador y actuador) puedan trabajar de manera discreta.

Como ya se ha descrito en el punto anterior, en este bloque aparecerá la condición del evento mediante la cual se hará la comparación entre la salida del sistema actual y la última salida que se envió al controlador para comprobar la condición de envío (*ecuación 10*).

Si se cumple dicha condición, entonces llegará la información de la salida al controlador para que así una nueva acción de control pueda ser calculada.

- **Controlador**



Este elemento recibe la señal de salida del sistema proveniente del sensor para así poder obtener el error que existe entre la referencia que debe seguir el sistema y su salida actual.

$$error(k) = Ref - y(k) \quad (12)$$

En este bloque del sistema estará implementado el regulador. Debido a la codificación de Truetime, para poder realizar el cálculo de la nueva acción de control, el regulador deberá estar implementado en forma de ecuaciones en diferencias, expresado de la siguiente forma:

$$u_1(k) = u_1(k - 2) + 0.04484 * error(k - 2) \quad (11)$$

$$u_2(k) = 0.7218 * u_2(k - 2) - 1.6365 * error(k - 3) \quad (12)$$

$$u_3(k) = 0.6137 * u_3(k - 1) + 1.0882 * error(k) + 1.84 * error(k - 1) \quad (13)$$

$$u(k) = u_1(k) + u_2(k) + u_3(k) \quad (14)$$

Siendo:

$u_1(k)$  la acción de control correspondiente al primer modo lento

$u_2(k)$  la acción de control correspondiente al segundo modo lento

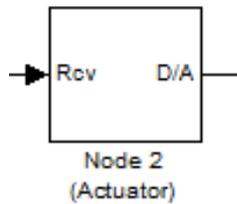
$u_3(k)$  la acción de control correspondiente al modo rápido

Los términos que dependen de una acción de control anterior serán los correspondientes a iteraciones en las que sí se ha realizado el cálculo de ésta.

Además, como se ha comentado anteriormente, este bloque también tendrá implementado la condición envío de la acción de control entre el controlador y actuador (*ecuación 11*)

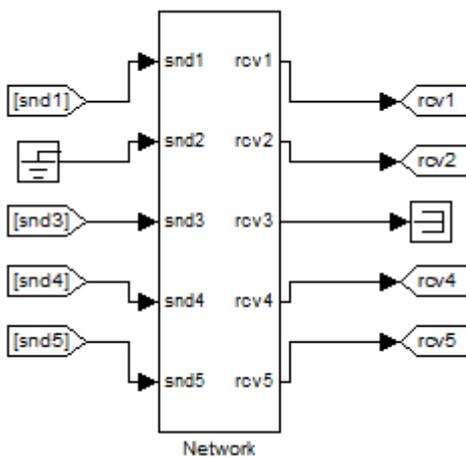
Por la cual, si la acción de control actual es aproximada a la última acción de control enviada al actuador no se produce el envío, consiguiendo el ahorro energético deseado ya que el actuador no realizará trabajo.

- **Actuador**



Este bloque es el encargado de aplicar la acción de control al sistema que recibe del controlador. Si éste no recibe información del controlador debido a que no se ha cumplido la condición de envío en este último, el actuador continúa aplicando la última acción de control recibida.

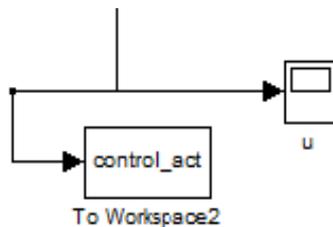
- **Red (Network)**



Este bloque es el que determina todas las características de la red del sistema. En él podemos modificar su velocidad, que para el presente trabajo se ha elegido un valor elevado, ya que se trata de un sistema de control convencional en el cual no hay ni retardos debidos a la saturación del bus de información ni retardos debidos al tiempo de cómputo de nueva información.

- **Bloques secundarios**

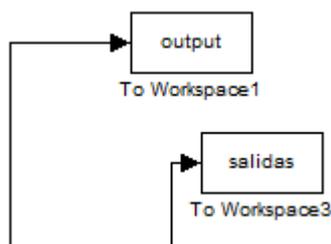
A continuación se mostrarán una serie de bloques secundarios que son necesarios para obtener las señales de respuesta del sistema y el diagrama temporal de las acciones de control aplicadas. Además servirán de conexión con el entorno de trabajo de Matlab para poder obtener información necesaria del sistema y poder tratarla con herramientas de Matlab.



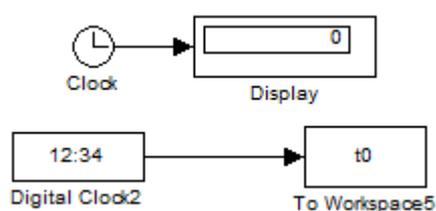
El primero de estos dos bloques (To Workspace) permitirá guardar todas las acciones de control aplicadas al sistema para poder trabajar con ellas desde Matlab.

al sistema.

El segundo de estos bloques es conocido en el lenguaje de Simulink como “scope”. Con él será posible visualizar en un diagrama temporal todas las acciones de control aplicadas



Estos dos bloques también son del tipo ToWorkspace, utilizándose ahora para guardar todas las salidas de la simulación del sistema y así poder ser utilizarlas en el entorno de Matlab.



Mediante estos bloques, es posible llevar la variable de tiempo de simulación continuo y discreto, respectivamente, al entorno de trabajo de Matlab.

### 3.4 SIMULACIONES

En el siguiente apartado se va a proceder a realizar simulaciones mediante la aplicación Truetime de Matlab con diferentes valores de las constantes  $\sigma_s$ ,  $\delta_s$ ,  $\sigma_c$  y  $\delta_c$  (si cambias la notación, aquí de debe cambiar también) para las condiciones de envío de información (condición de evento) entre sensor-controlador y controlador-actuador.

A fin de poder evaluar los resultados obtenidos en las siguientes simulaciones se va a definir una serie de índices, los cuales ya han sido utilizados en trabajos como [5]:

- **ISE:** se define como la integral del error al cuadrado, y es utilizado para definir el comportamiento de los sistemas de control. Es relativamente insensible a errores pequeños puesto que va al cuadrado, y por tanto, los errores grandes tendrá mayor peso en su valor.

$$ISE = \int_0^{\infty} e^2(t) dt \quad (13)$$

Para este caso de estudio, se rescribirá de la siguiente forma:

$$ISE = \sqrt{\sum_1^{464} (error(k))^2} \quad (14)$$

*Nota\*:* se ha tomado el sumatorio desde 1 hasta 464 ya que es el número total de iteraciones que se va a realizar en la simulación

Para el caso de estudio del trabajo, el valor de ISE es el correspondiente a la simulación basada en tiempo, el cual tiene un valor de **ISE = 3.81**

- **ISEP:** se define como el error integral absoluto al cuadrado entre la salida del sistema de una estrategia basada en tiempo y la salida del sistema con la estrategia basada en eventos.

$$ISEP = \int_0^{\infty} |y_{basado\ en\ tiempo}(t) - y_{basado\ en\ eventos}(t)|^2 dt \quad (15)$$

Para este caso de estudio, la *ecuación 15* se reescribirá de la siguiente forma:

$$ISEP = \sqrt{\sum_1^{464} |y_{basado\ en\ tiempo}(k) - y_{basado\ en\ eventos}(k)|^2} \quad (16)$$

*Nota\**: se ha tomado el sumatorio desde 1 hasta 464 ya que es el número total de iteraciones que se va a realizar en la simulación

- **NE**: este índice trata de comparar la calidad de la respuesta del sistema mediante la relación entre los índices anteriores.

$$NE = \frac{ISEP}{ISE} \quad (17)$$

Los tres índices anteriores cuantifican el rendimiento del sistema de control. No obstante, también se ha de utilizar otros índices que cuantifiquen la utilización de los dispositivos incluidos en el sistema de control (en definitiva, el ahorro computacional y energético que se puede lograr). En este caso se tendrá en consideración:

- **Uc**: la utilización del controlador

$$Uc = \frac{N_{SC}}{Iteraciones\ totales} \quad (18)$$

dónde N\_SC: es el número de envíos entre el sensor y controlador

- **Ua**: la utilización del actuador

$$Ua = \frac{N_{CA}}{Iteraciones\ totales} \quad (19)$$

dónde N\_CA: es el número de envíos entre el controlador y el actuador

Ambos índices de utilización siempre serán inferiores a la unidad.

- **GPI:** es el índice más significativo. Muestra el compromiso entre el rendimiento del control del sistema y el ahorro computacional y energético. Mediante este índice se va a poder determinar qué solución es la óptima para la problemática del trabajo.

$$GPI = W_1 * Uc + W_2 * Ua + W_3 * NE \quad (18)$$

dónde:

$W_1$ ,  $W_2$  y  $W_3$ : son coeficientes específicos de cada relación de la ecuación del GPI mediante los cuales es posible dar mayor peso a alguno de los aspectos presentes en este índice.

Cabe destacar que cuanto menor sean los dos primeros términos referidos a la utilización del controlador y actuador, mayor será el término referido al rendimiento del sistema, pues la respuesta del sistema será peor, y viceversa

La solución óptima al problema planteado en este trabajo será la que menor valor de GPI obtenga.

A continuación se van a mostrar las simulaciones realizadas para la obtención de la solución óptima al problema. Para ello, es muy importante la definición de los pesos de cada término dentro de la ecuación del GPI ( $W_1$ ,  $W_2$  y  $W_3$ ), ya que la solución dependerá de la elección de éstos.

Para una primera simulación se ha tomado como valores de estos pesos la unidad para todos ellos ( $W_1=W_2=W_3=1$ ). Esta es una solución en la que se está primando más la reducción de la utilización tanto del controlador como el actuador, debido a que hay dos términos con el mismo peso referidos a la utilización de los elementos (referidos al ahorro computacional y energético) frente a sólo un término referido al rendimiento del sistema.

La siguiente tabla mostrará los resultados obtenidos para las diferentes soluciones en las que se ha ido variando el valor de las constantes a, b, c y d. Estas constantes se relacionan con  $\sigma_s$ ,  $\delta_s$ ,  $\sigma_c$  y  $\delta_c$  de la siguiente manera:

$$\delta_s = 10 * a \quad (19)$$

$$\sigma_s = 10 * b \quad (20)$$

$$\delta_c = 0.01 * c \quad (21)$$

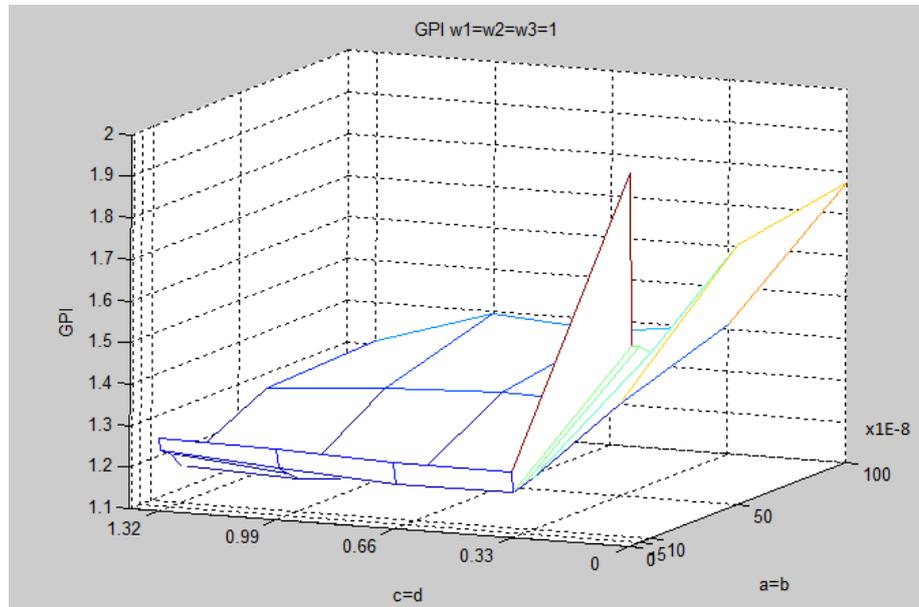
$$\sigma_c = 0.01 * d \quad (22)$$

	a=b	c=d	N_SC	N_SC (%)	N_CA	N_CA(%)	ISEP	NE	GPI
eb_0.0	0	0	464	1	464	1	0	0	2
eb_0.1	0	0,33	464	1	96	0,20689655	0,2030053	0,05327	1,260166
eb_0.2	0	0,66	464	1	83	0,17887931	0,3128587	0,082096	1,260975
eb_0.3	0	0,99	464	1	79	0,17025862	0,3835899	0,100656	1,270914
eb_0.4	0	1,32	464	1	75	0,16163793	0,434536	0,114024	1,275662
eb_1.0	1E-08	0	360	0,7758621	360	0,77586207	0,1086322	0,028506	1,58023
eb_1.2	1E-08	0,33	423	0,9116379	108	0,23275862	0,2436673	0,063939	1,208336
eb_1.3	1E-08	0,66	426	0,9181034	89	0,19181034	0,3622946	0,095068	1,204982
eb_1.4	1E-08	0,99	432	0,9310345	84	0,18103448	0,4218647	0,110699	1,222768
eb_1.5	1E-08	1,32	432	0,9310345	75	0,16163793	0,5779343	0,151653	1,244325
eb_2.0	5E-08	0	353	0,7607759	353	0,76077586	0,2013294	0,05283	1,574382
eb_2.2	5E-08	0,33	401	0,8642241	121	0,26077586	0,3346053	0,087802	1,212802
eb_2.3	5E-08	0,66	413	0,8900862	108	0,23275862	0,4127644	0,108311	1,231156
eb_2.4	5E-08	0,99	404	0,8706897	87	0,1875	0,5460237	0,143279	1,201469
eb_2.5	5E-08	1,32	418	0,9008621	79	0,17025862	0,6067287	0,159209	1,230329
eb_3.0	1E-07	0	339	0,7306034	339	0,73060345	0,3308271	0,086811	1,548018
eb_3.2	1E-07	0,33	387	0,8340517	130	0,28017241	0,4163917	0,109263	1,223487
eb_3.3	1E-07	0,66	392	0,8448276	104	0,22413793	0,5112296	0,134149	1,203115
eb_3.4	1E-07	0,99	391	0,8426724	86	0,18534483	0,5691303	0,149343	1,17736
eb_3.5	1E-07	1,32	397	0,8556034	78	0,16810345	0,627586	0,164682	1,188388
eb_4.0	5E-07	0	342	0,737069	342	0,73706897	0,9637879	0,252903	1,727041
eb_4.2	5E-07	0,33	342	0,737069	152	0,32758621	0,9802544	0,257223	1,321879
eb_4.3	5E-07	0,66	361	0,7780172	130	0,28017241	1,0295295	0,270153	1,328343
eb_4.4	5E-07	0,99	367	0,7909483	115	0,24784483	1,0575689	0,277511	1,316304
eb_4.5	5E-07	1,32	367	0,7909483	98	0,2112069	1,1091565	0,291048	1,293203
eb_5.0	1E-06	0	341	0,7349138	341	0,73491379	1,1626254	0,305078	1,774906
eb_5.2	1E-06	0,33	340	0,7327586	172	0,37068966	1,1751138	0,308355	1,411804
eb_5.3	1E-06	0,66	348	0,75	142	0,30603448	1,212515	0,31817	1,374204
eb_5.4	1E-06	0,99	355	0,7650862	136	0,29310345	1,2767279	0,33502	1,393209
eb_5.5	1E-06	1,32	346	0,7456897	109	0,23491379	1,2400358	0,325391	1,305995

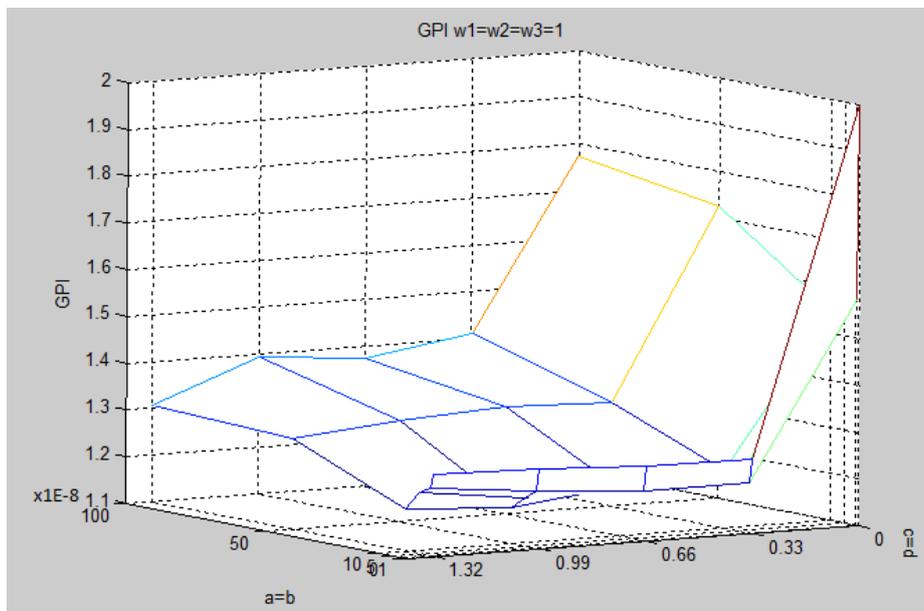
Tabla 2: Resultados para  $W_1=W_2=W_3=1$

Nota\*: para el cálculo de la utilización se ha realizado la relación del número de envíos respecto al número de iteraciones totales de la simulación (464).

A continuación se mostrará una gráfica 3D donde podemos ver los valores mostrados en la *tabla 2* y así comprender mejor la tendencia del proceso al variar las condiciones del evento.



Gráfica 2: GPI ( $W_1=W_2=W_3=1$ ).1



Gráfica 3: GPI ( $W_1=W_2=W_3=1$ ).2

*Nota\**: para el eje correspondiente a los valores  $a=b$  se ha decidido mostrarlos con exponencial para clarificar la gráfica.



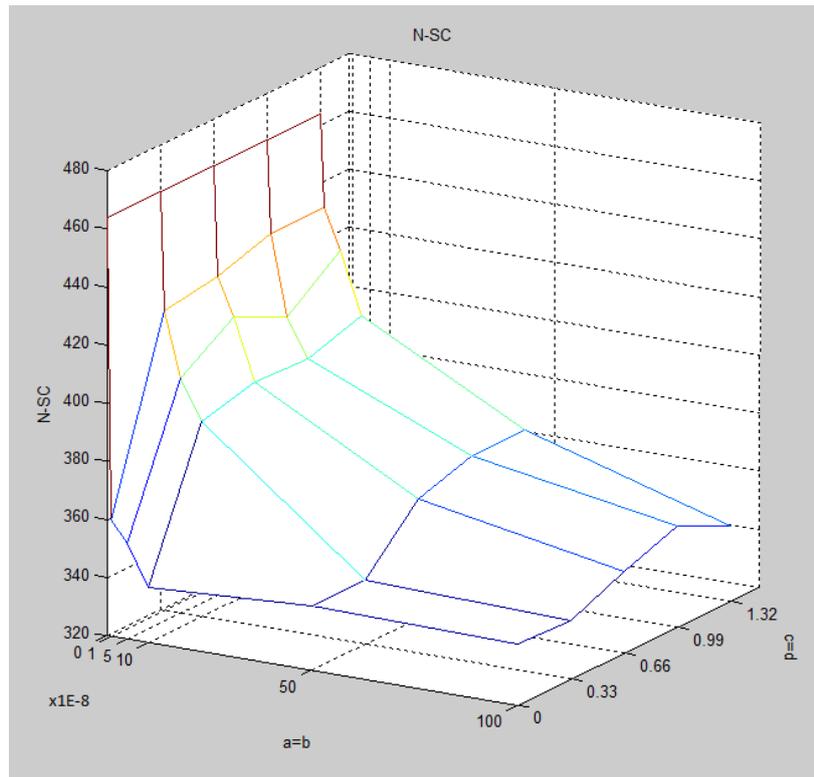
Para este caso, la simulación que presenta un menor valor de GPI es la obtenida cuando los valores  $a=b=1 \cdot 10^{-7}$  y  $c=d=0.99$ . En este caso el GPI es de 1.1776, con una utilización del controlador del 84,26%, del 18.53% del actuador y un error relativo (NE) del 14.93%.

Cabe destacar que la utilización del controlador en realidad no es del 84.26%, ya que al introducir la técnica de multifrecuencia con entrelazado se ha reducido hasta el 66% su utilización respecto a la técnica basada en tiempo. Por tanto, con la adición de la técnica de muestreo con evento la utilización real del controlador es de:

$$0.66 * 0.8426 = 0.5561 \approx 55.61\% \quad (23)$$

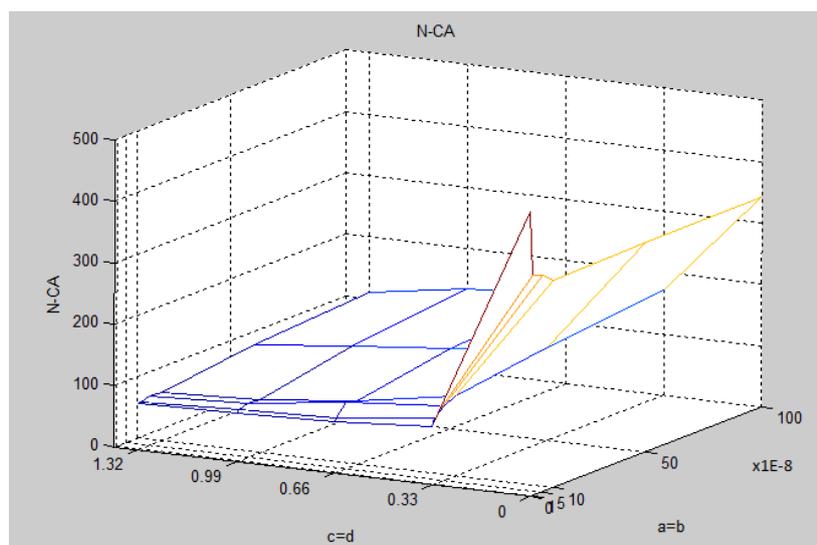
En conclusión, para el caso del controlador, gracias a la introducción de la técnica de multifrecuencia con entrelazado la carga de cómputo se ha reducido en un 33%, pero al hacer uso también de la técnica de muestreo por evento se ha conseguido un ahorro del 10.39% adicional.

Esta simulación va a ser también utilizada para poder comprender mejor el proceso al introducir los eventos como método de ahorro computacional y energético. Para ellos, se mostrarán las gráficas 3D de los diferentes términos de la ecuación del GPI ( $N_{SC}$ ,  $N_{CA}$  y  $NE$ ).



Gráfica 4:  $N\_SC (W_1=W_2=W_3=1)$

La anterior gráfica demuestra cómo se va reduciendo el número de envíos entre el sensor y controlador conforme va aumentando los valores de las constantes “a” y “b” (se va haciendo más restrictiva la condición del evento).



Gráfica 5:  $N\_CA (W_1=W_2=W_3=1)$



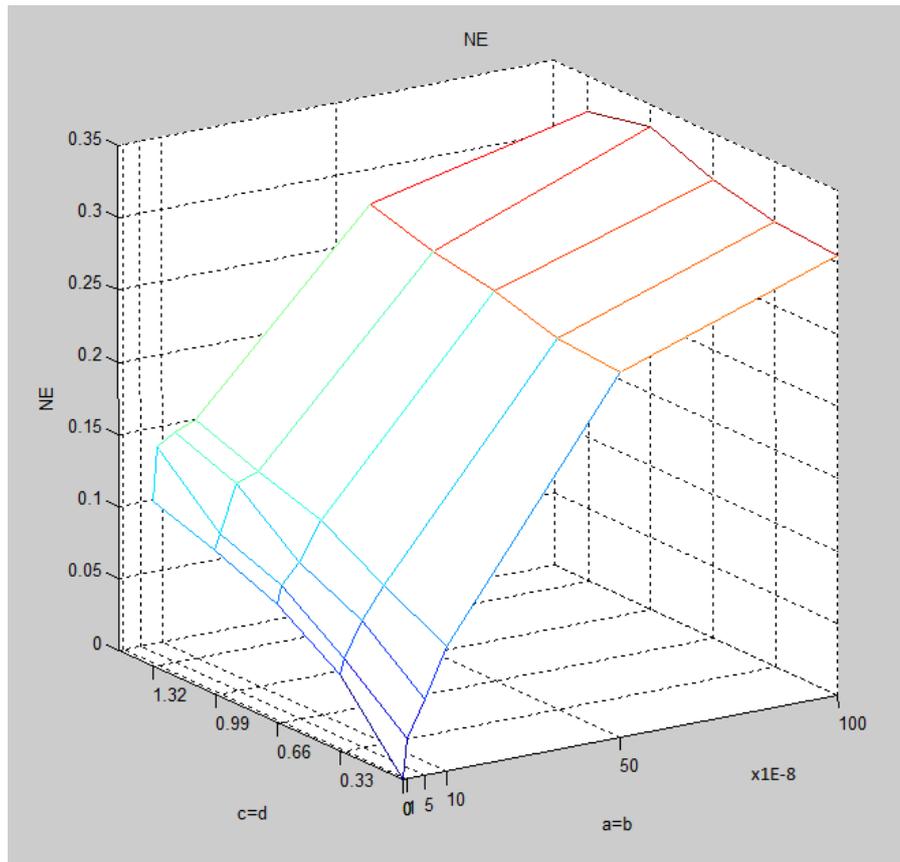
Al igual que para el caso anterior, en la *gráfica 5* se puede comprobar cómo los envíos entre el controlador y actuador se reducen conforme van aumentando los valores de “c” y “d”, lo que se traduce en un ahorro energético en el actuador,

Además de estas dos conclusiones extraídas de las gráficas anteriores hay una tercera conclusión que es muy importante comprender para entender bien la dinámica del proceso. La utilización del controlador no sólo va a ser función de los valores introducidos para los parámetros “a” y “b”. Puesto que se ha utilizado una condición del evento mixta, que se cumpla o no la condición del evento va a depender también de la respuesta del sistema para este caso, por lo tanto, esta utilización será función indirectamente de los parámetros “c” y “d”, que son los que van a marcar la aplicación de una nueva acción de control y condicionarán la respuesta.

Esta relación se puede ver con facilidad en la *gráfica 4*, en la cual, para un mismo valor de “a” y “b” se puede ver como la utilización del controlador aumenta conforme van aumentando los valores de “c” y “d”.

La misma relación ocurre también para el actuador, que se puede ver del mismo modo en la *gráfica 5*, aunque para este caso la relación no sea tan fuerte.

La conclusión que se extrae es que, al usarse menos el actuador, la acción de control se actualiza también menos haciendo empeorar la salida del sistema. Por ello, la condición del evento en el sensor se dispara más veces, lo que hace que se envíe más información hacia el controlador, aumentando su utilización.



Gráfica 6: NE ( $W_1=W_2=W_3=1$ )

La tendencia del índice NE, como se puede comprobar en la *gráfica 6*, es la de incrementar su valor con el aumento de los parámetros “a”, “b”, “c” y “d”. Como era de esperar, a mayor umbral menos eventos disparados y peor rendimiento de control.

Una segunda simulación será la correspondiente a la asignación de pesos para términos del GPI buscando una solución en la que sea más importante la consecución de un notable ahorro computacional y energético:  $W_1 = 0.45$ ,  $W_2 = 0.45$  y  $W_3 = 0.1$ .

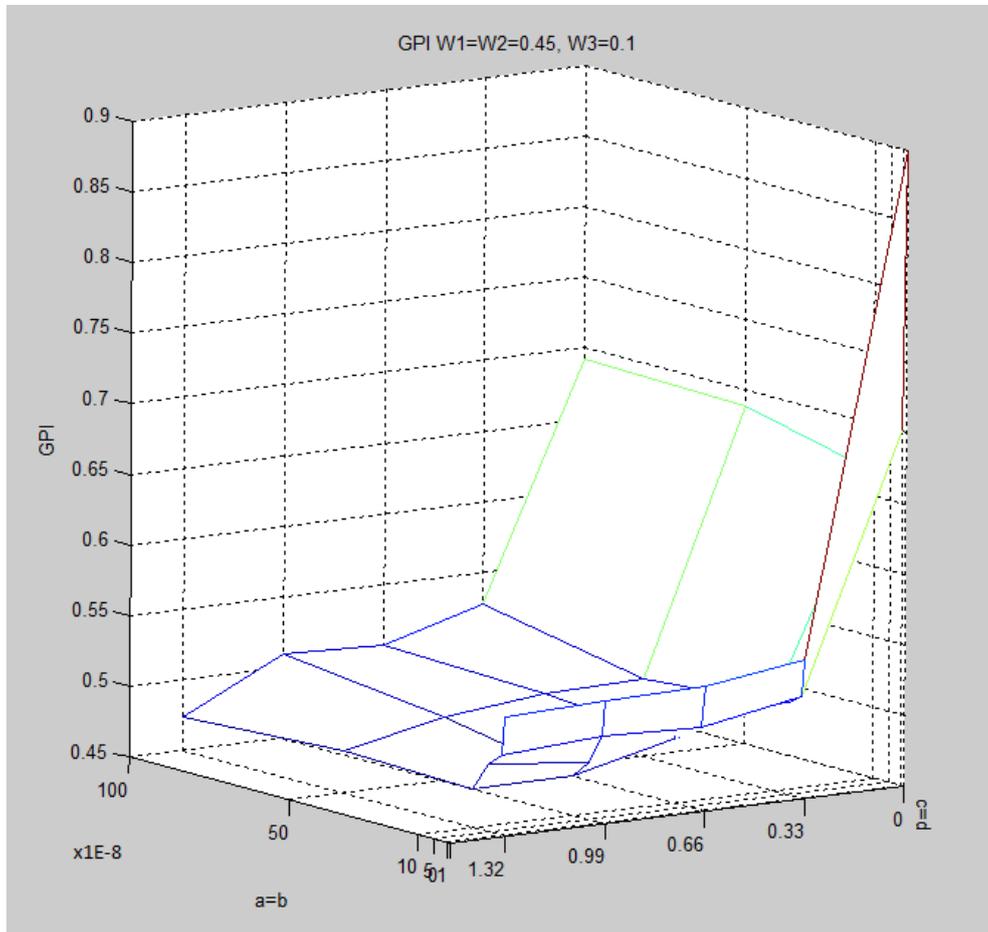
A continuación se muestra la tabla en la que están contenidos los datos de la simulación:

	a=b	c=d	N_SC	N_SC (%)	N_CA	N_CA(%)	ISEP	NE	GPI
eb_0.0	0	0	464	1	464	1	0	0	0,9
eb_0.1	0	0,33	464	1	96	0,206897	0,20301	0,05327	0,54843
eb_0.2	0	0,66	464	1	83	0,178879	0,31286	0,0821	0,538705
eb_0.3	0	0,99	464	1	79	0,170259	0,38359	0,10066	0,536682
eb_0.4	0	1,32	464	1	75	0,161638	0,43454	0,11402	0,53414
eb_1.0	1E-08	0	360	0,77586	360	0,775862	0,10863	0,02851	0,701126
eb_1.2	1E-08	0,33	423	0,91164	108	0,232759	0,24367	0,06394	0,521372
eb_1.3	1E-08	0,66	426	0,9181	89	0,19181	0,36229	0,09507	0,508968
eb_1.4	1E-08	0,99	432	0,93103	84	0,181034	0,42186	0,1107	0,511501
eb_1.5	1E-08	1,32	432	0,93103	75	0,161638	0,57793	0,15165	0,506868
eb_2.0	5E-08	0	353	0,76078	353	0,760776	0,20133	0,05283	0,689981
eb_2.2	5E-08	0,33	401	0,86422	121	0,260776	0,33461	0,0878	0,51503
eb_2.3	5E-08	0,66	413	0,89009	108	0,232759	0,41276	0,10831	0,516111
eb_2.4	5E-08	0,99	404	0,87069	87	0,1875	0,54602	0,14328	0,490513
eb_2.5	5E-08	1,32	418	0,90086	79	0,170259	0,60673	0,15921	0,497925
eb_3.0	1E-07	0	339	0,7306	339	0,730603	0,33083	0,08681	0,666224
eb_3.2	1E-07	0,33	387	0,83405	130	0,280172	0,41639	0,10926	0,512327
eb_3.3	1E-07	0,66	392	0,84483	104	0,224138	0,51123	0,13415	0,494449
eb_3.4	1E-07	0,99	391	0,84267	86	0,185345	0,56913	0,14934	0,477542
eb_3.5	1E-07	1,32	397	0,8556	78	0,168103	0,62759	0,16468	0,477136
eb_4.0	5E-07	0	342	0,73707	342	0,737069	0,96379	0,2529	0,688652
eb_4.2	5E-07	0,33	342	0,73707	152	0,327586	0,98025	0,25722	0,504817
eb_4.3	5E-07	0,66	361	0,77802	130	0,280172	1,02953	0,27015	0,503201
eb_4.4	5E-07	0,99	367	0,79095	115	0,247845	1,05757	0,27751	0,495208
eb_4.5	5E-07	1,32	367	0,79095	98	0,211207	1,10916	0,29105	0,480075
eb_5.0	1E-06	0	341	0,73491	341	0,734914	1,16263	0,30508	0,69193
eb_5.2	1E-06	0,33	340	0,73276	172	0,37069	1,17511	0,30836	0,527387
eb_5.3	1E-06	0,66	348	0,75	142	0,306034	1,21251	0,31817	0,507032
eb_5.4	1E-06	0,99	355	0,76509	136	0,293103	1,27673	0,33502	0,509687
eb_5.5	1E-06	1,32	346	0,74569	109	0,234914	1,24004	0,32539	0,473811

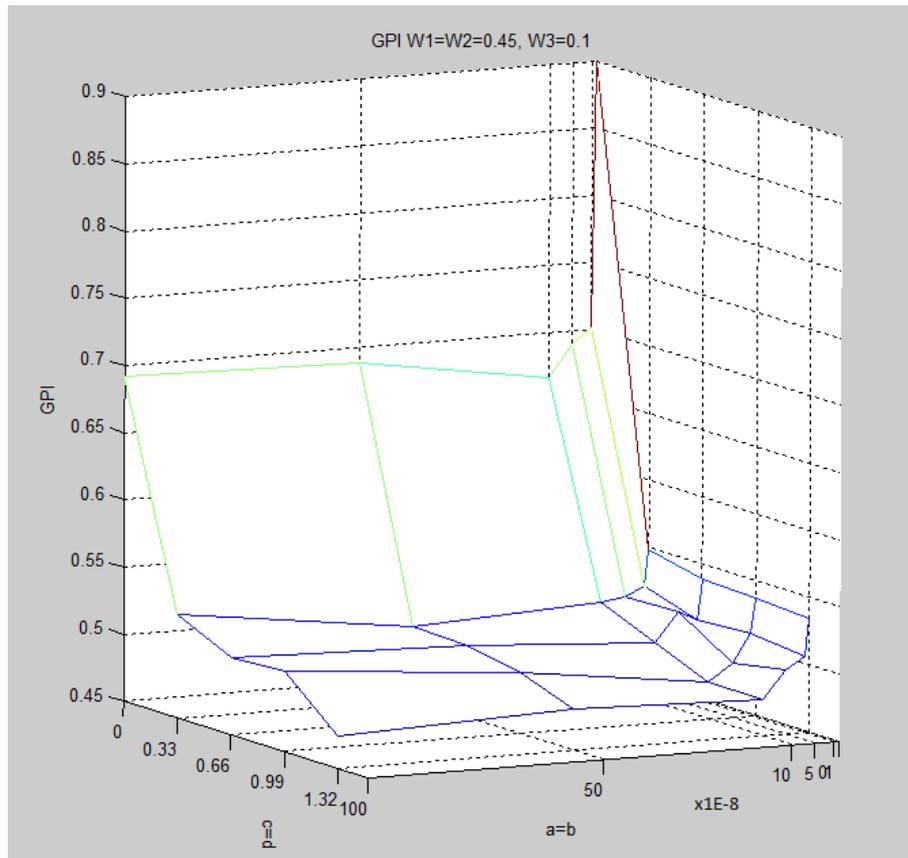
Tabla 3: Resultados para  $W_1=0.45$ ,  $W_2=0.45$ ,  $W_3=0.1$

Nota\*: para el cálculo de la utilización se ha realizado la relación del número de envíos respecto al número de iteraciones totales de la simulación (464).

La gráfica 3D obtenida para los valores de la *tabla 3* es la representada a continuación:



Gráfica 7: GPI ( $W_1=0.45, W_2=0.45, W_3=0.1$ ).



Gráfica 8: GPI ( $W_1=0.45$ ,  $W_2=0.45$ ,  $W_3=0.1$ ).2

Para esta simulación, la solución óptima será la correspondiente a los valores de  $a = b = 1 \cdot 10^{-6}$  y  $c = d = 1.32$  proporcionando un valor de GPI de 0.4738, con utilizaciones del 74.049256% para el controlador y del 23.49% para el actuador y un índice de rendimiento NE del 32.54%.

Como ya se ha comentado para el caso anterior, la utilización real para el controlador es de:

$$0.66 * 0.7456 = 0.492 \approx 49.2\% \quad (24)$$

El último caso de simulación será realizado con el objetivo de encontrar una solución en la que se prime más el índice del rendimiento de control, buscando que la respuesta del sistema no empeore mucho comparada con la respuesta para la estrategia basada en tiempo.

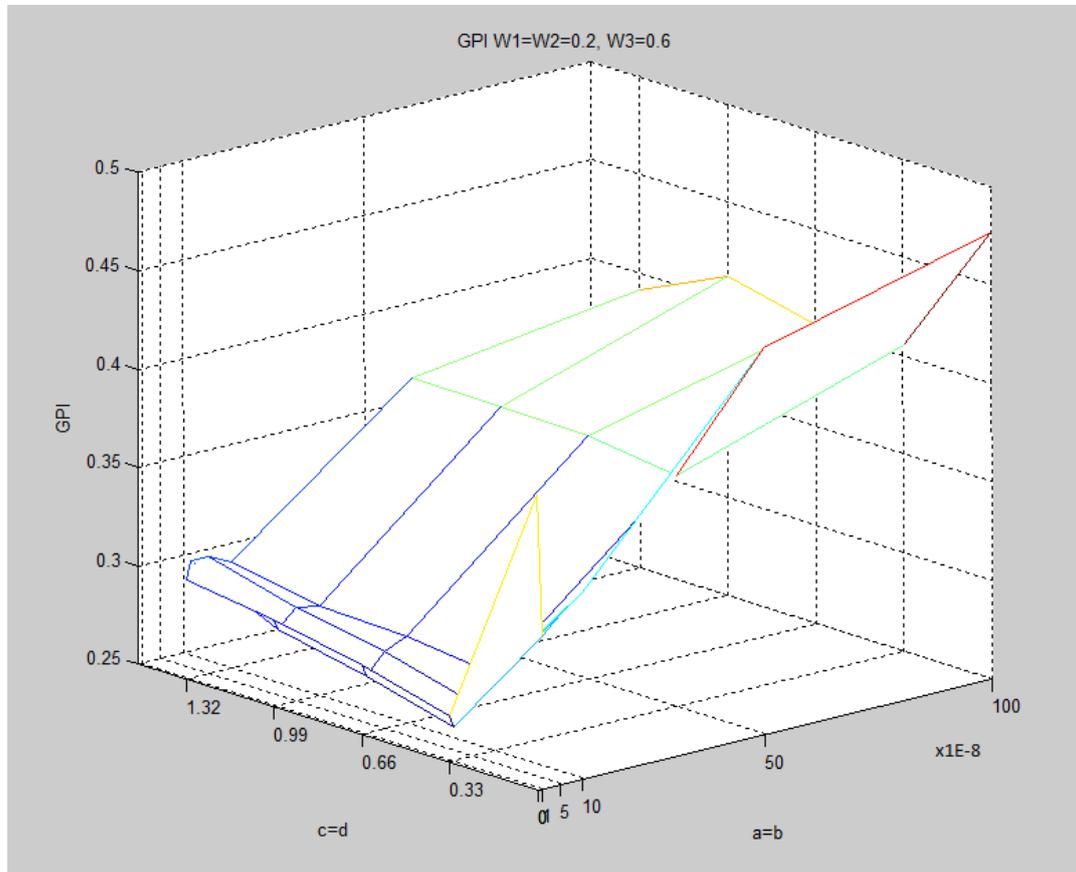
Para este caso los pesos serán:  $W_1 = 0.2$ ,  $W_2 = 0.2$  y  $W_3 = 0.6$ .

La siguiente tabla muestra los datos obtenidos para este caso de simulación:

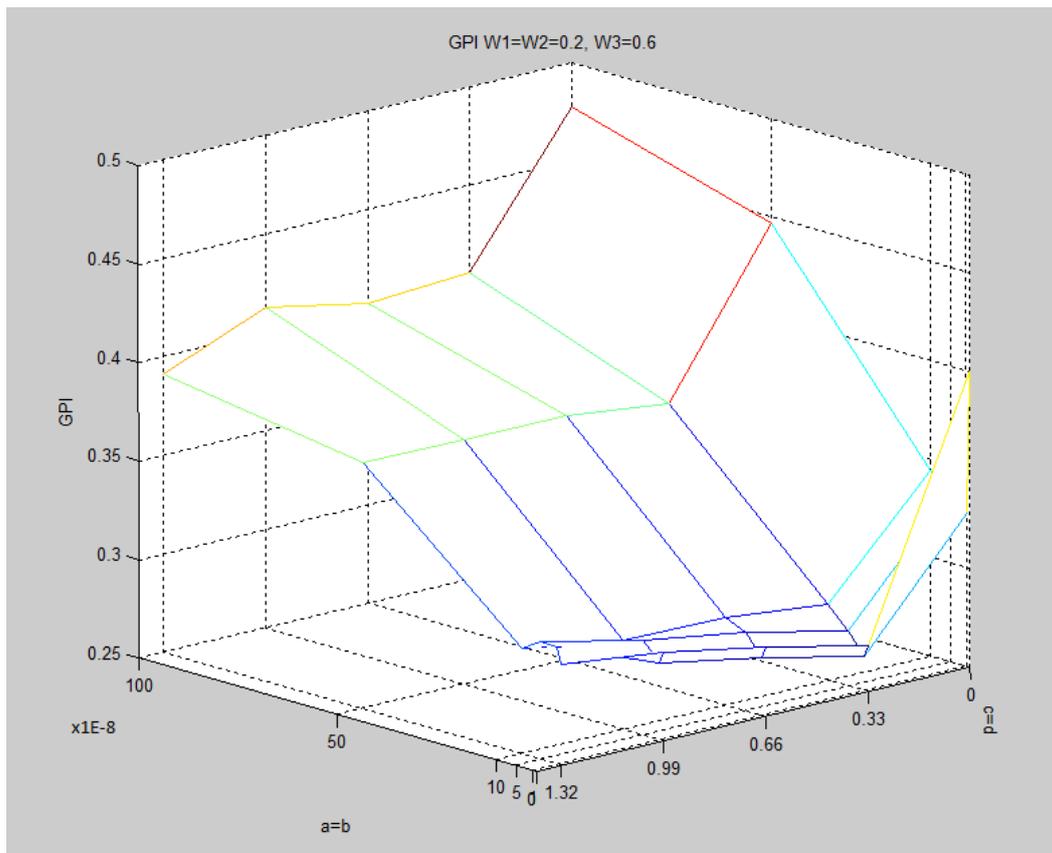
	a=b	c=d	N_SC	N_SC (%)	N_CA	N_CA(%)	ISEP	NE	GPI
eb_0.0	0	0	464	1	464	1	0	0	0,4
eb_0.1	0	0,33	464	1	96	0,2069	0,20301	0,0533	0,2733
eb_0.2	0	0,66	464	1	83	0,17888	0,31286	0,0821	0,285
eb_0.3	0	0,99	464	1	79	0,17026	0,38359	0,1007	0,2944
eb_0.4	0	1,32	464	1	75	0,16164	0,43454	0,114	0,3007
eb_1.0	1E-08	0	360	0,77586	360	0,77586	0,10863	0,0285	0,3274
eb_1.2	1E-08	0,33	423	0,91164	108	0,23276	0,24367	0,0639	0,2672
eb_1.3	1E-08	0,66	426	0,9181	89	0,19181	0,36229	0,0951	0,279
eb_1.4	1E-08	0,99	432	0,93103	84	0,18103	0,42186	0,1107	0,2888
eb_1.5	1E-08	1,32	432	0,93103	75	0,16164	0,57793	0,1517	0,3095
eb_2.0	5E-08	0	353	0,76078	353	0,76078	0,20133	0,0528	0,336
eb_2.2	5E-08	0,33	401	0,86422	121	0,26078	0,33461	0,0878	0,2777
eb_2.3	5E-08	0,66	413	0,89009	108	0,23276	0,41276	0,1083	0,2896
eb_2.4	5E-08	0,99	404	0,87069	87	0,1875	0,54602	0,1433	0,2976
eb_2.5	5E-08	1,32	418	0,90086	79	0,17026	0,60673	0,1592	0,3097
eb_3.0	1E-07	0	339	0,7306	339	0,7306	0,33083	0,0868	0,3443
eb_3.2	1E-07	0,33	387	0,83405	130	0,28017	0,41639	0,1093	0,2884
eb_3.3	1E-07	0,66	392	0,84483	104	0,22414	0,51123	0,1341	0,2943
eb_3.4	1E-07	0,99	391	0,84267	86	0,18534	0,56913	0,1493	0,2952
eb_3.5	1E-07	1,32	397	0,8556	78	0,1681	0,62759	0,1647	0,3036
eb_4.0	5E-07	0	342	0,73707	342	0,73707	0,96379	0,2529	0,4466
eb_4.2	5E-07	0,33	342	0,73707	152	0,32759	0,98025	0,2572	0,3673
eb_4.3	5E-07	0,66	361	0,77802	130	0,28017	1,02953	0,2702	0,3737
eb_4.4	5E-07	0,99	367	0,79095	115	0,24784	1,05757	0,2775	0,3743
eb_4.5	5E-07	1,32	367	0,79095	98	0,21121	1,10916	0,291	0,3751
eb_5.0	1E-06	0	341	0,73491	341	0,73491	1,16263	0,3051	0,477
eb_5.2	1E-06	0,33	340	0,73276	172	0,37069	1,17511	0,3084	0,4057
eb_5.3	1E-06	0,66	348	0,75	142	0,30603	1,21251	0,3182	0,4021
eb_5.4	1E-06	0,99	355	0,76509	136	0,2931	1,27673	0,335	0,4126
eb_5.5	1E-06	1,32	346	0,74569	109	0,23491	1,24004	0,3254	0,3914

Tabla 4: Resultados para  $W_1=0.2$ ,  $W_2=0.2$ ,  $W_3=0.6$

La gráfica 3D que representa los datos obtenidos en la tabla 4 se representa a continuación:



Gráfica 9: GPI ( $W_1=0.2, W_2=0.2, W_3=0.6$ ).1



Gráfica 9: GPI ( $W_1=0.2, W_2=0.2, W_3=0.6$ ).2

La solución obtenida para este caso de estudio es la representada por los valores  $a = b = 1 \cdot 10^{-8}$  y  $c = d = 0.33$  proporcionando un valor de GPI de 0.267, con utilizations del  $66\% \cdot 91.16\% = 60.16\%$  en el controlador, que como se puede comprobar sólo se consigue un 6% de ahorro adicional que con la utilización de solo la técnica de multifrecuencia con entrelazado, y del 23.27% en el actuador, además de un valor 6.39% para el índice NE.

## 4. CONCLUSIONES

En el siguiente apartado se va a analizar qué solución de las simuladas anteriormente es más adecuada para el caso de aplicación del presente trabajo, el cual tiene como objetivo realizar una mejora en el ahorro computacional y energético en el control de posicionamiento de los cabezales de lectura/escritura de un disco duro (Voice Coil Motor, VCM).

Para poder realizar la elección de la mejor solución al problema se va a analizar los resultados obtenidos en términos de utilización y rendimiento para los dos últimos casos simulados, dejando el primero de ellos como una simulación de referencia para entender la dinámica del control.

El **primero** de ellos es el correspondiente al caso en que se estimula la obtención de una solución con un mayor ahorro computacional y energético en decremento de una pérdida de rendimiento para la señal de respuesta del sistema respecto a la basada en tiempo. Los pesos para las variables serán  $W_1=0.45$ ,  $W_2=0.45$   $W_3=0.1$ , para los cuales, se han logrado los siguientes resultados:

- Utilización del controlador = 49.2%
- Utilización del actuador = 23.49%
- Índice ISEP = 1.24
- Índice NE = 32.54%

El segundo de los casos será ajustado para la adquisición de una solución en la que se persigue una menor pérdida en el rendimiento de la señal de respuesta del sistema. En este caso los valores de los parámetros serán  $W_1=0.2$ ,  $W_2=0.2$   $W_3=0.6$ , y se han alcanzado unos resultados:

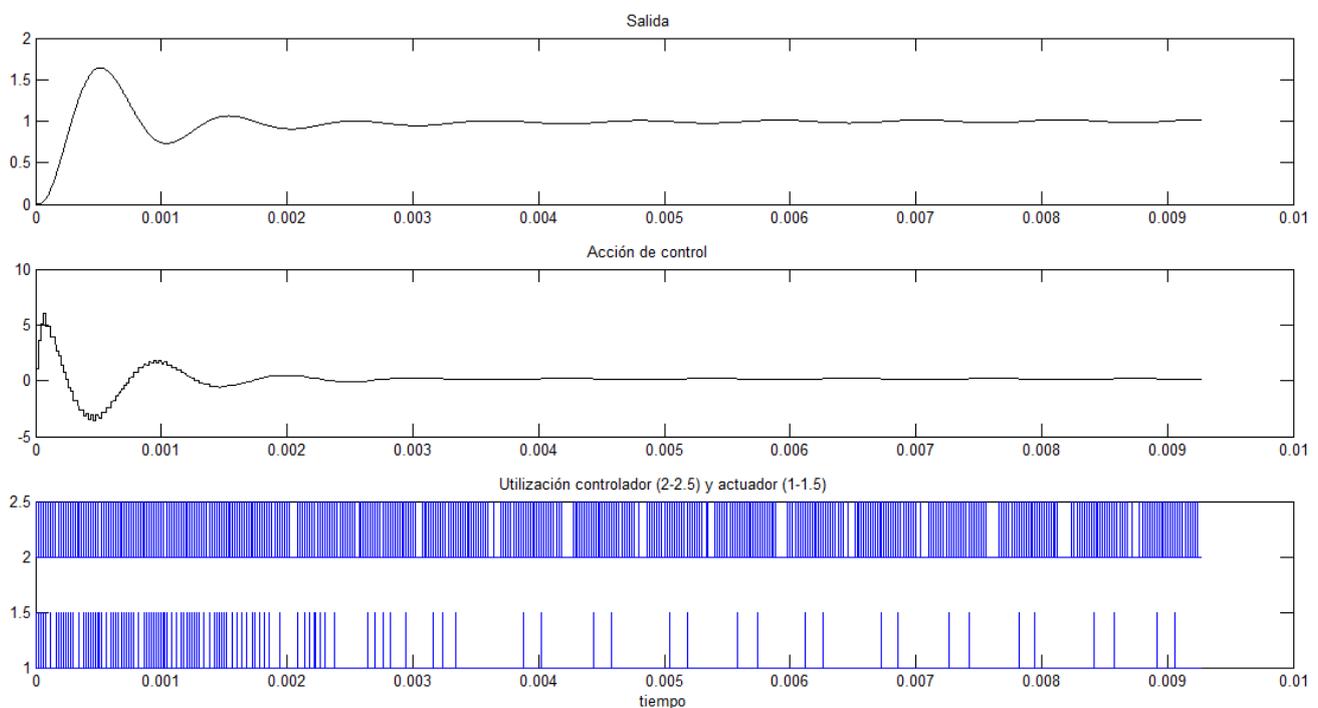
- Utilización del controlador = 60.16%
- Utilización del actuador = 23.27%
- Índice ISEP = 0.2436
- Índice NE = 6.39%

Analizando los resultados para las dos simulaciones anteriores se ha llegado a la conclusión de que la solución más adecuada es la obtenida para el caso en el que se presta una mayor importancia al rendimiento del sistema ( $W_1=0.2$ ,  $W_2=0.2$   $W_3=0.6$ ) por los siguientes motivos:

La utilización del actuador para ambos casos se ha conseguido unos valores muy parecidos, y aunque en la utilización del controlador si hay una diferencia del 10%, la limitación más significativa se encuentra en el rendimiento de la respuesta, ya que sí habrá una diferencia muy importante ente los dos casos. En caso de haber elegido la estrategia en la que se da mayor importancia al ahorro computacional y energético, la mejora que se habría conseguido respecto al caso elegido no hubiera sido muy reseñable, sin embargo, el rendimiento de la salida del sistema hubiera empeorado significativamente.

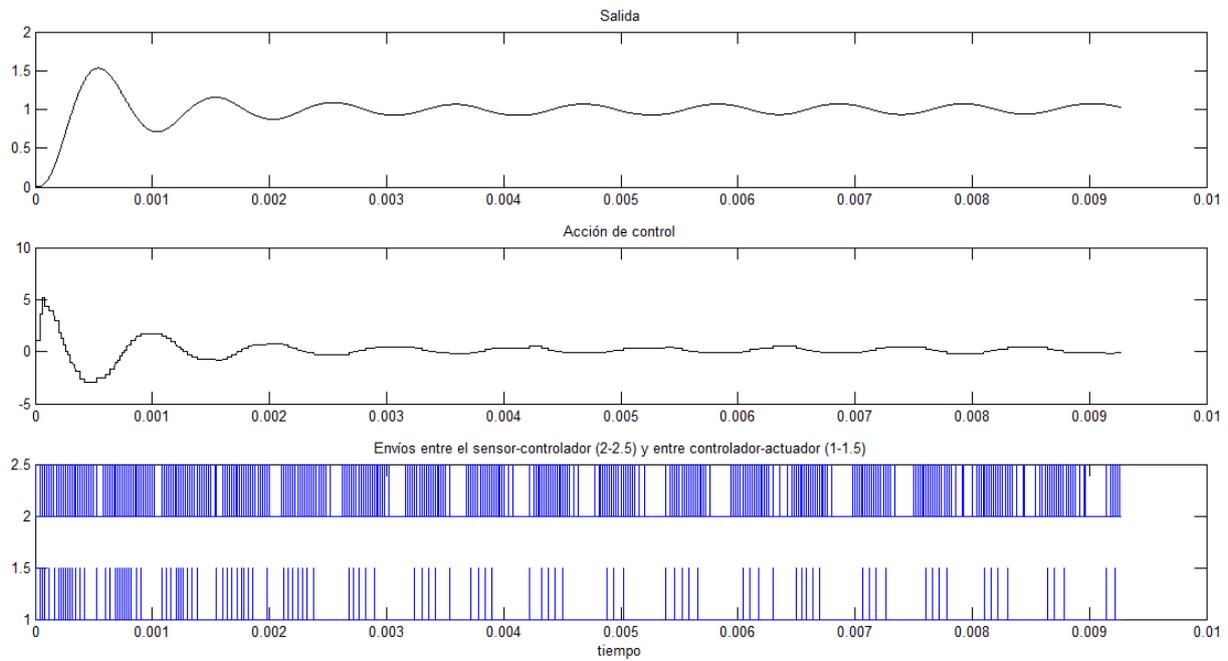
Además, el posicionamiento de los cabezales del disco duro requiere de un sistema de control que sea lo más preciso posible, por tanto la salida del sistema deberá ser con el mínimo error posible.

A continuación se mostrará la señal se salida del sistema para los valores de los parámetros “a”, “b”, “c” y “d” correspondientes a la solución adoptada ( $W_1=0.2$ ,  $W_2=0.2$   $W_3=0.6$ ).



Gráfica 10: Respuesta del sistema para estrategia basad en eventos, solución elegida.

Haciendo la comparación con la respuesta del caso en que se prima más la obtención de un mayor ahorro computacional y energético.



Gráfica 11: Respuesta del sistema para estrategia basada en eventos, solución descartada.

Se puede comprobar cómo se ha reducido sobre todo la utilización del controlador, no tanto la del actuador, a costa de tener una señal de respuesta con un mayor rizado en el permanente, lo que se traduce en un peor rendimiento.

Realizando una comparación con la señal entre la salida de la solución adoptada (Gráfica 10) y la señal de respuesta para la situación inicial basada en tiempo mostrada en la Gráfica 1, parece que la utilización del controlador se ha reducido en gran medida (un 40%) y la del actuador muy notablemente (un 77%), lo que supone un ahorro computacional y energético importante. Todo ello con una pérdida de rendimiento de la salida del sistema considerada como aceptable (un 6.39%) para las ventajas de ahorro conseguidas.



Para finalizar, del presente trabajo se pueden sacar las siguientes conclusiones de la utilización de las técnicas de control avanzadas aplicadas a éste:

Gracias a la implementación de un **sistema multifrecuencia con entrelazado** se ha conseguido reducir la carga de cómputo del controlador en un 33%, por tanto y como se ha mencionado anteriormente, ésta técnica es la que proporcionará una mayor reducción en la utilización de este elemento. Esta forma de control del sistema tendrá influencia en todo el proceso (tanto en el transitorio como en el permanente) sin provocar una pérdida de rendimiento en la respuesta.

Debido a la **introducción de las condiciones de evento** se ha logrado una reducción en la utilización muy importante en el actuador, lo que se traduce en un ahorro energético. Esta reducción no es tan relevante en el controlador, aunque ya se ha demostrado que los resultados son mejores que los reflejados en la *Gráfica 1*. Por el contrario, mediante el empleo de esta técnica de control sí que se va a obtener una reducción del rendimiento de la respuesta y será necesario llegar a un compromiso entre ahorro computacional y energético, y rendimiento. Este método de control avanzado tendrá influencia sobre todo en la parte permanente de la señal de respuesta (como se puede ver en la utilización de actuador en la *Gráfica 10*).

La utilización conjunta de estas dos técnicas avanzadas va a proporcionar un sistema de control el cual pueda obtener un ahorro computacional y energético trascendente, pero con la consecuencia de un empeoramiento del rendimiento de la señal de respuesta que podría llegar a ser significativo.





---

# PRESUPUESTO



## 5. PRESUPUESTO

El objetivo del presente apartado es el de calcular y mostrar el presupuesto basado en las horas de trabajo dedicadas para el cálculo y desarrollo del mismo así como de los materiales y documentación necesarios para su realización.

Con la finalidad de detallar de forma más precisa el trabajo realizado para la consecución del mismo, a continuación se mostrará un cronograma de trabajo:

Cronograma de trabajo										
Actividades realizadas	Enero		Febrero		Marzo		Abril		Mayo	
Estudio del problema planteado		X								
Revisión bibliográfica existente		X	X							
Estudio de las técnicas avanzadas de control			X	X	X					
Adquisición de la aplicación de simulación Truetime				X						
Familiarización con Truetime				X	X					
Implementación del código en Truetime						X	X	X		
Realización de simulaciones								X	X	
Estudio de las soluciones obtenidas									X	
Redacción del trabajo									X	
Entrega del informe final										X

Tabla 5: Cronograma de trabajo realizado

Con una ocupación media de 35 hora semanales de dedicación a la elaboración del trabajo, la suma total de las horas necesarias para la consecución de éste ha sido de **315 horas**.

## 5.1 COSTE DE REDACCIÓN DEL PROYECTO

De acuerdo con la tabla de niveles salariales del documento [11], el salario percibido por el ingeniero encargado de la realización del presente trabajo:

- Para un ingeniero de nivel 1 (Licenciados y titulados 2º y 3º ciclo universitario y Analista) el coste anual es de 23618.8 €.
- Teniendo en cuenta el coste de seguridad social empresa: 32%
- Horas de trabajo máximas anuales según [11] es de 1800 horas.

La tasa horaria final del ingeniero será:

$$Tasa\ horaria = \frac{23618,28 * 1.32}{1800} = 17,32\ \text{€/hora}\ (25)$$

Finalmente se ha tomado como tasa horaria para la realización del mismo de 18 €/hora.

	<i>Tasa horaria</i>	<i>Nº horas empleadas</i>	<i>Coste total</i>
<i>Elaboración del trabajo</i>	18 €/hora	315 horas	5670 €

Tabla 6: Coste de elaboración del trabajo

## 5.2 COSTE DE SERVICIOS FORMATIVOS

Para la realización del presente trabajo ha sido necesario la programación de visitas semanales con carácter meramente formativo al Departamento de Ingeniería de Sistemas y Automática (DISA) de la UPV.

Los servicios prestados han ocasionado un gasto para la realización del proyecto con una tasa horaria de 50 €/hora.

	<i>Tasa horaria</i>	<i>Nº horas empleadas</i>	<i>Coste total</i>
<i>Servicio formativo</i>	50€/hora	18 horas	900 €

Tabla 7: Coste del servicio formativo

### 5.3 COSTE MATERIAL

Los materiales que han sido necesarios para la realización del mismo han sido diversos; Ordenador portátil Asus X555L, licencia de Matlab y acceso a documentación relacionada con la temática del proyecto.

A continuación se muestra un informe detallado del coste total de cada material imputable a la relación del proyecto:

- **Ordenador portátil Asus X555L**

El coste debido al ordenador va a ser en función de la utilización de éste para la elaboración del proyecto.

Con un coste inicial de 599€ y una estimación de vida útil de 3 años, la amortización semanal para este quipo será:

$$\text{Amortización semanal} = \frac{599 \text{ €}}{52 \frac{\text{semanas}}{\text{años}} * 3 \text{ años}} = 3.84 \text{ €/semana} \quad (26)$$

Finalmente el coste del ordenador imputable al trabajo será de:

	<i>Tasa horaria</i>	<i>Nº semanas empleadas</i>	<i>Coste total</i>
<i>Ordenador Asus X555L</i>	3.84€/semana	9 semanas	34.55 €

Tabla 8: Coste amortización ordenador

- **Licencia Matlab**

Siguiendo con la misma metodología que para el material anterior, el coste imputable al proyecto correspondiente al uso del programa Matlab será en función al tiempo de dedicación para el desarrollo del mismo.

El coste por licencia asciende a 2000€ con derecho a uso por un período de 1 año, por tanto la amortización

$$\text{Amortización semanal} = \frac{2000 \text{ €}}{52 \frac{\text{semanas}}{\text{año}} * 1 \text{ año}} = 38,46 \text{ €/semana (27)}$$

El gasto imputable al proyecto derivado del uso de esta herramienta es de:

	<i>Tasa horaria</i>	<i>Nº semanas empleadas</i>	<i>Coste total</i>
<i>Licencia de Matlab</i>	38,46€/semana	9 semanas	346,14 €

Tabla 9: Coste de amortización programa Matlab



- **Acceso a bibliografía**

El coste imputable al proyecto debido al acceso a la documentación necesaria para la realización del mismo es:

	<i>Coste total</i>
<i>[7] Wu, S (2008). "Precision control for high-density and cost-effective hard disk drives" Tesis doctoral. Departamento de Ingeniería Mecánica. Universidad de California, Berkeley.</i>	30 €

*Tabla 10: Coste de acceso a bibliografía*

El resto de la bibliografía utilizada para la redacción del trabajo no imputara coste alguno debido a que es accesible por cortesía de la Universidad Politécnica de Valencia.

## 5.4 RESUMEN DEL PRESUPUESTO

<b>Redacción del proyecto</b>		5670 €
<b>Servicios formativos</b>		900 €
<b>Material</b>	Ordenador Asus X55L	34,55 €
	Licencia Matlab	346,14 €
	Acceso a bibliografía	30 €
	<b>Total Coste Material</b>	<b>410,69 €</b>
<b>Presupuesto de ejecución material</b>		<b>6980,69 €</b>
	6% de gastos generales	418,84€
	13% de beneficio industrial	907,5€
<b>Presupuesto ejecución por contrata</b>		<b>8307,03€</b>
	21% I.V.A	1744,47€
<b>Presupuesto Total</b>		<b>10051,5€</b>

Tabla 11: Resumen del presupuesto

Debido a la naturaleza del proyecto el porcentaje del coste, debido a los gastos generales, será del 6% y el correspondiente al beneficio industrial será del 13%.

**Asciende el presupuesto base de licitación a la expresada cantidad de: 10051,5 € (diez mil cincuenta y uno coma cinco EUROS).**



---

# ANEXOS Y BIBLIOGRAFÍA



## 6. ANEXOS

### 6.1 CÓDIGO PARA TRUETIME

#### CÓDIGO PARA EL SENSOR

##### CÓDIGO DE INICIAZIZACIÓN

```
function sensor_init
ttlNitKernel(1, 0, 'prioFP');

% Create sensor task
Ty = 0.00002; % Slow period
data.iter = 1; % to save the random pattern
data.total_iter = 232*2; % Number of total iterations
offset = 0;
period = Ty;
prio = 4;

% Variables to compute sensor usage
data.avisosens1 = zeros(data.total_iter+1,1); data.avisosens = 0;
data.salidaY(data.iter)=0;
ttCreatePeriodicTask('sens_task', offset, period, prio, 'senscode',data);

% Initialize network
ttCreateInterruptHandler('nw_handler', prio, 'msgRcvSensor');
ttlNitNetwork(3, 'nw_handler');
```

##### CÓDIGO DE EJECUCIÓN

```
function [exectime, data] = senscode(seg, data)

switch seg,
case 1,
    data.w1 = ttAnalogIn(1); % Read process outputs
    fprintf('Medidas del sensor\n');
    fprintf('salida actual\n');
    data.salidaY(data.iter)=data.w1;
```



```
data.ts = ttCurrentTime;
exectime = 0.0000001; %adquisition time

case 2,
a = 0.000001; %Relacionado con el parámetro  $\delta_s$ 
b = 0.000001; %Relacionado con el parámetro  $\sigma_s$ 

if data.iter==1 || data.iter==data.total_iter/2+1 %para muestrear al principio de la
ejecución y cuando cambia la referencia
data.w1_1 = data.w1;
fprintf('Envío del sensor\n');
ttSendMsg(4, data, 70); % send to the interference node 1
data.avisosens = 0;
else

if abs((data.w1-data.w1_1)^2) > 0.001*a*10000+0.001*b*(data.w1)^2*10000
data.w1_1 = data.w1;
fprintf('Envío del sensor\n');
ttSendMsg(4, data, 70); % send to the controller
data.avisosens = 0;

else
data.avisosens1(data.iter) = 1;
data.avisosens = data.avisosens+1; %Acumula iteración para incrementar luego en
slowctrlcode.m (pues no se ejecuta al no enviarle señal)
end

end
fprintf('Iteración de sensor\n');
data.iter

data.iter = data.iter+1;
if data.iter == data.total_iter+1
salidaY=data.salidaY;
suma1 = 0;
suma=0;
N_SC=0;
```



```
avisosens1 = data.avisosens1;
save avisosens avisosens1
save Salidas salidaY
for i = 1:data.total_iter+1
    suma = suma + data.avisosens1(i);

end
fprintf('valor de suma\n');
suma
N_SC=data.total_iter-suma;

suma1 = N_SC/(data.total_iter)*100;

fprintf('El número de envíos entre sensor y controlador es %f \n',N_SC);
fprintf('El canal 1 del sensor presenta una utilización del %f por cien \n',suma1);
save sensores suma1

end
exectime = 0.0000001;
case 3,
    exectime = -1; % finished
end
```



## CÓDIGO PARA EL CONTROLADOR

### CÓDIGO DE INICIAZIZACIÓN

```
function slow_controller_init
```

```
% Initialize TrueTime kernel
```

```
ttInitKernel(1, 0, 'prioFP'); % nbrOfInputs, nbrOfOutputs, fixed priority
```

```
% Create task data (local memory)
```

```
% Slow Controller (PI) definition
```

```
Ty = 0.00002; % Slow period
```

```
data.Ty = Ty;
```

```
data.iter = 1; % for the random pattern
```

```
data.total_iter = 232*2;
```

```
data.u1_1=0;
```

```
data.u1_2=0;
```

```
data.u2_1=0;
```

```
data.u2_2=0;
```

```
data.u3_1=0;
```

```
data.e_1=0;
```

```
data.e_2=0;
```

```
data.e_3=0;
```

```
% Reference to be followed
```

```
referenciaasc = ones(1,data.total_iter/2);
```

```
data.w_ref11 = [referenciaasc ones(1,data.total_iter/2)];
```

```
% Variables to compute sensor usage
```

```
data.avisoctrl1 = zeros(data.total_iter,1); %(234,1);
```

```
data.avisoact = 0;
```

```
data.uu_1=0;
```

```
% Create controller task
```

```
deadline = Ty;
```

```
prio = 3;
```



```
ttCreateTask('slow_ctrl', deadline, prio, 'slowctrlcode',data);
```

```
% Initialize network
```

```
ttCreateInterruptHandler('nw_handler', prio, 'msgRcvSlowCtrl');
```

```
ttlNitNetwork(4, 'nw_handler');
```

## CÓDIGO DE EJECUCIÓN

```
function [exectime, data] = slowctrlcode(seg, data)
```

```
switch seg,
```

```
case 1,
```

```
msg = ttGetMsg;    % Obtain sensor value
```

```
data.w1 = msg.w1;
```

```
data.w1_1 = data.w1;
```

```
data.iter = data.iter + msg.avisosens;
```

```
data.avisoctrl = msg.avisosens;
```

```
data.avisoaact = data.avisoaact + data.avisoctrl;
```

```
fprintf('Iteración en ctrl\n');
```

```
data.iter
```

```
fprintf('avisoctrl\n');
```

```
data.avisoctrl
```

```
fprintf('avisoaact\n');
```

```
data.avisoaact
```

```
%Cálculo del error
```

```
data.w_ref1(data.iter) = data.w_ref11(data.iter)-data.w1;
```

```
data.e(data.iter)=data.w_ref1(data.iter);
```

```
%MULTIFRECUENCIA ENTRELAZADO
```

```
if rem(data.iter,2)==0 %comprueba si el número es par
```

```
if data.iter==2
```

```
data.u1(data.iter)=0;
```

```
data.u2(data.iter)=0;
```



```
data.u3(data.iter)=0.6137*data.u3_1+1.0882*data.e(data.iter)+1.84*data.e_1;

end
if data.iter>2
    data.u1(data.iter)=data.u1_2+0.04484*data.e_2;
    data.u2(data.iter)=data.u2_1;
    data.u3(data.iter)=0.6137*data.u3_1+1.0882*data.e(data.iter)+1.84*data.e_1;
end
    fprintf('numero es par\n');
    data.iter

else
    if data.iter==1
        data.u1(data.iter)=0;
        data.u2(data.iter)=0;
        data.u3(data.iter)=1.0882*data.e(data.iter);
    end
    if data.iter==3
        data.u1(data.iter)=data.u1_1;
        data.u2(data.iter)=0.7218*data.u2_2;
        data.u3(data.iter)=0.6137*data.u3_1+1.0882*data.e(data.iter)+1.84*data.e_1;
    end
    if data.iter>3
        data.u1(data.iter)=data.u1_1;
        data.u2(data.iter)=0.7218*data.u2_2-1.6365*data.e_3;
        data.u3(data.iter)=0.6137*data.u3_1+1.0882*data.e(data.iter)+1.84*data.e_1;
    end
    fprintf('numero es impar\n');
    data.iter
end

data.u(data.iter)=data.u1(data.iter)+data.u2(data.iter)+data.u3(data.iter);
data.uu=data.u(data.iter);

data.e_3=data.e_2;
data.e_2=data.e_1;
data.e_1=data.e(data.iter);
```



```
data.u1_2=data.u1_1;
data.u1_1=data.u1(data.iter);

data.u2_2=data.u2_1;
data.u2_1=data.u2(data.iter);

data.u3_1=data.u3(data.iter);

data.ts = msg.ts;
exectime = 0.0000001; %computation time

case 2,
    c =1.33; %Relacionado con el parámetro  $\delta_c$ 
    d =1.33; % Relacionado con el parámetro  $\sigma_c$ 

if data.iter>1
    if abs((data.uu-data.uu_1))^2 >= 0.01*c+0.01*d*(data.uu)^2% || abs(data.w_ref2) >=

        ttSendMsg(2, data, 70);
        data.avisoact = 0;
        fprintf('Envío datos de control\n');
        % Si envío marco con un 1
        data.avisoctrl1(data.iter) = 1;
        data.avisoctrl1(data.iter)

    else
        fprintf('NO envío datos de control\n');
        data.avisoact = data.avisoact + 1;
    end
else
    data.uu_1 = data.uu;
    ttSendMsg(2, data, 70); % envío al actuador
    data.avisoact = 0;
    % Si envío marco con un 1
    fprintf('Envío datos de control\n');
    data.avisoctrl1(data.iter) = 1;
    data.avisoctrl1(data.iter)
```



end

```
fprintf('avisoctrl\n');  
data.avisoctrl
```

```
fprintf('avisocact\n');  
data.avisocact
```

```
data.iter = data.iter + 1;  
if data.iter == data.total_iter+1
```

```
avisoctrl1 = data.avisoctrl1;  
    eIAE_tb=data.e;  
    save avisoctrl avisoctrl1  
    save errorIAE_tb eIAE_tb
```

```
suma5 = 0;  
suma6 = 0;  
for i = 1:data.total_iter  
    suma5 = suma5 + data.avisoctrl1(i);  
end
```

```
suma6 = suma5/data.total_iter*100;  
fprintf('El número de envíos entre controlador y actuador es %f \n',suma5);  
fprintf('El canal 1 del controlador presenta una utilización del %f por cien \n',suma6);  
save controladores suma5
```

```
end
```

```
exectime = -1;
```

```
end
```



## CÓDIGO PARA EL ACTUADOR

### CÓDIGO DE INICIALIZACIÓN

#### FUNCTION ACTUATOR\_INIT

% Initialize TrueTime kernel

```
ttInitKernel(0, 1, 'prioFP'); % nbrOfInputs, nbrOfOutputs, fixed priority
```

% Create task data (local memory)

```
data.nca = 1; % Number of control actions inside the metaperiod
```

```
data.N = 1; % Multiplicity
```

```
data.Ty = 0.00002; % Slow period
```

```
data.Tu = data.Ty/data.N; % Fast period
```

```
data.factor = 0; % Factor in order to fix a time scale
```

```
data.iter = 1;
```

```
data.total_iter = 232*2; % Number of total iterations
```

% Create actuator task

```
deadline = 100;
```

```
prio = 1;
```

```
ttCreateTask('act_task', deadline, prio, 'actcode', data);
```

% Initialize network

```
ttCreateInterruptHandler('nw_handler', prio, 'msgRcvActuator');
```

```
ttInitNetwork(2, 'nw_handler'); % node #2 in the network
```

### CÓDIGO DE EJECUCIÓN

```
function [exectime, data] = actcode(seg, data)
```

```
switch seg,
```

```
case 1,
```



%Se recibe el mensaje y se actualizan los valores del error en caso de  
%ser necesario.

%Si no se ha superado el umbral en algún gdl, se mantiene el error de la  
%iteración anterior.

```
if data.nca==1
```

```
    msg = ttGetMsg;
```

```
    data.u=msg.uu;
```

```
    data.u
```

data.factor = data.factor + msg.avisoact; %Se aumenta el valor de data.factor y data.iter  
si no se ha producido envío desde slowctrl.

```
    data.iter = data.iter + msg.avisoact;
```

```
    fprintf('Iteración de actuador\n');
```

```
    data.iter
```

```
end
```

```
exectime = 0.0000001;
```

```
case 2,
```

```
ttAnalogOut(1, data.u)
```

```
    data.iter = data.iter + 1;
```

```
    exectime = 0.0000001;
```

```
case 3,
```

```
    if data.nca<data.N
```

```
        ttSleepUntil((data.factor*data.N+data.nca)*data.Tu,'act_task');
```

```
        data.nca = data.nca+1;
```

```
        ttSetNextSegment(2);
```

```
        exectime = 0.0000001;
```

```
    else
```

```
        data.nca = 1;
```

```
        data.factor = data.factor + 1;
```

```
        exectime = -1;
```



```
end  
fprintf('factor\n');  
data.factor  
end
```



## 7. BIBLIOGRAFÍA

- [1] Salt, J; Albertos,P (2005). Model-Based Multirate Controllers Design. IEEE Transactions on control systems technology, 13(6), 988-997.
- [2] Ding,J; Marcassa,F; Wu,S.C; Tomizuka,M (2006).Multirate control for computation saving. IEEE Transactions on control systems technology,14, 165-169.
- [3] Wu, S; Tomizuka, M (2003).Multi-rate Digital Control with Interlacing and its application to Hard Disk Drive Servo. Proc. of the American Control Conference, 4347-4352.
- [4] Dormido, S; Sánchez, J; Kofman, E (2008). Muestreo, control y comunicación basados en eventos. Revista Iberoamericana de Automática e Informática Industrial. 5(1), 5-26.
- [5] Sanchez,J; Guarnes,M.A; Dormido,S (2009). On the application of different Event-Based sampling strategies to the control of a simple Industrial Process. Sensors. 9, 6795-6818.
- [6] Heemels, M.; Johansson, K.H; Tabuada,P (2015). Event-triggered and Self-triggered Control. Encyclopedia of Systems and Control, 384-391.
- [7] Wu, S (2008). "Precision control for high-density and cost-effective hard disk drives" Tesis doctoral. Departamento de Ingeniería Mecánica. Universidad de California, Berkeley. 30€
- [8] Ogata, K.(1987). Discrete-time Control Systems, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 70€
- [9] Yang, T.C. (2006). Networked control system: a brief survey, IEE Proc. Control Theory Appl. 153 (4) 403–412
- [10] Donkers M., Heemels W. (2012). Output-based event-triggered control with guaranteed-gain and improved and decentralized event-triggering. Automatic Control, IEEE Transactions on, 57(6), 1362–1376,
- [11] España. Resolución de 9 de Octubre de 2013, de la Dirección General de empleo. [Internet] Boletín Oficial del Estado, 25 de Octubre de 2013, num. 256, pp 86811-86838.
- [12] Andersson,M; Henriksson, D; Cervin,A (2005). Truetime 1.3–Reference Manual. Recuperado el 24 de Mayo de 2016, de <http://www.control.lth.se/documents/2005/and+05tt.pdf>