

Arquitectura y desarrollo del robot humanoide microBIRO-II

Manuel Muñoz, Jordi Arjona *, Javier O. Coronel, Francisco Blanes, José Simó, Miguel Albero
{mmunoz, jacopa, pblanes, jsimo, mialgil} @ ai2.upv.es

* joralar@teleco.upv.es

Instituto de Automática e Informática Industrial, AI2
Universidad Politécnica de Valencia

Resumen

Los robots humanoides son uno de los desarrollos en sistemas robóticos que más auge están tomando hoy en día. Por un lado plantean retos a resolver (estabilidad, sensorización, integración de sistemas de visión) de gran interés desde el punto de vista de los sistemas empotrados de control, y por otro lado su natural orientación a la imitación del ser humano, ofrece un campo de trabajo en cuestiones como la inteligencia artificial, la interacción hombre-máquina y la cooperación entre robots como ninguna otra plataforma robótica ofrece.

En esa línea multidisciplinar se enmarca el presente trabajo en el cual se presenta un robot humanoide de tamaño pequeño diseñado cumpliendo con las restricciones impuestas en las pruebas de la competición RoboCup.

Palabras Clave: robots humanoides, sistemas empotrados de control, sensores, arquitecturas de control.

1 INTRODUCCIÓN

El robot microBIRO-II¹ es el tercero de una generación de robots humanoides desarrollados por el AI2, y que se inició con YABIRO[1][2], se miniaturizó con microBIRO-I[3] y finalmente con el objeto de permitir su participación en las pruebas de la RoboCup creció en tamaño y configuración hasta el robot que se presenta en este trabajo.

Las competiciones son hoy por hoy uno de los medios más usados para evaluar las capacidades de un sistema robótico ya que se basan en resultados sobre pruebas conocidas para todos los participantes por igual[4]. En el caso de la RoboCup y en concreto en la liga de fútbol de humanoides esta es además una competición entre robots, por lo cual dicha evaluación es además comparativa respecto al resto de robots.

En el primer apartado de este artículo se realizará una introducción a los aspectos hardware del robot,

presentando los distintos tipos de sensores y actuadores de que se dispone en microBIRO-II; en el segundo apartado serán comentados los diferentes sistemas de control que se encuentran implementados; por último, antes de pasar a las conclusiones, se explicará el funcionamiento y la importancia de la arquitectura FSA sobre este sistema concreto.

2 HARDWARE

La RoboCup como competición tiene por objetivo lograr que en el año 2050 el mejor equipo de robots humanoides se enfrente contra el mejor equipo de humanos en un partido de fútbol. Este objetivo planteado a largo plazo, guía el criterio de construcción de robots que año a año es más restrictivo. Las normas propias de la RoboCup[5] buscan asemejar desde el punto de vista sensorial y morfológico a los robots humanoides con los humanos. Es por ello, que no se pueden usar sensores como infrarrojos, láser, ultrasonidos, etc.,... por no encontrar estos semejanza con los sistemas de percepción humanos.

Toda la información sobre el entorno deberá provenir del sistema de visión que deberá ser activo, y tan solo se permiten sensores para la medición de valores sensoriales internos al robot como fuerzas, aceleraciones etc. Los sensores y actuadores que se pueden encontrar en microBIRO-II se pueden agrupar en los siguientes tipos:

- Servomotores
- Acelerómetros
- Sensores de presión
- Cámara de visión

Todos ellos deberán de cumplir dichas las restricciones impuestas anteriormente.

¹ Este trabajo se desarrolla en el marco del proyecto SIDIRELI DPI-2008-06737-C02-01/02 del Ministerio de Educación y Ciencia.

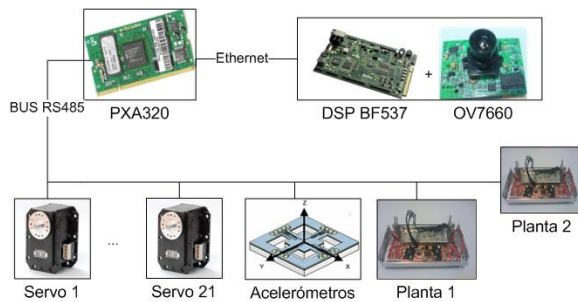


Figura 1: Arquitectura Hardware

Los 21 servomotores que conforman los 21 grados de libertad (GDL), serán utilizados en las articulaciones del robot, formando parte de la estructura del robot y dotando al robot de una gran movilidad y capacidad de realizar movimientos con una cierta complejidad, imitando lo mejor posible el comportamiento de un ser humano. Todo ello manteniendo las restricciones impuestas por la competición en dimensiones, peso, etc.,...

En la tabla 1 se puede observar la distribución de los grados de libertad del robot así como su movilidad.

Tabla 1: Distribución GDL y movilidad

| Ubicación | GDL | Movilidad |
|-----------|-----|---|
| Tobillo | 2 | $\pm 75^\circ$ 150° - 30° |
| Rodilla | 1 | 0° - 150° |
| Cadera | 3 | $\pm 75^\circ$ 30° - 150° 30° - 90° |
| Torso | 1 | $\pm 75^\circ$ |
| Brazos | 3 | $\pm 75^\circ$ $\pm 75^\circ$ $\pm 75^\circ$ |
| Cabeza | 2 | $\pm 75^\circ$ |

Los servomotores que se han utilizado son del tipo Dynamixel RX-10, RX-28 y RX-64 conectados al bus Dynamixel, que está formado por dos partes: una de ellas suministra la energía a los motores, la otra es un bus de comunicaciones Dynamixel-RS485 sobre el que se implementa el protocolo de proporcionado por el fabricante. Estos servos son específicos para robótica y comercializados por la empresa coreana Robotis.

Se ha dotado al robot de un sistema de sensorización capaz de medir inclinaciones. Este conjunto sensor está formado por un sistema integrado compuesto por dos acelerómetros y los elementos necesarios para la digitalización y procesamiento de las señales. Este elemento se encuentra ubicado en la cadera del robot.

Los acelerómetros son de 3 ejes y capaces de medir aceleraciones de hasta $\pm 3g$, además, en su montaje estarán rotados 45° entre sí, de modo que se puedan obtener medidas sin influencia de la inclinación. Este sistema de medida permitirá estimar las inclinaciones a las que se ve sometida la cadera.

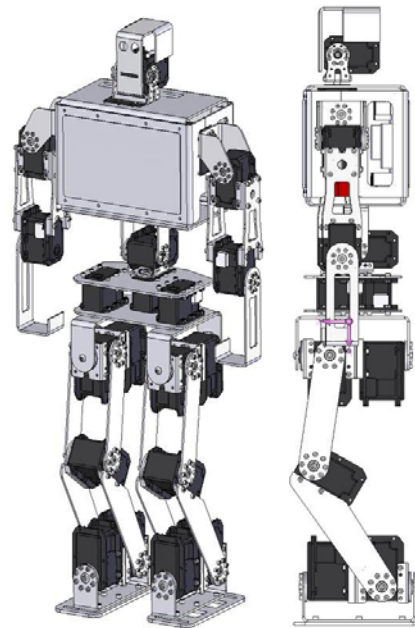


Figura 1: Vistas de microBIRO-II

Un microcontrolador dsPIC será el encargado de digitalizar y procesar las señales provenientes de los acelerómetros, permitiendo obtener medidas en bruto de los sensores o información más elaborada, liberando así de ese cómputo a la unidad de control principal.

Como segundo sistema de percepción, se ha incluido en los pies del robot un sistema de estimación del centro de masas del sistema. Del mismo modo que en el módulo de acelerómetros, se ha dotado al sistema de los elementos necesarios para instrumentar y procesar las señales procedente de los sensores, que en este caso son galgas extensiométricas situadas en unas lengüetas semiflexibles en los extremos del pie diseñadas a tal efecto.

Dicho módulo incluirá también un microcontrolador dsPIC que realizará las mismas funciones que en el módulo de acelerómetros.

Este módulo, permitirá obtener medidas en bruto desde los sensores o información más elaborada, como puede ser el saber si el pie está apoyado o no, medidas sobre la presión en cada uno de los puntos de apoyo, ubicación del centro de masas, etc.

Ambos módulos adicionalmente estarán comunicados con la unidad principal de control mediante el bus Dynamixel-RS485, el mismo que los servomotores, a través del cual se llevará a cabo, también la alimentación de los módulos.



Figura 2: Planta del pie de microBIRO-II

Para captar la información del entorno, se ha dispuesto un sistema de visión en la cabeza del robot. Este sistema utiliza una cámara Omnivision OV7660 basada en tecnología CMOS, que cuenta con una lente de 1/5", que abarca un campo de visión de 56° y permite obtener imágenes con un framerate de 15fps.

El sistema de visión realiza el procesamiento de las imágenes mediante un DSP Blackfin BF537 de Analog Devices. Éste realiza las funciones de adquisición de imágenes de la cámara y procesamiento de las mismas, generando matrices de características que puedan ser analizadas por el sistema para conseguir la información necesaria del entorno del robot y tomar las acciones pertinentes. Actualmente el módulo de visión se comunica mediante Ethernet, pero se está implantando la comunicación a través de un bus serie SSP-SPI. También está previsto sustituir este módulo por otro con visión estéreo.

Como unidad de control principal se utiliza un módulo que integra un procesador del tipo PXA320, que destaca por su alto rendimiento y bajo consumo. Éste permite gestionar los buses SSP-SPI, USB, Ethernet o una conexión WiFi entre otros. Actualmente se está desarrollando una FPGA que se mapeará en memoria y que permitirá derivar a ella la gestión de los buses de comunicaciones, liberando de esta tarea al microprocesador principal.

3 CONTROL BÁSICO DE MOVIMIENTOS

Como robot bípedo microBIRO-II debe cumplir con una serie de capacidades motrices básicas en la RoboCup como son andar, chutar, girar o realizar desplazamientos laterales. Para poder realizar estos de manera correcta es necesario implementar un sistema de control que permita que estos se realicen de modo adecuado.

Los bucles de control que se han implementado utilizan la información de los sensores para regular los movimientos del robot. Por ejemplo, los bucles de control de bajo nivel se encargan de mantener la estabilidad al caminar y para ello actúan en dos frentes, por una parte controlando la inclinación del torso/cadera del robot, y por otra manteniendo el centro de masas del robot dentro de un área de seguridad. Otros bucles de control de alto nivel toman decisiones basándose en la información del entorno aportada por el sistema de visión. En los párrafos siguientes se describirán con más detalle estos bucles de control.

El control de inclinación de torso/cadera se realiza utilizando como realimentación la información aportada por el sensor de inclinación. Es conocido que al caminar, la cadera del robot se ve expuesta a variaciones de inclinación en el transcurso de un paso provocadas por la incapacidad de los servomotores para mantener la posición cuando el robot se mantiene únicamente sobre una de las dos piernas, situación en la que deben soportar un elevado par. Por ello este sistema de control, se encarga de regular la posición de los motores de la cadera, evitando que el robot se balancee lateralmente o incluso que pudiera llegar a caerse.

El control de equilibrio trabajará, del mismo modo que el control de cadera, utilizando la información del módulo de acelerómetros. Su función principal es mantener erguido el torso de manera que éste permanezca perpendicular a la horizontal, independientemente de la superficie de apoyo, adaptando para esto la inclinación de pies y cadera y la longitud de las piernas.

A modo de test de este sistema de control se han realizado unas pruebas con el robot en posición erguida sobre una tabla. En esta situación se va inclinando ésta, de manera que el robot modifica su posición regulándola para mantenerse erguido respecto a la vertical.



Figura 3: Test para evaluar el control del equilibrio

Actualmente se encuentra en desarrollo el sistema de control que utiliza las medidas obtenidas del módulo de sensores de presión colocados en los pies del robot, para mejorar el apoyo del pie y con ello la estabilidad de la plataforma. Adicionalmente y utilizando la misma información el sistema de control intenta mantener la posición del centro de masas del robot dentro de una área de seguridad para evitar las caídas.

En ausencia de estos sistemas de control, el robot se controla en bucle abierto siguiendo unas tablas de movimientos generadas previamente atendiendo a criterios de estabilidad estática del robot.

La utilización de estos sistemas de control permitirán al robot reaccionar frente a un entorno cambiante que modifique los supuestos sobre los que se han generado las tablas para el bucle abierto, como pueden ser: un suelo rugoso o con pequeñas inclinaciones laterales, u oscilaciones imprevistas de la mecánica.

4 ARQUITECTURA DE CONTROL FSA EN microBIRO-II

EL desarrollo de un sistema de control para un robot humanoide es una labor que desde el inicio debe contar con una arquitectura clara dada su complejidad. Esta reside principalmente en:

- Se trata de un sistema con gran número de actuadores
- Los sensores proporcionan información de control a varios niveles de abstracción (control de motor, control de movimiento, control de trayectoria ...)
- En el caso de la visión la información es compleja y su uso dentro de los bucles de control pasa por varias etapas de procesado

Por todo ello es necesario dotar al robot de una arquitectura de control flexible, potente y que permita un desarrollo cíclico e incremental del sistema. En nuestro caso la arquitectura utilizada es FSA (Frame Sensor Adapter)[6]

La arquitectura FSA maneja 3 tipos de entidades, el Marco (Frame), los Adaptadores (Adapters) y los Sensores (Sensors) (escapa del objetivo de este artículo describir esta arquitectura por lo que se recomienda consultar el trabajo de Poza, Posadas, Simó, Pérez y Alberó[1]. A continuación se definirán estas entidades sobre microBIRO-II y se presentará como se ha adaptado el funcionamiento de éste a esta arquitectura, sin olvidar nunca los objetivos de tener una arquitectura modular, fácilmente escalable y, ante todo, portable.

4.1 CONSIDERACIONES PREVIAS

El primer punto a comentar es que, teniendo en cuenta el aspecto de la portabilidad, la implementación se realizó en C, esto nos permitirá portar todos los desarrollos a un gran repertorio de sistemas empujados.

Pasando a la identificación de las entidades, tal y como se ha podido ver en las secciones anteriores, se tiene una serie de dispositivos que son fácilmente identificables como sensores y actuadores (acelerómetros, cámara de visión, sensores de presión y servomotores) y un medio que será nuestro adaptador (el bus de campo Dynamixel-RS485 y el de Ethernet). El marco será la base desde la que se controle el funcionamiento la conexión y desconexión de sensores y adaptadores así como el arranque de las distintas operaciones de control que se deban lanzar en microBIRO-II.

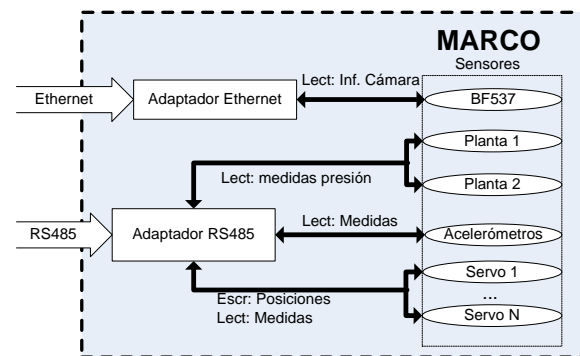


Figura 4: Arquitectura FSA

Una vez realizada esta identificación y trazado un esquema de cómo va a ser el sistema es el momento de realizar la implementación de cada una de las partes.

La ventaja que proporciona FSA es que se va a conseguir una gran flexibilidad acompañada de una gran independencia de cada una de las partes del sistema. Una de las piezas claves en cualquier sistema es la relación existente entre cada uno de los sensores y como se relacionan éstos con el medio de comunicación y los problemas que supone, a nivel software, realizar cualquier cambio a nivel de hardware. FSA proporciona independencia en el sentido de que los sensores se relacionarán con el adaptador/es en el que esté registrado mediante mensajes estándar, al igual que los adaptadores con ellos. Esto supone que si, citando el ejemplo de microBIRO-II, se cambia un sistema de cámara de visión de una sola cámara por un sistema estéreo, en la implementación de la aplicación sólo será necesario crear un nuevo sensor (o incluso sólo modificar ligeramente el previo) sin que esto afecte en absoluto a la relación de éste con el adaptador.

4.1 EL MARCO

El marco va a ser la pieza fundamental de la implementación del sistema de control. Su función básica es iniciar y detener el sistema, pero incluyendo la creación y definición de todos y cada uno de los sensores y adaptadores del sistema, así como de los bucles de control que harán uso de la información sensorial y de los actuadores.

En el caso de microBIRO-II, en el marco se encuentra definido todo el sistema de identificación de sensores y adaptadores, así como sus posiciones de memoria. También se definen y asignan en el marco a cada uno de los adaptadores aquellas operaciones que se desee que tengan disponibles y para que sensores se quiere que lo estén.

La importancia de una correcta implementación del marco es fundamental en el caso de microBIRO-II, donde se va a utilizar un bus Dynamixel-RS485 con un protocolo de comunicaciones dynamixel impuesto por el fabricante de los actuadores. Esto se debe a que todas las operaciones que quieran realizar los sensores dependientes de él deberán incluir las posiciones de memoria donde el protocolo deberá obtener la información cuando se le requiera. Esto puede parecer tedioso aparentemente, pero la ventaja es que, si hubiera que sustituir un sensor físico, con un protocolo de comunicación tan complejo como éste, los únicos parámetros a cambiar son los mapeos en memoria de las operaciones.

Además el marco proporciona otra ventaja a nivel de facilidad a la hora de incorporar nuevas partes al sistema puesto que lo único que será necesario será definirlos en el marco y asignarles un adaptador. Esto se debe a que al iniciar los adaptadores, estos leerán de la lista de sensores del sistema y registrarán aquellos que deban ir asociados a ellos y estos, a su vez, estarán preparados para interactuar con él. En resumen, el número de adaptadores y sensores activos va a depender del marco, los adaptadores simplemente se comportarán como una caja que recibe peticiones y las contesta, sin afectarle cuantos sensores dependen de él

Una vez el marco haya sido definido correctamente y se hayan conectado y activado los distintos adaptadores, el marco se convertirá en el punto de acceso a la información proporcionada por los distintos sensores de microBIRO-II. Además, desde aquí se podrán arrancar y manejar los distintos bucles de control implementados en el sistema permitiendo arquitecturas de control tanto jerárquicas como distribuidas.

4.2 IMPLEMENTACIÓN DE LOS ADAPTADORES

Los adaptadores implementados en el sistema tienen como función principal abstraer los canales de comunicación de la capa de sensores, ofreciendo para todos ellos una interfaz común de interacción basada en primitivas de lectura o escritura en el canal. En definitiva, debe de estar implementado de tal manera que su funcionamiento sea el de una caja que ofrece una serie de funciones (*connect*, *disconnect*, *start*, *stop*, *register sensor*, *write* y *read*) y que es capaz de trabajar con otra serie de funciones propias de los sensores en él registrados (*onStart*, *onStop* y *onMessage*). Si además tuviera que ser sustituido por otro adaptador distinto, capaz de ofrecer y realizar las mismas funciones, este cambio no afectará al sistema.

Citando de nuevo, por su complejidad, el ejemplo del bus Dynamixel-RS485, sin entrar a describir el funcionamiento de las funciones de conexión y desconexión, arranque y parada por su sencillez, la mayor complejidad al realizar su implementación adaptada a FSA consiste en decidir como realizar la gestión del bus y la planificación de operaciones. Finalmente se asumió que debían ser los sensores los que planificaran localmente cuando solicitar los datos, en base a la información de operaciones disponibles en el adaptador, y que realizaran las peticiones de datos al bus Dynamixel-RS485 cuando fuera pertinente mediante la llamada a sus métodos *read* o *write*. Una vez el bus recibe las peticiones, las ejecuta o almacena en un buffer circular en caso de que en ese instante este ocupado.

4.3 IMPLEMENTACIÓN DE LOS SENSORES

La función de los sensores a implementar en la arquitectura debe ser la de gestionar y almacenar la información obtenida de los sensores físicos a través del adaptador. Para esto, lo único que han de conocer son las operaciones que tienen disponibles para interactuar con el adaptador y conseguir información de los sensores físicos además de disponer de una representación del tipo de dato capaz de almacenar la información devuelta por los sensores.

Por su parte tendrán una serie de funciones estándar definidas, las conocidas *onStart*, *onStop* y *onMessage*, que serán invocadas por el adaptador en el que están registrados, las dos primeras para su activación y desactivación y la última que será a través de la que el adaptador intercambie información con el sensor (a partir de este momento siempre que se hable de sensor se estará haciendo referencia a un sensor software).

Un aspecto a resaltar es que, dado que son los propios sensores los que van a planificar como realizan las peticiones de datos al adaptador, han de ser capaces de enviar peticiones compuestas cuando tengan varios mensajes a enviar en un mismo instante temporal. Esto se puede ver claramente en el caso de los servomotores. Cuando se necesite actuar sobre ellos, muchas veces habrá que actuar sobre 12, 15 o más servomotores. Enviar esta cantidad de mensajes por el Dynamixel-RS485 a la vez no es adecuado, puesto que habría que encolarlos, pero si que podemos enviar un mensaje compuesto que les llegue a todos y así optimizar la respuesta temporal del sistema.

4.4 FUNCIONAMIENTO DEL SISTEMA

Como se ha comentado, habrá una serie de bucles de control que serán lanzados en el marco una vez se hayan activado los distintos adaptadores y sensores comprendidos en él. En el caso de microBIRO-II tenemos los bucles de control de cadera o de equilibrio, que necesitarán de la información proporcionada por estos sensores, accesible en el marco.

Empezando por el control de equilibrio, se deberán conocer los datos de posición e inclinación del robot, proporcionados por los acelerómetros, de modo que se pueda calcular la acción de control y a continuación modificar la posición de los servomotores para mantener el equilibrio. Esto se consigue activando las operaciones de recogida de datos del sensor acelerómetro, de modo que se le hagan peticiones al adaptador cada cierto periodo manteniendo los datos del robot actualizados a una frecuencia que evite que el bucle de control pueda trabajar en algún momento con datos obsoletos, ya que esto produciría resultados inesperados y posibles caídas del robot. Una vez el bucle ha calculado esta acción de control se ordenará a los sensores de los servomotores que modifiquen la posición de sus actuadores físicos mandando la orden a estos a través del adaptador y consiguiendo que el robot mantenga la estabilidad.

El funcionamiento del control cadera será semejante, obteniéndose también la información sensorial a partir de los acelerómetros, calculándose la correspondiente acción de control por el bucle de control de cadera y ejecutándose esta acción a través de los sensores relacionados con los servomotores.

4.5 VENTAJAS APORTADAS POR FSA

Se ha podido ver como FSA muestra ser una arquitectura altamente flexible y fácilmente escalable gracias a la capacidad de configuración que nos proporciona el marco. Esta flexibilidad queda

también patente al comportarse sus entidades como módulos que pueden ser reemplazadas por otras del mismo tipo sin que esto altere el funcionamiento del sistema.

También hay que destacar que, al igual que podemos aumentar el número de dispositivos, aumentar el número de bucles de control o aumentar la profundidad, complejidad y el detalle alcanzados por los sensores implementados no reviste dificultad.

Por último, gracias a FSA se consigue tener una visión de alto nivel del sistema, consiguiendo que los bucles de control se independicen de los sensores físicos en sí, pudiendo trabajar directamente con los datos, sin tener que preocuparse de cómo se han obtenido éstos ni de aspectos temporales, siendo todas estas cuestiones manejadas por los adaptadores.

5 RESULTADOS

En la actualidad tanto el marco, como el adaptador que representa al bus Dynamixel-RS485, como los sensores relativos al módulo de acelerómetros han sido implementados y funcionan de manera adecuada, habiéndose testado su funcionamiento con un módulo de acelerómetros conectado al PC mediante un bus Dynamixel-RS485 bucles de control de movimientos básicos.

A su vez en el robot se ha implementado el control de cadera y el control de equilibrio con resultados satisfactorios, como se pudo observar en la figura 3. Además de las pruebas realizadas para validar la arquitectura mediante los controladores comentados, se ha incluido en el robot la capacidad de teleoperarse, con lo que es posible realizar un guiado del mismo al tiempo que los reguladores asociados al control de cadera y de estabilidad aplican las acciones de control para compensar en dichos movimientos la dinámica del robot.

6 TRABAJO FUTURO

Una vez comprobada la viabilidad del empleo de la arquitectura FSA, las siguientes líneas de trabajo a desarrollar son:

- 1.- Incorporar el sensor de visión a microBIRO-II a los bucles de control. Esta labor implica un alto grado de complejidad en cuanto la información de tipo visual requiere de gran cantidad de análisis y tratamiento. Inicialmente y dada su orientación a participar en las pruebas de la RoboCup el planteamiento es el de trabajar en un entorno bien estructurado en el cual el robot pueda localizar

espacialmente los objetos de interés: pelota, líneas de campo, portería etc.

2.- Mejorar el comportamiento de microBIRO-II a partir de la incorporación de las funciones de sensorización de la planta del pie que estamos desarrollando actualmente.

3.- Incorporación de funciones de comunicación entre robots del mismo equipo mediante Wi-Fi. Este aspecto permitirá la integración de nuevas estrategias de control más orientadas a coordinación entre robots.

4.- Implementación y testeo de nuevos reguladores y nuevas funciones de control avanzado. Actualmente los reguladores utilizados se basan en una arquitectura jerárquica de controladores PID clásica. Dentro de las líneas de investigación del grupo se está trabajando en el desarrollo de una arquitectura de control que se adapta a las condiciones del sistema. Esta línea iniciada en el proyecto Kertrol [7] se continúa en el proyecto actual SIDIRELI en el cual los robots humanoides se han planteado como demostradores de la arquitectura en un escenario de cooperación con otros robots.

Referencias

- [1] Albero, M., Blanes, F., Benet, G., Simó, J., Perez, P., YABIRO, a new aproach to small biped robots. V Symposium on Intelligent Autonomous Vehicles. IAV'04. Lisboa. Portugal.
- [2] Albero, M., Blanes, F., Benet, G., Simó, J., Perez, P., YABIRO: prototipo de robot bipedo autónomo. XXIV Jornadas de Automática. León. Spain. 2003.
- [3] Albero, M., Nicolau, V., Blanes, F., Simó, J. E., (2007) microBiro: Un robot bípedo para la enseñanza y la investigación, XXVIII Jornadas de Automática. Huelva 2007.
- [4] Behnke, S., Robot competitions - ideal benchmarks for robotics research. In Proc. of IROS-2006 Workshop on Benchmarks in Robotics Research, October 2006.
- [5] Menegatti, E (2007) RoboCup Soccer Humanoid League Rules and Setup for the 2007 competition in Atlanta, USA April 4th.
- [6] Poza, J.L., Posadas, J.L., Simó, J.E., Pérez, P., Albero, M. (2004) Modelo de arquitectura de comunicaciones Frame-Sensor-Adapter (FSA). XXV Jornadas de Automática (JA2004) (ISBN 84-688-7460-4). Ciudad Real.
- [7] R. Simarro, J. Coronel, J. Simó, J.F. Blanes, Middleware for Control Kernel Implementation in Embedded Control Systems, 17th IFAC World Congress. Seoul: IFAC, 2008