

Plataforma Distribuida para la Realización de Prácticas de Robótica Móvil a través de Internet

Luis Payá, Oscar Reinoso, Arturo Gil y Luis M. Jiménez

Universidad Miguel Hernández, Departamento de Ingeniería de Sistemas Industriales,
Avda. de la Universidad s/n, Ed. Torreblanca, 03202 Elche (Alicante)-España
(e-mail: {lpaya, o.reinoso, arturo.gil, luis.jimenez}@umh.es)

Resumen

Este artículo presenta una plataforma distribuida que permite a los alumnos acceder a los robots disponibles en el laboratorio a través de Internet. Mediante esta plataforma, se crea un entorno remoto con el que los alumnos pueden llevar a cabo diferentes experimentos sobre varios equipos, con un horario flexible. De este modo, los usuarios pueden crear algoritmos básicos de control reactivo y de robótica colaborativa y probarlos sobre plataformas robóticas reales. El sistema implementado esta siendo utilizado actualmente para la realización de prácticas de robótica móvil por parte de estudiantes de ingeniería. Los estudiantes pueden probar los algoritmos que diseñan sobre equipos reales y pueden acceder a todos los servicios que ofrecen los robots, aprovechando además las ventajas que tiene la educación a través de Internet.

Palabras clave: robótica móvil, robótica colaborativa, educación en ingeniería, laboratorio remoto

Distributed Platform for Training in Mobile Robotics through Internet

Abstract

This paper presents a distributed platform that allows the students to access the available robots in the laboratory through Internet. With this platform, a remote environment is created so that the students are able to carry out several experiments using the laboratory equipment and with time flexibility. In this way, the users can create basic reactive control and collaborative robotics algorithms and test them on actual robotic platforms. The system is presently being used by engineering students in their labs on mobile robotics. The students can test the algorithms designed on real equipment and can access all the services provided by the robots. At the same time, the platform takes advantage of all the potentialities of the learning process using Internet.

Keywords: mobile robotics, collaborative robotics, engineering education, remote laboratory

INTRODUCCIÓN

La formación experimental de los estudiantes juega un importante papel en la educación en ingeniería. De este modo, se deben proporcionar todos los recursos necesarios que permitan al estudiante poner en práctica todo el conocimiento que ha adquirido durante el estudio teórico de las diferentes materias. Tradicionalmente, el elemento que hace posible este aprendizaje es el laboratorio, donde el estudiante puede realizar los experimentos necesarios para conocer los instrumentos y equipos que encontrará durante su vida laboral. Sin embargo, el laboratorio tradicional presenta una serie de carencias que dificultan el proceso de aprendizaje del alumno; requiere la presencia de los estudiantes y profesores en un horario predeterminado y el tiempo disponible para las sesiones prácticas está limitado. Además, el coste de establecimiento y mantenimiento del laboratorio suele ser alto, con lo cual, el contenido de las clases prácticas estará fuertemente condicionado por el número de equipos disponibles.

El uso de técnicas basadas en Internet supone varias ventajas, solucionando los problemas antes expuestos. Los estudiantes pueden acceder a los laboratorios en un horario totalmente libre, desde su propia casa, y disponiendo de todo el tiempo que requieran para alcanzar los objetivos de cada sesión práctica. Además, los estudiantes pueden acceder individualmente a los equipos, independientemente del número disponible de ellos. También, utilizando un sistema de evaluación online, el estudiante puede conocer los resultados de su evaluación en tiempo real y el profesor puede tener en cuenta no sólo los resultados finales sino el trabajo que el alumno ha llevado a cabo realmente para conseguir dichos resultados. Este sistema permitiría incluso que varios centros compartan los equipos de prácticas y por tanto, los gastos asociados, como muestran Berger y Topol (2001). Además, los estudiantes, están altamente motivados para hacer uso de los recursos disponibles en entornos remotos a través de Internet (Stafford, 2005).

Las tecnologías de la información y las comunicaciones se han venido aplicando en los últimos tiempos de forma exitosa a la docencia en control (Sánchez, 2001; Dormido, 2004). Una de las posibles aplicaciones son los laboratorios remotos, entornos distribuidos que permiten a los alumnos acceder remotamente a través de Internet a los equipos disponibles en el laboratorio. Cabe diferenciar este concepto del de laboratorio virtual, en el cual se simula el funcionamiento de dichos equipos, pero éstos no existen físicamente. Ejemplos de estos sistemas virtuales se presentan en los trabajos de Demetriou y Lambert (2006), que compara varios sistemas de modelado virtual y simulación de robots, proponiendo una nueva arquitectura para simulación de robots manipuladores y en el de Athanasiou et al. (2000), que presenta un entorno de simulación de robots móviles.

En este trabajo se presenta un laboratorio remoto que ha sido desarrollado con el objetivo de que los alumnos puedan hacer uso de los robots móviles disponibles en el laboratorio a través de Internet. Dentro del área de 'Educación en Control', se han desarrollado diferentes laboratorios remotos. Cassini et al. (2003) presentan un laboratorio remoto para control en automática que permite al usuario diseñar su propio controlador por medio del entorno MATLAB/Simulink y testarlo a través de una interfaz. Carusi et al. (2004), extienden este mismo concepto al uso, de forma remota, de un robot móvil. Pastor et al. (2003), muestran una estructura que permite laboratorios Web centrados en el aprendizaje de sistemas de control. RECOLAB (Jiménez et al., 2005) es otro laboratorio remoto similar en el cual se utiliza un servomotor para testar el controlador diseñado por los alumnos en un entorno remoto. Por otro lado, también han sido desarrollados varios laboratorios remotos con el objetivo de manejar robots en el entorno remoto. Candelas et al. (2003), presentan un laboratorio virtual para realización de prácticas en robótica que permite tanto la simulación de un brazo robot como la tele-operación del robot real equivalente. Thamma et al. (2004), describen una arquitectura cliente/servidor para el control remoto de un robot manipulador, haciendo uso de programación en Java utilizando TCP y sockets y accediendo al servidor usando cualquier navegador web. Khamis et al. (2003), han desarrollado la arquitectura 'Developer' para la teleoperación y control del robot B21r y por último, Siegwart y Saucy (1999), presentan y discuten varias aplicaciones que manejan robots móviles a través de Internet en EPFL, en Lausanne.

El aspecto novedoso de la plataforma propuesta en este trabajo consiste en que se dispone de un equipo heterogéneo de diferentes robots móviles que el alumno debe poder utilizar para testar los

algoritmos desarrollados. Por otro lado, también se debe permitir la interacción de un robot con el resto de los mismos llevando a cabo una monitorización de todos ellos mientras se está ejecutando una tarea. Cada robot debe poder interactuar con los algoritmos que están corriendo en el resto de robots activos en la plataforma.

En el presente trabajo se describe la plataforma desarrollada para la comunicación con los diferentes tipos de robots, mostrando los detalles de la implementación sobre el robot WiFiBot. Tras ello, se detalla el uso del sistema desde el punto de vista del usuario final, y para finalizar, se exponen las prácticas propuestas a los alumnos, los resultados obtenidos y las conclusiones del trabajo.

ARQUITECTURA DEL SISTEMA

El sistema de comunicaciones que se presenta a continuación ha sido diseñado con el objetivo de hacer más sencillo y estandarizar el acceso a los servicios que provee cada robot. Esta arquitectura admite la inclusión de nuevos agentes móviles sin tener que modificar la estructura global.

Agentes móviles.

La arquitectura de comunicaciones que se ha planteado tiene como objetivo monitorizar y controlar a un grupo heterogéneo de robots móviles con distintas habilidades sensoriales y arquitecturas internas. (Paya et al., 2006a) En concreto, se dispone de cuatro modelos diferentes de robots móviles, que se muestran en la figura 1.

B21r es un robot de 4 ruedas y cinemática síncrona. Lleva dos computadores a bordo con el sistema operativo Linux y el software Mobility (librerías basadas en CORBA 2.0 para comunicación con el robot). Dicha comunicación se realiza mediante un enlace ethernet inalámbrico. El robot cuenta con varios tipos de sensores: sonar, infrarrojos, láser y encoders. Además, lleva en su parte superior una unidad pan-tilt con dos cámaras Sony XC999 paralelas.

Robots WiFiBot: Cuentan con una cámara color DCS 900 y dos sensores infrarrojos de distancia. También están equipados con un procesador x86 AMD a 20 MHz encargado de controlar sus 4 motores de forma independiente. Además, se ha instalado un PC a bordo, con procesador Pentium III funcionando con sistema operativo Linux. Estos robots se comunican con el exterior mediante una comunicación WIFI 802.11b/g.

Robots EyeBot: Estos robots funcionan con un Microcontrolador Motorola 68332 a 25 MHz y cuentan con 1 MB de memoria RAM. Pueden capturar imágenes en color mediante una cámara con la que pueden extraer una representación simplificada del entorno por el que se están moviendo. También pueden obtener medidas de distancia utilizando dos sensores infrarrojos. Estos robots se comunican utilizando emisores/receptores de Radiometrix (BIM).

Robots RugWarrior: Estos robots cuentan con un microcontrolador Motorola 68HC11. Disponen de sensores infrarrojos, con los que detectan la presencia de obstáculos cercanos. Tienen capacidades sensoriales muy limitadas.

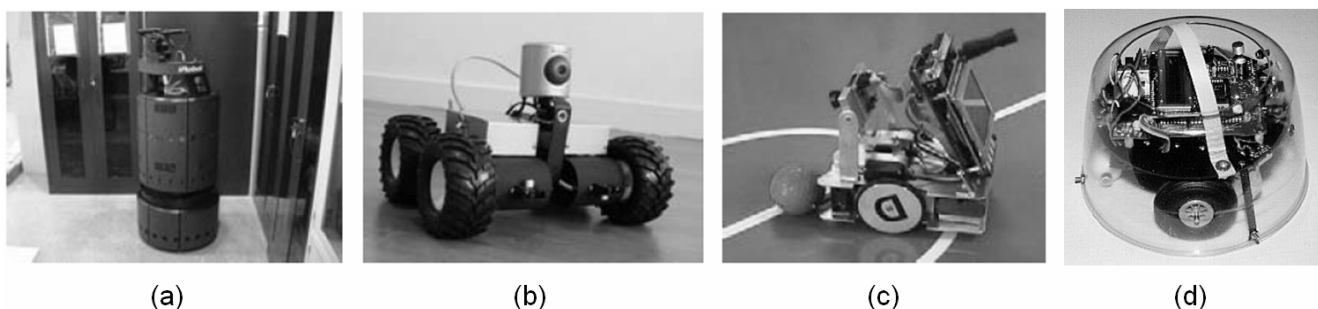


Fig. 1: Robots utilizados en la plataforma. (a) B21r, (b) WiFiBot, (c) EyeBot y (d) RugWarrior.

Arquitectura de la red de comunicaciones

Debido a las características heterogéneas del conjunto de robots móviles, se ha precisado el diseño de una arquitectura de red que integre diferentes tecnologías de comunicación vía radio y diferentes protocolos de red. La arquitectura, que se muestra en la figura 2, se estructura en tres subredes:

- 1) Red cableada Fast Ethernet a la que se conectan los computadores, servidores, routers y cámaras fijas. A esta red se conectará la herramienta de teleoperación que permitirá gestionar todos los agentes móviles del entorno remoto.
- 2) Red WIFI 802.11b/g que comunica los robots de tipo medio (iRobot B21r, y WifiBot).
- 3) Red radio diseñada e implementada específicamente para comunicar los robots de menor capacidad (Eyebot, RugWarrior) utilizando transceivers de Radiometrix (BIM) en la banda UHF de 433MHz.

Protocolos de comunicación

Se ha diseñado un protocolo específico de nivel aplicación orientado al control distribuido de robots móviles para la realización de tareas cooperativas. En la especificación del mismo se ha tenido en cuenta la posibilidad de trabajar de forma transparente con los diferentes dispositivos involucrados en este tipo de aplicaciones: minirobots con capacidad de procesamiento limitada, grandes robots móviles con CPUs embarcadas, cámaras fijas y móviles, computadores externos, servidores de datos, etc. Así, a la hora de programar una aplicación para el manejo de un robot en concreto, el alumno no debe preocuparse del funcionamiento interno de cada robot.

El carácter heterogéneo de los elementos utilizados ha precisado la utilización de diferentes interfases de programación pero manteniendo un protocolo común de aplicación. La figura 3 muestra la arquitectura básica diseñada. Los dispositivos más complejos que pueden conectarse a través de la red WIFI o Ethernet mediante TCP/IP utilizan para la comunicación de datos, dependiendo del dispositivo, la librería de sockets o bien una librería que da soporte al estándar CORBA. La arquitectura CORBA proporciona una metodología orientada a objetos bien definida para la implementación de aplicaciones distribuidas y constituye el modelo de referencia utilizado en el diseño general (OMG, 1995). El estándar CORBA ha sido utilizado para tareas similares de comunicación (Utz et al., 2004), demostrando ser eficiente para intercambiar información en equipos de robots heterogéneos de la liga RoboCup F2000.

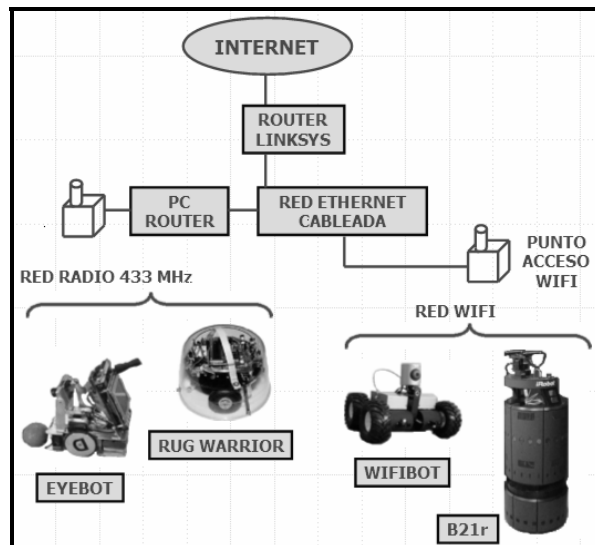


Fig. 2: Arquitectura general del sistema de comunicaciones

Los elementos principales de la red de comunicaciones se detallan a continuación:

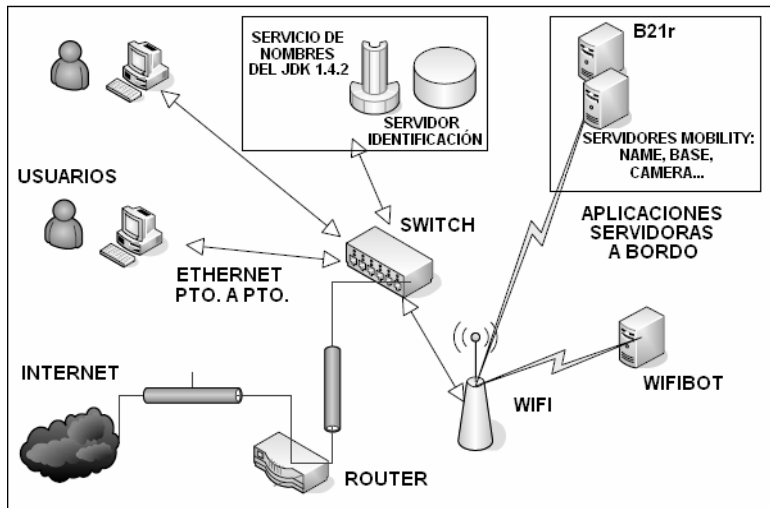


Fig. 3: Arquitectura básica diseñada

En primer lugar, a bordo de cada robot se encuentra un servidor que accede directamente al hardware de los robots (cámaras, sensores, motores, etc.). Estos servidores proporcionan una serie de servicios que se implementan mediante interfaces CORBA. En concreto, cada interfaz está implementada mediante un objeto de C++ que accede a los sensores del robot o comanda sus movimientos. El estándar CORBA permite acceder a cada servicio a través de una cadena de texto llamada IOR (Inter-operable Object Reference).

Por otro lado, se encuentra el servidor de identidad y ejecución, que se ejecuta en un PC conectado a la red y cuya dirección IP es conocida por todos los elementos del sistema. Este servidor tiene dos funciones principales. La primera de ellas consiste en mantener un listado de todos los robots que se encuentran activos en el sistema y qué interfaces ofrece cada uno de ellos. En concreto, para cada robot, el servidor de identidad y ejecución almacena una cadena de texto que identifica al robot, junto con una lista de los servicios que proporciona (cámara, sensores sónico, sensores infrarrojos, etc.) almacenando también una cadena IOR para cada uno de ellos. La segunda función principal es la de ofrecer un control de acceso de las aplicaciones cliente a los robots. Dada la naturaleza de la aplicación, un único cliente debe acceder al robot para comandarlo, debiendo quedar el resto de aplicaciones clientes a la espera. Además, una aplicación cliente no debe acaparar la ejecución indefinidamente. Este problema se ha resuelto de la siguiente manera: cuando una aplicación cliente desea acceder a un robot, el servidor de ejecución comprueba que esté libre y le devuelve las referencias IOR de las interfaces que solicite. A continuación, nuevas peticiones de otros clientes serán puestas en espera. Transcurrido cierto tiempo (10 min. en la implementación actual), el servidor de ejecución envía una orden al servidor del robot para reiniciar los servidores a bordo y le da control sobre el robot a la siguiente aplicación cliente que lo solicita.

Por último, las aplicaciones cliente se ejecutan en varios PC que funcionan bajo Linux Debian. Los alumnos se conectan a los PC Linux de forma remota y lanzan un módulo de monitorización/supervisión que centraliza la información recogida por todos los robots. Así, se recogen parámetros como, por ejemplo, el estado de cada robot, las lecturas de sus sensores, la trayectoria planificada en el entorno o la tarea que está llevando a cabo. Este módulo actúa como cliente CORBA de todos los elementos existentes en el sistema, haciendo peticiones de forma periódica a cada uno de ellos. Además, el módulo de monitorización o supervisión se encarga del control de la misión del equipo de robots. De esta manera, gestiona el reparto de tareas entre el grupo de robots móviles. En la figura 5 se muestra la pantalla principal del monitorizador, que se describe más profundamente más adelante.

APLICACIÓN AL ROBOT WIFIBOT

En las prácticas diseñadas hasta el momento, se utilizan principalmente los robots WiFiBot, debido a su tamaño, versatilidad y capacidad de procesamiento a bordo. Además de todos los servicios ya

comentados, en el propio robot WiFiBot se ha implementado un servidor puente multi-hilo que se ejecuta en la tarjeta PC del robot, y permite que varios programas accedan simultáneamente a los servicios que proporciona el controlador SC12. Ha sido implementado usando el mismo protocolo que el SC12 de modo que la misma clase cliente puede comunicarse con el servidor del controlador y con el servidor puente, especificando únicamente la IP y puerto correspondientes. De hecho, el servidor puente hace uso de esta clase para enlazar con el SC12. La ventaja de este servidor multihilo es que varios clientes pueden acceder a los servicios a bajo nivel que el robot ofrece. Por ejemplo, podrían estar ejecutándose simultáneamente el programa que controla el robot junto con un programa para monitorizar su evolución. Sin embargo, cuando un cliente está controlando el robot, el resto de clientes que se conecten únicamente pueden monitorizarlo.

Este servidor establece un enlace con el SC12 usando la clase cliente y se deja un socket abierto a la espera de posibles clientes. Cuando un nuevo cliente accede al sistema, se crea un hilo de ejecución independiente para atender a este cliente, como se muestra en la figura 4. Los diferentes hilos de ejecución se ejecutan en paralelo, compartiendo el mismo espacio en memoria. Para evitar posibles errores de comunicación con el SC12, que podrían producirse cuando varios clientes realizan una petición simultáneamente, debe utilizarse un mecanismo de sincronización. Para la construcción de este servidor, se ha utilizado un mecanismo de exclusión mutua (mutex), que bloquea los fragmentos de código donde se realizan las peticiones al SC12, de modo que únicamente un hilo puede estar ejecutando este fragmento de código al mismo tiempo. El resto de hilos son bloqueados hasta que el primero de ellos finaliza el proceso.

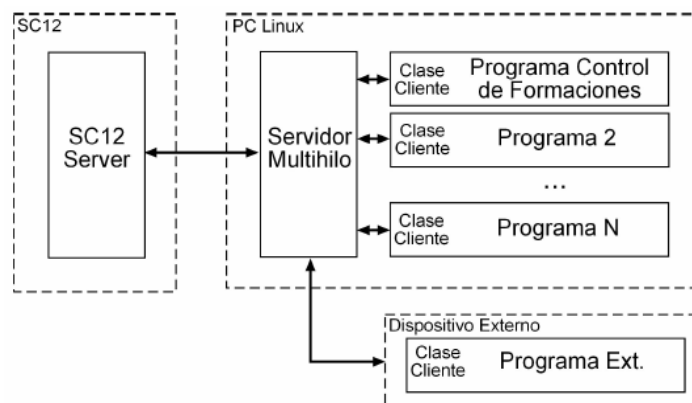


Fig. 4: Funcionamiento del servidor multihilo

USO DEL SISTEMA

Para permitir el acceso remoto a los robots se ha desarrollado una aplicación Java GUI. El sistema puede generar eventos basándose en tres causas diferentes; cuando el cliente interactúa con los componentes gráficos, a través de un temporizador que sincroniza los procesos y mediante un hilo de ejecución que, realizando peticiones CORBA al servidor de identificación, controla la presencia de los robots en el sistema. Esta aplicación actúa como un cliente CORBA, de modo que es capaz de requerir servicios a los actuadores y sensores de los robots. Cuando la aplicación cliente comienza su ejecución, el cliente debe configurar los datos necesarios para el servidor de identificación (IP y puerto del servidor de nombres e intervalo de tiempo para la sincronización de las peticiones). Dicho servidor realiza una inspección del sistema, en busca de los equipos disponibles, y tras ello, se muestra un árbol que contiene los robots disponibles y los servicios que ofrecen. Además, se asigna a los robots disponibles una interfaz gráfica en caso de que exista. Si no se encuentra ningún robot disponible, la aplicación queda a la escucha de una nueva conexión. La figura 5 muestra la interfaz gráfica correspondiente al robot WiFiBot. En la parte superior de la ventana se muestran diversos controles para mover el robot manualmente; con los botones de las flechas, las velocidades lineal y angular del robot pueden ser incrementadas o decrementadas, y el botón central detiene al robot. El incremento es configurable. Además, se muestran las velocidades lineal y angular actuales. En la parte inferior de la ventana, hay un componente panel con varias solapas. Estas solapas dan acceso a los siguientes paneles (que se muestran en la figura 6):

Panel Posición. El usuario puede resetear la odometría del robot e inicializarla a un valor predeterminado. Además, se posibilita el ir leyendo y almacenando continuamente el valor de la odometría en un fichero a elección del usuario.

Panel Infrarrojos. Se muestran las distancias frente a cada uno de los sensores infrarrojos y dichas distancias también se pueden ir almacenando continuamente en un fichero.

Panel Cámara Web. El usuario puede visualizar las imágenes capturadas por la cámara Web. Cuando se pulsa el botón 'Stop', la cámara deja de capturar y la última imagen tomada se puede almacenar en un fichero jpg.

Panel Algoritmo. A través de este panel, el usuario puede cargar el programa que ha desarrollado para controlar el robot de manera que pueda comprobar su funcionamiento. En primer lugar, el usuario tiene que buscar el fichero fuente C++ a enviar. Una vez lo tiene localizado, pulsando el botón 'Compile', el fichero es transferido y compilado en el servidor remoto (el usuario no posee las librerías necesarias para compilarlo en su PC), y aparece una ventana que muestra al cliente los resultados en la compilación. Si no se ha detectado ningún error en el proceso, al pulsar el botón 'Run', el algoritmo cargado comenzará su ejecución. En cualquier momento, el usuario puede detener esta ejecución y parar el robot usando el botón 'Stop'.

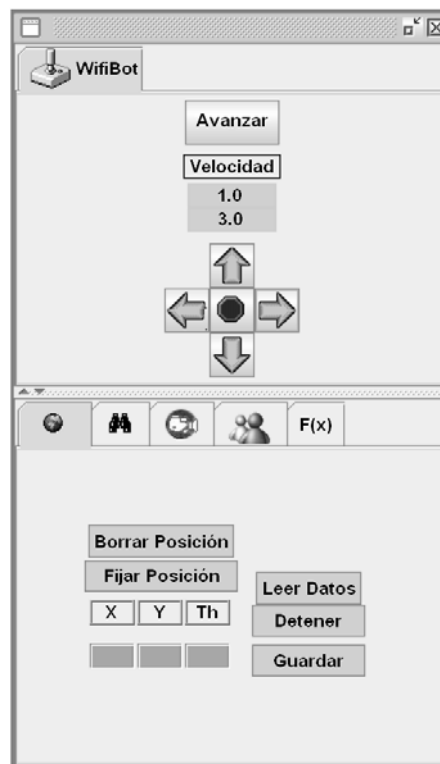


Fig. 5: Aplicación para comunicación con los robots

PRÁCTICAS DESARROLLADAS

Los servicios descritos en los apartados anteriores han permitido diseñar un nuevo conjunto de prácticas y experimentos a realizar por parte de los alumnos de la titulación de Ingeniería Industrial en la Universidad Miguel Hernández de Elche. En concreto, y dentro de la asignatura optativa de 4º curso "Control de Robots y Sistemas Sensoriales", se han venido utilizando durante el último curso académico este tipo de servicios, lo que ha posibilitado la realización de varias prácticas a distancia por parte de los alumnos.

Para el acceso a las prácticas vía Internet, el alumno únicamente debe registrarse y validarse dentro de la base de datos de que dispone para darle permiso de acceso. Una vez validado, el alumno

puede realizar las sesiones de prácticas disponibles. Los estudiantes pueden construir algoritmos de control reactivo básico y probarlos sobre los robots disponibles. Leyendo los valores proporcionados por los sensores y la cámara, y teniendo en cuenta la cinemática de cada robot, el estudiante debe programar su comportamiento de manera que realice una tarea concreta. Durante la navegación del robot, todos los datos relativos a su evolución pueden ser almacenados en varios ficheros de texto de manera que el estudiante, al final del proceso, será capaz de obtener gráficamente, la evolución de las variables necesarias (trayectoria, velocidad, etc.).

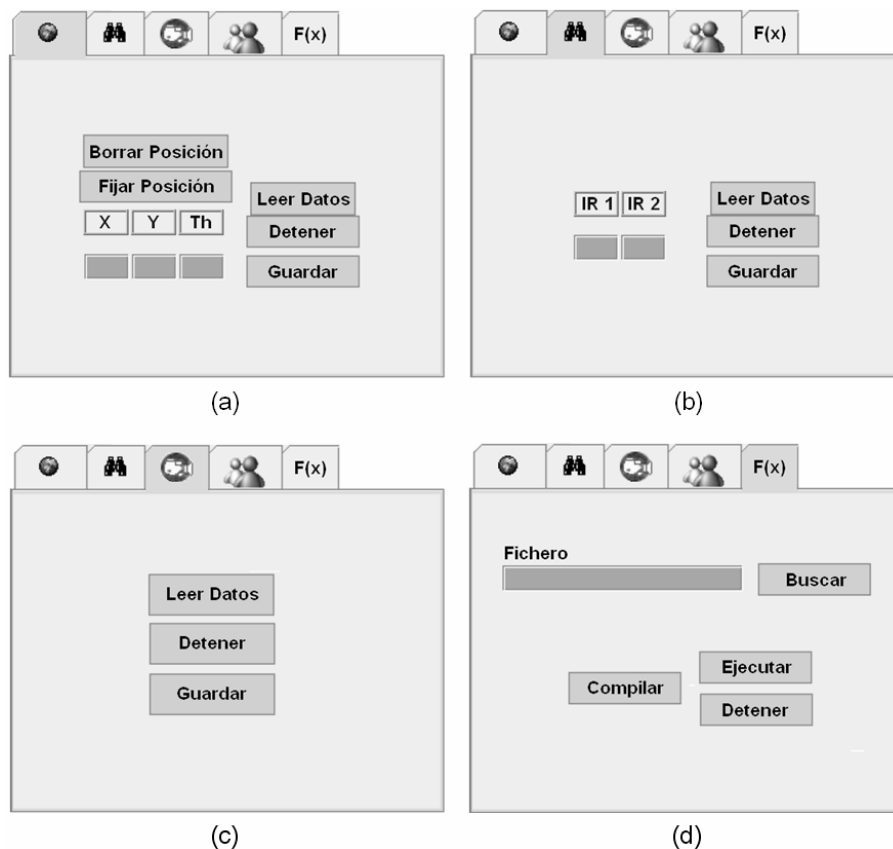


Fig. 6: Paneles para monitorización y control de WiFiBot. (a) Posición, (b) Sensores Infrarrojos, (c) Captura de imagen y (d) Descarga de un nuevo programa al robot

Actualmente se encuentran disponibles cuatro tipos de prácticas distintas, que abarcan distintos campos de conocimiento dentro de la robótica móvil. En una primera sesión introductoria presencial, se proporciona al alumno la información necesaria sobre los robots disponibles y los servicios de que dispone cada robot. Asimismo, se ponen a su disposición diversas plantillas y ejemplos que muestran el formato de utilización de los métodos implementados (lectura de sensores, captura de imágenes y control del robot). Asimismo, durante esta sesión se implementa un algoritmo sencillo, consistente en generar una trayectoria en línea recta del robot, girando a la derecha cuando encuentre un obstáculo. Este ejemplo permite al alumno introducirse en el manejo del equipo, creación del programa, descarga, compilación, etc. El resto de prácticas propuestas son:

Control reactivo del robot mediante coordinación competitiva

El alumno debe implementar un programa con el objetivo de mover el robot desde un punto inicial hasta un punto final, evitando varios obstáculos durante el recorrido hasta el destino, utilizando las lecturas de los infrarrojos. En todo momento, se trata de que el alumno haga que el robot se mueva en línea recta hacia el punto destino, girando cuando encuentra un obstáculo hacia el lado que esté más distante de dicho obstáculo y volviendo posteriormente a orientarse hacia el destino una vez superado el obstáculo (figura 7). Para ello, se utilizará exclusivamente información de la odometría. De este modo, se pretende que el alumno resuelva el problema mediante un control reactivo del robot utilizando dos comportamientos que se activan o no en función de la lectura de los infrarrojos: Ir

a destino y Evitar Obstáculo. La idea es que el problema se resuelve utilizando un problema de coordinación competitivo, con un único comportamiento activo en cada instante.

Comparación odometría – posición real

El objetivo de la segunda sesión de prácticas consiste en comparar los datos de trayectoria proporcionados por la odometría con la trayectoria real del robot. En la anterior práctica, los alumnos habrán comprobado la imprecisión de la odometría y la acumulación de error que supone si en ningún momento se toma una referencia externa. El objetivo de la segunda sesión consiste en resolver el mismo problema de la práctica anterior, pero para calcular la orientación necesaria hacia el punto destino, se utilizará la información proporcionada por una cámara situada en el techo y enfocando hacia el suelo. Sobre el robot se coloca una marca consistente en un círculo de color rojo, de modo que el objetivo del comportamiento Ir a destino será, en cada instante, extraer la posición del robot y proporcionar la acción de control que lo haga tender al destino (cuando esté activo el comportamiento Ir a destino).

Control reactivo mediante coordinación cooperativa

Dentro del entorno se encuentra una plantilla formada por cinco puntos negros situados en los vértices y en el centro de un cuadrado. Se trata de que el alumno consiga que el robot se mueva sobre sí mismo hasta que localice dicha plantilla a partir del sistema de visión embarcado y en ese momento tienda hacia ella situándose a cierta posición relativa y orientación respecto a la misma.

Se propone al alumno que resuelva el problema utilizando una arquitectura de control reactiva mediante un mecanismo de coordinación cooperativo entre los siguientes comportamientos: Ir a destino, Evitar obstáculos y Buscar referencia. Una futura ampliación de esta práctica consistiría en realizar un seguimiento de robots, en el que los alumnos mueven manualmente uno de los robots, que lleva una marca en su parte posterior y otro robot debe realizar un seguimiento del primero. Este problema implicaría un estudio de los retardos que se producen durante el control del robot.

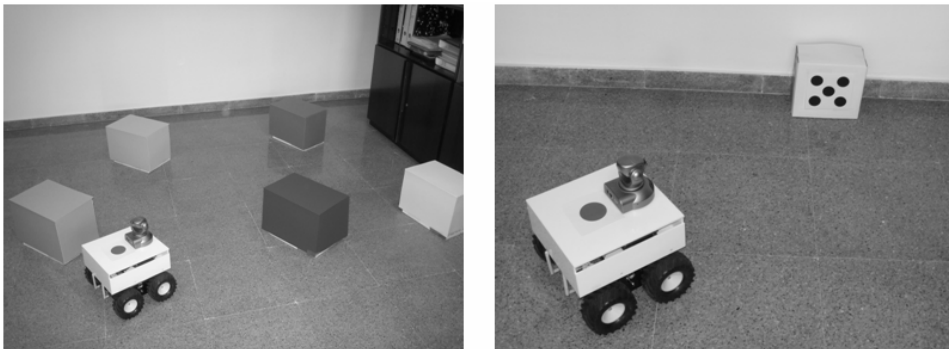


Fig. 7: Entorno por el que se mueve el robot.

Creación de una base de datos visual del entorno.

La última práctica disponible hasta el momento consiste en la creación, por parte del alumno, de una base de datos con imágenes desde varios puntos del entorno, comprimiendo esta información por medio de PCA. Posteriormente, se debe realizar una localización topológica dentro del entorno, proyectando la imagen actual en la base y comprobando cual de las almacenadas en la base de datos es la más cercana. Esta práctica se plantea como optativa.

RESULTADOS ALCANZADOS

La realización de prácticas a distancia permite que los alumnos puedan realizar este tipo de experimentos en cualquier momento del día y cualquier día que los dispositivos físicos se encuentren operativos, incluidos los fines de semana sin atender a un horario previamente planificado a principios de curso. En el gráfico de la figura 8 se observan los datos acerca del número de horas

promedio por sesión dedicado por parte de los alumnos en cada una de las cuatro prácticas diseñadas y configuradas. Como se observa, el valor promedio del número de horas es muy superior al destinado en las sesiones presenciales del curso previo (dos horas para cada una de las dos primeras prácticas y 4 horas para el resto). Esto se justifica dado que los alumnos se encuentran más predispuestos a la realización de las prácticas sin atender en demasía al tiempo que le dedican.

En el gráfico de la figura 9 se observa el número de sesiones de conexión que han realizado los alumnos para completar una práctica. Esto permite evidenciar que los alumnos (en promedio) prefieren realizar la práctica en diferentes intervalos de tiempo en lugar de dedicarle un tiempo excesivamente largo. Obviamente estos datos pueden verse alterados en función del número de conexiones realizadas a la vez dado que el sistema acota el tiempo de acceso al usuario en estas condiciones. Si bien no es posible constatar a largo plazo la ventaja de realizar prácticas a distancia por parte de los alumnos dado que únicamente se disponen de datos acerca del último curso académico, sí que se observa una mayor predisposición por parte de los alumnos para la realización de las prácticas docentes. Además las encuestas periódicas realizadas a los estudiantes muestran la alta predisposición por parte de los alumnos a la realización de este tipo de prácticas a distancia.

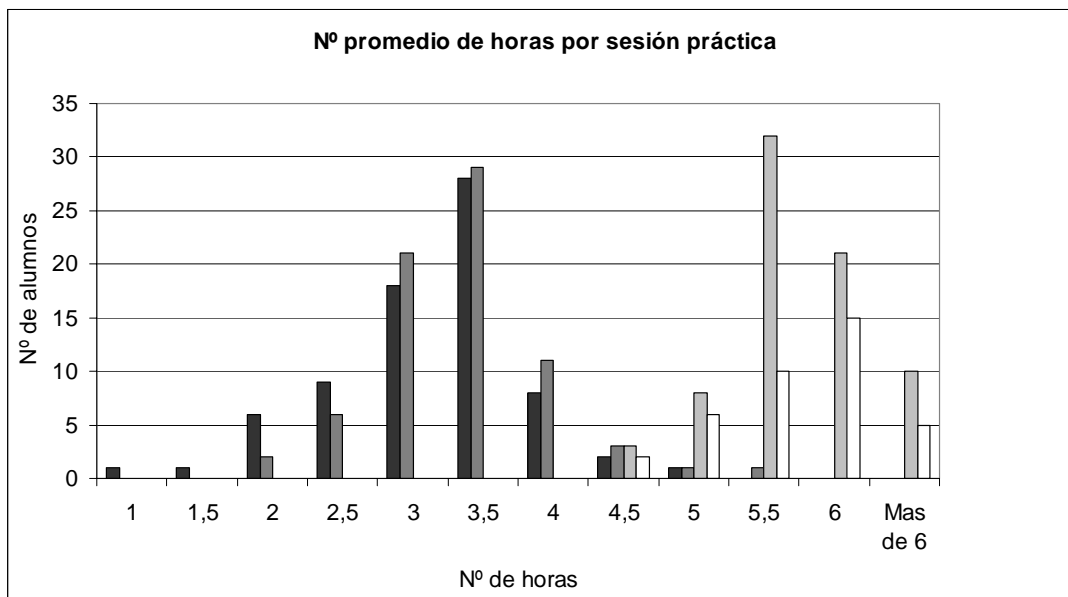


Fig. 8: Número promedio de horas dedicadas por parte de los alumnos a cada una de las sesiones (■ sesión 1, ■ sesión 2, ■ sesión 3, □ sesión 4)

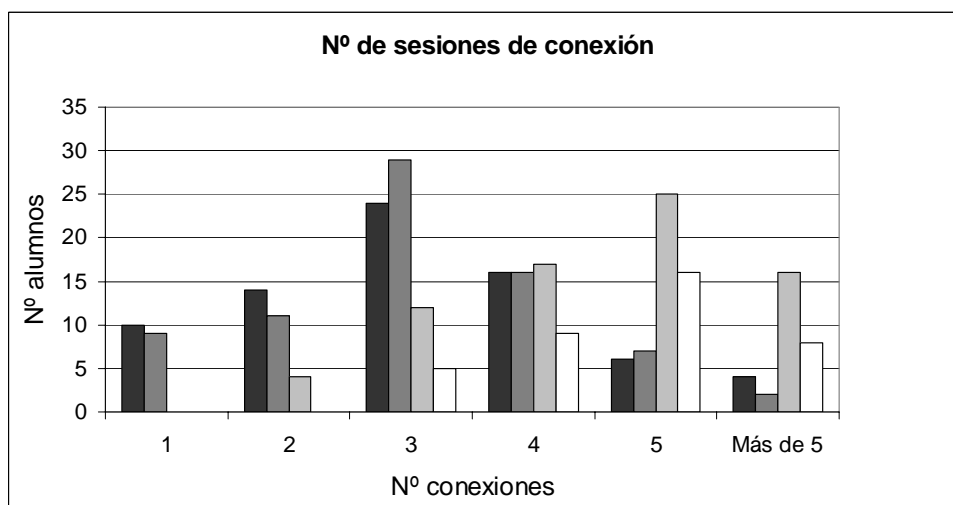


Fig. 9: Número de sesiones de conexión realizadas por los alumnos en cada práctica (■ sesión 1, ■ sesión 2, ■ sesión 3, □ sesión 4)

Para finalizar, cabe destacar que la plataforma implementada ha sido utilizada en una tarea de mantenimiento de formaciones en un equipo de varios robot WiFiBot, basándose en comportamientos. Para esta tarea, que implica colaboración entre robots, la plataforma ha resultado altamente eficiente, proporcionando muy buenos resultados (Payá et al., 2006b).

CONCLUSIONES

En este trabajo se ha presentado una plataforma distribuida para la comunicación con y entre los miembros de un equipo heterogéneo de robots. Esta plataforma permite la monitorización y control de estos robots, de forma remota, a través de Internet, y de forma totalmente transparente para el usuario, que maneja todos los robots usando la misma interfaz gráfica. Además, se posibilita la realización de tareas que impliquen colaboración entre robots, lo cual requiere que se comparta información entre los miembros del equipo.

El sistema implementado esta siendo utilizado actualmente para la realización de prácticas de robótica móvil por parte de estudiantes de ingeniería. Los estudiantes pueden probar los algoritmos que diseñan sobre equipos reales y puede acceder a todos los servicios que ofrecen los robots, aprovechando además las ventajas que tiene la educación a través de Internet.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por el Ministerio de Educación y Ciencia a través del proyecto DPI2004-07433-C02-01 'Herramientas de Teleoperación Colaborativa. Aplicación al Control Cooperativo de Robots', y por el proyecto PCT-G54016977-2005 'Robots Cooperativos para la vigilancia e inspección de edificios e instalaciones industriales'.

REFERENCIAS

- Athanasiou, S., I. Poulakis, P. Tsanakas y N. Koziris; *TALOS: An interactive Web-enabled educational environment on mobile robot technology*, Actas del 10º Congreso Electrotécnico del Mediterráneo, 1, 387-390, Lemesos, Chipre, 29 al 31 de Mayo (2000).
- Berger, K.A. y M.T. Topol; *Technology to enhance learning. Use of a web site platform in traditional classes and distance learning*, Marketing education review: 11, 15-26 (2001).
- Candelas, F.A. y otros cinco autores; *A virtual laboratory for teaching robotics*, Int. J. Engineering Education: 19(3), 363-370 (2003).
- Carusi, F., M. Casini, D. Prattichizzo y A. Vicino; *Remote control of a Lego mobile robot through the web*, Actas del Taller de Educación en Control basada en Internet, Grenoble, Francia, 5 al 7 de Septiembre (2004).
- Casini, M., D. Prattichizzo y A. Vicino; *The automatic control telelab: a user-friendly interface for distance learning*, IEEE Transactions on Education: 46(2), 252-257 (2003).
- Demitriou, A.G. y A.H. Lambert; *Virtual environments for robotics education: an extensible object-orientation platform*, IEEE Robotics and Automation: 12(4), 75-91 (2006).
- Dormido, S.; *Control learning: present and future*, Annual Reviews in Control: 28(1), 115-136 (2004).
- Jiménez, L.M. y otros cuatro autores; *RECOLAB: Laboratorio remoto de control utilizando Matlab y Simulink*, Revista Iberoamericana de Automática e Informática Industrial: 2(2), 64-72 (2005).
- Khamis, A., D.M. Rivero, F. Rodríguez y M. Salichs; *Pattern-based architecture for building mobile robotics remote laboratories*, Actas del Congreso Internacional de Robótica y Automatización del IEEE, 3, 3284-3289, Taipei, Taiwan 14 al 19 de Septiembre (2003).

- OMG, Object Management Group; *Common Object Request Broker: Architecture and Specification*. Revision 2.0. (1995).
- Pastor, R., J. Sánchez y S. Dormido; *A XML Framework for Development of Web-based Laboratories focused on Control Systems Education*, Int. J. of Engineering Education: 19(3), 445-454 (2003).
- Payá, L. y otros cuatro autores; *Distributed platform for the control of the WiFiBot robot through Internet*, 7º Simposio IFAC de Avances en Educación en Control, Madrid, España, 21 al 23 de Junio (2006a).
- Payá, L. y otros cuatro autores; *Behaviour-based multi-robot formations using computer vision*, Actas de la 6ª Conferencia Internacional en Visualización, Creación y Procesamiento de Imágenes, 488-493, Palma de Mallorca, España 28 al 30 de Agosto (2006b).
- Sánchez, J.; *Un nuevo enfoque metodológico para la enseñanza a distancia de asignaturas experimentales: análisis, diseño y desarrollo de un laboratorio virtual y remoto para el estudio de la automática a través de Internet*, Tesis Doctoral, Universidad Nacional de Educación a Distancia, Madrid, España (2001).
- Siegwart, R. y P. Saucy; *Interacting mobile robots on the Web*. Actas del Taller de Retos actuales en Robótica a través de Internet, Conferencia Internacional de Robótica y Automatización, Detroit, USA 10 al 15 de Mayo (1999).
- Stafford, R. F.; *Understanding motivations for Internet use in distance education*, IEEE Transactions on Education: 49(2), 301-306 (2005).
- Thamma, R., L.H. Huang, S.J. Lou, y C.R. Diez; *Controlling robot through Internet using Java*, J. Industrial Technology: 20(3), 25-33 (2004).
- Utz, H., F. Stulp y A. Moeld; *Sharing belief in teams of heterogeneous robot*, Actas del simposio de RoboCup-2004, 508-515, Lisboa, Portugal 4 a 5 de Julio (2004).