

## Un Banco de Pruebas Remoto para Experimentación en Robótica Ubicua

Adrián Jiménez-González, José Ramiro Martínez-de Dios\*, Alberto de San Bernabé, Gabriel Núñez, Aníbal Ollero

*Grupo de Robótica, Visión y Control, Departamento de Ingeniería de Sistemas y Automática, Escuela Técnica Superior de Ingeniería, Camino de los Descubrimientos s/n, 41092 Sevilla, España.*

### Resumen

La escasez de herramientas experimentales adecuadas ha sido señalada como un factor crítico en el desarrollo de la robótica ubicua. Este artículo describe un banco de pruebas remoto para experimentación en robótica ubicua integrando robots con redes inalámbricas de sensores. El banco de pruebas está compuesto por 5 robots Pioneer 3-AT, por una red de cámaras y por una red inalámbrica de sensores, todos ellos equipados con una amplia variedad de sensores. La arquitectura del banco de pruebas permite una gran diversidad de esquemas de cooperación entre sus elementos abstrayendo sus particularidades y capacidades sensoriales, computacionales y de comunicaciones. Permite ejecutar experimentos con distintos grados de descentralización. El banco de pruebas, instalado en la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla, es abierto y hace posible el acceso de forma remota mediante una interfaz gráfica intuitiva. Copyright © 2014 CEA. Publicado por Elsevier España, S.L. Todos los derechos reservados.

### Palabras Clave:

Bancos de pruebas, robótica ubicua, redes inalámbricas de sensores

### 1. Introducción

En los últimos años han surgido en el ámbito de los sistemas distribuidos un cierto número tecnologías que permiten desarrollar sistemas, denominados ubícuos, que permiten integrar dispositivos de pequeño tamaño y de bajo coste que cooperan formando redes para realizar una misión común. En la robótica ubicua se integran sistemas robóticos y tecnologías ubicuas, tales como las redes de cámaras o las redes inalámbricas de sensores, en inglés Wireless Sensor Networks (WSN).

La cooperación entre robots móviles y redes de sensores distribuidos ha atraído recientemente importantes esfuerzos de investigación en los ámbitos académico e industrial. La movilidad y capacidad de carga de los robots permite mejorar los rangos, robustez y precisión de la percepción obtenida mediante redes de sensores estáticos. Por otro lado, las medidas tomadas por sensores distribuidos permiten mejorar la percepción de los robots. Los robots también pueden emplearse para desplegar

sensores, e.g. cuando la infraestructura estática es inexistente o ha sido dañada. La cooperación entre robots y sensores distribuidos se ha empleado en vigilancia (Li and Parker, 2008), monitorización (Maza et al., 2011), localización (Sichitiu and Ramadurai, 2004), despliegue de la red (Popa and Lewis, 2008), reparación de la red (Corke et al., 2006), recogida de datos de sensores dispersos (Shah et al., 2003), entre muchos otros.

La cooperación entre dispositivos muy heterogéneos, tales como robots, redes inalámbricas de sensores y redes de cámaras, presenta retos importantes. Sin embargo, es dicha heterogeneidad la que origina complementariedades de interés en muchas aplicaciones. Una de las principales dificultades en la investigación en robótica ubicua reside en la escasez de bancos de pruebas adecuados para validar y comparar métodos. Aunque existen un buen número de bancos de pruebas para sistemas multi-robot y para redes de sensores, el número de los que permiten experimentos de robótica ubicua, donde cooperan robots y sistemas ubícuos, es aún muy escaso.

Este artículo describe un banco de pruebas remoto para probar, comparar y validar técnicas y algoritmos de robótica ubicua. El banco de pruebas integra robots móviles, redes de cámaras y redes inalámbricas de sensores. Su arquitectura *software* permite la comunicación bidireccional entre todos ellos, lo que le dota con la capacidad para implantar un gran número de experimentos cooperativos entre robots y la infraestructura ubicua. Tiene una arquitectura modular y emplea herramientas de

\* Autor en correspondencia.

Correos electrónicos: [adrianjmnz@gmail.com](mailto:adrianjmnz@gmail.com) (Adrián Jiménez-González), [jdedios@cartuja.us.es](mailto:jdedios@cartuja.us.es) (José Ramiro Martínez-de Dios), [adesanbernabe@us.es](mailto:adesanbernabe@us.es) (Alberto de San Bernabé), [gabnungle@alum.us.es](mailto:gabnungle@alum.us.es) (Gabriel Núñez), [aollero@cartuja.us.es](mailto:aollero@cartuja.us.es) (Aníbal Ollero)

URL: <http://www.esi.us.es/~jdedios/> (José Ramiro Martínez-de Dios), <http://grvc.us.es/aollero/> (Aníbal Ollero)

código abierto e interfaces estandarizadas que permiten la incorporación sencilla de nuevos componentes. Es accesible de forma remota mediante una interfaz gráfica que permite la programación, configuración, ejecución y monitorización de experimentos. Este banco de pruebas, instalado en una sala de 500 m<sup>2</sup> en el sótano del edificio de la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla, ha sido empleado para experimentar los métodos desarrollados en la Red de Excelencia CONET, financiada por el FP7 (INFSO-ICT-224053).

La estructura del artículo es la siguiente. La Sección 2 repasa el estado del arte en bancos de pruebas para robótica ubicua. La Sección 3 presenta su diseño y especificaciones. Las Secciones 4 y 5 describen brevemente sus principales componentes *hardware* y *software*. La Sección 6 presenta las funcionalidades del banco de pruebas para facilitar su empleo remoto. La Sección 7 ilustra algunos de los experimentos que se han llevado a cabo y presenta las principales lecciones aprendidas. Las conclusiones se presentan en la última sección.

## 2. Estado del arte

Los bancos de pruebas son herramientas de gran importancia para evaluar y comparar algoritmos y métodos en igualdad de condiciones. Aunque en algunas áreas es frecuente emplear bancos de datos y/o simuladores, en robótica los bancos de pruebas permiten obtener información práctica que difícilmente puede obtenerse empleando simuladores. Los bancos de pruebas también son herramientas muy extendidas en las áreas tecnológicas que componen los sistemas ubicuos.

La necesidad de validación experimental ha originado la aparición de una amplia variedad de bancos para experimentación en robótica. Una gran parte de ellos han sido desarrollados para cubrir necesidades concretas, tales como la exploración planetaria (Miller et al., 2000), la coordinación multi-agente (Jin et al., 2004), operaciones de búsqueda y rescate (Deloach et al., 2003) o la conectividad en redes móviles (Stubbs et al., 2006). Estos bancos de pruebas fueron diseñados para experimentar en aplicaciones específicas por sus desarrolladores. No emplean herramientas estandarizadas y pocos de ellos son accesibles de forma remota. HoTDeC (Stubbs et al., 2006) y el banco descrito en (De et al., 2006) son algunas excepciones.

También existen bancos de pruebas de robótica de carácter generalista, desarrollados para ser empleados en un amplio rango de experimentos. Entre otros destacan el propuesto en (Michael et al., 2008), que estudia el despliegue de grandes enjambres de robots móviles, o el propuesto en (Riggs et al., 2010) para experimentos en control y percepción cooperativa. También se han desarrollado bancos de pruebas que integran robots móviles no terrestres, como (Hoffmann et al., 2004), que incluye robots aéreos tipo *quadrotor* o (Brundage et al., 2006), que permite la cooperación de robots marinos. En (How et al., 2008) se propone un banco de pruebas que integra robots terrestres y aéreos.

Los bancos de pruebas antes citados no incluyen sistemas ubicuos. En la comunidad científica de las redes inalámbricas de sensores, los bancos de pruebas son una de las herramientas experimentales más ampliamente utilizadas. Esta comuni-

dad mantiene diversos bancos de pruebas. Gran parte de su éxito se debe a que son de uso público mediante acceso remoto y permiten la reutilización de código ya que emplean librerías, lenguajes y sistemas operativos estándares y, en gran medida, de código abierto. TWIST (Handziski et al., 2006) es un buen ejemplo de banco de pruebas para redes inalámbricas de sensores. Tiene un enfoque generalista y puede albergar un amplio rango de experimentos. Es público, permite acceso remoto y mantiene una arquitectura que se ha reutilizado en otros bancos de pruebas como WUSTL<sup>1</sup>. También se han desarrollado bancos de pruebas para aplicaciones concretas, perdiendo generalidad pero ganando eficiencia. Este es el caso de: Imote footnote <http://www.ee.washington.edu/research/nsl/Imote/>, que se centra en métodos de localización basados en potencia de señal de radio (RSSI); el banco de pruebas descrito en (Albesa et al., 2007), focalizado en la monitorización ambiental; y SmartSantander<sup>2</sup>, que trata sobre ciudades inteligentes. Algunos de estos bancos de pruebas integran nodos móviles portados por personas para investigar los efectos de la movilidad. Una tendencia reciente es federar grupos de bancos de pruebas bajo la misma interfaz (Chatziannakis et al., 2010).

Existen algunos bancos de pruebas que integran sistemas ubicuos con cámaras, como por ejemplo el desarrollado para el proyecto *Meerkats* con el objetivo de encontrar un compromiso entre eficiencia energética y rendimiento en una red de cámaras (Boice et al., 2006). Se compone de ocho nodos de sensores visuales unidos a una estación base. Otro ejemplo es *SensEye* (Kulkarni et al., 2005) que integra una red heterogénea de nodos inalámbricos y cámaras. *SensEye* está compuesto por varios niveles. El más bajo está formado por nodos inalámbricos con sensores de vibración. El segundo nivel contiene nodos inalámbricos con cámaras de bajas prestaciones, tales como CMUcam o Cyclops. El tercero está formado por estaciones base conectadas a *webcams*, que pueden capturar imágenes con mayor fidelidad que las del segundo nivel. El cuarto nivel está compuesto por cámaras de alta resolución orientables y directamente conectadas a computadores. El objetivo es el reconocimiento y seguimiento de objetos de manera eficiente mediante la progresiva activación de niveles más altos según la necesidad.

También existen bancos de pruebas que integran parcialmente sistemas robóticos con sistemas ubicuos. A veces se emplean robots móviles para aportar movilidad a los nodos, ya que los robots pueden realizar movimientos con mayor precisión y repetibilidad que los seres humanos (Rensfelt et al., 2011). Como ejemplo, Mobile Emulab (Johnson et al., 2006) utiliza 5 robots como nodos móviles. Los usuarios pueden, de forma remota, asignar trayectorias, ejecutar programas y guardar registros de datos. Igualmente, algunos sistemas multi-robot utilizan la infraestructura ubicua como un mero sensor distribuido. Por ejemplo, iMouse (Tseng et al., 2007) emplea una red inalámbrica de sensores para detectar alarmas que generan acciones de vigilancia coordinada en el equipo de robots. En estos bancos de pruebas la integración entre tecnologías robóticas y ubicuas

<sup>1</sup><http://www.cs.wustl.edu/wsn/index.php?title=Testbed>

<sup>2</sup><http://www.smartsantander.eu/>

es sólo parcial, lo que no permite explotar las sinergias entre ambas. Están sesgados hacia una de las tecnologías (sistemas multi-robot o sistemas ubicuos) y no pueden realizar experimentos que requieren una interacción estrecha, por lo que no son adecuados para muchos problemas de robótica ubicua.

Por último, se han desarrollado algunos (pocos) bancos de pruebas que permiten esquemas de cooperación de robótica ubicua. El banco de pruebas PEIS (*Physically Embedded Intelligent Systems*) fue diseñado de forma específica para experimentos de domótica (Saffiotti and Broxvall, 2005). El escenario de PEIS es un pequeño apartamento equipado con electrodomésticos automáticos, robots móviles y sensores distribuidos. La arquitectura *software* desarrollada para el proyecto es modular, flexible y abstrae la heterogeneidad del *hardware*. ISROBOTNET (Barbosa et al., 2009) es un banco de pruebas para sistemas multi-robot y redes de sensores que fue desarrollado en el marco del Proyecto Europeo URUS (*Ubiquitous Robotics in Urban Settings*)<sup>3</sup>. Está enfocado hacia robótica urbana e incluye algoritmos para seguimiento de personas, detección de actividades humanas y percepción cooperativa mediante redes de cámaras estáticas y móviles, entre otros. Aunque PEIS e ISROBOTNET permiten experimentos de robótica ubicua, han sido diseñados para aplicaciones específicas, lo que restringe los experimentos que pueden albergar. UBIROBOT (*Ubiquitous Robotic*) es un banco de pruebas que integra una red WSN, varios robots móviles, PDAs y teléfonos móviles inteligentes con Bluetooth. No ha sido posible encontrar publicaciones ni ninguna otra información más allá de la disponible en su sitio web<sup>4</sup>.

El banco de pruebas que se describe en este artículo integra robots móviles, redes de cámaras y redes WSN. Tiene un enfoque generalista, modular y abierto, lo que permite implantar una gran variedad de experimentos y explotar las sinergias entre sistemas multi-robot y sistemas ubicuos. Es accesible de forma remota y dispone de herramientas para facilitar la implantación y ejecución de experimentos.

### 3. Especificaciones y diseño

Este banco de pruebas fue diseñado para cumplir con dos objetivos principales. Desde un punto de vista de la investigación y desarrollo, debe servir de plataforma remota para probar, comparar y validar técnicas que combinen robots y redes de sensores distribuidos. Desde el punto de vista de la aplicación industrial debe servir como paso intermedio en el desarrollo de métodos antes de probarlos en la aplicación definitiva. Para identificar los requisitos de diseño se realizaron encuestas a 84 usuarios potenciales (universidades y empresas). Los resultados de la encuesta han puesto de manifiesto los siguientes requerimientos:

- **Generalidad.** Debe permitir realizar una amplia variedad de experimentos en tantas aplicaciones como sea posible.

- **Flexibilidad.** Su arquitectura debe ser modular y flexible para permitir la fácil incorporación de nuevos componentes.
- **Posibilidad de expansión.** Debe emplear interfaces estándares y código re-usable, preferiblemente abierto.
- **Facilidad de empleo.** Debe tener una interfaz amigable que permita acceder y programar los recursos de forma sencilla y directa. Además, debe incluir funcionalidades pre-programadas que puedan ser empleadas por los usuarios.
- **Acceso remoto.** Debe ser accesible de forma remota garantizando la seguridad y privacidad.

El banco de pruebas propuesto integra plataformas ampliamente empleadas en experimentos de robótica ubicua: robots móviles, redes de cámaras y redes inalámbricas de sensores. Estas redes constituyen un sector tecnológico en notable expansión. Están formadas por un alto número de nodos distribuidos que se organizan de forma autónoma en redes para cumplir una determinada tarea. Entre sus características destacan sus reducidos coste, tamaño y consumo energético. La estandarización de protocolos, como el IEEE 802.15.4, y de sistemas operativos, como TinyOS, ha facilitado su extensión a un número creciente de aplicaciones (Akyildiz et al., 2002).

Para permitir la interoperabilidad entre robots, redes de cámaras y redes WSN, con grandes diferencias en capacidades sensoriales, de procesamiento y de comunicaciones, el banco de pruebas incluye una capa de abstracción, a través de la cual cada módulo se comunica empleando interfaces estandarizadas. Este banco de pruebas utiliza Player como capa de abstracción.

Player<sup>5</sup> es una herramienta de código abierto muy extendida en robótica. Se basa en un diseño modular de tipo cliente-servidor. El Servidor Player interactúa con los dispositivos y emplea interfaces abstractos (llamados Interfaces Player) para comunicarse con el Cliente Player, ver Fig. 1. Player incluye soporte para una gran variedad de dispositivos, robots y sensores. Además, permite incorporar nuevos componentes *hardware* y *software* de forma directa y flexible. Player es independiente del Sistema Operativo y soporta programas en varios lenguajes, como C/C++, Python, Java. Dispone del simulador Stage que permite la simulación del experimento antes de su realización. Además, Player/Stage permite migrar código del simulador a la implementación experimental de forma sencilla.

Player es uno de los predecesores de ROS (*Robot Operation System*)<sup>6</sup>, de gran éxito reciente en la comunidad robótica. Se eligió Player porque cuando se diseñó este banco de pruebas, en 2007, ROS aún no estaba suficientemente maduro. Existe un alto grado de compatibilidad entre ambos, lo que permitirá la migración del banco de pruebas a ROS sin mucho esfuerzo.

Además, para facilitar el uso del banco de pruebas, se desarrolló una interfaz gráfica que permite la programación y configuración de dispositivos así como la ejecución y monitorización

<sup>3</sup><http://urus.upc.es>

<sup>4</sup><http://ubirobot.ucd.ie/>

<sup>5</sup><http://playerstage.sourceforge.net/>

<sup>6</sup><http://www.ros.org>

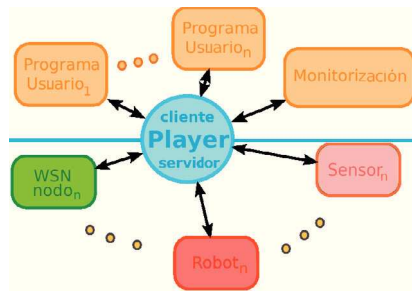


Figura 1: Esquema del modelo de interoperabilidad del banco de pruebas.

|                             | Generalista | Flexible | Abierto | Fácil | Remoto |
|-----------------------------|-------------|----------|---------|-------|--------|
| Componentes <i>hardware</i> | •           | •        | •       | •     |        |
| Player/Stage                | •           | •        | •       | •     |        |
| Red virtual privada         |             |          |         | •     | •      |
| Interfaz gráfica            |             |          |         | •     | •      |
| Funciones básicas           |             |          |         | •     |        |

Tabla 1: Correspondencia entre requisitos de diseño y los componentes principales del banco de pruebas propuesto.

experimentos de forma remota. El acceso remoto se realiza por medio de una red virtual privada (VPN) que garantiza la seguridad mediante autenticación y cifrado. En el banco de pruebas se desarrolló una librería de funciones básicas seleccionables que pueden ser utilizadas de forma directa por los usuarios en sus experimentos. En la Tabla 1 se muestra la correspondencia entre los componentes principales del banco de pruebas y los requisitos planteados en el diseño.

#### 4. Componentes *hardware*

El banco de pruebas integra un equipo de robots móviles, una red inalámbrica de sensores y una red de cámaras, ver Fig. 2. Contiene un equipo de 5 robots basados en la plataforma *Mobilerobotics* Pioneer 3-AT, con tracción diferencial y movimiento holónimo. Cada robot, ver Fig. 3, está equipado con un láser *Hokuyo* UTM-30LX 2D, con 30 m y 270 grados de rango y 0,25 grados de resolución. También, cada robot integra un sensor *Microsoft* Kinect, que proporciona imágenes RGB-D. Para su localización en exteriores los robots están equipados con receptores GPS y unidades de medida inercial (IMU). Cada robot ha sido equipado con un computador portátil tipo *Netbook* y un punto de acceso inalámbrico IEEE 802.11 a/b/g/n. Todos los robots están conectados por una red WiFi a 5 GHz para mejorar el aislamiento frente a la red WiFi de la Univ. de Sevilla, que opera en la banda de 2,4 GHz.

El banco de pruebas contiene una red inalámbrica de sensores con nodos estáticos y móviles. Los nodos estáticos están colocados en disposición reticular colgando del techo a una al-



Figura 2: Fotografía del banco de pruebas instalado en una sala de 500 m² en los sótanos del edificio principal de la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla. Pueden observarse 5 robots Pioneer 3-AT y los nodos de la red WSN colgando del techo.

tura de 1,6 m sobre el suelo. Los nodos se encuentran conectados mediante una topología tipo árbol. En total se dispone de 30 nodos de cada uno de los modelos Iris, MicaZ, Mica2 y TelosB (120 en total), todos de *Crossbow*. En cada experimento es posible utilizar todos los nodos del mismo modelo o de varios modelos (redes heterogéneas). Los Iris, MicaZ y TelosB tienen comunicaciones compatibles al utilizar el protocolo estándar IEEE 802.15.4. Los nodos TelosB integran sus propios acelerómetros, sensores de temperatura, humedad e intensidad luminosa. Para el resto de nodos se dispone de placas *Crossbow* MTS400, MTS420 o MTS310, que contienen magnetómetros, sensores de presión atmosférica, micrófonos e incluso GPS en la versión MTS420. Algunos nodos WSN han sido equipados con detectores de humo y sensores de concentración de gases. Además, el banco de pruebas también contiene nodos WSN móviles, que pueden ser portados por personas o por robots. En este último caso, cada nodo móvil está físicamente conectado a un robot por medio de una conexión USB, mediante la cual se establece un protocolo bidireccional que permite el intercambio de datos, órdenes y peticiones.

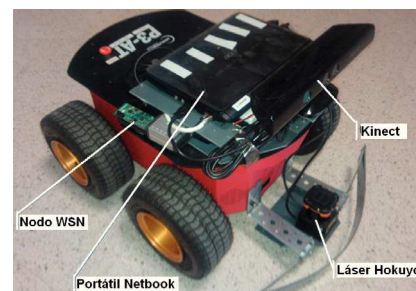


Figura 3: Detalle de los robots móviles y sus componentes principales.

El banco de pruebas contiene una red de 4 cámaras estáticas distribuidas en la sala. Además, cada robot lleva a bordo un sensor Kinect que proporciona imágenes RGB-D. Esta red de cámaras fijas y móviles puede ser utilizada como un sensor distribuido, accediendo a las imágenes y aplicándoles métodos de

procesamiento. Las imágenes también pueden emplearse para la visualización remota de experimentos.

Se dispone, además, de una red de micro-cámaras. Cada nodo está formado por un módulo CMUcam3 conectado a un nodo WSN, ver Fig. 4. En cada módulo CMUcam3 se realiza la captura de imágenes RGB de 352x288 píxeles y se les aplica métodos simples de procesamiento de imágenes. Las medidas obtenidas en cada módulo CMUcam3 se transmiten al nodo WSN por medio de un protocolo bidireccional sobre RS-232. Los nodos WSN pueden emplear su infraestructura de comunicaciones propia para implementar esquemas de percepción distribuida. A pesar de que actualmente disponen de reducida capacidad computacional y de ancho de banda, las redes de cámaras WSN han tenido un notable auge en los últimos años debido a su fácil despliegue y gran autonomía.



Figura 4: Detalle de los nodos de la red de cámaras WSN. Cada nodo está integrado por un módulo CMUcam3 conectado a un nodo WSN.

El banco de pruebas incluye también un PC de monitorización y registro de datos que tiene acceso a todos los dispositivos y sensores. Se encuentra conectado al nodo base de la red WSN y tiene acceso inalámbrico a todos los robots.

## 5. Componentes software

### 5.1. Arquitectura

La Fig. 5 muestra la arquitectura *software* del banco de pruebas. En cada robot se ejecuta el *Servidor Player Robot* que permite la interacción con todos sus sensores y actuadores y los hace accesibles al *Programa de Usuario* mediante Interfaces Player. Los programas de usuario pueden acceder a los dispositivos mediante interfaces de programación de aplicaciones (APIs) que abstraen sus singularidades. Igualmente, el *Servidor Player WSN* puede interaccionar con la red WSN. Cada *Programa de Usuario*, mediante Interfaces Player, puede interaccionar con el *Servidor Player* de cualquier robot o con el *Servidor Player WSN*.

La arquitectura permite implementar experimentos con diversos grados de descentralización. En los experimentos centralizados, el *Programa Centralizado de Usuario* accede a los *Servidores Player* de cada robot mediante Interfaces Player. La arquitectura también permite esquemas completamente descentralizados, donde los *Programas de Usuario* que se ejecutan en cada robot comparten información y cooperan mediante comunicaciones interproceso. La arquitectura permite también experimentos donde un *Programa de Usuario Remoto* accede a los *Servidores Player* mediante Interfaces Player o a los *Programas*

*de Usuario* locales por medio de comunicaciones interproceso. Cada usuario remoto tiene control sobre el programa que se ejecuta en su computador, lo que facilita la depuración del experimento. El banco de pruebas provee la infraestructura necesaria para realizar experimentos (módulos de color azul) mientras que el usuario debe ocuparse únicamente de los *Programas de Usuario* que desea ejecutar en los robots (en naranja) y en los nodos de la red inalámbrica de sensores (en verde).

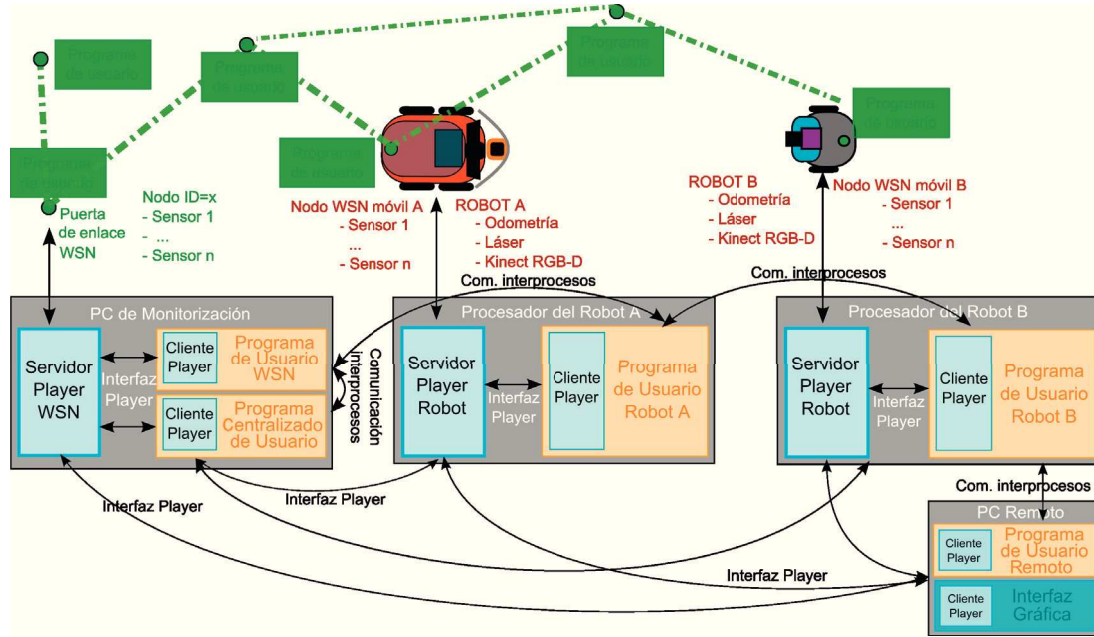
La cooperación entre robots y la red inalámbrica de sensores se potencia mediante una interfaz entre esta y Player que consigue comunicación bidireccional de datos, órdenes y peticiones entre dicha red y los robots. Esta interfaz representa una novedad significativa de este banco de pruebas, permitiendo que el robot acceda a los datos de los sensores del nodo WSN que porta y, viceversa, el nodo puede acceder a la información del robot. Por ejemplo, el robot puede calcular su posición empleando las lecturas RSSI que su nodo de la red inalámbrica mide de los paquetes que recibe de los nodos estáticos. La interfaz también permite que el nodo WSN móvil dé órdenes a su robot, e.g. acudir a una posición para que el nodo pueda comunicarse con otros nodos de la red inalámbrica. Esta interfaz ha sido admitida para formar parte de Player y se encuentra actualmente integrada en la versión 3.1.x, que puede descargarse de la web de Player<sup>7</sup>.

### 5.2. Localización de robots móviles

La localización y orientación de los robots en experimentos en exteriores se obtiene mediante receptores GPS y unidades IMU. En interiores se ha desarrollado un sistema de visión por computador basado en balizas. Se descartaron soluciones basadas en cámaras en el techo o paredes, porque en escenarios amplios (como en nuestro caso) se requiere un alto número de cámaras y capacidad de procesamiento. Se optó por un método sencillo y eficiente equipando cada robot con una cámara de bajo coste apuntando hacia el techo, donde se colocaron balizas dispuestas de forma reticular en localizaciones conocidas. La óptica de las cámaras se eligió para que, desde cualquier posición del robot, la imagen siempre contenga al menos un cuadrado de balizas, i.e. cuatro balizas cada una en una esquina del cuadrado.

La Fig. 6 resume las etapas del método empleado. Tras la segmentación de imágenes mediante umbrales, para cada baliza se calcula el centroide en el plano de imagen, se extraen las características de su geometría (área, excentricidad de la frontera) y se clasifica empleando un método de mínima distancia. Las balizas fueron diseñadas para que la clasificación pueda realizarse empleando características sencillas y robustas ante rotaciones. Las cámaras están calibradas internamente para evitar distorsiones ópticas. Seguidamente, se identifica el cuadrado de balizas presente en la imagen. Las balizas están colocadas de tal forma que cada uno de los cuadrados de balizas es único. Así, identificando las balizas se puede determinar unívocamente el cuadrado. Las posiciones en coordenadas sobre el techo de las balizas desplegadas son conocidas. Por tanto, identificando

<sup>7</sup><http://sourceforge.net/projects/playerstage/files/>

Figura 5: Esquema general de la arquitectura *software* del banco de pruebas.

el cuadrado, se pueden determinar las posiciones de las balizas que lo conforman. Finalmente, conocidas las posiciones reales de las balizas y sus posiciones en el plano de imagen, es posible calcular la posición y orientación del robot empleando homografías, que permiten calcular las transformaciones entre dos planos:

$$sp_i = \mathbf{H}_i P_i = \mathbf{A}_i T_i P_i, \quad (1)$$

donde  $p_i$  son las coordenadas homogéneas del centroide de la baliza  $i$  en el plano de imagen y  $P_i$  es la posición del centro de la baliza en coordenadas reales.  $\mathbf{A}_i$  representa la calibración interna de la cámara  $i$  y  $T_i$  la matriz de transformación entre el sistema de referencias del robot  $i$  y el sistema de referencia que consideramos base. Suponiendo que el sistema de referencia del robot móvil tiene ángulos de alabeo (*roll*) y elevación (*pitch*) nulos,  $T_i$  puede expresarse según la expresión:

$$T_i = \begin{bmatrix} c\theta & s\theta & 0 & x_r \\ -s\theta & c\theta & 0 & y_r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

De forma sencilla se puede calcular la posición  $(x_r, y_r)$  y orientación del robot ( $\theta$ ) empleando sólo dos correspondencias. En nuestro caso se disponen como mínimo de 4, una por baliza del cuadrado, lo que permite mejorar la estimación empleando métodos de mínimos cuadrados. El método ha sido validado experimentalmente con un error inferior a 7 mm de media y una desviación estándar menor a 5 mm, suficiente para los experimentos y aplicaciones deseados. Además, es computacionalmente eficiente y puede ejecutarse a tasas adecuadas (10 imágenes

por segundo) en los procesadores de los robots, donde otros procesos se ejecutan simultáneamente.

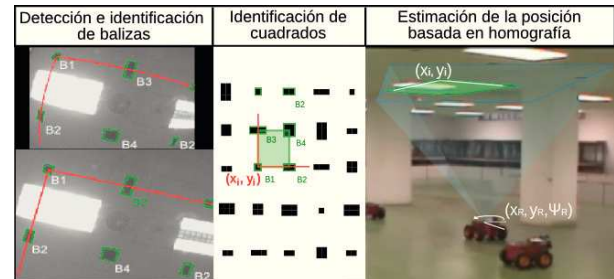


Figura 6: Izquierda) Detección de balizas. Centro) Identificación de cuadrados de balizas. Derecha) Estimación de posición y orientación mediante homografía.

### 5.3. Módulos pre-programados

El banco de pruebas incluye un conjunto de funciones básicas para facilitar el desarrollo de experimentos. Los usuarios pueden incluirlas en su proyecto, evitando tener que programar todas las funcionalidades de los robots, nodos WSN y procesamiento de imágenes de cámaras.

#### 5.3.1. Componentes para robots móviles

El banco de pruebas ofrece métodos para planificación de trayectorias, seguimiento de trayectorias con evitación de obstáculos, generación de movimientos aleatorios, así como un método de localización del robot.

Se incluye un planificador de trayectorias basado en *Wavefront propagation*. El método supone que se dispone de un

mapa del entorno discretizado en celdas. Se asigna cada celda con un coste inversamente proporcional a la distancia a su obstáculo más cercano. Cuando al planificador se le asigna un destino, busca un camino partiendo del punto de destino asignando costes a las celdas contiguas. El método selecciona las celdas con menor coste desde el origen hasta el destino, encontrando líneas rectas que no cruzan ningún obstáculo. Los puntos finales de dichas rectas se emplean como referencias del método de seguimiento de trayectorias. Se incluyen dos métodos de seguimiento de trayectorias con evitación de obstáculos: el método *smooth nearness diagram* (Durham and Bullo, 2008) específico para vehículos no holónomos y el método *vector field histogram plus* (Ulrich and Borenstein, 1998), para vehículos con tracción diferencial. También se ha desarrollado un método de generación de movimientos aleatorios.

El banco de pruebas incluye el método de localización AMCL *Adaptive Monte-Carlo Localization* (Fox et al., 1999), que estima la posición y orientación del robot empleando medidas de odometría y contrastando las medidas láser con el mapa del entorno. Se basa en un Filtro de Partículas que ajusta de forma adaptativa el número de partículas para equilibrar incertidumbre y carga computacional.

### 5.3.2. Componentes para sistemas ubicuos

Se han desarrollado funciones de procesamiento de imágenes bajo nivel para la red de cámaras CMUcam3. Se ofrecen funciones de segmentación de imágenes por nivel de gris y por color en RGB y en HSI, así como funciones de segmentación por diferencia respecto a una imagen de referencia. También se incluyen métodos de crecimiento de regiones. La Fig. 7 muestra los resultados de aplicar sobre una imagen los métodos de segmentación de color y segmentación por diferencia respecto a la imagen de referencia. Dichos algoritmos se han implementado de forma eficiente. La ejecución consecutiva de ambos métodos en imágenes de 352x288 píxeles se realiza a una tasa de 2-3 imágenes por segundo, que es suficiente para un buen número de aplicaciones.



Figura 7: Arriba) Imagen tomada por una cámara CMUcam3; Centro) Imagen tras aplicar segmentación por diferencia respecto a la imagen de referencia; Abajo) Imagen resultante de aplicar segmentación de color rojo.

Para las redes inalámbricas de sensores se implantaron métodos para formación de redes y recogida de datos mediante el protocolo *Collection Tree Protocol* (CTP) (Gnawali et al., 2009). CTP calcula dinámicamente rutas hasta el nodo WSN base empleando canales multi-salto entre los nodos WSN estáticos. Se han desarrollado funciones de monitorización donde cada nodo estático toma periódicamente las lecturas de sus sensores y las transmite al nodo base empleando los canales multi-salto. Los nodos móviles transmiten al nodo base empleando los canales estáticos. Cada nodo móvil selecciona el nodo estático con el

que tiene mejor conectividad (e.g. en términos de potencia recibida o de probabilidad de recepción de paquete *PRR*) y le transmite sus lecturas. El nodo estático seleccionado enruta los paquetes del móvil hacia el nodo base empleando los canales multi-salto.

## 6. Acceso remoto

El acceso remoto es importante para su empleo por parte de la comunidad. El acceso al banco de pruebas se proporciona mediante una red virtual privada (VPN) que permite protegerlo de accesos no controlados. Como se sabe, una red virtual privada (VPN) es una red de ordenadores en la que algunos enlaces se llevan a cabo a través de circuitos virtuales utilizando redes mayores (en este caso, Internet). La configuración de VPN seleccionada contiene mecanismos de autenticación y cifrado para garantizar la seguridad de las comunicaciones. Permite a los usuarios remotos conectarse al banco de pruebas de una forma transparente y simple: una vez conectados al servidor de VPN, los usuarios tienen acceso al banco de pruebas como si se estuvieran en la misma red de área local.

Mediante la VPN es posible acceder en modo consola a todos los ordenadores del banco de pruebas. Para facilitar el uso remoto se ha desarrollado una interfaz gráfica, ver Fig. 8. Está integrada en la arquitectura del banco de pruebas, ver Fig. 5, y tiene acceso a cada componente del experimento empleando Interfaces Player. La interfaz gráfica permite la gestión integral del experimento de forma remota. Permite visualizar todas las medidas de los sensores gráficamente (cámaras, láser, posición de robots y nodos) o en modo texto (lecturas de sensores). La interfaz gráfica permite la programación remota de nodos y robots, seleccionando las funciones básicas que deben ejecutar o subiendo directamente los archivos ejecutables en cada computador. Desde ella se puede controlar el inicio/parada de los programas de cada robot y el inicio/parada del registro de datos de cualquiera de los sensores o dispositivos que se emplean en el experimento.

Además, se ha diseñado una página web para la reserva y gestión de experimentos<sup>8</sup>. En ella se incluye documentación de los robots, sensores y nodos disponibles, manuales, tutoriales y consejos prácticos. La sección de descargas permite acceder a las distribuciones, librerías y aplicaciones necesarias para el desarrollo de experimentos. También está disponible una máquina virtual que puede ejecutarse en cualquier sistema operativo y que incluye todo el *software* necesario para realizar experimentos.

## 7. Experimentos

En la Tabla 2 se muestra un resumen de algunos de los experimentos realizados en el banco de pruebas, indicando las plataformas y sensores empleados. Se han ejecutado experimentos de sistemas multi-robot cooperando con infraestructura ubicua. También, experimentos que sólo involucran sistemas

<sup>8</sup><http://conet.us.es>

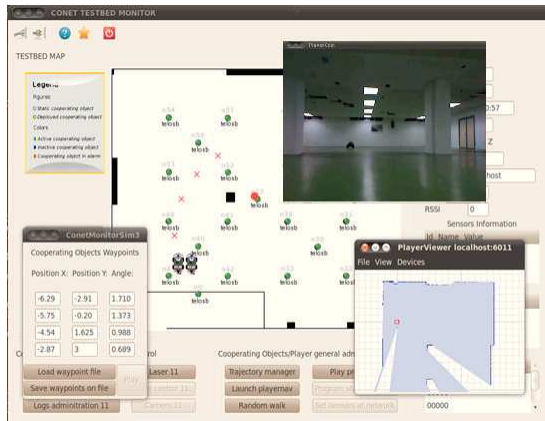


Figura 8: Interfaz gráfica del banco de pruebas durante la monitorización de un experimento. La interfaz permite, entre otras cosas, mostrar en vivo imágenes tomadas por el sensor Kinect (ventana de arriba) o lecturas de láser Hokuyo (ventana de abajo) tomadas por el robot.

multi-robot y otros que involucran sólo a tecnologías ubicuas. A continuación se describen brevemente algunos de los experimentos realizados que ponen de manifiesto las capacidades del banco de pruebas. Por último, en la Sección 7.4 se resume brevemente nuestra experiencia y lecciones aprendidas en el empleo práctico del banco de pruebas.

### 7.1. Reparación de redes WSN mediante robots

La robustez de una red WSN puede verse comprometida en el caso de fallos de los nodos. Cuando un nodo falla, no sólo deja de tomar datos de sus sensores, sino que también deja de retransmitir paquetes pudiendo deteriorar el funcionamiento de toda la red. Las variaciones en las condiciones de transmisión también pueden provocar zonas sin cobertura. Los robots móviles son de gran utilidad para diagnosticar y reparar las redes inalámbricas de sensores. Se pueden utilizar para identificar límites de conectividad y para detectar/confirmar agujeros en la cobertura. Una vez que se han detectado los fallos, se pueden utilizar robots para reparar la red WSN aplicando diversas estrategias.

Se pueden emplear robots para recolocar nodos WSN cuando la funcionalidad de la red no está comprometida, e.g. en caso de colocación incorrecta o no óptima de los nodos. Cuando los daños comprometen la misión, la opción es desplegar nuevos nodos para sustituir los dañados o para reparar los agujeros de conectividad (Corke et al., 2006). Si hay daños importantes que impiden la recogida de datos mediante rutado multi-salto entre los nodos estáticos, una opción es emplear robots para recogida de datos de la red inalámbrica de sensores. Los robots se controlan para moverse cerca de los nodos estáticos y asegurar comunicación sin errores empleando bajas potencias de transmisión y consumo energético. Esta opción es particularmente útil en escenarios amplios o con nodos separados entre sí (Martínez-deDios et al., 2012).

Seguidamente se describe brevemente un experimento donde se emplean robots móviles para reparar la conectividad de

una red inalámbrica de sensores. Supongamos un escenario donde se han desplegado nodos estáticos pero no se conoce su posición exacta. La red debe recoger datos de todos los nodos y hacerlos accesibles en un computador central, para lo cual debe garantizarse la conectividad y robustez de la red. La solución propuesta consiste en tres pasos: diagnóstico, reparación de la conectividad y reparación de la robustez.

El primer paso es la **diagnóstico** de la red WSN. Para ello se obtiene su matriz de conectividad, donde se refleja qué nodos están accesibles para cada uno de los nodos de la red WSN. Los nodos intercambian paquetes y cada uno registra los identificadores de los nodos con los que tiene conexión. En esta fase se emplean robots móviles para recorrer el escenario mientras que el nodo a bordo de cada robot transmite mensajes solicitando a los nodos estáticos su información de conectividad. El nodo a bordo recibe paquetes con la información de conectividad y se la transmite al robot mediante el protocolo WSN-Player. El Cliente Player, que se ejecuta en el computador del robot, recibe la información y la registra generando la matriz de conectividad. El Cliente Player incluye también un método de localización basado en RSSI que le permite ubicar los nodos estáticos de la red inalámbrica de sensores. La Fig. 9 muestra la red WSN identificada incluyendo los nodos detectados (círculos) en su ubicación estimada. Se indica mediante líneas la conectividad entre los nodos estáticos. En esta fase se detecta que la red está fragmentada en tres agrupaciones: algunos nodos están desconectados del nodo base unido físicamente al computador central.

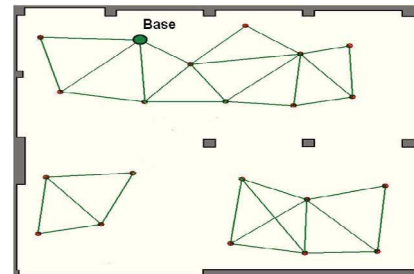


Figura 9: Red detectada en la fase de diagnóstico. La red está fragmentada.

Seguidamente se emplean técnicas basadas en la llamada **conectividad-k** para la **reparación de la conectividad** de la red inalámbrica de sensores. La **conectividad-k** expresa el número  $k$  de rutas disjuntas entre cualquier par de nodos de una red. **Conectividad-0** significa que en la red existen nodos aislados. Se ha desarrollado un algoritmo que calcula el número mínimo de nodos (y las localizaciones donde deben desplegarse) necesarios para que la red tenga una **conectividad-k** deseada. En nuestro ejemplo, ver Fig. 9, es necesario desplegar un nodo entre las tres agrupaciones. Tras la ejecución del algoritmo, el robot despliega el nodo en la ubicación calculada. El robot ha sido equipado con un pequeño brazo manipulador que le permite desplegar nodos WSN, ver Fig. 10. Una vez que el robot ha desplegado el nodo, la red WSN establece nuevas rutas y, como resultado, todos los nodos estáticos son accesibles al computador central, ver Fig. 11-izquierda. En rojo se muestran

Tabla 2: Resumen de experimentos principales realizados en el banco de pruebas que se propone.

|                                            | Plataformas         |                |         | Sensores principales |         |      | Funciones básicas | Remoto |
|--------------------------------------------|---------------------|----------------|---------|----------------------|---------|------|-------------------|--------|
|                                            | Sistema multi-robot | Red de cámaras | Red WSN | Láser                | Cámaras | RSSI |                   |        |
| Localización basada en firmas RSSI         | •                   |                | •       |                      |         | •    | •                 | •      |
| Reparación de redes WSN con robots         | •                   |                | •       |                      |         | •    | •                 |        |
| Localización con entrenamiento de RSSI     |                     |                | •       |                      |         | •    | •                 |        |
| Seguimiento mediante redes de cámaras      | •                   | •              |         |                      | •       |      | •                 | •      |
| Guiado de robots mediante WSN              | •                   |                | •       |                      |         |      | •                 | •      |
| Navegación/generación de mapas multi-robot | •                   |                |         | •                    |         |      | •                 |        |
| Percepción activa en redes CMUcam3         | •                   | •              |         |                      | •       |      | •                 |        |
| Seguimiento multi-robot basado en POMDP    | •                   | •              |         | •                    |         |      | •                 |        |
| Navegación multi-robot distribuida         | •                   |                |         | •                    |         |      | •                 | •      |
| Recogida de datos WSN mediante robots      | •                   |                | •       |                      |         | •    | •                 | •      |
| Seguridad en cooperación WSN-robots        | •                   |                | •       |                      |         | •    | •                 | •      |

los nuevos enlaces originados por el despliegue del nodo central.

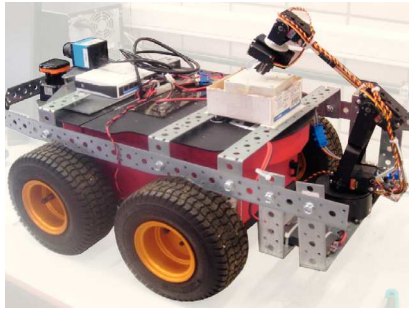


Figura 10: Robot Pioneer 3-AT dotado con un pequeño brazo robótico.

Seguidamente se procede a la **reparación de la robustez** de la red WSN. La *redundancia- $k$*  de un nodo  $i$  expresa el número  $k$  de nodos vecinos al nodo  $i$  que pueden fallar sin que la red pierda conectividad (Atay and Bayazit, 2009). Los nodos con *redundancia-0* son puntos frágiles de la red: en caso de que el nodo falle, la red queda desconectada. Se considera que una red es robusta si todos sus nodos tienen *redundancia-2*. Siguiendo el ejemplo de la Fig. 9, la Fig. 11-izquierda muestra la red WSN en el comienzo de esta fase. Se indican los valores de *redundancia- $k$*  de cada nodo. Hay tres nodos con *redundancia-0* y un alto número de nodos con *redundancia-1*. Se ha desarrollado un algoritmo que calcula el número mínimo de nodos (y de localizaciones donde deben desplegarse) necesarios para que todos los nodos de la red tengan una *redundancia- $k$*  deseada. Para conseguir que la red tenga *redundancia-1* es necesario desplegar dos nodos más. La Fig. 11-centro muestra la red resultante tras el despliegue de dichos nodos (en rojo los nuevos enlaces originados por los dos nodos desplegados). Para conseguir que la red tenga *redundancia-2* es necesario desplegar un nodo más. La parte derecha de la Fig. 11 muestra la red resultante tras el despliegue.

El experimento fue ejecutado mediante la interfaz gráfica empleando funciones básicas para el seguimiento de trayectorias. En la fase de diagnóstico se emplearon varios robots coor-

dinados. Las matrices de conectividad locales creadas por cada robot se fusionaron en un programa central. Se puede observar un vídeo del experimento en el enlace<sup>9</sup>.

## 7.2. Localización de robots empleando sistemas ubicuos

Se han desarrollado diversos experimentos de localización de robots empleando la infraestructura ubicua entre los que destaca la localización mediante medidas de la intensidad de la señal radio (*Received Signal Strength Indication*, RSSI) basada en *fingerprinting*. En escenarios interiores las medidas de RSSI contienen altos niveles de ruido originado por reflexiones y otras interacciones con el entorno. Se han desarrollado experimentos basados en enfoques *fingerprinting* consistentes en contrastar las medidas RSSI que recibe el robot con un mapa de RSSI del escenario generado previamente. En la primera fase uno o varios robots recorren el escenario y periódicamente envían mensajes baliza. Los nodos estáticos responden y el nodo de a bordo mide la potencia RSSI de los mensajes que recibe. El nodo de a bordo le comunica al robot dichas medidas por medio del protocolo WSN-Player y el Cliente de Player las registra en un mapa que asocia localización del robot con las medidas RSSI que ha recibido de los nodos estáticos. En la segunda fase, cuando el robot desea conocer su localización envía un mensaje baliza. Los nodos estáticos responden y el robot integra las medidas en un Filtro de Partículas que coteja las medidas en el mapa RSSI previamente generado.

El experimento fue realizado de forma local y también de forma remota. No se apreciaron diferencias significativas entre ambas quizás motivado por el escaso ancho de banda de las medidas RSSI que se transmiten en este experimento. Se emplearon funciones básicas para la navegación del robot. El mapa RSSI se generó empleando simultáneamente varios robots en cooperación, lo que requirió configurar adecuadamente el módulo de evitación de obstáculos. Cada robot generó la matriz de conectividad correspondiente a su porción del escenario. La matriz fue calculada de forma centralizada a partir de las matrices de cada robot. El usuario desarrolló los programas de los nodos WSN (nodos estáticos y el nodo a bordo del robot) y

<sup>9</sup><http://www.youtube.com/watch?v=1rKjdT-dCjc>

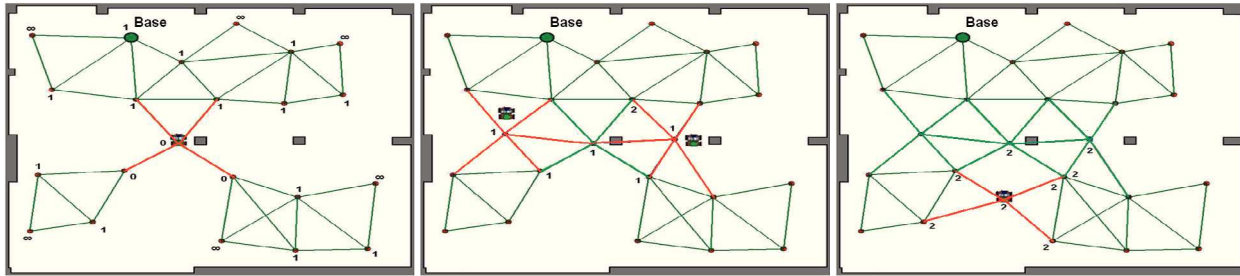


Figura 11: Izquierda) Estado de la red antes del comienzo de la fase de reparación de redundancia, en rojo los enlaces originados en la fase de reparación de conectividad; Centro) Estado de la red después de la corrección de *redundancia-1*, en rojo los enlaces originados en esta fase por el despliegue de dos nodos; Derecha) Estado de la red después de la corrección de *redundancia-2*, en rojo los enlaces originados en esta fase por el despliegue de un nodo.

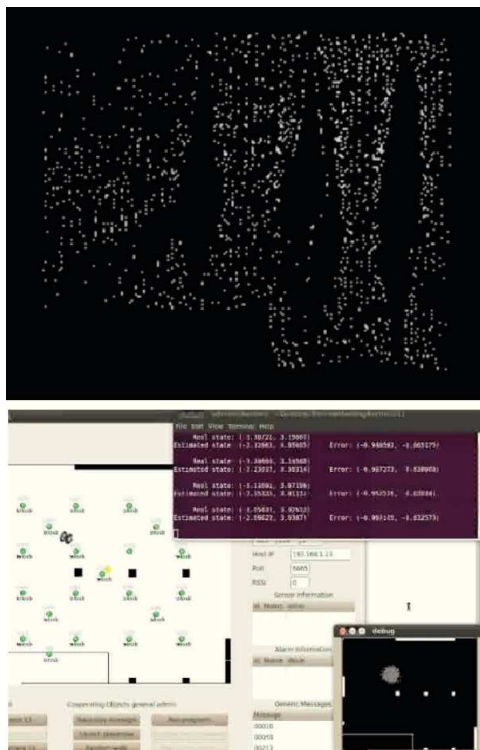


Figura 12: Arriba) Mapa RSSI (valores entre 0 y 255) obtenido mediante un robot móvil. Abajo) Captura de pantalla tomada durante el experimento de localización RSSI basada en *fingerprinting*.

el Cliente Player con el método de localización basado en Filtro de Partículas. El Cliente Player fue ejecutado en un ordenador remoto y monitorizado empleando la interfaz gráfica y la red de cámaras estáticas. Este experimento fue ejecutado en vivo en el tutorial *Remote Open Testbed for Cooperation of Wireless Sensor Networks and Mobile Robots*, impartido en el congreso EWSN'2012 en Trento (Italia)<sup>10</sup>. En esta demostración se utilizó un enlace Wi-Fi por medio de *Eduroam* en la Universidad de Trento. La Fig. 12 muestra el mapa RSSI y capturas

de pantalla del experimento donde puede apreciarse la nube de partículas.

### 7.3. Vigilancia de perímetros mediante robótica ubicua

El banco de pruebas para robótica ubicua ha sido utilizado para realizar experimentos de vigilancia de perímetros mediante algoritmos descentralizados (Acevedo et al., 2011). Dicho algoritmo utiliza múltiples robots para vigilar un perímetro cerrado y asegura que todos los tramos del perímetro serán visitados periódicamente por algún robot. El objetivo de los experimentos era caracterizar dicho algoritmo analizando sus limitaciones y eficiencia.

Formalmente se pretende resolver el problema de vigilancia de un perímetro de longitud  $L$  con  $N$  robots. La información detectada por cada robot (supuesto con rango de comunicaciones limitado) debe compartirse con el resto en el menor tiempo posible. Los robots se mueven por el perímetro con una velocidad entre 0 y  $v_{max}$ , que no tiene por qué ser la misma para cada robot. El algoritmo consiste en forzar que los robots vecinos se muevan en sentidos opuestos, encontrándose periódicamente y calibrando su velocidad para asegurar que cada posición del perímetro es visitada con un único período. En cada encuentro, la información intercambiada incluye la distancia recorrida y la información conocida sobre otros robots, permitiendo a cada robot realizar una estimación de la longitud del perímetro y la suma total de velocidades de los robots. Así, cuando el algoritmo converge, el perímetro se visita de manera óptima y los robots se encuentran en los puntos intermedios adecuados.

Se realizaron numerosas repeticiones de los experimentos, tanto en interiores como en exteriores, para caracterizar el algoritmo. Así, la fase experimental de esta investigación se benefició de la capacidad del banco de pruebas para operar tanto en interiores como en exteriores. Uno de los principales resultados obtenidos fue el límite superior en el tiempo de convergencia en relación con el tiempo que tardaría el robot más lento en cubrir el perímetro. También se obtuvo el límite superior del tiempo en que el último robot obtiene la información completa del sistema. En otra serie de experimentos se demuestra que el algoritmo es robusto ante perturbaciones, en el sentido de que se recupera y vuelve a converger a una solución cuando un robot queda inservible o se añade un nuevo robot al equipo.

<sup>10</sup><http://ewsn12.disi.unitn.it/tutorials.html>

#### 7.4. Lecciones aprendidas

Hasta la fecha en torno al 30 % de los experimentos remotos, los cuales requieren de restricciones adicionales relacionadas con la calidad del acceso a Internet. El servidor VPN del banco de pruebas está conectado mediante un enlace dedicado de fibra óptica al Centro de Datos de la Escuela Técnica Superior de Ingeniería, lo que garantiza una calidad de acceso a Internet. En las pruebas realizadas hemos percibido que el acceso del usuario debe tener un ancho de banda de 10 MB/s nominales aproximadamente para poder recibir simultáneamente lecturas de láser e imágenes de las cámaras sin retrasos significativos. Esta restricción tiene un bajo impacto práctico, permitiendo que los usuarios puedan acceder sin dificultades desde redes Wi-Fi públicas con niveles medios de tráfico. No hemos recibido notificaciones de usuarios donde esta restricción haya limitado ningún experimento. Sin embargo, el ancho de banda necesario podría reducirse adaptando adecuadamente la implementación del experimento.

Los programas de usuario pueden ejecutarse de forma remota en el *PC Remoto* de la Figura 5 o de forma local centralizada en el *PC de Monitorización* o distribuida en los procesadores de cada robot. El primer caso requiere la transmisión por Internet de todas las lecturas láser, imágenes y datos. Sin embargo, en el segundo, el procesamiento se realiza de forma local, y sólo sería necesario transmitir por Internet lecturas e imágenes para la monitorización del experimento, que pueden adaptarse con menor resolución y tasa configurando adecuadamente el Servidor Player. Indicar que todos los datos de cada experimento se registran con su tasa y resolución originales y el interfaz gráfico permite que un usuario pueda descargárselos de forma remota tras el experimento.

Hasta ahora el banco de pruebas ha dado un bajo nivel de incidencias en experimentos remotos. El servidor, que está implementado con un PC convencional, o puntos de acceso Wi-Fi han fallado ocasionalmente tras días o semanas de inactividad. Dichos fallos se resuelven de forma sencilla, reiniciando los dispositivos. Este bajo nivel de incidencias se debe al apoyo del ingeniero del banco de pruebas, el cual está presente en todos los experimentos remotos. El ingeniero del banco de pruebas da apoyo para evitar incidencias como colisiones entre robots, puede mover los robots, recargar sus baterías y puede reiniciar todos los sistemas en apenas 5 minutos. Sin embargo, el ingeniero del banco no realizará ninguna labor propia del método que se pretende experimentar. El banco de pruebas ha sido empleado durante más de 6 meses de forma interna por los participantes del proyecto CONET y dichas pruebas han sido de gran apoyo para probar todos los módulos y funciones.

Debido a la presencia del ingeniero del banco de pruebas cada sesión de experimentos se restringe a 4 horas. El ingeniero del banco indica la disponibilidad del banco de pruebas mediante un calendario web accesible a todos los usuarios registrados. Con cierta frecuencia han surgido conflictos con usuarios de otros continentes o usuarios europeos con otros horarios laborales. Cada solicitud de experimento debe ser validada por el ingeniero del banco de pruebas antes de que aparezca en el calendario. Nos han ocurrido casos en que dos usuarios solicitaron utilizar la misma sesión al mismo tiempo, sin dar tiempo

a que el calendario fuera actualizado. El orden de llegada de la solicitud fue criterio seguido para resolver estos conflictos.

#### 8. Conclusiones y trabajos futuros

El artículo describe un banco de pruebas para la experimentación en robótica ubicua. Integra actualmente una flota con 5 robots móviles Pioneer 3-AT, una red de cámaras y una red inalámbrica de sensores, ambos con los sensores más frecuentemente utilizados en experimentos de robótica ubicua.

La arquitectura *software* modular y flexible considera todos los elementos al mismo nivel, permitiendo una gran variedad de experimentos. Puede albergar experimentos con diferentes grados de heterogeneidad, desde centralizado hasta completamente distribuido. El banco de pruebas contiene numerosos componentes de ayuda para la experimentación tales como un sistema de localización mediante balizas y librerías con funciones básicas para evitar tener que programar y configurar todos los sistemas. Está dotado de una interfaz gráfica para la gestión integral del experimento de forma remota. Permite: programar robots y nodos WSN, monitorizar y visualizar el experimento, iniciar/parar cada programa y gestionar el registro de datos entre otros. Este banco de pruebas ha sido empleado para validar experimentos de cooperación multi-robot y de cooperación entre robots móviles e infraestructura ubicua desarrollados en la Red de Excelencia CONET, financiada por el FP7 (INFOS-ICT-224053).

Entre los trabajos actuales se incluyen el desarrollo de una herramienta de simulación adecuada, que integre robots, redes de cámaras y redes inalámbricas de sensores. Se pretende integrar también teléfonos inteligentes y otros dispositivos móviles. Además, entre las mejoras sugeridas por los usuarios destacan la recarga automática de los robots móviles y la integración de otros robots móviles, en particular robots aéreos basados en plataforma *quadrotors*. También se han sugerido nuevas funcionalidades básicas, que de hecho se han ido agregando, por ejemplo una función de mapeado basada en láser.

#### English Summary

##### A Remote Testbed for Ubiquitous Robotics Experimentation

#### Abstract

The lack of adequate experimental tools has been highlighted as a major critical issue in the development of ubiquitous robotics integrating mobile robots and wireless sensor networks. This paper describes a remotely operated testbed for ubiquitous robotics experimentation. It is currently comprised of 5 Pioneer 3-AT robots, a camera network and a wireless sensor network WSN, each equipped with widespread off-the-shelf sensors. The testbed architecture allows wide variety of cooperation schemes among its elements, abstracting their particularities and sensory, computational and communication capabilities. It also allows performing experiments with different degrees of decentralization. The testbed, currently deployed in the

School of Engineering at the University of Seville, is open and can be accessed remotely through a friendly graphical user interface.

#### Keywords:

testbeds, networked robots, wireless sensor networks

#### Agradecimientos

Este trabajo ha sido financiado parcialmente por la red de excelencia CONET, Cooperating Objects Network of Excellence (INFSO-ICT-224053) de la Comisión Europea en el Programa ICT del VII Programa Marco y por el proyecto de Excelencia Robótica Urbica en Entornos Urbanos financiado por la Junta de Andalucía. Aníbal Ollero agradece además la financiación parcial por parte de EC-SAFEMOBIL (ICT-288082). Los autores agradecen a los participantes en CONET así como a D. Victor Vega y a D. José Manuel Sánchez-Matamoros por su apoyo y contribuciones al trabajo.

#### Referencias

- Acevedo, J., Arrúe, B., Maza, I., Ollero, A., november 2011. Optimal distribution of multiple robots for perimeter surveillance. In: 3rd National Conference on Experimental Robotics. pp. 228 – 232.
- Akyildiz, I., Su, W., Sankarasubramanian, Y., Cayirci, E., aug 2002. A survey on sensor networks. *IEEE Communications Magazine* 40 (8), 102 – 114.
- Albesa, J., Casas, R., Penella, M., Gasulla, M., oct. 2007. Realnet: An environmental wsn testbed. In: Intl. Conf. on Sensor Technologies and Applications, SensorComm2007. pp. 502 – 507.
- Atay, N., Bayazit, B., january 2009. Mobile wireless sensor network connectivity repair with k-redundancy. In: Algorithmic Foundation of Robotics VIII. Vol. 57 of Springer Tracts in Advanced Robotics. Springer Berlin / Heidelberg, pp. 35–49.
- Barbosa, M., Bernardino, A., Figueira, D., Gaspar, J., Goncalves, N., Lima, P., Moreno, P., Pahliani, A., Santos-Victor, J., Spaan, M., Sequeira, J., oct. 2009. Isrobotnet: A testbed for sensor and robot network systems. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems. pp. 2827 – 2833.
- Boice, J., Lu, X., Margi, C., Stanek, G., Zhang, G., Obraczka, K., 2006. Meer-kats: A power-aware, self-managing wireless camera network for wide area monitoring.
- Brundage, H., Cooney, L., Huo, E., Lichter, H., Oyebode, O., Sinha, P., Stanway, M., Stefanov-Wagner, T., Stiehl, K., Walker, D., sept. 2006. Design of an rovs to compete in the 5th annual mate rovs competition and beyond. In: OCEANS. pp. 1 – 5.
- Chatzigiannakis, I., Fischer, S., Koninis, C., Mylonas, G., Pfisterer, D., 2010. Wisebed: An open large-scale wireless sensor network testbed. In: Sensor Applications, Experimentation, and Logistics. Vol. 29 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecom. Engineering. Springer Berlin Heidelberg, pp. 68–87.
- Corke, P., Hrabar, S., Peterson, R., Rus, D., Saripalli, S., Sukhatme, G., 2006. Deployment and connectivity repair of a sensor net with a flying robot. In: Experimental Robotics IX. Vol. 21 of Springer Tracts in Advanced Robotics. Springer Berlin / Heidelberg, pp. 333–343.
- De, P., Raniwala, A., Krishnan, R., Tatavarthi, K., Modi, J., Syed, N. A., Sharma, S., Chiehue, T.-c., 2006. Mint-m: an autonomous mobile wireless experimentation platform. In: 4th Intl. Conf. on Mobile Systems, Applications and Services. ACM, New York, NY, USA, pp. 124–137.
- Deloach, S. A., Matson, E. T., Li, Y., 2003. Exploiting agent oriented software engineering in cooperative robotics search and rescue. *Intl. Journal of Pattern Recognition and Artificial Intelligence* 17 (5), 817 – 835.
- Durham, J., Bullo, F., sept. 2008. Smooth nearness-diagram navigation. In: IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems. pp. 690 – 695.
- Fox, D., Burgard, W., Dellaert, F., Thrun, S., 1999. Monte carlo localization: efficient position estimation for mobile robots. In: AAAI 16th Conf. on Artificial Intelligence and 11th Conf. on Innovative Applications of Artificial Intelligence. pp. 343–349.
- Gnawali, O., Fonseca, R., Jamieson, K., Moss, D., Levis, P., November 2009. Collection Tree Protocol. In: 7th ACM Conference on Embedded Networked Sensor Systems.
- Handziski, V., Köpke, A., Willig, A., Wolisz, A., 2006. Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks. In: 2nd ACM Intl. Workshop on Multi-hop ad hoc Networks: from theory to reality. New York, NY, pp. 63–70.
- Hoffmann, G., Rajnarayan, D., Waslander, S., Dostal, D., Jang, J., Tomlin, C., oct. 2004. The stanford testbed of autonomous rotorcraft for multiagent control. In: 23rd Digital Avionics Systems Conf. Vol. 2. pp. 12.E.4 – 12I–10.
- How, J., Bethke, B., Frank, A., Dale, D., Vian, J., april 2008. Real-time indoor autonomous vehicle test environment. *IEEE Control Systems* 28 (2), 51 – 64.
- Jin, Z., Waydo, S., Wildanger, E., Lammers, M., Scholze, H., Foley, P., Held, D., Murray, R., Jul. 2004. Mvwt-ii: the second generation caltech multi-vehicle wireless testbed. In: American Control Conf. Vol. 6. pp. 5321 – 5326.
- Johnson, D., Stack, T., Fish, R., Flickinger, D. M., Stoller, L., Ricci, R., Lepreau, J., april 2006. Mobile emulab: A robotic wireless and sensor network testbed. In: 25th IEEE Intl. Conf. on Computer Communications. pp. 1 – 12.
- Kulkarni, P., Ganesan, D., Shenoy, P., Lu, Q., 2005. Senseye: a multi-tier camera sensor network. In: 13th annual ACM international conference on Multimedia. MULTIMEDIA '05. ACM, New York, NY, USA, pp. 229–238. DOI: 10.1145/1101149.1101191
- Li, Y. Y., Parker, L., april 2008. Intruder detection using a wireless sensor network with an intelligent mobile robot response. In: IEEE Southeastcon. pp. 37 – 42.
- Martínez-deDios, J., Lferd, K., de San Bernabé, A., Núñez, G., Torres-González, A., Ollero, A., 2012. Cooperation between uas and wireless sensor networks for efficient data collection in large environments. *Journal of Intelligent and Robotic Systems*, 1 – 18.
- Maza, I., Caballero, F., Capitán, J., Martínez-de Dios, J., Ollero, A., 2011. Experimental results in multiuav coordination for disaster management and civil security applications. *Journal of Intelligent & Robotic Systems* 61, 563–585.
- Michael, N., Fink, J., Kumar, V., march 2008. Experimental testbed for large multirobot teams. *IEEE Robotics Automation Magazine* 15 (1), 53 – 61.
- Miller, D., Seanz-Otero, A., Wertz, J., Chen, A., Berkowski, G., Brodel, C., Carlson, S., Carpenter, D., Chen, S., january 2000. Spheres - a testbed for long duration satellite formation flying in micro-gravity conditions. In: AAS/AIAA Space Flight Mechanics Meeting. pp. 167 – 179.
- Popa, D. O., Lewis, F. L., 2008. Algorithms for robotic deployment of wsn in adaptive sampling applications. In: Wireless Sensor Networks and Applications. Signals and Communication Technology. Springer US, pp. 35–64.
- Rensfelt, O., Hermans, F., Gunningberg, P., Larzon, L.-Å., Björnmö, E., 2011. Repeatable experiments with mobile nodes in a relocatable wsn testbed. *The Computer Journal* 54 (12), 1973–1986.
- Riggs, T., Inanc, T., Zhang, W., may 2010. An autonomous mobile robotics testbed: construction, validation, and experiments. *IEEE Transactions on Control Systems Technology* 18 (3), 757–766.
- Saffiotti, A., Broxvall, M., 2005. Peis ecologies: ambient intelligence meets autonomous robotics. In: Joint Conf. on Smart Objects and Ambient Intelligence: innovative context-aware services: usages and technologies. ACM, New York, NY, USA, pp. 277–281.
- Shah, R. C., Roy, S., Jain, S., Brunette, W., 2003. Data mules: modeling and analysis of a three-tier architecture for sparse sensor networks. *Ad Hoc Networks* 1 (2-3), 215 – 233.
- Sichitiu, M., Ramadurai, V., oct. 2004. Localization of wireless sensor networks with a mobile beacon. In: IEEE Intl. Conf. on Mobile Ad-hoc and Sensor Systems. pp. 174 – 183.
- Stubbs, A., Vladimerou, V., Fulford, A., King, D., Strick, J., Dullerud, G., june 2006. Multivehicle systems control over networks: a hovercraft testbed for networked and decentralized control. *IEEE Control Systems* 26 (3), 56 – 69.
- Tseng, Y.-C., Wang, Y.-C., Cheng, K.-Y., Hsieh, Y.-Y., june 2007. imouse: An integrated mobile surveillance and wireless sensor system. *Computer* 40 (6), 60 – 66.
- Ulrich, I., Borenstein, J., may 1998. Vfh+: reliable obstacle avoidance for fast mobile robots. In: IEEE Intl. Conf. on Robotics and Automation. Vol. 2. pp. 1572 – 1577.