

XLIII Jornadas de **AUTOMÁTICA 2022**

Universidad de La Rioja
Escuela Técnica Superior de Ingeniería Industrial
7, 8 y 9 de septiembre de 2022
Logroño (La Rioja)

Libro de Actas



XLIII JORNADAS DE AUTOMÁTICA : LIBRO DE ACTAS

UNIVERSIDAD DE LA RIOJA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INDUSTRIAL

7, 8 Y 9 DE SEPTIEMBRE DE 2022

LOGROÑO (LA RIOJA)

EDITAN:

Servizo de Publicacións. Universidade da Coruña, A Coruña

Comité Español de Automática, Barcelona

Universidad de La Rioja, Logroño

2022

ORGANIZAN:

Comité Español de Automática
Universidad de La Rioja

EDITORES:

Carlos Balaguer Bernaldo de Quirós

José Manuel Andújar Márquez

Ramon Costa Castelló

Carlos Ocampo Martínez

Jesús Fernández Lozano

Matilde Santos Peñas

José Enrique Simó Ten

Montserrat Gil Martínez

Jose Luis Calvo Rolle

Raúl Marín Prades

Eduardo Rocón de Lima

Elisabet Estévez Estévez

Pedro Jesús Cabrera Santana

David Muñoz de la Peña Sequedo

José Luis Guzmán Sánchez

José Luis Pitarch Pérez

Oscar Reinoso García

Oscar Déniz Suárez

Emilio Jiménez Macías

Vanesa Loureiro Vázquez

DISEÑO DE CUBIERTA:

Universidad de La Rioja

DOI: <https://doi.org/10.17979/spudc.9788497498418>

ISBN: 978-84-9749-841-8

DEPÓSITO LEGAL: C 944-2022

THEMA: TJFM, TJF

CDU: 681.05(063)



© de esta edición: UDC, CEA, Universidad de La Rioja
© de los textos: los autores

ORGANIZADORES Y PATROCINADORES

Organizadores

Comité Español de Automática
Universidad de La Rioja, Escuela Técnica Superior de Ingeniería Industrial

Patrocinadores

Alecop
Ayuntamiento de Logroño
Colegio Oficial de Ingenieros Industriales de Aragón y La Rioja
Colegio Oficial de Ingenieros Técnicos Industriales de La Rioja
Comité Español de Automática (CEA)
Departamento de Ingeniería Eléctrica (UR)
Electromatic
Electrónica de Luis
Escuela Técnica Superior de Ingeniería Industrial (UR)
Fundación CEA
Fundación San Millán de la Cogolla
Gobierno de La Rioja
MathWorks
Omron
Recirsa
Smc
Universidad de La Rioja (UR)

Patrocinadores de CEA

Asti Mobile Robotics
Automática e Instrumentación
Idom
Ikerlan
Infaimon
Iqua Robotics
Isa Sección Española
MathWorks
Prodel
Robotnik
Schneider Electric
Siemens
Springer
Ubotica
Universidade da Coruña

Sistema portátil de bajo coste para la medición y representación de parámetros cinemáticos en 3D.....	926
<i>Juan Francisco Navarro Iribarne, David Moreno Salinas and José Sánchez Moreno</i>	
Módulo pasarela ROS/OPC UA para integración de datos de robot.....	934
<i>María del Mar Plaza Cano, Emima Ioana Jiva, Francisco Blanes Noguera, Patricia Balbastre Betoret and Francisco Fraile Gil</i>	
Plataforma Código Abierto para Generar Gemelos Digitales	941
<i>Alejandro Sánchez García, Elisabet Estévez Estévez, Ildefonso Ruano Ruano, Juan Gómez Ortega and Javier Gámez García</i>	
Integración de FACTORY IO y 4DIAC-FORTE para la validación de software de control en la norma IEC 61499	949
<i>Andrés Tendero-Vegas, Oscar Miguel-Escrig, Julio-Ariel Romero-Pérez, Bianca Wiesmayr, Virendra Ashiwal and Alois Zoitl</i>	

Visión por Computador

Predicción automática de la carga frutal de olivos empleando UAV y redes convolucionales	956
<i>Pablo Asensio Jimenez, Diego Manuel Martínez Gila, Silvia Satorres Martínez, Elisabet Estévez Estévez, Juan Gómez Ortega and Javier Gámez García</i>	
Deep learning classification applied to traffic accidents prediction.....	964
<i>Richard Coll-Josifov, Albert Masip-Álvarez and David Lavèrnia-Ferrer</i>	
Assessment of age estimation methods for forensic applications using non-occluded and synthetic occluded facial images.....	972
<i>Elouan Derenee Jeuland, Aitor Del Río Ferreras, Deisy Chaves, Eduardo Fidalgo Fernandez, Victor González-Castro and Enrique Alegre</i>	
Towards a Framework for the Democratisation of Deep Semantic Segmentation Models ...	980
<i>Rubén Escobedo and Jónathan Heras</i>	
Creación y análisis de vistas 360 a partir de un par de imágenes fisheye.....	985
<i>María Flores, David Valiente, Arturo Gil, Oscar Reinoso and Luis Payá</i>	
Evaluación por deep learning de la trayectoria de un asistente robotizado en cirugías nasales transesfenoidales	993
<i>Carlos Fontúrbel, Adrián Ayuso and Eusebio de la Fuente</i>	
Segmentación en imagen de frutos de granado usando deep learning con aplicación en agricultura de precisión	1001
<i>Jaime Giménez-Gallego, Juan D. González-Teruel, Ana B. Toledo-Moreo, Manuel Jiménez-Buendía, Fulgencio Soto-Valles and Roque Torres-Sánchez</i>	
Hacia los gemelos digitales térmicos de edificios con robots	1007
<i>Alejandro López-Rey, Amanda Ramón, Antonio Adán, Blanca Quintana Galera and Juan García Aguilar</i>	
Visión artificial para detección automática de altura del cultivo.....	1015
<i>Fatima Martínez, Jose Maria Manzano, James Romaine and Pablo Millan</i>	

PRÁCTICAS DE CONTROL DE ROBOTS PARALELOS EN EL ESPACIO DE ESTADOS USANDO UN LABORATORIO REMOTO

Adrián Peidró*, Luis Payá, Óscar Reinoso, Mónica Ballesta, Arturo Gil
 Grupo de Automatización, Robótica, y Visión por Computador, Universidad Miguel Hernández
 03202 Elche (Alicante), Spain

*Autor de correspondencia. Email: apeidro@umh.es

Resumen

Este artículo presenta un laboratorio remoto para realizar prácticas de control en el espacio de estados con dos robots paralelos reales, a través de Internet. Se presenta la arquitectura cliente-servidor de este laboratorio, en la que el cliente es una interfaz Java de control usada por el estudiante para comandar experimentos y visualizar resultados, mientras que el servidor es un PC industrial que controla ambos robots paralelos mediante una tarjeta de control dSPACE 1103 y una serie de componentes software que incluyen Matlab, Jmserv, y un gestor de reservas y de credenciales para garantizar un acceso seguro a los robots remotos. Se describen varias prácticas que pueden realizarse con este laboratorio remoto, como la identificación del modelo dinámico multivariable de los robots paralelos, el diseño de controladores integrales con estado realimentado, o el estudio de la pérdida de controlabilidad de dichos robots a medida que se aproximan a singularidades.

Palabras clave: robot paralelo, laboratorio remoto, control en el espacio de estado, controlabilidad, singularidades

1 INTRODUCCIÓN

Los laboratorios virtuales y remotos son plataformas software y ciberfísicas que permiten al estudiantado realizar prácticas y experimentos a través de Internet, evitando las limitaciones de los laboratorios presenciales tradicionales, que únicamente permiten al estudiantado realizar prácticas en horarios específicos y desplazándose en persona a dichos laboratorios. Los laboratorios virtuales permiten al estudiantado experimentar con un sistema completamente simulado, mientras que los laboratorios remotos le permiten experimentar con equipamiento físico real que se encuentra en las dependencias de la universidad. Cuando se desea que el estudiantado reciba una experiencia de aprendizaje más próxima a la que recibiría al realizar prácticas presenciales con equipos reales, se opta por los laboratorios remotos ya que éstos incorporan el manejo de dichos equipos a

distancia, y de este modo incorporan al experimento factores reales importantes que los laboratorios virtuales suelen omitir, como ruido en sensores, rozamientos difíciles de modelar, etc.

Los laboratorios remotos pueden incorporarse a la enseñanza y aprendizaje a distancia de prácticamente cualquier disciplina científico-técnica, pero la ingeniería de control y la robótica son algunas de las disciplinas en las que resulta más apropiado su uso. En el caso de los robots manipuladores industriales de tipo serie, se han implementado laboratorios remotos que permiten al estudiantado programar robots reales para tareas de pick-and-place [8], programación mediante teach pendant virtual [13], o planificación de trayectorias y visual servoing [12]. Incorporando la ingeniería de control, también pueden encontrarse laboratorios remotos para experimentar a distancia con el control de robots de tipo serie, ya sea control PID desacoplado para las distintas articulaciones [14,11,2], o controladores más generales, definidos con mayor libertad por el estudiantado [11,2].

Los laboratorios remotos de robots paralelos son más escasos que los de tipo serie. Los robots paralelos son manipuladores en los que la pinza está controlada por dos o más cadenas cinemáticas conectadas en paralelo. Estos robots poseen mayor rigidez y capacidad de carga que los de tipo serie, pero tienen un espacio de trabajo más limitado, y pueden encontrar singularidades en las que el robot no es controlable. En [10] se presenta un laboratorio remoto para experimentar con el control desacoplado de cada uno de los pistones de un robot paralelo. En [6] se describe otro laboratorio remoto para experimentar con el control de un robot paralelo de tipo cable, usando controladores PD, de pre-alimentación de su dinámica inversa, o combinaciones entre éstos. Por último, en [7] se presenta un laboratorio virtual que, pese a no incorporar equipos reales, permite estudiar la cinemática, dinámica y control de varios robots paralelos de tipo delta dispuestos en una línea de producción, con un elevado grado de realismo y similitud con una planta real, gracias a Unity 3D.

En este artículo presentamos un laboratorio remoto para experimentar con el control de robots paralelos

reales en el espacio de estado. Dado que los robots paralelos son sistemas multivariable con un elevado grado de acoplamiento entre distintos grados de libertad (dada su estructura de cadenas cinemáticas cerradas), su control clásico desacoplado no siempre proporciona el mejor comportamiento dinámico. Por el contrario, el control por realimentación del estado permite considerar el acoplamiento existente entre varias entradas y salidas, así como la controlabilidad del sistema, que en un robot paralelo se ve degradada a medida que el robot se aproxima a singularidades.

Este artículo está organizado como sigue. La Sección 2 describe los dos robots paralelos implementados en el laboratorio remoto. La Sección 3 describe en detalle la arquitectura cliente-servidor de dicho laboratorio. A continuación, la Sección 4 describe algunas prácticas que pueden realizarse a distancia con los robots implementados. Por último, la Sección 5 resume las conclusiones y trabajos futuros.

2 ROBOTS IMPLEMENTADOS

En esta sección se presentan los dos robots paralelos implementados en el laboratorio remoto desarrollado.

2.1 El robot 5R

El robot 5R se muestra en la Figura 1(a). Se trata de una cadena cinemática cerrada formada por cinco barras (una de ellas fija) unidas mediante cinco articulaciones de rotación (de ahí el nombre “5R”, donde “R” denota una articulación de rotación tipo bisagra). También puede interpretarse como dos brazos planares de tipo RRR unidos en su extremo final, en cuyo caso podríamos denotar este robot como “2RRR”. Las dos articulaciones conectadas con la barra fija están actuadas por motores de corriente continua. En este artículo, las longitudes de las barras son $l_1=l_2=l_3=l_4=0.2m$, y $l_5=0.48m$. Los ángulos (θ_1 y θ_2) que forman las barras l_1 y l_4 con la horizontal son las coordenadas articulares actuadas. Dichos ángulos están directamente controlados mediante los mencionados motores.

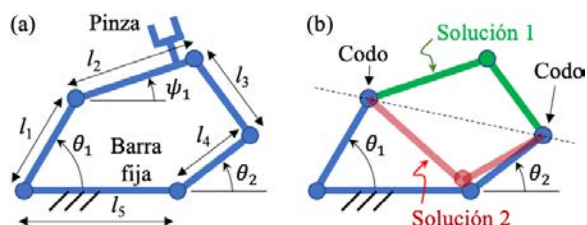


Figura 1: (a) Robot paralelo 5R. (b) Sus 2 soluciones.

Este robot se utiliza cuando se necesita controlar la posición de una pinza en el plano, sin importar su orientación. Se emplea, por ejemplo, en líneas de

producción para tareas de envasado o empaquetado de piezas o productos cuya proyección tiene silueta circular, cuya orientación es irrelevante para ser introducidos en su envase (por ejemplo, huevos).

La cinemática directa de este robot consiste en determinar su configuración cuando se conocen los ángulos θ_1 y θ_2 . Si se conocen dichos ángulos, el robot puede adoptar dos posibles soluciones que sitúan la pinza en posiciones simétricas respecto a la línea que pasa por los codos de ambos brazos (véase la Figura 1b). Si únicamente se miden los ángulos θ_1 y θ_2 , no se puede conocer cuál de las dos soluciones adopta el robot sin usar algún sensor redundante.

2.2 El robot 3RRR

El robot 3RRR se muestra en la Figura 2. Este robot consta de tres brazos de tipo RRR. Los extremos ABC de cada brazo están conectados a una base fija, mientras que los otros están conectados a una plataforma móvil DEF que transporta la herramienta o pinza para realizar tareas o transportar productos. En este artículo, todas las barras tienen una longitud de 0.2m, los puntos ABC que unen los brazos con la base fija forman un triángulo equilátero de lado 0.595m, y los puntos DEF que unen los brazos con la plataforma móvil forman otro triángulo equilátero de lado 0.2m. Las articulaciones unidas a la base fija están actuadas por motores de corriente continua, y dichos motores se encargan de controlar los ángulos θ_1 , θ_2 y θ_3 .

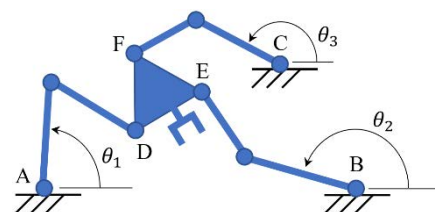


Figura 2: Robot paralelo 3RRR

Al igual que el anterior robot, éste también puede emplearse en tareas de pick-and-place y envasado o empaquetado de productos en el plano, pero el robot 3RRR puede controlar tanto la posición de la pinza como su orientación, de modo que le permite ajustar la orientación para insertar el producto transportado en su alojamiento en el envase, pudiendo así envasar productos con cualquier geometría (no solo circular).

De forma análoga al anterior robot, la cinemática directa del robot 3RRR consiste en determinar la configuración del robot cuando se conocen los ángulos θ_1 , θ_2 y θ_3 . No obstante, la cinemática directa del robot 3RRR es considerablemente más complicada que la del robot 5R. Esto se debe a que resolver la cinemática directa del robot 5R se reduce a resolver una ecuación cuadrática (de la que surgen las dos soluciones

simétricas ilustradas en la Figura 1b), mientras que la cinemática directa del robot 3RRR requiere resolver un polinomio de grado 6 que puede admitir 2, 4 o 6 soluciones reales distintas [5], dependiendo del valor que tengan los ángulos θ_1 , θ_2 y θ_3 . Por tanto, midiendo únicamente dichos ángulos no es posible saber cuál de esas soluciones es la adoptada por el robot, siendo necesario utilizar algún sensor redundante, como se explicará posteriormente en la Sección 3.3.

3 ARQUITECTURA

La Figura 3 muestra la arquitectura de este laboratorio, que va desde la aplicación cliente usada por el estudiante para visualizar el robot y comandar sus movimientos, hasta el propio robot real situado en las instalaciones de la universidad. Entre ambos extremos (estudiante y robot remoto) hay varios elementos que se describirán en las siguientes subsecciones.

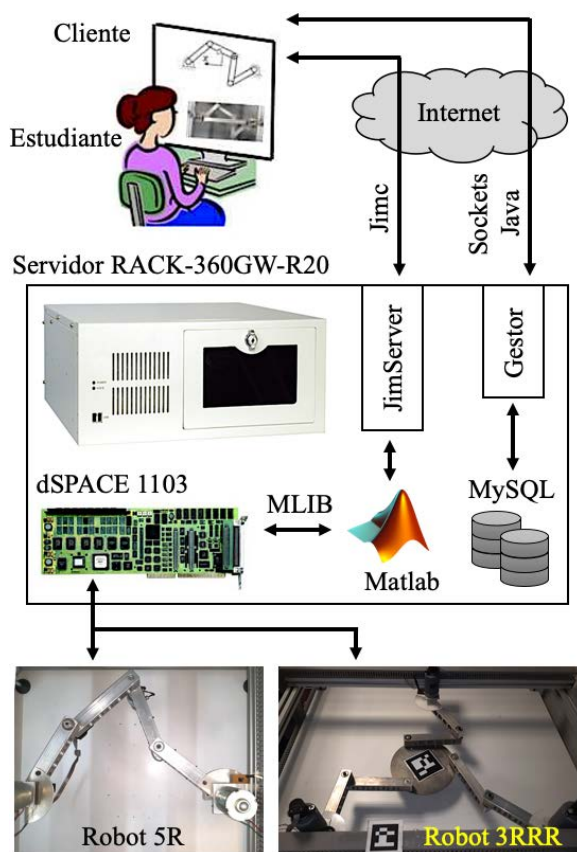


Figura 3: Arquitectura del laboratorio remoto

3.1 APLICACIÓN DEL CLIENTE

3.1.1 Autenticación y reserva de franja horaria

La aplicación del cliente consiste en una interfaz en Java, desarrollada mediante Easy Java Simulations [3], que consta de los siguientes elementos. La primera

pantalla mostrada al estudiante es una ventana de autenticación, donde cada estudiante debe ingresar su usuario y contraseña antes de poder utilizar el robot (Figura 4). Esta autenticación es necesaria por motivos de seguridad y exclusividad de uso, ya que el usuario manejará equipos reales que no pueden ser manejados de forma simultánea por más de un usuario, o de otro modo el robot podría recibir órdenes contradictorias a la vez. Cuando el usuario ingresa sus credenciales, éstas son enviadas al servidor, que las contrasta con una base de datos en MySQL.

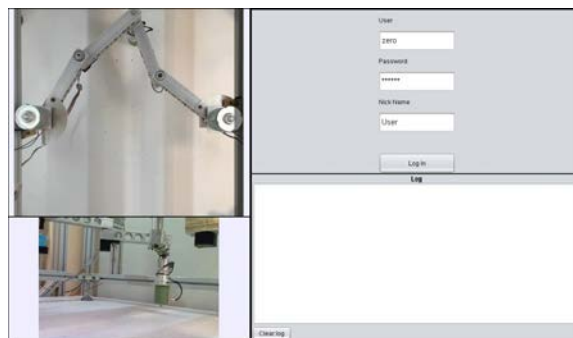


Figura 4: Pantalla de autenticación del estudiante

Cabe destacar que, a la izquierda de la pantalla de inicio de sesión, se dispone de una imagen en tiempo real del robot situado en las instalaciones de la universidad, captada por una webcam instalada sobre el robot. Esta imagen de webcam puede visualizarse incluso aunque el usuario no disponga de credenciales de acceso o, incluso teniéndolas, no disponga de una reserva de uso exclusivo del robot en la franja horaria actual. De este modo, el usuario puede ver el robot en movimiento si algún otro estudiante se encuentra utilizándolo, aunque solo podrá ver la imagen de la webcam del robot, y no las gráficas de resultados de los experimentos, que solo podrán ser visualizadas por el estudiante que tenga una reserva activa (estas gráficas se explicarán posteriormente).

Tras producirse el ingreso correcto al servidor, aparece una pantalla como la mostrada en la Figura 5, que permite al estudiante realizar la reserva de una franja horaria para la utilización exclusiva del robot (botón "Book"), o proceder a utilizar el robot si se dispone ya de una reserva para la franja actual de tiempo (botón "Use the robot"). Al clicar en el botón de reserva, se muestra una tabla horaria que permite al estudiante seleccionar un mes, día y hora para el uso exclusivo del robot. Se permite reservar una hora por día y usuario. Estas reservas se guardan y consultan en la misma base de datos MySQL utilizada para validar las credenciales de los usuarios.

Tras reservar una franja horaria y llegado el momento de la reserva, el usuario puede clicar en el botón "Use the robot", en cuyo caso el servidor procederá a

inicializar el robot (posicionándolo en una postura de referencia y calibrando sus sensores) para permitir al usuario la realización de experimentos remotos, mediante la interfaz de control que se describirá en el siguiente apartado.

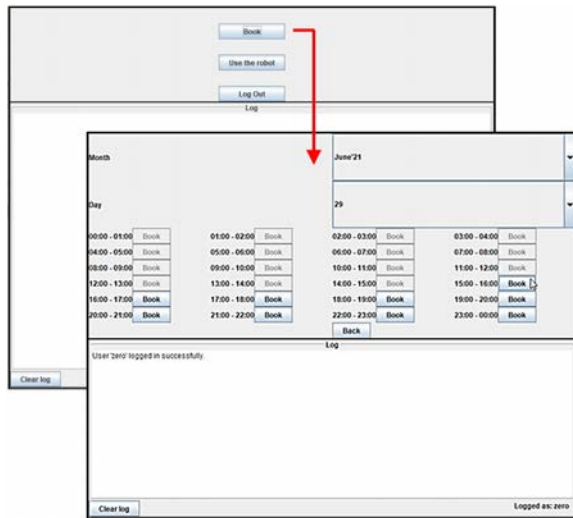


Figura 5: Reserva de una franja horaria

3.1.2 Interfaz de control del usuario

Una vez el usuario ha ganado control sobre el robot remoto, puede realizar experimentos mediante una interfaz gráfica de control como la mostrada en la Figura 6. Esta interfaz consta de tres secciones. En la sección superior se muestran las gráficas de las salidas del sistema. Para ambos robots, considerándolos como sistemas dinámicos multivariable o MIMO (Multiple Input - Multiple Output), las salidas son los ángulos girados por las articulaciones actuadas (θ_i en las Figuras 1 y 2). La sección intermedia de la interfaz muestra las correspondientes acciones de control, que son los voltajes aplicados a los motores de corriente continua encargados de mover las respectivas articulaciones actuadas. Haciendo clic derecho sobre estas gráficas, es posible desplegar una nueva ventana donde estas gráficas aparecen en formato de una tabla que puede exportarse a Excel, Matlab o cualquier otro programa para procesar los datos con el objetivo de realizar una identificación paramétrica u otros fines, según se explicará en la Sección 4 de este artículo.

Por último, la sección inferior dispone de varios campos numéricos en los que el estudiante puede introducir los parámetros de control para la realización del experimento. En primer lugar, puede introducir los incrementos deseados para las coordenadas articulares actuadas respecto a su valor actual, dichos incrementos se tratan como referencias en escalón a seguir por el lazo de control. A continuación, el usuario debe especificar las ganancias de control, pudiendo elegir dos opciones. La primera es un control

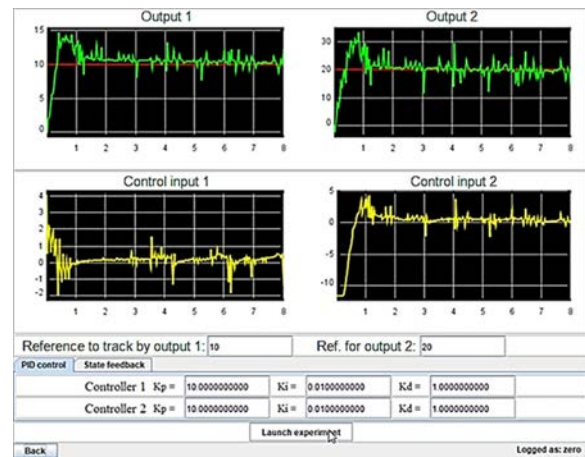


Figura 6: Interfaz para experimentos de control

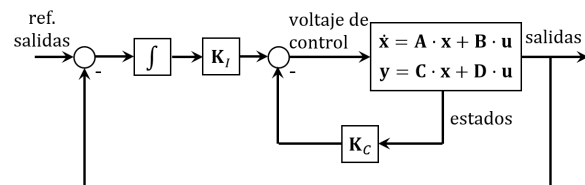


Figura 7: Control integral con estado realimentado

desacoplado PID clásico de cada articulación actuada. La segunda opción es un control multivariable por realimentación del estado (con ganancia matricial K_C) con regulador integral K_I para ajuste del régimen permanente, donde ya no se asume que las articulaciones están desacopladas (como sí asume el control PID) sino que existe acoplamiento dinámico entre ellas. El control multivariable por realimentación del estado es el mostrado en la Figura 7.

En ambos esquemas de control, todas las variables son incrementales respecto a la configuración adoptada por el robot antes de realizar el experimento de control. El trabajar con variables incrementales permite que el estudiantado utilice técnicas lineales de diseño clásico y moderno, una vez tenga identificado el modelo dinámico del robot. Evidentemente, el usar técnicas lineales de diseño provocará que el robot se comporte como se desea solo si las referencias introducidas al lazo de control son relativamente pequeñas. En efecto, si el estudiante introduce referencias demasiado grandes, estará solicitando al controlador que aleje el robot demasiado de su punto de trabajo inicial (en torno al cual se realiza el diseño del regulador), por lo que el regulador diseñado podría no funcionar como se espera.

3.2 SERVIDOR

El servidor es un ordenador industrial IEI Technology Corp., modelo RACK-360GW-R20, que incorpora una tarjeta de control dSPACE 1103 que se conecta directamente a los robots remotos que se describirán en la Sección 3.3. Dicha tarjeta dispone de numerosas entradas y salidas analógicas/digitales con las que leer encoders y otras señales digitales o analógicas, ejecutando el algoritmo de control y enviando a los drivers de los motores las tensiones de control necesarias. Además, el servidor ejecuta un conjunto de programas que permiten gestionar el acceso de los estudiantes, sus reservas, y los flujos de información entre el estudiante y el robot. Estos programas y flujos de información se resumen en la Figura 3, y a continuación se describen con mayor detalle.

3.2.1 Gestor de credenciales y reservas + MySQL

Un programa en Java, de elaboración propia, se ejecuta continuamente en el servidor a modo de gestor de credenciales y reservas, escuchando peticiones del cliente para validar sus credenciales (introducidas por el estudiante en la pantalla mostrada en la Figura 4), o para realizar la reserva de una franja horaria para uso exclusivo del robot. La comunicación entre este gestor y la interfaz de control del cliente se realiza mediante sockets de Java. El gestor de credenciales y reservas accede a dos bases de datos MySQL almacenadas en el propio servidor: una base de datos contiene las credenciales de los usuarios para poder validarse en el sistema, y la otra contiene todas las reservas realizadas por los usuarios.

3.2.2 JimServer

JimServer es un programa escrito en Java [4] que se ejecuta en el servidor. JimServer permite comunicar una sesión de Matlab corriendo en el servidor con la interfaz de usuario que se ejecuta en el ordenador del estudiante, de modo que dicha interfaz puede enviar comandos de Matlab para que se ejecuten en la sesión de Matlab del servidor, pudiendo intercambiar, además de dichos comandos, variables y datos de forma bidireccional. Los comandos de Matlab que se envían desde la interfaz de usuario del cliente para ser ejecutados en la sesión de Matlab del servidor, se usan para modificar parámetros del esquema de control de los robots remotos, como por ejemplo las ganancias de control o los setpoints, y también para registrar los resultados de ejecución de cualquier experimento de control (gráficas de las salidas del sistema y de las acciones de control) y devolverlas al usuario, para que éste pueda visualizarlos en forma de gráficas como se ha ilustrado en la Figura 6.

3.2.3 Matlab y Simulink

Por último, el último programa necesario para completar la comunicación entre los robots remotos y el usuario es Matlab y Simulink. El programa de control encargado de controlar ambos robots se encuentra programado en forma de diagrama de Simulink que alberga, entre muchos otros bloques, las ganancias de control y el setpoint a seguir por el robot. Este esquema de control se carga y ejecuta directamente sobre la tarjeta dSPACE 1103 antes mencionada. Cuando el usuario desea introducir un nuevo setpoint o referencia para las articulaciones del robot, o cuando se desean modificar las ganancias de control de acuerdo a lo descrito en la Sección 3.1.2, esos cambios se producen mediante la ejecución de comandos de la librería MLIB, que comunica Matlab con la tarjeta dSPACE. Dicha librería contiene comandos que permiten modificar cualquier parámetro del esquema de control Simulink, o extraer datos de la tarjeta. Estos comandos se envían desde la interfaz de control del cliente a través de JimServer, y se ejecutan en el Matlab local que corre en el servidor.

3.3 ROBOTS REMOTOS

Como se ha comentado anteriormente en la Sección 2, los dos robots paralelos reales implementados en el laboratorio remoto son los robots 5R y 3RRR. Ambos robots se muestran en la parte inferior de la Figura 3. Las dimensiones de los robots construidos son las que se han indicado en la Sección 2, siendo la masa de cada barra de 0.5 kg. Los robots están movidos por motores de corriente continua de 24 V, que están alimentados por servocontroladoras ESCON 50/5 del fabricante Maxon. Estas servocontroladoras tienen sus pines digitales y analógicos directamente conectados a la tarjeta dSPACE 1103 instalada en el ordenador servidor, y reciben de dicha tarjeta la tensión analógica de control que debe ser aplicada a las bornas de cada motor, según determine el esquema de control programado en dicha tarjeta a través de Simulink.

Como se ha comentado en la Sección 2, el robot 5R dispone de dos articulaciones actuadas, que son las conectadas directamente a la base fija del robot, mientras que el robot 3RRR dispone de tres articulaciones actuadas, que también son las situadas sobre la base fija del robot. En robots paralelos, es habitual elegir como articulaciones actuadas aquellas conectadas directamente a la base fija del robot para minimizar las masas e inercias en movimiento, que serían mucho mayores si los motores se colocaran sobre otras articulaciones. Para cualquiera de los dos robots, cada articulación actuada consta de un motor de corriente continua de 24 V y un encoder óptico modelo HEDR-55L2-BY09, que tiene una resolución de 3600 pulsos por vuelta.

Se ha prescindido de reductoras entre los motores y las articulaciones actuadas para evitar la holgura que normalmente introducen los engranajes de dichas reductoras, que pueden llegar a empeorar la precisión del robot y generar un comportamiento demasiado vibratorio. El hecho de prescindir de reductoras, además de suprimir las holguras, también permiten al robot alcanzar mayores velocidades, a costa de reducir el par motriz que pueden proporcionar los motores, aunque esto último no supone un problema ya que ambos robots se han construido para trabajar en planos horizontales, de modo que los motores no han de vencer el peso propio de las barras del robot.

Como se ha comentado anteriormente, cada articulación actuada está sensorizada mediante un encoder óptico que mide la posición angular de dicha articulación, con el fin de realimentar dicha posición. Los sensores ópticos empleados son incrementales, por lo que es necesario establecer un protocolo de calibración durante el arranque del robot, para encontrar la posición de referencia cero de cada articulación y así resetear los contadores encargados de contar pulsos de los encoders incrementales, para conocer el ángulo absoluto de cada articulación actuada (ángulo absoluto θ_i respecto a la horizontal, según se ha representado en las Figuras 1 y 2). Dicho protocolo de calibración inicial se basa en incorporar un sector circular solidario a cada articulación actuada, en combinación con sensores ópticos modelo CNY70. En el arranque del robot, si el sensor óptico está tapado, se controla la articulación en velocidad, haciéndola girar a baja velocidad en un sentido hasta que el sensor óptico quede destapado. O si se encuentra destapado, se hace girar en sentido contrario hasta que el sensor óptico queda tapado. Cuando se produce el cambio entre tapado y destapado, se resetea el contador del encoder y se fija así la posición cero de cada articulación, finalizando la calibración.

Por último, cada robot dispone de un sensor redundante adicional para conocer qué solución de la cinemática directa adopta, y así poder controlarlo correctamente. Como se ha comentado en la Sección 2, cada uno de estos robots posee varias soluciones a la cinemática directa, de modo que, conociendo únicamente los ángulos θ_i de las articulaciones actuadas, no es posible saber cuál de las soluciones es la adoptada actualmente por el robot. En el robot 5R, conociendo los ángulos de las articulaciones actuadas, el robot puede adoptar dos soluciones simétricas (Figura 1b). Para determinar su solución, se ha colocado un tercer encoder en el codo izquierdo, que permite conocer el ángulo ψ_1 y así conocer sin ninguna ambigüedad la solución adoptada. En el caso del robot 3RRR esto resulta más complicado, ya que éste puede adoptar hasta 6 soluciones distintas para un mismo valor de las articulaciones actuadas. En este caso, como se puede apreciar en la Figura 3, se han colocado

marcas ArUco en la base fija del robot y en su efector final, para estimar la pose relativa entre ambos, y se compara dicha pose relativa con las seis posibles soluciones teóricas que se obtendrían al resolver la cinemática directa a partir de los ángulos de las articulaciones actuadas medidos por los sensores.

4 PRÁCTICAS PROPUESTAS

El laboratorio remoto presentado está pensado para realizar prácticas de control con robots paralelos, especialmente prácticas en el contexto de asignaturas de robótica o control multivariable en el espacio de estados. A continuación, se describen tres posibles actividades prácticas que pueden realizarse con este laboratorio en el marco de dichas asignaturas.

4.1 PRÁCTICA 1: IDENTIFICACIÓN

La interfaz de usuario diseñada permite al estudiante registrar las entradas y salidas del robot paralelo remoto, entendido éste como un sistema dinámico multivariable, donde las entradas son los voltajes aplicados a los motores de corriente continua y las salidas son los ángulos de las articulaciones actuadas (θ_i). Disponiendo de las señales de entradas y salidas de un experimento, es posible realizar una identificación del modelo dinámico del robot en el espacio de estados, ya sea de un modelo no-lineal más preciso, o bien de un modelo linealizado en torno a la posición inicial del experimento.

Para realizar este experimento de identificación con la interfaz de control diseñada, se puede proceder de la siguiente manera. Primero, se debe diseñar un regulador R_0 que permita estabilizar el robot en la configuración en la que se desea estimar el modelo dinámico linealizado. Dicho regulador R_0 puede diseñarse por ensayo y error, y no tiene por qué satisfacer especificaciones dinámicas concretas, bastará con que logre estabilizar al robot en la configuración deseada con error nulo en régimen permanente. Partiendo de dicha configuración, y con el mismo controlador, se aplica un pequeño escalón a las referencias del lazo de control, desplazando así ligeramente el robot de su posición inicial. La interfaz de usuario registrará y mostrará las gráficas de evolución de las salidas del sistema (ángulos de las articulaciones actuadas), así como las gráficas de los voltajes que el regulador R_0 ha aplicado a los motores para controlar este desplazamiento. Disponiendo de la entrada de control al sistema multivariable, y de su salida, es posible exportar esas gráficas como tablas de datos, que pueden procesarse para identificar un modelo dinámico linealizado del robot en el espacio de estado (matrices **A**, **B**, **C**, **D**), en torno a la configuración de partida del experimento. Tal

identificación puede realizarse, por ejemplo, de forma automatizada mediante el comando “ssest” de Matlab, o puede realizarse una identificación intentando ajustar los datos registrados a un modelo construido a mano que quede en función de masas, inercias o rozamientos a estimar por mínimos cuadrados.

Nótese que este experimento de identificación se realiza con el sistema en bucle cerrado con un regulador R_0 que estabilice al robot, ya que los robots son inestables en bucle abierto: al introducirles un voltaje constante, sus articulaciones adquieren una velocidad constante en régimen permanente, por lo que las salidas (que son la integral de dichas velocidades) tendrían un comportamiento no acotado.

4.2 PRÁCTICA 2: CONTROLABILIDAD

Los robots paralelos implementados en este laboratorio remoto, como la mayoría de robots, exhiben un comportamiento no-lineal entre sus entradas y salidas dinámicas (voltajes de los motores y desplazamientos angulares de sus articulaciones). No obstante, resulta de interés, en asignaturas de control moderno, estudiar dichos robots en torno a una configuración determinada, lo cual permite aproximar su comportamiento mediante un modelo linealizado que podría estimarse mediante un procedimiento de identificación en bucle cerrado como se ha descrito en la Sección 4.1.

El estudiar la dinámica de un robot paralelo como un modelo linealizado en torno a un punto de trabajo también tiene otro interés en asignaturas de robótica y de control en el espacio de estados, y es que permite estudiar cómo varía la controlabilidad del robot en distintas configuraciones de éste. Es bien sabido que un robot paralelo puede encontrarse con singularidades típicamente llamadas de “tipo 2” o de “tipo paralelo”, en las que no todos los estados son controlables [9]. La teoría de control moderno para sistemas lineales permite estudiar la controlabilidad calculando la matriz de controlabilidad Q y estudiando su número de condición. Cuanto mayor sea dicho número de condición, más cerca estará Q de no tener rango máximo. En el límite, cuando el robot alcanza una singularidad de tipo 2, Q pierde rango, su número de condición tiende a infinito, y existen direcciones del espacio de estados en las que el robot no puede controlarse mediante los voltajes de control [9].

Teniendo en cuenta lo anterior, esta segunda práctica que se propone consiste en que el estudiante mueva el robot a varias configuraciones de su espacio de trabajo, identifique el modelo dinámico linealizado del robot en torno a cada una de esas configuraciones, y estime el número de condición de la matriz de controlabilidad en cada configuración, comprobando que dicho número de condición aumenta en la

proximidad de las singularidades, denotando una degradación en la controlabilidad del robot.

4.3 PRÁCTICA 3: DISEÑAR REGULADOR

Por último, y partiendo también de un modelo dinámico lineal identificado en torno a una configuración de partida, en esta tercera práctica se debe diseñar un regulador por realimentación del estado con ajuste del régimen permanente mediante control integral, como el mostrado en la Figura 7. El objetivo es diseñar las ganancias K_c y K_i de dicho esquema para lograr que, en bucle cerrado, el sistema exhiba un determinado comportamiento dinámico (tiempo de establecimiento y sobreoscilación) y un error nulo en régimen permanente. El método sugerido para el diseño de las ganancias es el de asignación de polos mediante la fórmula de Bass-Gura [1]. Pueden realizarse distintas pruebas a la hora de elegir los polos dominantes y no-dominantes a fijar, así como a la hora de asignar cuántos estados controlará cada entrada, con el fin de comprobar cómo algunas elecciones se acercan más al comportamiento deseado que otras.

5 CONCLUSIONES

En este artículo hemos descrito la arquitectura de un laboratorio remoto diseñado para realizar prácticas de control multivariable en el espacio de estados con dos robots paralelos planos. Se ha descrito la arquitectura del laboratorio, identificando sus diversos componentes, y se han descrito varias prácticas susceptibles de realizarse con dichos robots a través de Internet. Las prácticas propuestas pretenden experimentar con la identificación de robots paralelos como sistemas MIMO, estudiar su controlabilidad en distintos puntos de su espacio de trabajo, y diseñar reguladores por realimentación del estado.

En el futuro, incorporaremos más robots paralelos a este laboratorio remoto, así como nuevas prácticas que trabajen con modelos no-lineales de dichos robots.

Agradecimientos

Este trabajo ha sido financiado por el Centro de Inteligencia Digital de Alicante (CENID) a través del proyecto “Hacia la formación práctica ubicua y digital en robótica mediante laboratorios remotos”.

English summary

PRACTICES OF STATE-SPACE CONTROL OF PARALLEL ROBOTS USING A REMOTE LABORATORY

Abstract

This paper presents a remote laboratory for doing practices of state-space control with two real parallel robots over the Internet. The client-server architecture of this laboratory is presented, where the client is a Java control interface used by the student to command experiments and visualize results, whereas the server is an industrial PC that controls both parallel robots through a dSPACE 1103 control board and a series of software components including Matlab, Jmsserver and a book manager that guarantees safe access to the remote robots. Different lab practices that can be done with this remote laboratory are described, such as the identification of the multivariate dynamic model of the parallel robots, the design of integral controllers with state feedback, or the study of loss of controllability of the robots when they approach singularities.

Keywords: parallel robot, remote laboratory, state-space control, controllability, singularities

Referencias

- [1] Bourlès, H., & Kwan, G. K. (2013). Linear systems. John Wiley & Sons.
- [2] Castellanos, A. S., Santana, L. H., Rubio, E., Ching, I. S., & Santonja, R. A. (2006). Virtual and remote laboratory for robot manipulator control study. *International Journal of Engineering Education*, 22(4), 702.
- [3] Esquembre, F. (2015). Facilitating the creation of virtual and remote laboratories for science and engineering education. *IFAC-PapersOnLine*, 48(29), 49-58.
- [4] Farias, G., De Keyser, R., Dormido, S., & Esquembre, F. (2010). Developing networked control labs: A matlab and easy java simulations approach. *IEEE Transactions on Industrial Electronics*, 57(10), 3266-3275.
- [5] Gosselin, C. M., & Sefrioui, J. A. O. U. A. D. (1991, June). Polynomial solutions for the direct kinematic problem of planar three-degree-of-freedom parallel manipulators. In *Fifth Intl. Conf. on Advanced Robotics* (pp. 1124-1129). IEEE.
- [6] Korayem, M., Maddah, S. M., Taherifar, M., & Tourajizadeh, H. (2014). Design and programming a 3D simulator and controlling graphical user interface of ICaSbot, a cable suspended robot. *Scientia Iranica*, 21(3), 663-681.
- [7] Li, C., Fu, L., & Wang, L. (2018, June). Innovate engineering education by using virtual laboratory platform based industrial robot. In *2018 Chinese Control and Decision Conf.* (pp. 3467-3472).
- [8] Marín, R., Sanz, P. J., Nebot, P., & Wirz, R. (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.
- [9] Peidro, A., Paya, L., Valiente, D., Gil, A., & Reinoso, O., (2022) Design of a simulation tool to study the controllability and state-space control of a parallel robot, *INTED2022 Proceedings*, pp. 3790-3800.
- [10] Santana, I., Ferre, M., Izaguirre, E., Aracil, R., & Hernandez, L. (2012). Remote laboratories for education and research purposes in automatic control systems. *IEEE transactions on industrial informatics*, 9(1), 547-556.
- [11] Temelta, H., Gokasan, M., & Bogosyan, S. (2006). Hardware in the loop robot simulators for on-site and remote education in robotics. *International Journal of engineering education*, 22(4), 815.
- [12] Torres, F., Candelas, F. A., Puente, S. T., Pomares, J., Gil, P., & Ortiz, F. G. (2006). Experiences with virtual environment and remote laboratory for teaching and learning robotics at the University of Alicante. *International Journal of Engineering Education*, 22(4), 766.
- [13] Tzafestas, C. S., Palaiologou, N., & Alifragis, M. (2005, June). Experimental evaluation and pilot assessment study of a virtual and remote laboratory on robotic manipulation. In *Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005.* (Vol. 4, pp. 1677-1684). IEEE.
- [14] Wu, M., She, J. H., Zeng, G. X., & Ohyama, Y. (2008). Internet-based teaching and experiment system for control engineering course. *IEEE Transactions on Industrial Electronics*, 55(6), 2386-2396.



© 2022 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC-BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).