

Integración de dispositivos físicos en un laboratorio remoto de control mediante diferentes plataformas: Labview, Matlab y C/C++

R. Costa-Castelló * M. Vallés ** L. M. Jiménez *** L. Diaz-Guerra *
A. Valera ** R. Puerto ***

* Institut d'Organització i Control de Sistemes Industrials, Escola Tècnica Superior d'Enginyeria Industrial de Barcelona, Universitat Politècnica de Catalunya, Av. Diagonal 647,11, 08028-Barcelona

** Departamento de Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia, Camino de Vera, nº14, 46022, Valencia, España, (e-mail: {mvalles,giuprog}@isa.upv.es)

*** Departamento de Ingeniería de Sistemas Industriales, Avda. Universidad s/n, 03202 Elche (Alicante), España (e-mail: {luis.jimenez}@umh.es)

Resumen: Los laboratorios remotos de control posibilitan la realización de prácticas con equipos físicos reales que se encuentran ubicados en entornos remotos siendo accesibles por parte de los alumnos en cualquier instante de tiempo y desde cualquier lugar con conexión a Internet. En este artículo se presenta un conjunto de dispositivos físicos incorporados dentro de una esquema global de laboratorios remotos (AutomatL@bs) que posibilitan la realización de prácticas a distancia con estos dispositivos. El rasgo principal común a estos dispositivos consiste en que posibilitan la actuación con sistemas físicos remotos para la realización de prácticas de control clásico con los mismos. Se prestará una especial atención a las diferentes posibilidades de comunicación entre la herramienta visual que dispone el usuario para su interconexión basada en Easy Java Simulations, y la conexión directa con los dispositivos físicos ubicados en los laboratorios remotos. Estos sistemas se han empleado para la realización de prácticas dentro del sistema AutomatL@bs desde el curso 1997/98. Copyright © 2010 CEA.

Palabras Clave: Educación en Control, Laboratorio remoto, Laboratorio virtual, Control remoto, Control PID, Técnicas de Laboratorio.

1. INTRODUCCIÓN

Durante los últimos años el uso de las nuevas tecnologías en la educación ha experimentado un extraordinario crecimiento. Estas nuevas tecnologías han permitido modificar paulatinamente la forma en que los alumnos acceden a los contenidos docentes de cada una de las materias que reciben prestando cada vez más atención a nuevos contenidos pedagógicos que aumenten la calidad del aprendizaje al posibilitar esquemas de educación mucho más flexibles. Hoy en día, la cantidad de información a la que los alumnos pueden acceder a través de Internet es inmensa, pudiendo seleccionar y contrastar los documentos más adecuados en la materia de estudio. El acceso a través de Internet a cualquier tipo de documentación que permita a los alumnos asentar los conocimientos de la materia cursada es un proceso que se ha impuesto durante los últimos años y que ha permitido avanzar en el clásico proceso de aprendizaje. De esta forma los alumnos pueden adquirir los conocimientos precisos de cada una de las materias sin necesidad de acudir de forma presencial a las típicas clases teóricas donde se imparten las lecciones magistrales. Por ejemplo (Gomes and Bogosyan, 2007; Dormido and Torres, 2005) proporcionan aplicaciones prácticas en empresas que ofrecen cursos en línea (*ISA Distance Learning*, 2003), (*Customer Training Courses*, 2003). El aprendizaje basado en la web también es de gran interés para compañías en las que el aprendizaje continuo de sus empleados es importante (Badersten and Avasjö, 1997), (*Online Manu-*

facturing Courses from the Manufacturing Training Network (MTN), 2003). Además de las aplicaciones educativas, la monitorización y el control remoto de procesos industriales a través de Internet ofrecen diferentes ventajas (optimización de las infraestructuras, gestión del mantenimiento preventivo, reducción de los costes de personal, etc.) y también son un paso adelante en el tele-trabajo (López, 1999), (Crossan and Burton, 1993), (Norman, 1993). Algunas aplicaciones relevantes se pueden encontrar en áreas tan diversas como la medicina (Katsura *et al.*, 2007), las plantas químicas (Cushing, 2000) o la robótica (Rösch *et al.*, 2002; Luo and Chen, 2000; Marin *et al.*, 2005; Birk and Kenn, 2002; Safaric *et al.*, 2001).

En las titulaciones de ingeniería y más concretamente en la disciplina de control, además de los conocimientos teóricos de cada una de estas materias es imprescindible que el alumno realice un conjunto de prácticas que le permita asentar y validar los conocimientos teóricos adquiridos. En este sentido, se ha realizado durante los últimos años un esfuerzo considerable en la utilización de las nuevas posibilidades que se ofrecen para el desarrollo de prácticas a distancia por parte de los alumnos que cursan materias relacionadas con el control.

Dos son los tipos de laboratorio que permiten a los alumnos realizar prácticas a distancia a través de Internet: los laboratorios virtuales, y los laboratorios remotos. Un laboratorio virtual permite, por ejemplo, acceso continuo a un proceso simulado en un computador (Vallés *et al.*, 2001; Zuluaga *et*

al., 2005; McKee and Gatward, 2005; Guzmán *et al.*, 2005). Bajo esta estructura, es posible una interacción remota entre el laboratorio virtual y el estudiante. A mitad de camino entre los laboratorios tradicionales y los virtuales están los laboratorios remotos que ofrecen un experimento real a los usuarios remotos (Valera *et al.*, 2005b,a; Domínguez *et al.*, 2005; Gasa *et al.*, 2005; Jimenez *et al.*, 2005). La incorporación de cámaras Web permite la observación de la evolución del sistema real además de las variables de interés.

En los laboratorios remotos se dispone de dos opciones distintas de operar: modo por lotes y modo en línea. Un modo de operación en línea puro considera que los algoritmos de control se encuentran en el computador remoto y que las acciones de control así como la información de los sensores se transmiten a través de Internet (Yang *et al.*, 2003; Sala *et al.*, 2006). Se puede cambiar la referencia y los parámetros mientras se lleva a cabo el experimento, pero aparecen retardos variables como consecuencia del tráfico de Internet. En cambio, el modo de operación por lotes evita los retardos de Internet (Valera *et al.*, 2005a; Jimenez *et al.*, 2005; Domínguez *et al.*, 2005) ya que la referencia y los parámetros del controlador se envían al servidor antes de que el experimento empiece y, una vez finaliza, la salida del proceso se envía al computador remoto.

Sin embargo, el modo por lotes limita totalmente la interacción con el usuario durante la ejecución con el experimento. Para evitar los retardos en lo posible y aumentar la interactividad, se puede realizar una implementación del modo en línea en la que los controladores están en servidor, pero desde el ordenador remoto se puede cambiar la referencia y los parámetros mientras se ejecuta el experimento. Esta última opción permite evitar los retardos y aumentar la interactividad con el estudiante (Dormido *et al.*, 2005; Guzmán *et al.*, 2005).

A pesar de la madurez de las técnicas, y que la mayor parte del equipamiento necesario para poner en marcha un laboratorio virtual/remoto se encuentra disponible en los laboratorios tradicionales (el único elemento adicional necesario es una interfaz entre la aplicación local y el Servidor Web) a los problemas inherentes a la red comentados anteriormente se suelen unir la complejidad y el coste como problemas habituales en la implementación de laboratorios remotos (Marange *et al.*, 2007; Hercog *et al.*, 2007; Casini *et al.*, 2007), lo que limita en gran medida su expansión y generalización.

En este artículo se presenta y analiza la experiencia llevada a cabo por parte de tres universidades españolas en el desarrollo de un laboratorio remoto de control que posibilita a los alumnos la realización de prácticas mediante al acceso a diferentes dispositivos y maquetas localizadas físicamente en diferentes ubicaciones. Esta experiencia se incorpora dentro del proyecto *AutomatL@bs* (AutomatL@bs, 2008), en el que participan un total de siete universidades españolas incorporando diferentes contenidos tanto en equipos como en tipos de prácticas docentes y que permiten a los alumnos de estas siete universidades la realización de prácticas de control con cualquiera de los equipos englobados. Los alumnos que participan en la realización de estas prácticas lo hacen bajo un entorno común independiente del sistema físico con el que interactúen. Este entorno común que utilizan se encuentra basado en *Easy Java Simulations* (EJS) ((Dormido, 2004)). Los sistemas físicos que se han incorporado al proyecto anteriormente referenciado y que se comentarán con más detalle posteriormente, son los siguientes: un motor de corriente continua, un sistema de bola

y viga, y un sistema constituido por un rotoimán. Cada uno de estos sistemas se ha integrado mediante arquitecturas diferentes que son las que se presentarán en apartados posteriores.

EJS se ha convertido en una potente herramienta que permite generar applets Java de una forma sencilla y cómoda accesibles a través de internet simulando los sistemas deseados, a la vez que permite una interfaz de usuario completamente configurable. Esta herramienta ha sido ampliamente utilizada en el diseño de laboratorios remotos de control como por ejemplo en (Dormido *et al.*, 2008) o (Candelas *et al.*, 2006), así como en laboratorios virtuales (Diaz *et al.*, 2000; Christian and Esquembre, 2007). Tomando como base esta herramienta se han integrado diferentes laboratorios remotos utilizando varias opciones para el control a bajo nivel del prototipo deseado. En concreto en este artículo se presentan tres opciones diferentes: Matlab-Simulink, Labview o C/C++. De esta manera el tiempo de desarrollo se reduce considerablemente al poder reutilizarse los diferentes algoritmos de control previamente desarrollados.

El resto del artículo se estructura de la siguiente manera. En la sección 2 se analizan las diferentes posibilidades de conexión entre los sistemas físicos reales y Easy Java Simulations (EJS) que se han implementado: conexión vía Labview, conexión a través de Matlab, y conexión directa mediante C/C++. En la siguiente sección se presentan las diferentes prácticas y sistemas físicos implementados en cada una de estas conexiones, en concreto se presentan los sistemas: rotoimán, bola y viga, y motor de corriente continua, así como los diferentes experimentos de control que se pueden realizar con los mismos. Finalmente se presentan las principales conclusiones que se pueden alcanzar así como los trabajos futuros a desarrollar.

2. CONECTIVIDAD EJS - SISTEMAS FÍSICOS REMOTOS

En esta sección se van a analizar diferentes posibilidades de conexión entre la interfaz de usuario desarrollada en EJS y el sistema real ubicado en el laboratorio remoto. El acceso a los diferentes dispositivos físicos que constituyen los sistemas de prácticas de manera que se puedan modificar los parámetros de control sobre los mismos, se puede realizar de múltiples formas. En este artículo se analizan las siguientes conectividades:

- Interacción del sistema físico a través de Matlab.
- Interacción del sistema físico a través de Labview.
- Interacción directa del sistema físico a través de librerías programadas en C/C++.

2.1 Conectividad EJS-Matlab

Matlab es una herramienta ampliamente utilizada en el entorno universitario que permite analizar y simular multitud de procesos de toda índole. Constituye una plataforma que goza de un soporte técnico y un nivel de desarrollo muy elevado, lo que ha permitido una amplia y creciente difusión en el entorno educativo y más concretamente en las asignaturas relacionadas con el control automático. Uno de los aspectos más significativos radica en la posibilidad de incorporar elementos que permiten la ejecución en tiempo real de los sistemas físicos a través de un sistema de adquisición de datos. De esta manera el tiempo de desarrollo de las aplicaciones que interactúan con los sistemas físicos se reduce considerablemente. Así, Matlab y Simulink han sido previamente utilizados en diferentes sistemas de desarrollo de laboratorios remotos de control como en (Jimenez *et al.*, 2005).

Como se ha comentado previamente la interfaz de usuario de cualquiera de los laboratorios remotos, así como el laboratorio virtual que simula el comportamiento de los equipos físicos reales, se ha desarrollado dentro del entorno de EJS. En la figura 1 se observa la consola que permite acceder tanto a la interfaz de experimentación virtual como a la interfaz de experimentación remota para el caso de un motor de corriente continua.

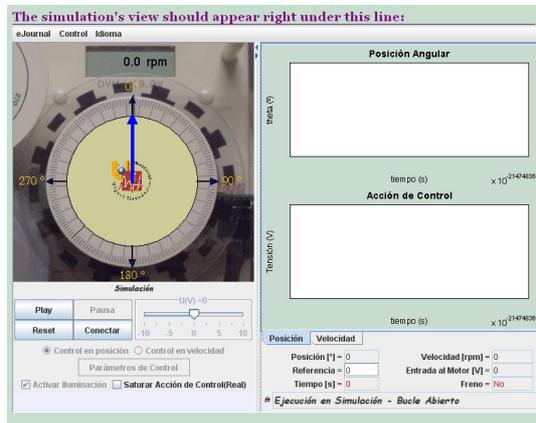


Figura 1. Consola de acceso a la interfaz virtual y remota del motor de corriente continua.

Sin embargo para el acceso remoto al sistema físico, se hace uso de Matlab y más concretamente de Simulink junto con el Real Time Workshop para acceder al sistema físico de manera que se puedan modificar los parámetros de control así como las condiciones de partida en el desarrollo de nuevos experimentos. El enlace entre Matlab y el entorno desarrollado en EJS se realiza a través de un servidor dedicado a tal efecto que se ha denominado *JimServer* (Vargas *et al.*, 2009). En la figura 2 se observa el esquema de comunicaciones que se ha desarrollado.

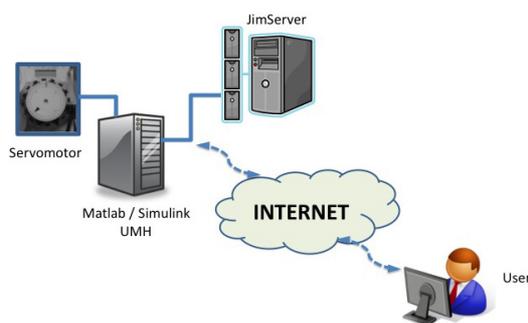


Figura 2. Comunicación entre el entorno local con interfaz EJS y el sistema remoto. El sistema físico se controla a través de Matlab/Simulink.

De esta manera la conexión entre el sistema remoto a través de la interfaz que dispone el usuario y el sistema físico se puede realizar de dos maneras (ambas comunicadas a través del servidor *JimServer*):

- A través de Matlab: La comunicación se realiza a través de las variables que se encuentran disponibles en el Workspace de Matlab, lanzando desde aquí la aplicación desarrollada en Simulink
- Directamente con Simulink: La comunicación se realiza directamente a través de Simulink.

El sistema físico (en este caso un motor de corriente continua) es controlado a partir de un esquema simulink que interacciona con el motor mediante la correspondiente tarjeta de entradas salidas. Dado que se dispone de dos tipos de experimentos a realizar sobre el motor de corriente continua (control en velocidad y control en posición), se han desarrollado dos esquemas de control a través de simulink. A modo de ejemplo se puede observar el esquema implementado que posibilita el control en posición del motor de corriente continua a través de la tarjeta de entradas/salidas (ver figura 3). Los parámetros del regulador se pueden sintonizar modificándolos a través de Matlab. De esta manera el usuario a través de la interfaz desarrollada en EJS modifica estos parámetros del regulador, enviándolos a Matlab a través del servidor *JimServer*.

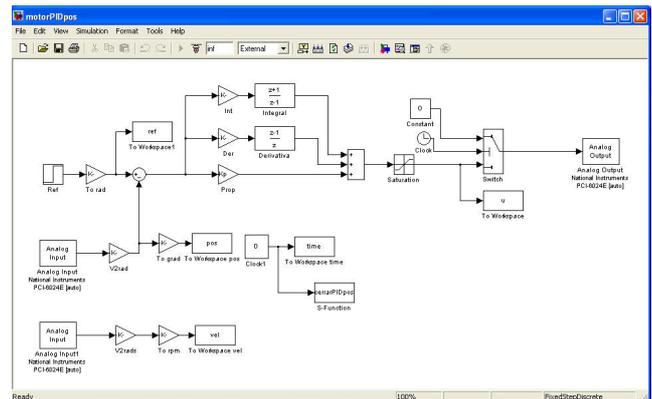


Figura 3. Esquema simulink de acceso al motor para la realización de un control en posición tipo PID.

Uno de los inconvenientes que presenta el modo de acceso planteado a través de Simulink mediante la comunicación con Matlab, consiste en que iniciado el experimento en tiempo real, no se tiene acceso a las variables de entorno del sistema hasta el momento en que Simulink termina de ejecutar el experimento. El tiempo de ejecución de cada uno de los experimentos (control en posición y control en velocidad) se ha establecido en 8 segundos. De esta forma una vez iniciada la ejecución en Simulink, no existe posibilidad de refresco de las variables asociadas (salida, acción de control, etc.) hasta que no se finaliza por completo esta ejecución. Esto es consecuencia de que desde el entorno de trabajo de Matlab no se puede acceder a las variables que maneja Simulink en las que se van almacenando los valores que recibe del proceso. Finalizada esta ejecución, al disponer ya de estas variables en el entorno de Matlab, se envían estos resultados, refrescando de esta manera las gráficas en la consola de la aplicación donde el usuario visualiza todos los resultados.

2.2 Conectividad EJS - Labview

LabVIEW (*Laboratory Virtual Instrumentation Engineering Workbench*) es un entorno de desarrollo de aplicaciones de procesamiento de señales, control, adquisición de datos e instrumentación desarrollado y mantenido por NI (*National Instruments*). Desde la primera versión (1986), LabVIEW ha focalizado sus esfuerzos en el desarrollo de soluciones integrales para la industria y, sobre todo, ha dedicado especial atención al mundo académico cuando se incluyen nuevas características al entorno. Por estas razones, LabVIEW es hoy en día (versión LabVIEW 8.6, agosto 2008) un estándar de *facto* en diferentes ámbitos de la automática y la academia.

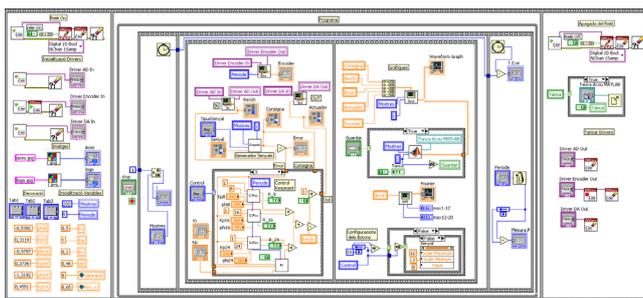


Figura 4. Programa principal de la aplicación de control local.

LabVIEW es una herramienta multiplataforma disponible para sistemas operativos Windows, Linux y MAC. Las aplicaciones desarrolladas por los usuarios reciben el nombre de instrumentos virtuales o simplemente, VI.

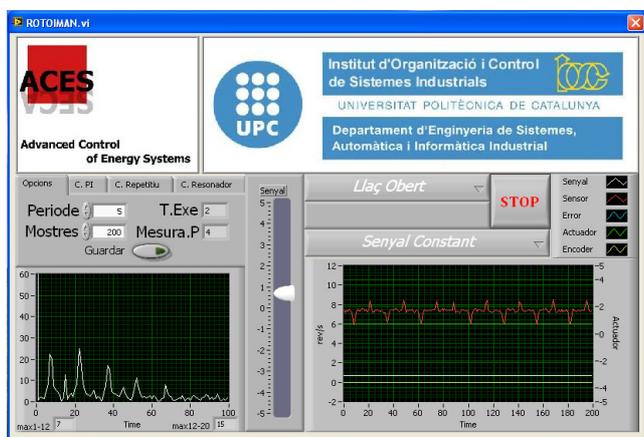


Figura 5. Interfaz de usuario (*Frontal Panel*) de la aplicación de control local.

LabVIEW presenta algunas ventajas adicionales en relación al uso de otras herramientas de desarrollo de aplicaciones en el ámbito del control. Algunas de ellas son:

- El lenguaje de programación, conocido como *G*, es totalmente gráfico. A modo de ejemplo, la figura 4 muestra la forma que toma el programa principal del sistema de control de uno de los laboratorios remotos desarrollado mediante LabVIEW.
- El entorno integra el uso de múltiples dispositivos hardware, especialmente los de NI, permitiendo un desarrollo rápido y eficiente. También puede interactuar con dispositivos no propietarios de la marca, gracias a la utilización de librerías internas de LabVIEW para comunicar con ellos vía *dlls*, *C code* o *scripts* de Matlab.
- El manejo de hilos de ejecución concurrentes es realmente simple, fácil e intuitivo.
- Un conjunto de librerías proporcionadas por el entorno facilita el desarrollo de aplicaciones. Por ejemplo, librerías para procesamiento de señales, control e instrumentación, comunicación, manejo de archivos, control de flujo, etc.
- El desarrollo de la interfaz gráfica de usuario o *Frontal Panel* está totalmente integrado. Todos los elementos de visualización e interacción están incluidos en el propio entorno y no introducen complejidad adicional. La figura 5 muestra la interfaz de usuario de la aplicación mostrada en la la figura 4.

En el ámbito de la experimentación remota con fines académicos, LabVIEW ha integrado soluciones que permiten a sus desarrolladores implementar de manera rápida y con unas pocas pulsaciones de ratón un laboratorio remoto. Esta tecnología conocida como Paneles Remotos permite publicar el panel frontal de una aplicación LabVIEW a través de Internet utilizando el servidor web interno de la herramienta. Sin embargo, una de las desventajas de este enfoque es que el usuario remoto necesita tener instalado un software adicional (distribuido por NI) en la máquina del cliente llamado LabVIEW *run-time engine*. Sin embargo, esto no origina complicaciones adicionales para la ejecución de la herramienta en modo remoto. Por otra parte, desde un punto de vista pedagógico se persigue también que los laboratorios remotos/virtuales incorporen mecanismos de simulación y diseño con la misma interfaz de usuario lo que se traduce en una complejidad extra de desarrollo.

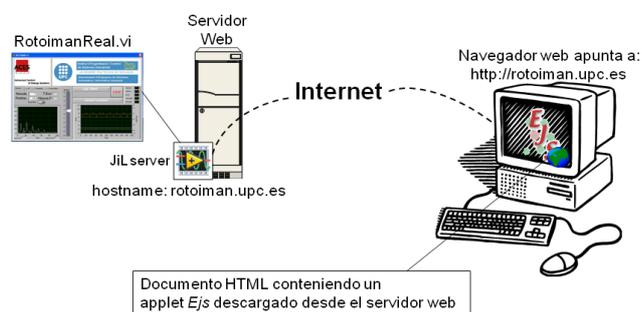


Figura 6. Arquitectura del servidor JIL.

Con el fin de solventar estas limitaciones y sin renunciar al potencial de LabVIEW como herramienta de desarrollo, el Departamento de Informática y Automática de la UNED ha desarrollado un conjunto de aplicaciones informáticas que permiten el desarrollo de laboratorios virtuales y remotos de manera estructurada utilizando EJS y LabVIEW. La Figura 6 presenta un diagrama de comunicación de este enfoque.

Este esquema se basa en el uso de un módulo de software genérico, desarrollado en LabVIEW denominado JIL Server. JIL Server integra la gestión automática de usuarios, permite acceder desde un *applet* o aplicación JAVA a cualquiera de las variables que componen una aplicación LabVIEW ejecutada en una máquina remota (como por ejemplo, Rotoiman.vi en la Figura 6), ya sea para monitorizar su evolución o modificar su valor. Este marco de trabajo permite, de forma simple, integrar en una aplicación EJS la capacidad de monitorizar el estado de una aplicación LabVIEW remota y modificar el valor de algún parámetro de entrada a la misma. Para ello, una librería JAVA llamada *jil.jar* ha sido desarrollada para ser importada en aplicaciones del lado del cliente. Esta librería contiene clases y métodos que permiten a estas aplicaciones controlar la conexión con un VI remoto publicado por JIL Server.

Mediante este procedimiento se han desarrollado algunas de los laboratorios virtuales y remotos dentro del entorno de experimentación del proyecto Automatl@bs.

2.3 Conectividad EJS - C/C++

Otra alternativa a la hora de implementar el servidor de la aplicación correspondiente al laboratorio remoto es utilizar un lenguaje de programación de alto nivel como es el C++.

El proceder de esta manera tiene la ventaja de que el código desarrollado resulta altamente eficiente dado que se desarrolla específicamente para la aplicación en cuestión, por lo que los tiempos de cómputo son menores que con otras herramientas. Sin embargo, esta solución tiene la desventaja de que se necesita un mayor desarrollo de programación ya que no existen funciones predefinidas para la aplicación a desarrollar.

A la hora de implementar el servidor debe tenerse en cuenta cuál es su funcionalidad. El servidor deberá estar esperando peticiones entrantes de trabajar con el equipo. En cuanto esto de produzca debe consultar en la base de datos en donde se alojan las reservas para trabajar con la planta gestionada por el servidor si se trata de una reserva correcta. En caso de no ser así, la aplicación queda de nuevo a la espera de nuevas peticiones.

En el caso de tratarse de una petición correcta deberá iniciarse la aplicación que permita al usuario remoto controlar la planta gestionada por el servidor. Para ello, se deberá, por un lado, controlar la planta usando el controlador proporcionado por el usuario remoto, pero, al mismo tiempo se le debe enviar a dicho usuario información de cómo está funcionando su controlador, así como permitir la posibilidad de que éste cambie dinámicamente tanto la referencia como los parámetros del controlador. Por otro lado, además, la aplicación debe controlar que el usuario no exceda en su uso el tiempo que tenía asignado.

Para permitir todas estas ejecuciones de manera simultánea se ha desarrollado una aplicación basada en hilos de ejecución con la estructura que se muestra en la figura 7.

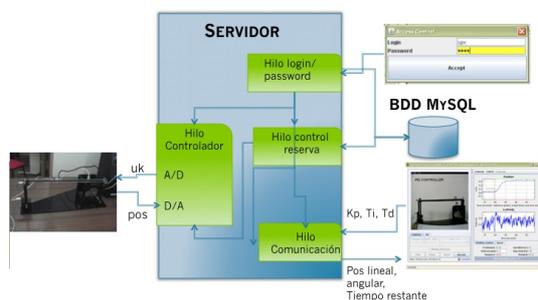


Figura 7. Estructura del controlador basado en C++.

A continuación detallaremos la funcionalidad de cada uno de estos hilos.

Hilo Login/Password Tal como se ha comentado anteriormente, la aplicación principal estará continuamente a la espera de que un usuario quiera conectarse a la planta. En el momento en que esto suceda, se crea este hilo el cual únicamente se encarga de recoger el usuario y la contraseña introducidos por el cliente remoto.

Esta información se guarda en unas variables que se comparten con la aplicación principal, de manera que ésta es la que se encarga de acceder a la base de datos y verificar si el usuario dispone de una reserva válida, es decir, que el usuario y contraseña sean válidos y que dicho usuario disponga de una reserva para el día y hora en que nos encontremos.

Si es así, la aplicación procede a dotar al usuario de las funcionalidades suficientes que le permitan controlar de manera

remota la planta para lo cual lanzará la ejecución simultánea de tres hilos: el del controlador, el de comunicación y el de control de reserva que aparecen en la figura anterior. En caso contrario, informa al usuario del tipo de error y queda a la espera de una nueva conexión entrante.

Hilo Controlador Este hilo, como su nombre indica será el encargado de ejecutar el controlador diseñado por el usuario remoto, para lo cual deberá acceder a las tarjetas de adquisición de datos y manejar el periodo de muestreo aunque para el usuario remoto esto resulte transparente.

Algunos de los parámetros del controlador podrán ser cambiados de manera dinámica aunque para el hilo Controlador esto resulta transparente, ya que esto se consigue a través de la compartición de variables entre este hilo y el de comunicación.

Además, debido a la funcionalidad que tiene a su cargo, este hilo será el que tenga la prioridad más alta frente a los otros dos hilos que se ejecutan de manera concurrente junto con él.

Hilo de Comunicación Este hilo tendrá dos objetivos. Por un lado, permitir que el usuario pueda cambiar dinámicamente tanto los parámetros del controlador como el valor de la referencia y, por otro lado, que el servidor pueda enviar al usuario remoto información en tiempo real de cómo está funcionando su controlador.

Hilo de Control de Reserva Además de lo realizado por los hilos comentados anteriormente, será necesario también controlar que el usuario que se encuentre actualmente usando la planta no exceda el tiempo que tiene asignado para su sesión, para lo cual de manera periódica verifica dicha información (que además se envía también al usuario remoto para que tenga conocimiento de ella).

3. PROTOTIPOS EXPERIMENTALES

3.1 Rotoiman

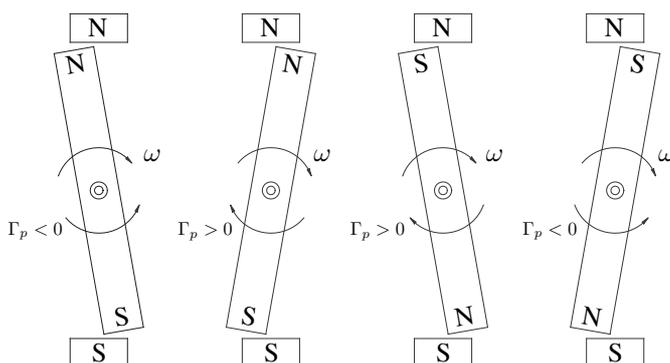


Figura 8. Esquema de la estructura de la planta.

La teoría de control clásica se ha centrado en el estudio de esquemas para el seguimiento/rechazo de consignas/perturbaciones constantes, por ello las plantas didácticas habitualmente se centran en ilustrar este tipo de problemática. En la práctica existen sistemas que presentan otro tipo de referencias o perturbaciones, un claro ejemplo de ello son los sistemas con elementos rotatorios. En este régimen de funcionamiento, cualquier fricción, desequilibrio o asimetría existente en el sistema genera una perturbación periódica sobre el mismo.

El período de dicha perturbación está directamente relacionado con la velocidad angular que lo genera. La figura 8 muestra el esquema de un prototipo didáctico diseñado para ilustrar este concepto (Costa-Castello *et al.*, 2005). La idea básica consiste en acoplar un elemento que genere un par pulsante (Γ_p) sobre el eje de un motor de corriente continua. El medio utilizado para generar el par es un sistema de imanes (figura 8). Una barra central, solidaria al eje del motor, dispone de un imán en cada uno de sus extremos (los imanes presentan polaridades inversas por lo que toda la barra se asemeja a un imán de mayor tamaño). El movimiento del motor de corriente continua genera un par debido a la interacción de los diferentes imanes que componen el sistema, el par depende del ángulo (θ) del eje del motor. Cuando la velocidad angular del eje es constante ($\dot{\omega} = \ddot{\theta} = 0$) el par sobre el eje de rotación es una señal periódica con un período relacionado con la velocidad angular. El objetivo de control para esta planta es que gire a velocidad constante pese a la existencia de este par de perturbación (Xargay and Castelló, 2004). La figura 9 muestra el prototipo construido, este presenta un *encoder* que permite medir instantáneamente la posición y velocidad del eje de giro, esta medida es la utilizada para cerrar los lazos de control.

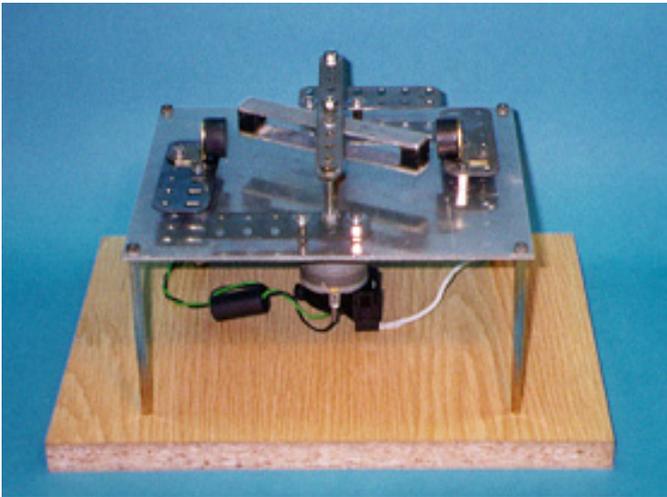


Figura 9. Maqueta de la planta analizada.

Recientemente este prototipo ha sido incorporado en la red *Automat@abs*, para ello se tomó como punto de partida un laboratorio virtual previamente existente (Mata and Castelló., 2005) y una implementación del controlador de bajo nivel (Costa-Castello *et al.*, 2005). Con el fin de simplificar la conexión entre el laboratorio virtual y el controlador local se ha recodificado mediante *LabVIEW* el controlador local. La figura 4 muestra la estructura del instrumento virtual que se ha desarrollado. Esta aplicación local permite desarrollar diferentes tipos de ensayos sobre el dispositivo real, además de ofrecer una interfaz de usuario que puede utilizarse en entornos docentes presenciales (figura 5).

Asumiendo que la dinámica dominante es la mecánica, la planta puede modelarse mediante un sistema de primer orden $\omega(s)/V(s) = \frac{K}{\tau s + 1}$ con una perturbación aditiva sobre la salida.

3.2 Motor de corriente continua

Otro de los equipos físicos incorporados al conjunto está constituido por un motor de corriente continua. Sobre este equipo se

pueden realizar prácticas de control clásico tanto en velocidad como en posición. Los componentes electromecánicos de esta unidad comprenden el motor de corriente continua *Feedback 32-002*, un tacómetro analógico, un potenciómetro analógico como señal de posición, encoders incrementales absolutos y un freno magnético. En la figura 10 se observa una imagen de este dispositivo. Asimismo el esquema eléctrico del motor de corriente continua se detalla en la figura 11.



Figura 10. Servomotor Feedback 32-002.

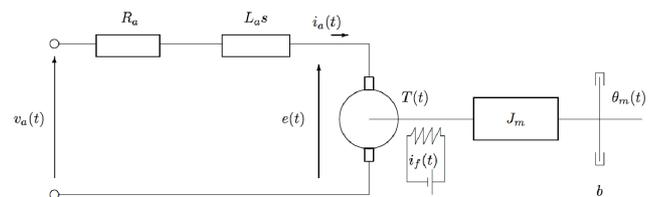


Figura 11. Esquema eléctrico de un motor de corriente continua.

Este tipo de sistemas de servoposicionamiento basado en motor de corriente continua consta de tres subsistemas: el subsistema eléctrico, el magnético y el mecánico. Su análisis, y por tanto la obtención del modelo matemático, se suele realizar estudiando estos tres subsistemas:

- Subsistema magnético: Una de las partes más importantes del motor, el devanado de inducido, consiste en un arrollamiento de varias espiras que puede girar inmerso en un campo magnético constante. Dicho campo magnético se puede generar mediante un imán permanente o por un devanado consistente en una bobina por la que circula una corriente de excitación $i_f(t)$, que supondremos constante para que el campo sea también constante. Al circular una corriente $i_a(t)$ por el devanado inducido, como resultado de la interacción con el campo magnético se ejerce sobre él un par $T(t)$ que es directamente proporcional al campo magnético y a la propia corriente de inducido $i_a(t)$. Dado que se supone constante el campo magnético, el par del motor será proporcional a la corriente de inducido, por lo que:

$$T(t) = K_t i_a(t) \quad (1)$$

Por otra parte, el giro de las espiras del devanado de inducido en presencia del campo magnético, produce en el mismo una fuerza contraelectromotriz, $e(t)$, proporcional a su velocidad de giro:

$$e(t) = K_e \dot{\theta}_m(t) \quad (2)$$

- Subsistema eléctrico: Asimismo, el devanado de inducido es, a todos los efectos, un conductor, con una resistencia R_a y una inductancia L_a , sobre la que hay que considerar, además, la fuerza contraelectromotriz como una fuente de tensión dependiente de la velocidad de giro. La ecuación en la malla será por tanto:

$$v_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K_e \dot{\theta}_m(t) \quad (3)$$

Aplicando la transformada de Laplace a la ecuación anterior, se obtiene:

$$V_a(s) = (R_a + sL_a)I_a(s) + K_e \dot{\theta}_m(s) \quad (4)$$

- Subsistema mecánico: El par mecánico $T(t)$ desarrollado por el motor se emplea para imprimir aceleración angular $\dot{\theta}_m(t)$ a la carga y en vencer la fuerza de fricción $b \dot{\theta}_m(t)$, siendo J_m la constante inercial del motor y b el coeficiente de fricción viscosa. Por tanto:

$$T(t) = J_m \ddot{\theta}_m(t) + b \dot{\theta}_m(t) \quad (5)$$

Aplicando la transformada de Laplace a la ecuación anterior:

$$T(s) = (J_m s + b) \dot{\theta}_m(s) \quad (6)$$

La función de transferencia $\theta_m(s)/V_a(s)$ se puede obtener agrupando términos en las expresiones anteriores, llegando a la siguiente expresión:

$$\frac{\theta_m(s)}{V_a(s)} = \frac{K_t}{s[(J_m s + b)(L_a s + R_a) + K_t K_e]} \quad (7)$$

Ahora, asumiendo que la inductancia de la armadura es generalmente pequeña, podemos despreciarla en cuyo caso, la función de transferencia se reduce a:

$$\frac{\theta_m(s)}{V_a(s)} = \frac{K_m}{s(\tau s + 1)} \quad (8)$$

siendo $K_m = \frac{K_t}{(R_a b + K_t K_e)}$ y $\tau = \frac{R_a J_m}{(R_a b + K_t K_e)}$

Si consideramos $\omega(s)$ la salida del sistema, la función de transferencia que se obtiene es la siguiente:

$$\frac{\omega(s)}{V_a(s)} = \frac{K_m}{\tau s + 1} \quad (9)$$

donde $\omega(t) = \frac{d\theta(t)}{dt}$ representa la velocidad angular del motor.

De esta forma el modelo experimental del motor de corriente continua representado en un diagrama Simulink es el que aparece en la figura 12.

3.3 Sistema Bola y Viga

El sistema de bola y viga es otro de los experimentos integrados en *AutomatL@bs*, en concreto se utiliza una maqueta de la compañía Quanser Consulting (Figura 13). Según se muestra en la figura 14, este prototipo consiste en una barra formada por dos rieles sobre los que rueda una bola de acero, producto de la inclinación de la misma. La inclinación α de la barra se puede manipular actuando sobre la posición angular θ del

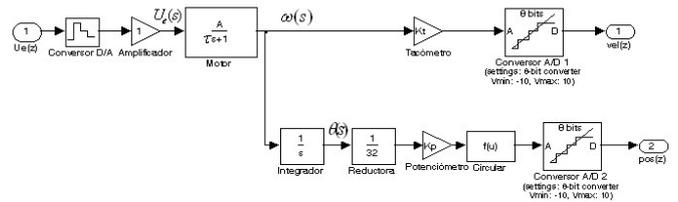


Figura 12. Modelo experimental del motor de corriente continua en Simulink.

motor, que está acoplado mecánicamente mediante engranajes al extremo móvil de la barra. Se puede medir la posición x de la bola sobre la barra mediante un potenciómetro situado en los rieles. La bola hace las veces de cursor cortocircuitando los elementos resistivos de dicho potenciómetro. Además, se puede medir también la posición angular del motor mediante un segundo potenciómetro conectado a su eje.

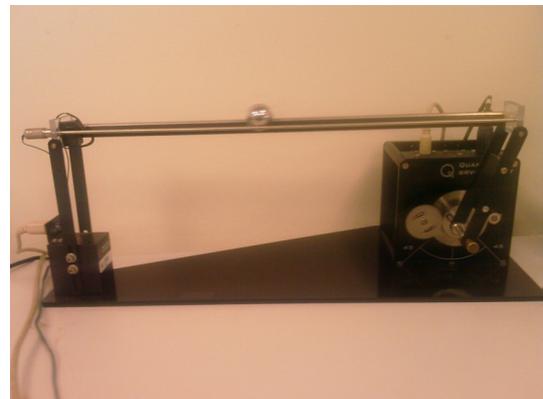


Figura 13. Sistema de la bola y la viga.

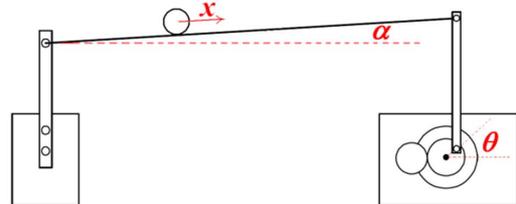


Figura 14. Esquema del sistema de la bola y la viga.

El modelo matemático de este proceso se obtiene realizando un balance de las fuerzas que actúan sobre la bola:

$$m_b \frac{d^2 r}{dt^2} = -m_b g \sin(\alpha) - B_b \frac{dr}{dt} \quad (10)$$

donde r corresponde a la distancia desde el pivote hasta la bola, m_b a la masa de la bola, B_b al coeficiente de fricción rozamiento entre la bola y la viga, y g a la aceleración debida a la gravedad. Esta ecuación se puede linealizar en torno a $\alpha=0$, obteniendo la expresión siguiente:

$$m_b \frac{d^2 r}{dt^2} = -m_b g \alpha - B_b \frac{dr}{dt} \quad (11)$$

Dado que la barra está fija al engranaje, y considerando $r_e \ll L$, siendo r_e el radio del engranaje y L la longitud de la barra; la relación entre el ángulo de inclinación de la barra y el ángulo del eje del motor puede aproximarse como $\alpha = \frac{r_e}{L} \theta$. Teniendo en cuenta estas consideraciones y asumiendo en el origen en el

centro de la barra es posible formular la siguiente función de transferencia:

$$\frac{r(s)}{\theta(s)} = \frac{-r_e g m_b}{L s (m_b s + B_b)} \quad (12)$$

4. PRÁCTICAS PROPUESTAS

En esta sección se describen las actividades pedagógicas diseñadas para realizar sobre los prototipos experimentales descritos anteriormente. Los laboratorios virtuales y remotos que se van a presentar tienen en común unos modos de funcionamiento predefinidos, así como unos entornos de experimentación con interfaces muy similares de forma que los alumnos los observen con un único entorno.

El uso combinado de todos estos laboratorios permiten realizar prácticas de los principales conceptos introducidos en un curso de control clásico y diferentes conceptos avanzados sobretodo los relacionados con el uso de las técnicas de diseño en el campo frecuencial.

4.1 Modos de funcionamiento

Todos los laboratorios ofrecen dos modos de funcionamiento claramente diferenciados:

- **Modo Virtual:** Aunque es transparente para el estudiante, en este modo el estudiante trabaja utilizando un modelo numérico del comportamiento del sistema. La vista corresponde a un modelo geométrico y un conjunto de gráficas cuantitativas que se van actualizando con los resultados del modelo numérico.

Este modo es de gran interés como herramienta de primer contacto y validación de cálculos analíticos. Dado que el uso de este modo no consume recursos de la red no dispone de restricciones de acceso.

- **Modo Remoto:** En este modo el estudiante trabaja con una planta real situada en alguna de las universidades de la red. El acceso a la planta remota se realiza mediante el mismo entorno utilizado en el Modo Virtual lo que simplifica el acceso y uso del laboratorio, pues los estudiantes ya lo conocen.

Únicamente se permite a los estudiantes acceder al laboratorio remoto una vez hay completado satisfactoriamente las diferentes actividades en el Modo Virtual. Esta manera permite reducir el tiempo que necesita el estudiante la planta real.

En cada uno de estos dos modos existen, a su vez, dos modos complementarios referidos al control:

- **Modo Manual:** El modo manual permite realizar ensayos de lazo abierto. Este tipo de ensayos tiene dos objetivos pedagógicos, el primero de ellos es visualizar la necesidad de un control en lazo y el segundo es la obtención de modelos del comportamiento del sistema que sirvan de base para el diseño de controladores.

Aunque Automatl@bs incorpora plantas de diferente naturaleza, las utilizadas en estos laboratorios son plantas mecánicas simples, por ello su comportamiento puede ser descrito por modelos de primer orden cuando la salida es velocidad, $\frac{\omega(s)}{V(s)} = \frac{k_\omega}{\tau s + 1}$ o bien de segundo orden cuando la salida es posición $\frac{\theta(s)}{V(s)} = \frac{k_\theta}{s(\tau s + 1)}$.

Los parámetros de estos modelos se obtienen de forma diferente en función de la práctica a realizar, así en el

servomotor se obtiene mediante identificación paramétrica a partir de la observación de la respuesta del sistema frente a diferentes señales de entrada, en la *bola y viga* se calculan a partir de los parámetros físicos y se analiza la influencia de ciertos parámetros como la fricción o la inercia, y finalmente en el *rotoimán* el modelo se ofrece ya diseñado y se utiliza este modo para analizar las características frecuenciales de la resulta del sistema cuando este está sometido a perturbaciones de excitación permanente.

- **Modo Automático:** El modo automático permite realizar ensayos de lazo cerrado utilizando diferentes controladores.

A partir de los modelos obtenidos mediante los ensayos en Modo Manual es posible analizar la estabilidad de lazo cerrado y diseñar controladores para que cumpla diferentes especificaciones.

En las prácticas del *servomotor* y la *bola y viga* se utilizan los polos de lazo cerrado como herramienta de estudio de la estabilidad. Dado que se utilizan controladores de bajo orden P, PI, PID es posible realizar este tipo de estudio analíticamente. Todo el proceso se realiza utilizando el formalismo de tiempo continuo. En contrapartida, en la práctica del *rotoiman* se utiliza el campo frecuencial y el formalismo de tiempo discreto para realizar el estudio de estabilidad y el diseño de controladores.

En las prácticas del *servomotor* y la *bola y viga* se analiza el tipo de control necesario para alcanzar las especificaciones necesarias en función del tipo del sistema. En estas dos prácticas se pretende seguir referencias constantes, para ello es suficiente la introducción de la parte integral en los sistemas de tipo 0, salida velocidad, (control PI o PID) y el P o PD en las plantas de tipo 1, salida posición. En contrapartida las prácticas del *rotoiman* se centran en mostrar las limitaciones del control PID para el rechazo de perturbaciones no constantes, y la introducción del control resonante y el control repetitivo con el fin de rechazar perturbaciones de carácter periódico.

4.2 Entornos de experimentación

Las aplicaciones desarrolladas tienen dos partes bien diferenciadas (ver Figura 15). En la parte de la izquierda (que se denominará a partir de ahora ventana de representación) se puede observar un esquema del sistema (en este caso, la bola y la viga). En la parte de la derecha (que se denominará ventana de evolución) se muestra la evolución de las principales variables del proceso.

En la parte de la superior de la ventana de representación se puede observar un esquema del sistema que cambiará en función del estado del proceso. En la parte inferior hay un panel con una serie de botones que se van a utilizar para actuar sobre el sistema, y que serán diferentes según el proceso concreto que se vaya a emplear. Aunque el panel de botones es diferente según la planta, todas ellas tienen una serie de opciones comunes.

Entre esas opciones comunes nos encontramos con los botones que permiten al usuario controlar las operaciones principales sobre la evolución del sistema. En concreto puede realizar las acciones siguientes:

- **Play.** Sirve para iniciar la simulación.
- **Pause.** Sirve para establecer una pausa en la simulación.

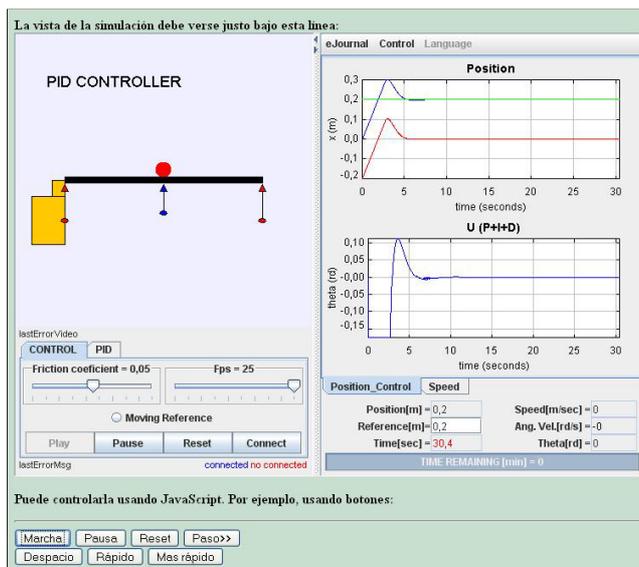


Figura 15. Interfaz de la aplicación para el sistema de la bola y la viga.

- **Reset.** Sirve para resetear la simulación o conexión remota que se esté llevando a cabo e iniciar de nuevo el trabajo desde cero.
- **Connect.** Si no se presiona este botón, al pulsar el Play la aplicación funciona en modo simulación, por lo tanto, se está ejecutando la aplicación como laboratorio virtual. Si se presiona, permite la conexión con la planta en modo remoto (utilizando el sistema real que se encuentra en el laboratorio) una vez superado el control de acceso. Las claves de acceso se facilitarán cuando se permita el acceso a la planta a un determinado estudiante. Es importante notar también que cuando se está trabajando en modo remoto los botones de Play, Pause y Reset no están activos.
- **Disconnect.** Durante el tiempo que la planta está trabajando en modo remoto, el botón Connect se sustituye por el de Disconnect que sirve para parar el funcionamiento del laboratorio y regresar a modo simulación.

Gracias a estos dos últimos botones se permite el cambio entre los modos de funcionamiento anteriormente descritos, que en la interfaz se traduce en dos formas de visualización de la planta (ver figura 1):

- En modo virtual. Las plantas se muestran de forma simulada tal y como aparece en la figura 15. Cuando se trabaja en este modo, el sistema opera localmente y evoluciona en base a un modelo matemático del proceso.
- En modo remoto (usando la conexión por Video). Se muestra la planta real que está en el laboratorio remoto. Este modo de funcionamiento se muestra la imagen tomada por vídeo de la planta real que se encuentra en el laboratorio. En la parte central de la ventana de representación, aparece una descripción que indica el estado del vídeo.

Además de otros botones específicos para la operación de cada planta, por debajo de las pestañas de operación del sistema, aparece una línea informativa donde se explicita el modo de operación del sistema en todo momento. Si se encuentra en modo simulación, tal y como ocurre en la figura 15, aparecerá la frase: Status Lab: Simulation Mode. No obstante, en caso de

operar en forma remota será éste el estado que se muestre (Status Lab: Remote Mode with labserver).

A la derecha de la ventana de representación está la ventana de evolución del sistema, en la que se muestra, como su propio nombre indica la evolución de las variables más importantes del proceso. En su menú de archivos (parte superior de la ventana de evolución) se encuentra la opción Control, cuyo menú desplegable permite realizar dos posibles acciones sobre el sistema (modos de control complementarios anteriormente descritos): manual o automático. Debajo del menú de archivos de la ventana se encuentran gráficas en las que se recoge la evolución de las variables más importantes del proceso. Debajo de ambas gráficas se muestran un conjunto de campos numéricos que indican el valor de las distintas variables de interés del proceso.

Finalmente, debajo de estos campos numéricos aparece un mensaje que indica el tiempo restante en minutos que nos queda para operar con la planta si estamos trabajando en modo remoto. Este tiempo irá decreciendo mientras dure el transcurso de la práctica, para que el alumno conozca en todo momento el tiempo de conexión a planta del que dispone. No obstante, en caso de trabajar en modo simulación este valor aparecerá a 0.

A continuación se muestran brevemente los diferentes entornos desarrollados:

Rotoiman El entorno de desarrollado para este proceso, permite tanto virtual como remotamente realizar ensayos de lazo abierto, ensayos de lazo cerrado mediante controladores PI, controladores resonantes (Malo and Griño (2008)) o controladores repetitivos (Costa-Castello *et al.* (2005)). La figura 16 muestra el aspecto del entorno resultado de todo este proceso y que está integrado en *AutomatL@bs*. En la parte izquierda se puede observar la imagen del proceso simulado, así como las diferentes pestañas con opciones que permiten realizar los experimentos ya enumerados. En la parte derecha se muestran las variables de interés del sistema (velocidad y acción de control), de forma gráfica y numérica. Tanto virtual como remotamente el entorno permite realizar ensayos de lazo abierto, ensayos de lazo cerrado mediante controladores PI, controladores resonantes (Malo and Griño, 2008) o controladores repetitivos (Costa-Castello *et al.*, 2005).

Motor de corriente continua Sobre este sistema se han implementado los siguientes experimentos:

- Control en posición
- Control en velocidad
- Identificación del sistema en lazo abierto

Cada uno de estos experimentos se puede realizar a través de la interfaz (véase figura 1) tanto en modo virtual como en modo remoto. Como se observa, la interfaz es similar al del resto de sistemas físicos integrados dentro del laboratorio remoto de control. En la parte izquierda de la consola se observa la imagen del motor y superpuesto en realidad aumentada las señales de referencia así como la posición o velocidad a la que gira el motor. En la parte derecha de esta consola se presentan las variables del sistema (posición o velocidad en función del tipo de experimento seleccionado), y la acción de control. En la parte inferior izquierda se disponen un conjunto de botones y barras de desplazamiento de manera que se posibilita el acceso a la configuración de los diferentes parámetros de control en cada uno de los experimentos implementados, así como los

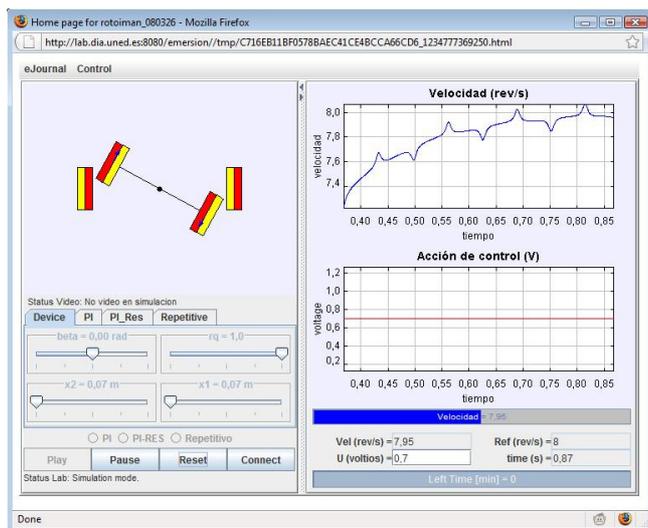


Figura 16. Laboratorio virtual/remoto del rotoiman.

botones que permiten lanzar el experimento bien en modo virtual o en modo remoto tras efectuar la conexión.

Sistema Bola y Viga La aplicación tiene en la parte superior izquierda un esquema del sistema de la bola y la viga (ver figura 15), en donde la posición de la bola sobre la viga cambiará en función del estado del proceso. En la parte inferior izquierda, como peculiaridad de este proceso, tienen dos pestañas: CONTROL y PID. En la pestaña CONTROL se proporciona un conjunto de sliders, o barras de desplazamiento, así como una serie de botones que permiten definir diversas situaciones en la dinámica del proceso (por ejemplo, considerar o no que existe rozamiento). En la pestaña PID se proporcionan tres campos numéricos que permiten variar los parámetros de los controladores asociados al control automático de posición. Se puede por lo tanto variar la ganancia proporcional (K_p), el tiempo integral (T_i) y el tiempo derivativo (T_d). Inicialmente tendrán asignados unos valores que permiten operar con el sistema de una forma razonable.

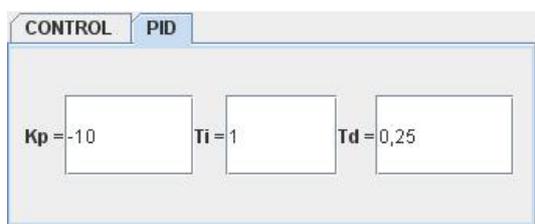


Figura 17. Pestaña PID.

En la parte de la derecha se muestra la evolución de las principales variables del proceso (posición, velocidad, velocidad angular, etc.), tanto de forma gráfica como numérica.

El problema de control que se propone en el sistema de la bola y la viga es controlar la posición donde se encuentra la bola sobre la viga. Para ello tanto en el laboratorio virtual como en el remoto (Figura 15) se permiten realizar los siguientes experimentos:

- Operación en lazo abierto. En este modo el usuario va consignando manualmente el valor de θ , ello permite analizar la respuesta del sistema en lazo abierto.

- Control utilizando un controlador de posición. El controlador que se propone para controlar la posición de la bola es un PD, $C(s) = K_p(1 + T_d s)$, o un PID, $C(s) = K_p(1 + \frac{1}{T_i s} + T_d s)$. Si la elección de los parámetros del controlador es adecuada, el sistema será estable y no existirá error en régimen estacionario. Concretamente, las raíces de la ecuación característica del sistema cuando se emplea un regulador de tipo PD se corresponden con la siguiente expresión:

$$Lm_b s^2 + (LB_b - K_p T_d r_{e g m_b}) s - K_p r_{e g m_b} = 0 \quad (13)$$

5. CONCLUSIÓN

En este artículo se han presentado diferentes laboratorios remotos de control para la realización de prácticas docentes a través de internet. Una de las características principales de estos laboratorios remotos de control es que todos ellos se han configurado mediante una interfaz de acceso común desarrollada en *Easy Java Simulations*, de manera que el estudiante realiza las prácticas con los diferentes laboratorios de la misma manera independientemente de cómo se tenga configurado el laboratorio físico real. En este sentido, y a modo de ejemplo, se proponen en este artículo tres formas diferentes de acceso a los laboratorios reales: mediante Matlab, a través de Labview, y directamente con C/C++. La principal ventaja de esta forma de actuación consiste en que se reducen los tiempos de desarrollo de laboratorios remotos de control, al independizar el acceso remoto mediante una plataforma común desarrollada en EJS, del acceso local de cada una de las maquetas o dispositivos físicos que puede implementarse mediante el uso de diferentes herramientas como las presentadas en este artículo. En este sentido, el acceso a los diferentes equipos físicos, así como las prácticas y experimentos a desarrollar sobre los mismos, se encuentran homogéneamente estructurados tanto en su acceso e interfaz permitiendo que el estudiante dedique el máximo tiempo posible a los aspectos relacionados con el análisis e interpretación de los resultados alcanzados al realizar las prácticas remotas con los diferentes equipos. Los equipos utilizados en el laboratorio remoto han sido un Rotoimán, un Motor de Corriente continua, y un sistema Bola y Viga.

AGRADECIMIENTOS

Este trabajo ha sido realizado parcialmente gracias a los proyectos CICYT DPI2007-62582 (Ministerio de Ciencia y Tecnología) e Incorporación de un laboratorio remoto en la red de laboratorios remotos y virtuales: Automat@Labs (UPC 2008).

REFERENCIAS

- AutomatL@bs (2008). Red de laboratorios de automatica, <http://lab.dia.uned.es/automatlab/>.
- Badersten, L. and A. Avastj  (1997). Ericsson company sweden and lund university. In: *13 EUCEN European Conference*.
- Birk, A. and H. Kenn (2002). Roboguard, a tele-operated mobile security robot. *Control Engineering Practice* **10**, 1259–1264.
- Candelas, F.A., C.A. Jara and F. Torres (2006). Flexible virtual and remote laboratory for teaching robotics. *IV International Conference on Multimedia and Information and Communication Technologies in Education* **3**, 1959–1963.
- Casini, M., D. Prattichizzo and A. Vicino (2007). Operating remote laboratories through a bootable device. *Trans on Industrial Electronics* **54**(6), 3134–3140.

- Christian, W. and F. Esquembre (2007). Modelling physics with easy java simulations. *The Physics Teacher* **45**(10), 475–480.
- Costa-Castello, R., J. Nebot and R. Griño (2005). Demonstration of the internal model principle by digital repetitive control of an educational laboratory plant. *IEEE Trans. Educ.* **48**(1), 73–80.
- Crossan, G. and P. F. Burton (1993). Teleworking stereotypes: a case study. *Journal of Information Science* **19**(5), 349–362.
- Cushing, M. (2000). Process control across the internet. *Chemical Engineering* **107**(5), 80–82.
- Customer Training Courses (2003). Siemens Energy & Automation.
- Diaz, J., C. Queiruga, C. Villar and L. Fava (2000). A virtual java simulation lab for computer science students. Proceedings of WebNet World Conference on the WWW and Internet pp. 982–984.
- Domínguez, M., P. Reguera and J. J. Fuertes (2005). Laboratorio remoto para la enseñanza de la automática en la universidad de león (españa). *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Dormido, R., H. Vargas, N. Duro, J. Sánchez, S. Dormido-Canto, G. Farias, F. Esquembre and S. Dormido (2008). Development of a web-based control laboratory for automation technicians: The three-tank system. *IEEE Transactions on Education* **58**(1), 35–45.
- Dormido, S. (2004). Control learning: Present and future. *Annual Reviews in Control* **28**(1), 115–136.
- Dormido, S. and F. Torres (2005). Aplicación de las tic's a la educación en automática. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Dormido, S., S. Dormido-Canto, R. Dormido and J. Sanchez (2005). The role of interactivity in control learning. *International Journal of Engineering Education* **21**(6), 1122–1133.
- Gasa, D., I. Garrido, R. Costa and L. Basañez (2005). Plataforma de ensayo remoto de controladores basados en autómatas finitos. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Gomes, L. and S. Bogosyan (2007). Ss on on e-learning and remote laboratories within engineering education - first part. *Trans. on Industrial Electronics* **54**(6), 3054–3056. Gomes:2007.
- Guzmán, J. L., F. Rodríguez, M. Berenguel and S. Dormido (2005). Laboratorio virtual para la enseñanza de control climático de invernaderos. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Hercog, D., B. Gergic, S. Uran and K. Jezernik (2007). A dsp-based remote control laboratory. *Trans. on Industrial Electronics* **54**(6), 3057–3068.
- ISA Distance Learning
- ISA Distance Learning (2003). *ISA Training Institute*.
- Jimenez, L.M., R. Puerto, O. Reinoso, C. Fernandez and R. Neco (2005). Recolab: Laboratorio remoto de control utilizando matlab y simulink. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 64–72.
- Katsura, S., K. Ohnishi and K. Ohishi (2007). Transmission of force sensation by environment quarryer based on multi-lateral control. *IEEE Transactions on Industrial Electronics* **54**(2), 898 – 906.
- Luo, R. C. and T. M. Chen (2000). Development of a multi-behaviour based mobile robot for remote supervisory control through the internet. *IEEE Transactions on Mechatronics* **5**(5), 376–385.
- López, J. (1999). Teleworking in a multinational environment. In: *Proceedings of the Sixth International Conference on Intelligence in Services and Networks*.
- Malo, Shane and Robert Griño (2008). Adaptive feed-forward cancellation control of a full-bridge dc-ac voltage inverter. In: *The International Federation of Automatic Control. Proceedings of the 17th World Congress*. Seoul, Korea. pp. 4571–4576.
- Marange, P., F. Gelot and B. Riera (2007). Remote control of automation systems for des courses. *Trans. on Industrial Electronics* **54**(6), 3103–3111.
- Marin, R., P. Sanz, P. Nebot and R. Wirz (2005). A multimodal interface to control a robot arm via the web: a case study on remote programming. *IEEE Transactions on Industrial Electronics* **52**(6), 1506–1520.
- Mata, Enric Xargay and Ramon Costa Castelló. (2005). Laboratorio virtual ejs para el estudio del control repetitivo.. In: *Actas de las IV Jornadas de Enseñanza a través de Internet/Web de la Ingeniería de Sistemas y Automática, EIWI-SA2005 (CEA-IFAC)*. Thomson. Granada. pp. 33–39. ISBN: 84-9732-451-X.
- McKee, G. T. and A. J. Gatward (2005). Escenarios online para robots: Diseño y utilización. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Norman, S. (1993). Will telework for us?. *Library Association Record*. **95**(2), 90–91.
- Online Manufacturing Courses from the Manufacturing Training Network (MTN)
- Online Manufacturing Courses from the Manufacturing Training Network (MTN) (2003). SAE International.
- Rösch, O. J., K. Schilling and H. Roth. (2002). Haptic interfaces for the remote control of mobile robots. *Control Engineering Practice* **10**, 1309–1313.
- Safaric, R., M. Debevc, R. M. Parkin and S. Uran (2001). Telerobotics experiments via internet. *IEEE Transactions on Industrial Electronics* **48**(2), 424–431.
- Sala, A., M. Vallés and J. L. Díez (2006). Remote tcp/ip-based process control with time-varying sampling period. In: *Proc. IEEE International Conference on Control Applications - CCA'06*. pp. 2081–2086.
- Valera, A., M. Vallés and J. L. Díez (2005a). Simulación y control de procesos físicos de forma remota. *Revista Iberoamericana de Automática e Informática Industrial* **2**(2), 3–7.
- Valera, A., M. Vallés, J. L. Díez and P. Albertos (2005b). Dynamic virtual and remote control laboratory development. *IEEE Control Systems Magazine* **25**(1), 35–39.
- Vallés, M., J. L. Díez and A. Valera (2001). Setting up a virtual matlab control laboratory. In: *IFAC-Workshop ED-COM2001: Education in Automatic Control*. Johannes Kepler University Linz (Austria).
- Vargas, H., J. Sánchez-Moreno, S. Dormido, C. Salzmann, D. Gillet and F. Esquembre (2009). Web-enabled remote scientific environments. *Computing in Science and Engg.* **11**(3), 36–46.
- Xargay, Enric and Ramon Costa Castelló (2004). Modelado de una planta diseñada para ilustrar el principio del modelo interno.. In: *XXV Jornadas de Automática (Editado por J.A. Somolinos, Ed.)*. Ciudad Real. ISBN 84-688-7460-4. Dep Legal: CR-388-2004.
- Yang, S.H., X. Chen and J.L. Alty (2003). Design issues and implementation of internet-based process control systems. *Contr. Eng. Practice* **11**, 709–720.

Zuluaga, C. A., C. G. Sánchez and E. A. Rodríguez (2005).
Laboratorio de automática vía internet (lavi). Revista Iberoamericana de Automática e Informática Industrial **2**(2), 3–7.