

VI Jornadas CEA de enseñanza a través de Internet-Web de la Ingeniería de Sistemas y Automática

EIWISA'10

León, 2-4 de Junio 2010



CEA
I F A C



Título: Actas de las VI Jornadas de Trabajo Enseñanza vía Internet / Web de la Ingeniería de Sistemas y Automática

León, 2-4 de Junio 2010

COMITÉ ORGANIZADOR

Manuel Domínguez González

Perfecto Reguera Acevedo

Juan José Fuertes Martínez

Antonio Morán Álvarez

Serafín Alonso Castro

Diego Fernández Durán

Roberto García Valencia

Oscar Reinoso García

Luis Miguel Jiménez García

ISBN-10: 10/38430 -- ISBN-13: 978-84-693-1779-2

D.L.: LE-940-2010

© del texto: los autores de cada ponencia

© de ilustraciones y tablas: los autores de cada ponencia

© de la edición: J.M. Sánchez Moreno

O. Reinoso García

M. Domínguez González

Printed in Spain
Imprime: LOGOTÓPICO:comunicación positiva
www.logotopico.es

Reservados todos los derechos. Ni la totalidad ni parte de este libro puede reproducirse o transmitirse por ningún procedimiento electrónico o mecánico, incluyendo fotocopia, grabación magnética o cualquier almacenamiento de información o sistema de reproducción, sin permiso previo y por escrito de los titulares del Copyright.

Presentación

Las I Jornadas EIWISA tuvieron lugar en la Universidad Politécnica de Valencia, las II Jornadas EIWISA, en la Universidad Nacional de Educación a Distancia, las III Jornadas EIWISA, en la Universidad de Alicante. Las IV Jornadas EIWISA ya se enmarcaron dentro del CEDI'05 celebrado en Granada. En 2007, las V Jornadas se integraron de nuevo dentro del Congreso Español de Informática CEDI'07. Desde sus inicios, en todas las celebradas se pudo apreciar el alto grado de inquietud que despertaban las nuevas tecnologías de la información en el campo de la Ingeniería de Sistemas y Automática, el buen nivel en que se encontraba la docencia española en relación con estos temas y el deseo generalizado, tanto de mantener foros abiertos de debate, como de la integración de las tecnologías de la información y la innovación tecnológica al ámbito del proceso enseñanza-aprendizaje en general.

Aunque las Jornadas están centradas fundamentalmente en las aplicaciones de innovación tecnológico-educativa en el campo de la Ingeniería de Sistemas y Automática, las aportaciones esperadas en las mismas están abiertas a otros campos de conocimiento, en especial, aquellos relacionados con el ámbito de la ingeniería. En esta ocasión, las ponencias se han dividido en 4 grandes áreas temáticas:

- Laboratorios Virtuales
- Nuevas técnicas educativas en control
- Laboratorios remotos
- EJS en laboratorios vía web.

Dicho esto, no queda más que agradecer la confianza depositada por el Comité Español de Automática (CEA) en el grupo SUPPRESS de investigación de la Universidad de León para la organización de estas VI Jornadas EIWISA, y también agradecer el apoyo de la Junta de Castilla y León y la propia Universidad de León, sin el cual esta iniciativa difícilmente hubiera podido ver la luz.

EL COMITÉ ORGANIZADOR

Índice de contenidos (I)

PONENCIAS EJS EN LABORATORIOS VÍA WEB

Simulación en Easy Java de Prácticas de Control Realizadas con Equipo Feedback MS-150 005

Autores: Manuel Domínguez, Juan José Fuertes, Perfecto Reguera, Antonio Moran, Serafín Alonso y Diego Fernandez

Diseño e implementación de una biblioteca gráfica 3D para Easy Java Simulations 014

Autores: Carlos A. Jara, Francisco Esquembre, Francisco A. Candelas, Fernando Torres y Sebastián Dormido

Laboratorio Remoto para Prácticas de Control en Espacio de Estado mediante EJS 020

Autores: Luis M. Jiménez, Oscar Reinoso, Miguel Juliá, Arturo Gil y José M. Marín

Librería Digital de EJS para la Enseñanza en Ingeniería de Control y Ciencias Físicas 026

Autores: Luis de la Torre, Héctor Vargas, José Sánchez, Sebastián Dormido y Francisco Esquembre

LABORATORIOS REMOTOS

Sistema de teleoperación cooperativa con retorno sensorial de fuerza 032

Autores: D. Bustelo , R. Ramos, E. Delgado, M. Díaz-Cacho y A. Barreiro

Simulaciones robóticas avanzadas mediante la plataforma EJS+EjsRL 039

Autores: Carlos A. Jara, Francisco A. Candelas y Fernando Torres

Laboratorio Virtual y Remoto para la Enseñanza de Robótica Paralela 046

Autores: Francisco J. Martínez, Ramón González, Francisco Rodríguez y José Luis Guzmán

Desarrollo de Herramientas y Técnicas de Realidad Aumentada para la Mejora de Laboratorios Remotos: Laboratorio Remoto Aumentado 054

Autores: Andrés Mejías Borrero, José Manuel Andújar Márquez y Marco A. Márquez Sánchez

Índice de contenidos (II)

LABORATORIOS VIRTUALES

Herramienta interactiva para el aprendizaje del control en modo de deslizamiento en R^2 062

Autores: Niliana Carrero Candelas, Ramón Costa-Castelló, Sebastián Dormido y Enric Fossas

Laboratorio De Control Virtual068

Autores: Fco. Javier García y Eduardo J. Moya

Laboratorio virtual para la exploración de entornos mediante un conjunto de robots móviles073

Autores: O. Reinoso, M. Juliá, L. Payá, L.M. Jiménez y D. Úbeda

NUEVAS TÉCNICAS EDUCATIVAS EN CONTROL

Un Nuevo Enfoque para el Diseño de Sistemas Tutores Inteligentes con Aprendizaje079

Autores: V. Amela, J. L. Díez y M. Vallés

Aplicación de Tecnologías Semánticas y de la Web 2.0 para la Representación, Recuperación e Interacción con Ejercicios de Control085

Autores: Isaías García Rodríguez, Carmen Benavides Cuéllar, Héctor Aláiz Moretón, Francisco J. Rodríguez Sedano y Ángel Alonso Álvarez

Autocorrección vía web para asignaturas de ingeniería basadas en Matlab 092

Autores: Fabio Gómez-Estern y Manuel Ruiz Arahal

Estudio interactivo de los conceptos fundamentales del control mediante objetos de aprendizaje 099

Autores: José Luis Guzmán, Ramón Costa-Castelló, Sebastian Dormido y Manuel Berenguel

FTPack: un framework para la programación de sistemas físicos educativos basados en kits Fischertechnik™106

Autores: Jesús Salido, Oscar Déniz, M^a Gloria Bueno y M^a del Milagro Fernández

Enseñando EJS a través de juegos113

Autores: Javier Vegas Regidor, José María Pérez Ramos y Eladio Sanz García

Laboratorio Remoto para Prácticas de Control en Espacio de Estado mediante EJS

Luis M. Jiménez, Oscar Reinoso, Miguel Juliá, Arturo Gil, José M. Marín
 luis.jimenez@umh.es, o.reinoso@umh.es, mjulia@umh.es, arturo.gil@umh.es, jmarin@umh.es
 Dpto. de Ingeniería de Sistemas Industriales, Univ. Miguel Hernández
 Avda. Universidad s/n 03202 Elche (Alicante)

Abstract- En este artículo se presenta la aplicación de un laboratorio remoto para la docencia práctica de teoría de control, presentando un ejemplo de estudio de las técnicas de diseño en espacio de estado. El laboratorio basado *EJS*, y que forma parte de la plataforma colaborativa *AutomatL@bs*, permite el acceso remoto a una planta física a través de Internet. Se describe la implementación del laboratorio remoto y virtual, la integración de la simulación con la ejecución en tiempo real mediante Matlab/Simulink, y se presta una especial atención a la integración de recursos docentes para mejorar la docencia práctica en las asignaturas de control.

I. INTRODUCCIÓN

La docencia en el campo de la teoría de control ha experimentado en los últimos años la incorporación de nuevas tecnologías que permiten al alumno acceder a través de Internet a una inmensa cantidad de información y recursos, modificando paulatinamente el esquema docente tradicional y mejorando la calidad de los procesos de enseñanza y aprendizaje. El alumno puede no solo acceder a una enorme variedad de documentación y experiencias, contrastando y seleccionando las más adecuadas en la materia específica estudiada; sino que, así mismo, se ponen a su disposición recursos normalmente escasos y poco accesibles para la docencia práctica necesaria en la enseñanza del control [1].

En enseñanzas técnicas y más concretamente en la disciplina de control, además de los contenidos teóricos es fundamental la realización de prácticas con equipos reales de forma que el alumno asimile de forma adecuada los conceptos teóricos. Los laboratorios presenciales son la herramienta tradicional en la docencia práctica, pero están sometidos a una serie de factores y limitaciones:

- El alto coste que suponen los equipos de prácticas que limita la variabilidad de sistemas físicos y el número de puestos para trabajar en pequeños grupos.
- El elevado número de alumnos normalmente presentes en las aulas, lo cual eleva el tiempo necesario para que el alumno complete las tareas y la reducción consiguiente en los contenidos que el alumno puede experimentar.
- La limitada disponibilidad temporal de laboratorios que condiciona el número de experimentos a realizar a lo largo del curso.

Por este motivo desde principios de los 90 educadores en ingeniería han demostrado la capacidad de realizar experimentos en laboratorios docentes mediante el uso de Internet y tecnologías Web. Así, la educación a distancia mediante el uso de laboratorios a través de Internet es un tema de total actualidad como consecuencia del enorme potencial que supone para mejorar la calidad del proceso enseñanza-aprendizaje [2][3][4]. Los laboratorios a distancia pueden dividirse en dos clases: laboratorios virtuales y laboratorios remotos [5]. En los primeros, los estudiantes pueden ejecutar simulaciones remotas junto con posibles animaciones. En los segundos, los estudiantes hacen uso remoto de los equipos reales disponibles en los laboratorios.

Sin duda es este último tipo de laboratorio, con la realización de prácticas con sistemas físicos reales, el que permite al estudiante adquirir y afianzar los conocimientos de una forma más eficiente que al realizar únicamente experimentos simulados. A través de estos laboratorios remotos, los usuarios pueden realizar y planificar experimentos, analizar los datos experimentales alcanzados como si estuvieran físicamente presentes en el laboratorio en cualquier momento del día [4].

En este sentido, en los últimos años se han propuesto diferentes laboratorios de sistemas de control para la realización de prácticas mediante acceso remoto [5][6][7][8]. La mayoría de estos laboratorios utilizan plataformas específicas de desarrollo dando lugar a una disparidad de implementaciones. Es de destacar la herramienta para simulación *EJS (Easy Java Simulations)* [9], que permite implementar de forma sencilla laboratorios virtuales y puede enlazarse con otras plataformas de control de sistemas físicos en laboratorios remotos [8][10]. El laboratorio propuesto en este trabajo está basado en la herramienta *EJS* y en la plataforma *Matlab/ Simulink* [7] que constituye una herramienta fiable y ampliamente utilizada para implementar esquemas de control sobre sistemas físicos reales.

Destacar por último la plataforma *AutomatL@bs* [10] supone proyecto pionero para construir una red de laboratorios remotos distribuida, que permite unir los recursos de varias Universidades para la docencia en control automático. Ofrece una plataforma abierta para crear laboratorios remotos y virtuales con una interfaz unificada al alumno independiente del sistema utilizado, potenciando

especialmente las características colaborativas aplicadas al proceso de aprendizaje.

El resto del artículo presenta los aspectos fundamentales del laboratorio remoto propuesto, describiendo en la sección 2 más en detalle la herramienta *EJS* y la estructura de la aplicación desarrollada. En la sección 3 se presenta una práctica completa aplicada a la docencia en control en espacio de estado utilizando la plataforma propuesta. Finalmente se presentan las principales conclusiones y resultados.

II. DESCRIPCIÓN DEL LABORATORIO REMOTO

El laboratorio propuesto (Fig. 1) está basado en la experiencia previa de los autores en laboratorios remotos [7] y en la colaboración con otras universidades en el proyecto *AutomatL@bs* [12], que abarca a siete universidades implicadas en la docencia de asignaturas de control utilizando laboratorios remotos. El principal objetivo del proyecto ha sido mejorar el acceso a sistemas físicos distribuidos en laboratorios remotos, eliminando las limitaciones impuestas por la asignación de tiempo en los laboratorios y posibilitando el acceso a una enorme variedad de sistemas físicos gracias al uso compartido de equipos entre Universidades.

El laboratorio remoto permite al alumno acceder a través de Internet a un sistema físico que es controlado desde un servidor (figura 1). El sistema físico presentado es un motor de corriente continua, pudiendo el alumno configurar y manejar todos los parámetros de la ejecución del sistema de control monitorizando las variables de sistema.

El laboratorio remoto está construido sobre tres componentes principales:

- *Easy Java Simulations (EJS)* [9]. Es una plataforma software que permite implementar simulaciones gráficas avanzadas actuando tanto en laboratorios virtuales como establecer la conexión con sistemas físicos reales en un laboratorio remoto. Es una

herramienta sencilla que ayuda a no programadores a crear simulaciones interactivas en Java, habitualmente con fines de enseñanza o aprendizaje

- Sistema de control en tiempo real implementado sobre un servidor basado en *Matlab/Simulink* y la toolbox *Real-Time Windows Target* [10]. Los algoritmos de control se implementan como esquemas Simulink precompilados conectados con *EJS* mediante un interfaz software en red.
- El tercer componente es *eMersion* [11], desarrollado por el EPFL de Lausanne, implementa el entorno docente colaborativo para que el estudiante pueda acceder al material, planificar sus prácticas, acceder a los recursos compartidos y enviar y compartir los resultados con profesores y otros estudiantes.

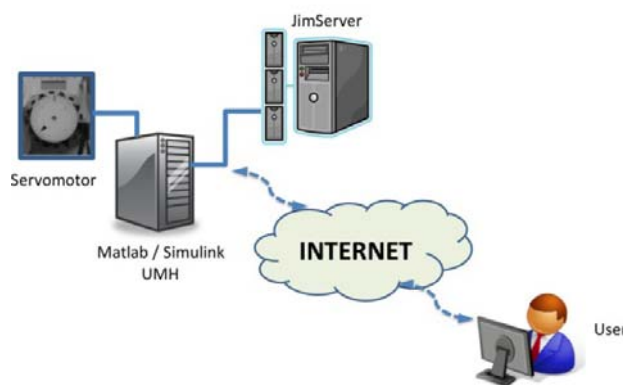


Fig. 1. Laboratorio remoto para prácticas de control Recolab

La aplicación basada en *EJS* dispone de dos modos de acceder al laboratorio: *simulación virtual* y *ejecución real* sobre el sistema físico de algoritmos de control. La ejecución sobre el sistema físico real utiliza *Matlab/Simulink* junto a la *Real-Time Workshop toolbox* para conseguir que la ejecución se realice en tiempo-real sin los retardos impuestos por la conexión a Internet. La conexión entre *EJS* y *Matlab* es gestionada por el módulo denominado *JimServer*. La figura 2 muestra esquemáticamente esta comunicación que permite la modificación de los parámetros de control, la ejecución de

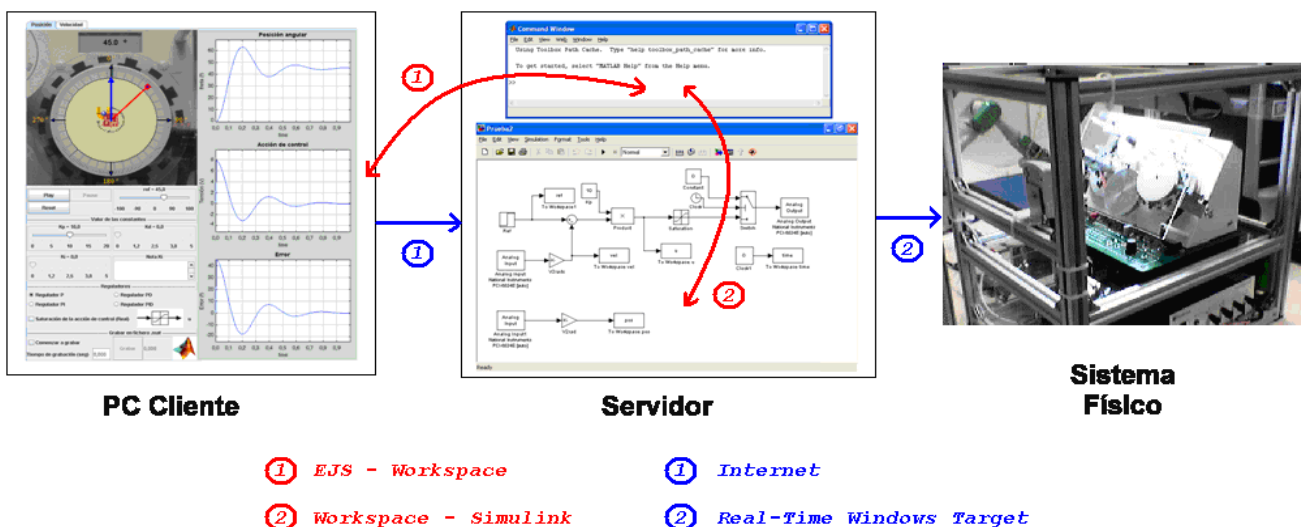


Fig. 2. Arquitectura laboratorio remoto: comunicación *EJS/Matlab*

comandos, y el acceso a las variables de ejecución de los esquemas de control implementados en *Matlab/Simulink*. El módulo formado por *JimServer/Matlab* puede residir en diferentes computadores permitiendo múltiples sistemas físicos y diferentes interfases para su implementación (*Matlab/Labview/C*) [13] conectados a un mismo laboratorio *EJS*.

A. Sistema físico: motor de corriente continua

El sistema físico conectado actualmente al laboratorio remoto es un motor de corriente continua modelo Feedback 33-002 (Fig. 3). Esta maqueta de laboratorio está integrada por un motor de corriente continua, amplificador de potencia, tacómetro, encoder absoluto y diferencial, potenciómetro de posición y freno magnético. Los sensores de velocidad están conectados directamente al eje del motor, mientras que los sensores de posición angular están conectados a través de una reductora.

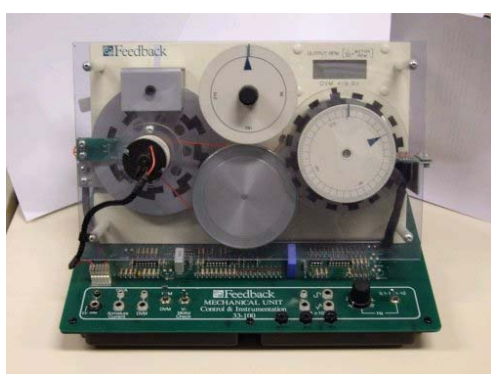


Fig. 3. Motor de corriente continua Feedback

El motor está conectado al computador que ejecuta el módulo *JimServer/Matlab* a través de una tarjeta de entrada salidas National Instruments *NI 6024E* utilizada, no solo para cerrar el bucle de control sobre el sistema físico, sino para controlar asimismo otros elementos del laboratorio: iluminación, supervisión, control del freno magnético, ...).

En la sección 3 se describe con más detalle la interfaz de usuario de la aplicación *EJS* sobre un ejemplo práctico de estudio de sistemas de control en espacio de estado.

B. Conexión Simulación *EJS*- *Matlab/Simulink*

EJS ejecuta las simulaciones que se realizan localmente en PC del estudiante y permiten de forma sencilla visualizar y comparar el comportamiento de un sistema físico a partir de las ecuaciones que lo modelan. La ejecución de los algoritmos de control en tiempo real se realiza mediante esquemas *Simulink* precompilados en un servidor ubicado en el laboratorio remoto. Estos esquemas están parametrizados, pudiendo modificar los parámetros de ejecución desde la aplicación *Matlab*. Se han desarrollado múltiples esquemas *Simulink* y un conjunto de aplicaciones en lenguaje *Matlab* que permiten controlar la ejecución y obtención de los resultados con los datos de evolución de las variables del proceso.

La conexión con la interfaz de usuario y simulación se realiza mediante un servidor (*JimServer*) que se ejecuta en el laboratorio remoto y que permite, mediante una interfaz de *sockets*, enviar directamente comandos *Matlab* al controlador del sistema físico. Estos comandos permiten ejecutar los scripts de ejecución de los esquemas *Simulink*, su sincronización, y la recuperación de los datos de evolución de las variables del proceso.

III. EJEMPLO: PRÁCTICAS DE DISEÑO DE REGULADORES EN ESPACIO DE ESTADOS

La idea básica del Control en el Espacio de Estados es disponer de la máxima información posible del sistema para, de esta forma, obtener un control más preciso y eficiente; para ello, se busca conocer el comportamiento interno de los sistemas a partir de las variables que influyen en su dinámica surgiendo, de este modo, el concepto de *estado* del sistema.

Entre las prácticas que se plantean a los alumnos utilizando el laboratorio remoto se incluye el control por realimentación de estado de la salida de un motor de corriente continua controlado en posición. En este sistema, tanto la velocidad como la posición son medibles y constituyen el vector de estado que define el comportamiento del sistema.

Esta práctica tiene como objetivos docentes que el alumno sea capaz de identificar un sistema de primer orden con integrador (sistema de tipo 1), asimilar los métodos de diseño de servosistemas combinando la metodología de diseño de realimentación de estado y realimentación de la salida para implementar sistemas de seguimiento. Por otro lado, se plantea un método de diseño específico de servosistemas cuando el proceso a controlar posee un integrador en bucle abierto, es decir, cuando el proceso es de tipo uno.

El alumno debe trabajar inicialmente en *modo simulación* y, posteriormente una vez comprobado el funcionamiento del sistema de control propuesto testarlo en *modo remoto*, contrastando los resultados.

A. Modelo del sistema

El modelo experimental que se va a utilizar en el laboratorio se muestra en la figura 4, los parámetros del modelo se muestran en la tabla I. El motor está conectado a un computador mediante una tarjeta de entradas salidas que implementa los convertidores Analógico/Digital y Digital/Analógico indicados en el mismo. La acción de control (U_e) actúa sobre el voltaje del inducido a través del amplificador integrado en el propio servomotor. Entre el eje del motor y la carga existe una reductora para incrementar el par motor. El sensor de velocidad (tacómetro) está situado antes de la reductora, el sensor de posición (potenciómetro) está situado a la salida de la reductora. $f(u)$ modela el comportamiento circular del potenciómetro multivuelta (presentará una respuesta en diente de sierra volviendo al valor inicial al completar cada vuelta).

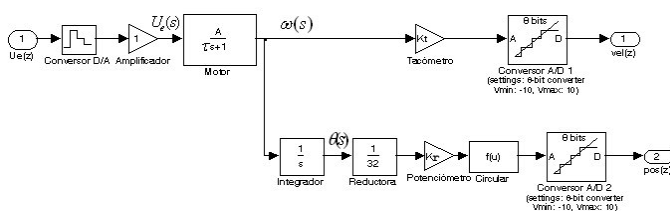


Fig. 4. Diagrama de bloques del modelo del servomotor

TABLA I

PARÁMETROS DEL MODELO DEL SERVOMOTOR

A	Ganancia del motor
τ	Constante de tiempo del motor
K_t	Ganancia del tacómetro
K_r	Ganancia del sensor de posición
$K_1 = A * K_t$	Ganancia del sistema en velocidad
$K_2 = A * 1/32 * K_r$	Ganancia del sistema en posición

Durante la identificación el alumno debe calcular las ganancias del sistema en velocidad y posición K_1, K_2 . Aunque el alumno puede probar métodos avanzados de identificación (RLS), en esta etapa se propone que el alumno aplique diferentes escalones al motor y obtenga las curvas de respuesta de velocidad y posición. A partir de la respuesta del sistema en velocidad (tacómetro) el alumno calculará la ganancia K_1 (régimen permanente) y la constante de tiempo τ del motor (tiempo en que se alcanza el 63% del valor final).

A partir de la pendiente de la respuesta del potenciómetro de posición se calculará la ganancia K_2 que incluye la reductora y la calibración del potenciómetro. El valor de la ganancia del motor A se puede obtener a partir de los datos de calibración del tacómetro.

B. Control por realimentación de estado con seguimiento de la salida

En el desarrollo de este apartado se estudia la forma de obtener un sistema con los polos en las posiciones deseadas y error de posición nulo por medio de la realimentación de la salida. Para el cálculo del regulador, hay que diseñar la matriz de realimentación del estado con objeto de fijar el comportamiento dinámico del sistema, de forma que se asignen directamente todos los polos de la parte controlable y observable del proceso. Puesto que se pretende diseñar un sistema de seguimiento de la salida, la matriz de realimentación del estado no puede calcularse de manera independiente al factor integral que se pretende introducir para eliminar el error en régimen permanente.

Los pasos que seguirá el alumno para diseñar el control en posición del servosistema son los siguientes:

- Obtener el modelo de estado discretizado del servomotor a partir de la identificación del sistema. El modelo utilizará variables de estado medidas con sentido físico (posición y velocidad).

- Utilizando el modelo discreto y considerando las especificaciones deseadas, se diseñará la matriz de realimentación del estado correspondiente mediante una técnica aplicable únicamente a servosistemas de tipo 1.
- Se comprobará, mediante una simulación, que el comportamiento del sistema con el regulador seleccionado es el deseado.
- Por último, tras comprobar que se cumplen las especificaciones de diseño y se alcanza la posición deseada en el sistema simulado, se implementará el regulador en el sistema real y se comprobará, con conexión remota, que el comportamiento del sistema con el regulador seleccionado es el deseado.

El sistema a controlar es modelado por las siguientes ecuaciones de estado y de salida (Fig. 5) (Ecuación 1):

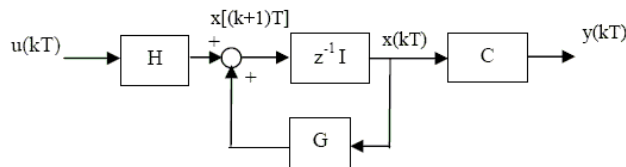


Fig. 5. Modelo de Estado Discreto equivalente

$$\begin{aligned} \bar{x}(k+1) &= G \cdot \bar{x}(k) + H \cdot u(k) \\ y(k) &= C \cdot \bar{x}(k) \end{aligned} \quad (1)$$

Dado que la salida del sistema es igual a la primera variable de estado (en el caso del motor, la posición), y que dicha variable de estado es la salida del integrador, se cumple que:

$$y(k) = x_1(k) \quad (2)$$

En los sistemas de tipo 1 (motor controlado en posición) el error ante escalón en régimen permanente es nulo si la salida está realimentada negativamente. Por ello se plantea un esquema de control por realimentación de estado para ubicar los polos del sistema y realimentación de la salida para eliminar el error en régimen permanente (Fig 6).

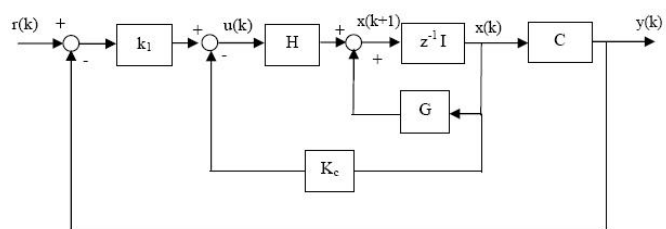


Fig. 6. Esquema de control por realimentación del estado y la salida

Donde la matriz de realimentación del estado Kc tiene la siguiente estructura (ecuación 3):

$$K_c = [0 \quad k_2] \quad (3)$$

La estructura de la realimentación de estado preserva el tipo del sistema garantizando que el error en régimen permanente tenderá a cero. Reduciendo el diagrama de bloques se obtiene la acción de control equivalente (ecuación 4):

$$u(k) = -Kx(k) + k_1 r(k), \quad K = [k_1 \quad k_2] \quad (4)$$

Con esta acción de control la dinámica del sistema realimentado y del error se modela mediante la ecuación 5. El vector de error en régimen permanente tiende a cero desde cualquier condición inicial siempre que diseñemos un regulador con un comportamiento estable para el sistema realimentado.

$$\begin{aligned} \bar{x}(k+1) &= [G - H \cdot K] \cdot \bar{x}(k) + k_1 \cdot r(k) \\ \bar{e}(k+1) &= [G - H \cdot K] \cdot \bar{e}(k) \\ \bar{e}(k) &= \bar{x}(k) - \bar{x}(\infty) \end{aligned} \quad (5)$$

Utilizando las herramientas de diseño mediante ubicación de polos se calculará la matriz de ganancias de realimentación K a partir de las especificaciones de diseño (régimen transitorio).

Para ajustar la matriz de ganancias de realimentación de estado se calcula la matriz de transformación a la forma canónica controlable T_c a partir de la matriz de controlabilidad (ecuación 6). A partir de los coeficientes del polinomio característico del sistema (ecuación 7) y los coeficientes del polinomio característico deseado, fijado a partir de las especificaciones (ecuación 8) podemos obtener la matriz de realimentación de estado (ecuación 9)

$$Q = [H \quad GH] = \begin{bmatrix} e_1^T \\ e_2^T \end{bmatrix}, \quad T_c^{-1} = \begin{bmatrix} e_2^T \\ e_1^T \cdot G \end{bmatrix} \quad (6)$$

$$P(z) = |z \cdot I - G| = |s \cdot I - \tilde{G}| = z^2 + a_1 \cdot z + a_0 \quad (7)$$

$$\phi(z) = |z \cdot I - G + H \cdot K| = z^2 + \alpha_1 \cdot z + \alpha_0 \quad (8)$$

$$\tilde{K} = [\alpha_1 - a_1, \quad \alpha_0 - a_0], \quad K = \tilde{K} \cdot T_c^{-1} \quad (9)$$

A partir de las dos componentes de K (ecuación 4) obtendremos la matrices de ganancias para la realimentación de estado K_c y de la salida k_1 .

C. Simulación del sistema de control en EJS

La figuras 7, 8 y 9 muestran la interfaz de usuario del laboratorio desarrollado sobre EJS. Está dividido en tres ventanas principales:

- *Ventana de simulación:* contiene los controles, representación del sistema, imágenes y gráficos de simulación y muestreo para las variables del proceso (Fig. 7)
- *Ventana eMersion:* permite acceder a los recursos docentes colaborativos como manuales, protocolos, foros,... (Fig. 8)
- *Ventana e-Journal:* da acceso al almacén de informes y datos de simulación/ejecución (Fig. 8)

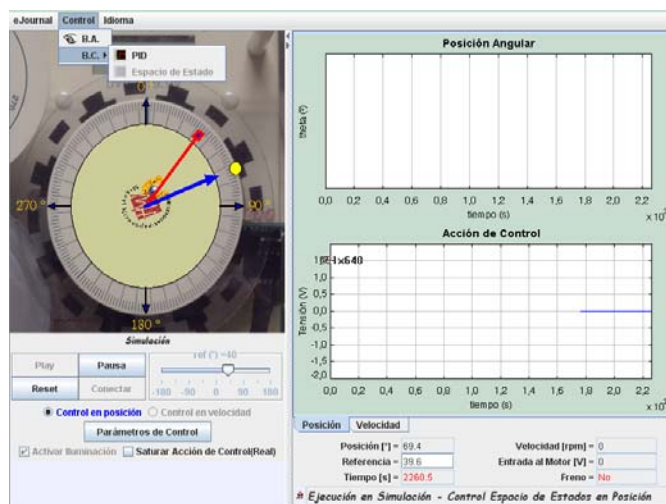


Fig. 7. Interfaz de usuario aplicación EJS: ventana simulación



Fig. 8. Interfaz de usuario aplicación EJS: ventana eMersion



Fig. 9. Interfaz de usuario aplicación EJS: ventana e-Journal

El laboratorio dispone de varios modos de visualización:

- *Modo Simulación:* en este modo el sistema actúa de forma local sin necesidad de conexión remota con el servidor. Utiliza el modelo matemático del proceso simulando la ejecución de los algoritmos de control. Los parámetros y referencias se muestra superpuestos sobre una imagen estática del motor
- *Modo Remoto (con conexión de video):* el motor real localizado en el laboratorio remoto se muestra mediante una cámara Web que transfiere las imágenes al equipo local requiriendo de una conexión a Internet con un ancho de banda adecuado. La ejecución de los algoritmos de control se realiza sobre el sistema real y no sobre el modelo matemático.
- *Modo Remoto (utilizando realidad aumentada):* este modo añade a la visualización y ejecución sobre el sistema físico remoto la superposición de la simulación de la posición calculada a partir del

modelo matemático, pudiendo comprobar las discrepancias.

El acceso a los modos remotos se realiza a través del botón “Conectar” presente en la ventana principal y requiere previamente haber realizado una reserva horaria y haber cumplimentado previamente las tareas de diseño en modo simulación, que será validada por el profesor.

A la derecha de la ventana principal está ubicada la ventana de evolución (Fig. 7) que muestra las variables del proceso, en el ejemplo la posición del eje del motor o la acción de control aplicada.

Los parámetros de control pueden ser ajustados manualmente a través de una ventana de diálogo (Fig. 10)

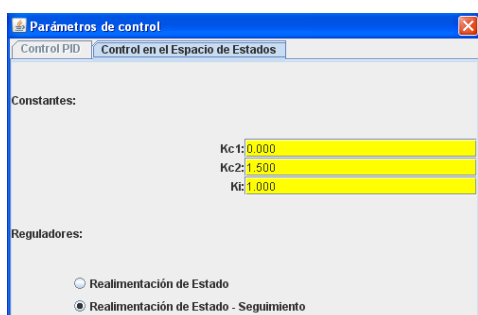


Fig. 10. Ventana de ajuste de parámetros

IV. CONCLUSIONES

En este trabajo se ha presentado una plataforma para la realización de prácticas de control remotas. El sistema desarrollado ha sido integrado dentro de una red de laboratorios remotos *AutomatL@bs* que permiten al alumno acceder a una amplia variedad de sistemas físicos bajo un entorno unificado. El laboratorio ha sido desarrollado sobre la herramienta de simulación *EJS* que permite de forma sencilla implementar laboratorios virtuales y su conexión con plataformas de ejecución real en sistemas físicos. La implementación del control en tiempo real se ha realizado en *Matlab/Simulink* mediante un mecanismo de comunicación remota con la aplicación del cliente.

La aplicación docente del sistema propuesto se muestra sobre un ejemplo de estudio de sistemas de control en espacio de estado describiendo en detalle la realización de las tareas asignadas utilizando la herramienta propuesta.

Los resultados en términos de participación de estudiantes en los dos últimos años académicos, han validado el uso del laboratorio remoto para asimilar conocimientos y habilidades en el campo de la ingeniería de control. Este tipo de laboratorios en los que el alumno puede interactuar con equipos reales sin las limitaciones físicas y temporales, es cada vez más importante en los estudios técnicos, especialmente en los nuevos planes de estudio donde la importancia del trabajo autónomo del alumno está siendo potenciada.

La adecuada combinación de prácticas presenciales y remotas junto a las posibilidades que provee la plataforma de interacción entre el alumno y el profesor, así como entre alumnos de diferentes Universidades, constituye uno de los pilares que proporciona el sistema propuesto y que permiten seguir avanzando en la mejora de la calidad docente en la disciplina de control.

AGRADECIMIENTOS

Los autores desean agradecer a los profesores e investigadores de las siete Universidades que participan en el proyecto *AutomatL@bs* y sin cuya cooperación y esfuerzo no habría sido posible realizar este trabajo. Deseamos agradecer asimismo al Vicerrectorado de Ordenación Académica y el Consejo Social de la Universidad Miguel Hernández de Elche el soporte para la implantación de este proyecto de innovación docente.

REFERENCIAS

- [1] Dormido, S., Torres F., “Aplicación de las TICs a la educación en automática”. *Revista Iberoamericana de Automática e Informática Industrial* 2 (2), 3-7 (2005)
- [2] Ausserhofer, A. “Web-based teaching and learning: A panacea?”. *IEEE Communications Magazine* pp. 92-96. (1999)
- [3] Nguyen, A., Gillet, D., Sire, S., “Sustaining collaboration within a learning community in flexible engineering education”. *Proc. World Conf. Educational Multimedia, Hypermedia and Telecommunications*, 21–26. June (2004)
- [4] Shor M., “Remote-access engineering educational laboratories: Who, what, when, why and how?”. *Proc. American Control Conference*, 2949-2950 (2000)
- [5] Robson R. and Shor M., “A student-centered feedback control model of the educational process”, *Frontiers in Educational Conference*, 2, 14-19 (2000)
- [6] Casini, M., D. Prattichizzo and A. Vicino. “The automatic control telelab: A user-friendly interface for distance learning” *IEEE Transactions on Education* 46(3), 252-257 (2003)
- [7] Jimenez, L.M., R. Puerto, O. Reinoso, C. Fernandez and R. Neco. “Recolab: Laboratorio remoto de control utilizando matlab y simulink”. *Revista Iberoamericana de Automática e Informática Industrial* 2(2), 64–72 (2005)
- [8] Dormido, R., Vargas, H., Duro, N., Sanchez, J., Dormido-Canto, S., Farias, G., Esquembre, F., Dormido, S., “Development of a web-based control laboratory for automation technicians: The three-tanks system”. *IEEE Transactions on Education* 58(1),3-7 (2008)
- [9] Esquembre, F., June. “Easy java simulations: A software tool to create scientific simulations in java”. *Comput. Phys. Commun.* 156 (2), 21–26. (2004)
- [10] O. Reinoso, L.M. Jiménez, L. Payá, L. Fernández, M. Juliá , “Using a remote laboratory for teaching laboratory practices in engineering courses”. *International Conference of Education, Research and Innovation* (2009)
- [11] H.1 Vargas, J. Sánchez Moreno, C. Salzman, F. Esquembre, S. Dormido, and D. Gillet. “Web-Enabled Remote Scientific Environments”. *Computing in Science and Engineering*, 11(3):36..46, (2009)
- [12] *AutomatL@bs* Red de laboratorios de Automática. (2008) <http://lab.dia.uned.es/automatlab>
- [13] R. Costa-Castelló, M. Vallés, L. M. Jiménez, L. Díaz-Guerra, A. Valera, R. Puerto, “Integración de Dispositivos Físicos en un Laboratorio Remoto de Control Mediante Diferentes Plataformas: Labview, Matlab y C/C++”. *Revista Iberoamericana de Automática e Informática Industrial* 7(1), 22-34 (2010)