COOPERACIÓN ENTRE ROBOTS BASADA EN UN PLANTEAMIENTO MULTI-AGENTE CON HERRAMIENTAS CACSD

A. Oller[1], R. García[2], J.A. Ramón[1], A. Figueras[1], y J.Ll. de la Rosa[2]
Grup d'Enginyeria de Control i Sistemes Intel·ligents, eXiT
1.- Departament d'Electrònica, Informàtica i Automàtica
2.-I nstitut d'Informàtica i Aplicacions (IIiA)
Laboratori Europeu Associat - Sistemes Intel·ligents i Control Avançat (LEA-SICA)
Universitat de Girona
{oller, rafa, josepa, figueras, peplluis}@eia.udg.es

Resumen

Las estructuras de decisión multi-agente pueden ser aplicables en sistemas industriales complejos, como pueden ser los sistemas de distribución de energía o de material, o en grandes plantas con diversas estaciones o con procesos complicados y con grandes cantidades de variables. Las competiciones futbolísticas multi-robot son un fórum adecuado donde tanto las reglas como las limitaciones técnicas proporcionan una plataforma de aplicabilidad de dichas estructuras. En este artículo se muestran nuevos desarrollos en hardware y software que permiten la decisión entre agentes que, en este caso, cooperan para jugar a fútbol.

1. INTRODUCCIÓN

Para facilitar que la comunidad del control automático desarrolle utilice los agentes en el análisis, diseño y implementación de sistemas de control, este trabajo propone la incorporación de las capacidades y el potencial de acción de los agentes en MATLAB/SIMULINK (M/S a partir de ahora). Dicho potencial fue inicialmente introducido en [7] a través del lenguaje orientado a agentes (AOL) llamado AGENT-0 que ofrece la posibilidad de trabajar con las capacidades, las creencias, los objetivos y los compromisos de los agentes. Tanto el sistema de visión basado en el reconocimiento de colores como la estructura de control de los micro-robots están incluidos en este artículo. El sistema micro-robótico está preparado para conseguir controlar movimientos de hasta 1 m/s, lo cual requiere el uso de técnicas de control avanzado para sistemas dinámicos. En las secciones 2, 3, 4 , 5 se describe el sistema de control global. En la sección 6 se describe someramente los dos niveles en los que se basa el control del comportamiento individual de los robots así como su comportamiento cooperativo.

Los robots han sido diseñados para competir en MIROSOT'97 [3] y en posteriores ediciones. Debido a las normas de dicha competición, los robots están limitados a las medidas de 7.5x7.5x7.5 cm. Y deben moverse autónomamente en un campo rectangular de 130x90 cm. (véase *"Fig.1"*). Además, los equipos pueden usar un sistema de visión central para obtener las posiciones y las orientaciones de los robots asi como de la pelota. Un ordenador central (host) deberá calcular los movimientos individuales y transmitirlos a fin de que sean ejecutados por cada robot.

El diseño y el control de los robots se ha realizado teniendo en cuenta principalmente dos problemas :

- Control a alto nivel . Las tareas más importantes consistirán en asignar posiciones a los robots y generar los comandos que deberán ejecutar. Todo esto se llevará a cabo en el host.
- Control a bajo nivel . La tarea principal consistirá en recibir y ejecutar con exactitud los comandos del host. Estas operaciones de ejecutarán en el mismo robot.

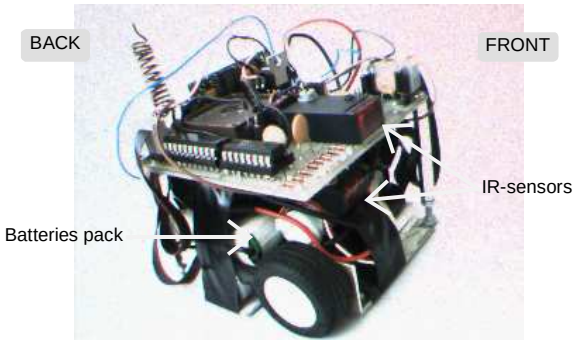
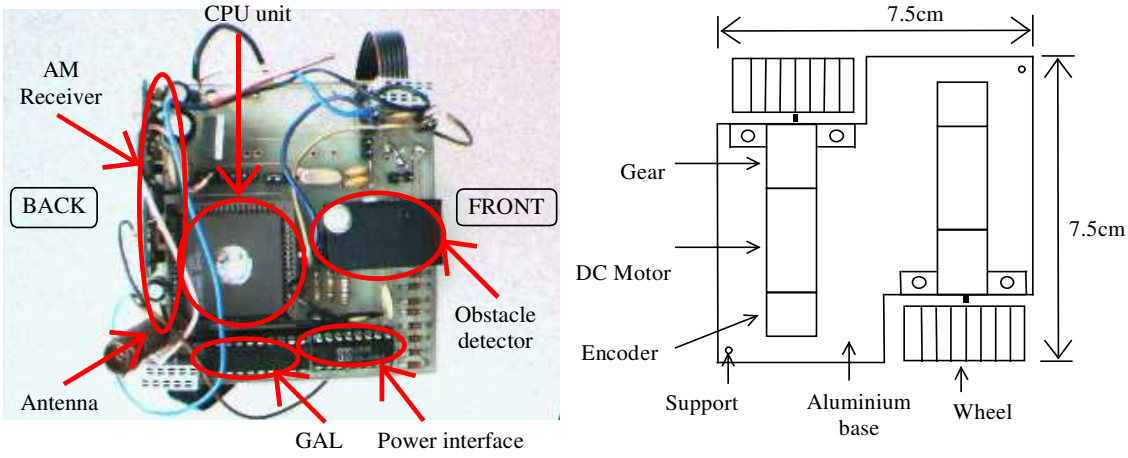


Fig. 1 Vista general de uno de los robots

2. ESPECIFICACIONES TÉCNICAS DE LOS ROBOTS

De acuerdo a las limitaciones físicas de los robots, las capacidades de automoción y de comunicación han sido implementadas usando pequeños dispositivos. A partir de *"Fig.1"* y de *"Fig.2"*, puede verse que los robots se componen de un receptor, una unidad CPU de proceso, una unidad de potencia, dos micro-motores, y baterías tanto para la automoción como para las unidades de proceso y de comunicación. A continuación se describen brevemente algunos componentes de los robots :

87C552 : Este microcontrolador contiene una ROM no volátil de 8K*8, una memoria de datos volátil de 256*8, 6 puertos E/S configurables, dos timers internos, 8 entradas ADC y un pulso dual con salida modulada.
L293 : Contiene dos puentes y drivers, con un componentes externo capaz de controlar la velocidad y el sentido de rotación de los motores.
Sensores : Dos detectores fotoeléctricos, E3S de OMROM, con una distancia de detección ajustable. La salida es de colector abierto.
Motores : Dos motores MAXON DC, 6V de 1.5W, velocidad nominal de 12.300 rpm, con reductor (16 :1).
Encoder : Encoder digital magnético con 16 pulsos por vuelta.
Receptor : Receptor super-heterodine con oscilador monolítico MOD STD 433 SIL, a 433,92 MHz. Totem Line AUR·EL S.p.A.
Baterías : 9 baterías de 1.2V - 1000 mA/h para cada robot.



(a) ***(b)***

Fig.2 : Vista superior del robot. (a) Detalles de la placa, (b) Esquema del chasis

3. EL SISTEMA EN SU CONJUNTO

Los robots no disponen de una cámara propia de forma que se utiliza una sistema de visión global centralizado. En este trabajo, el sistema de visión [2] está implementado en un Pentium PC que procesa 7

imágenes por segundo. Los datos que se obtienen son las posiciones de todos los objetos que se hallan en el campo de juego (pelota y robots) y las orientaciones de los robots del equipo local. Estos datos se comparten con el ordenador host usando el protocolo de comunicación TCP/IP el cual contiene diversos parámetros de estrategia, y algoritmos de decisión y de planificación de trayectorias. Todo ello le permite generar tramas para enviarlas a los robots con el formato siguiente :

ID_ROBOT | #comandos | comm_1 | comm_2 | .. | byte_comportamiento | ETX (1)

Esta trama permite que el robot pueda ejecutar rápidos movimientos en el campo de juego y evitando los obstáculos. Aunque la velocidad máxima del robot sea de 120 cm/sec (en línea recta), el valor usual de la velocidad es de 50 cm/sec para evitar problemas de patinaje en las ruedas. De esta manera, el robot está preparado para moverse trazando el camino obtenido por el algoritmo de planificación de trayectorias. Teniendo en cuenta que tanto los obstáculos (robots contrarios) como la pelota son móviles, el camino dictado por el host puede convertirse en erróneo. Por ello, con la ayudo de los sensores del robot y del byte_comportamiento de la trama (1) es posible hacer que el robot reaccione autónomamente ante la presencia de obstáculos, paredes o sencillamente para detectar la pelota y chutarla

.

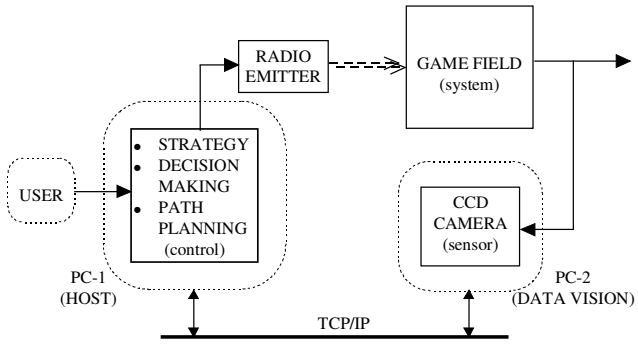
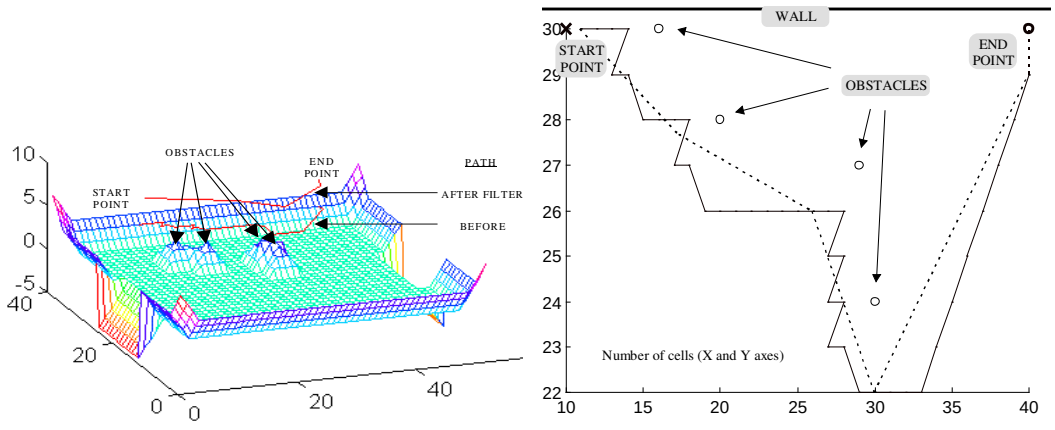


Fig. 3 Estructura de Control a Alto Nivel

4. ALGORITMO DE PLANIFICACIÓN DE TRAYECTORIAS

El algoritmo está basado en la idea de los campos de potencial artificiales [3] (APF) y las modificaciones propuestas por N. Clavel [1]. La totalidad del campo de juego se asocia a una matriz de forma que cada uno de sus elementos se corresponde a una pequeña cuadrícula real (o celdas). Se utiliza un sistema cartesiano de coordenadas para definir las posiciones de los centros de las celdas las cuales contienen el valor del potencial asociado. Una vez la matriz está calculada, el algoritmo busca sistemáticamente cual de las tres celdas vecinas (véase *"Fig. 5"*) contiene un valor de potencial inferior al de la celda ocupada por el robot. El trio de celdas dependerá de la dirección relativa en que se halle el objetivo : N, NE, E, SE, S, SW, W y NW. De esta manera el algoritmo sigue un método de búsqueda orientado a encontrar la celda objetivo la cual tendrá asociado un valor de potencial nulo.



(a) ***(b)***

Fig. 4 Ejemplo (a) Vista 3-dimensional del campo de potencial asociado a una situación en el campo de juego con 4 obstáculos. (b) Área ampliada: la línea discontinua corresponde al camino a seguir

Con este método se calcula una matriz de guía con la que se obtiene un camino libre de obstáculos y dirigido al objetivo. Dicho camino contiene un alto número de pequeñas oscilaciones a causa del proceso de búsqueda lo cual dificulta que el robot lo ejecute correctamente. Por ello, se aplica un proceso de filtraje a fin de suavizarlo (véase *"Fig.4"*) y se obtiene una trayectoria que puede traducirse directamente a la

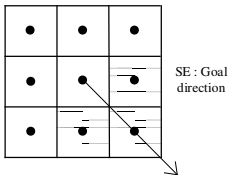


Fig. 5 Ejemplo : sólo se busca entre las celdas en gris para la dirección SE

secuencia definida en la trama (1).

5. CONTROL A BAJO NIVEL

En esta sección de describe el hardware del robot y se explica cómo funciona el programa principal, el cual permite el control de distancia, ángulo y velocidad.

Descripción del hardware

El hardware se ha diseñado basándose en el microcontrolador 87C552 (véase *"Fig. 6"*), el cual dispone de dos salidas PWM para controlar la velocidad de los motores. Los actuadores son dos motores DC, incluyendo un reductor y un encoder. La conexión entre el microcontrolador y estos motores se realiza a través de un IC L293. Los sensores consisten en un par de detectores fotoeléctricos situados en la parte frontal : uno en la parte alta para detectar sólo obstáculos y el otro en la parte baja orientado para detectar la pelota en una zona suficientemente cercana. Por lo tanto, la fusión de la información obtenida de los sensores permite saber al robot el tipo de objeto que tiene delante. Finalmente, la comunicación entre el host y los robots se realiza a través de radio de forma que cada robot tiene un código identificativo (ID_ROBOT) tal como muestra la expresión (1).

Programa principal

Cuando el robot recibe la secuencia de comandos (1), las ejecuta de inmediato. Cada comando contiene la siguiente información en un solo byte:

comm_X := ángulo (4 bits) | distancia (4 bits) (2)

Así, el robot está preparado para ejecutar trayectoias como las mostradas en la *"Fig.5b"* con los siguientes valores para el ángulo y la distancia :

- ángulo = {+90, +67.5, +45, +22.5, 0, -22.5, .., -90} grados → 9 valores diferentes
- distancia = {0, 5, 10, .., 45} cm → 10 valores diferentes

El movimiento del robot empieza con una rampa en velocidad y cuando finaliza la ejecución del último comando, se ejecuta otra rampa para detenerlo. El control del movimiento se realiza con la medida de tres señales : los dos encoders y la tensión de las baterías. Los encoders proporcionan la información en relación a la distancia y al ángulo. Sin embargo, la medida del voltaje de las baterías se utiliza para asegurar la alimentación a los motores : si las baterías están poco cargadas se incrementa el tiempo de ciclo del pulso PWM suministrado a los motores.

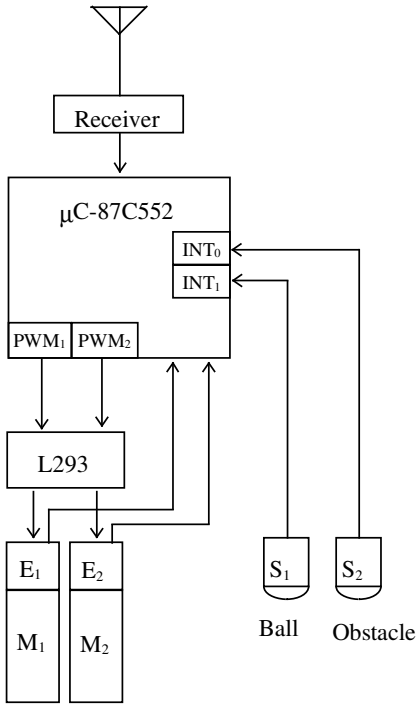


Fig. 6 Elementos básicos de control de los robots

Mientras los comandos se van ejecutando, los sensores pueden interrumpir el programa principal ya que van directamente conectados a las líneas de interrupción del microcontrolador.

6. COOPERACIÓN

La estructura de decisión en forma de agente se ha diseñado en dos bloques jerárquicos : el que contiene las tareas relativas a la colectividad (alta jerarquía, *bloque cooperativo*), es decir, acciones comunicativas como informar y pedir, y el *bloque privado* en donde el agente ejecuta sus acciones privadas.

- Bloque Privado : Cada agente puede ejecutar diferentes acciones de acuerdo a su conjunto de capacidades, según lo propuesto en el lenguaje AGENT0. En este bloque, este conjunto también incluye la capacidad de persistencia debido al hecho de que el entorno es dinámico y varía constantemente. Se han definido tres niveles para gestionar diversos tipos de información según el grado de abstracción : nivel agente (decisiones de comportamiento), nivel supervisor (planificación de trayectorias - intenciones ocultas - y cálculo de consignas), y nivel de control (ejecución de consignas).

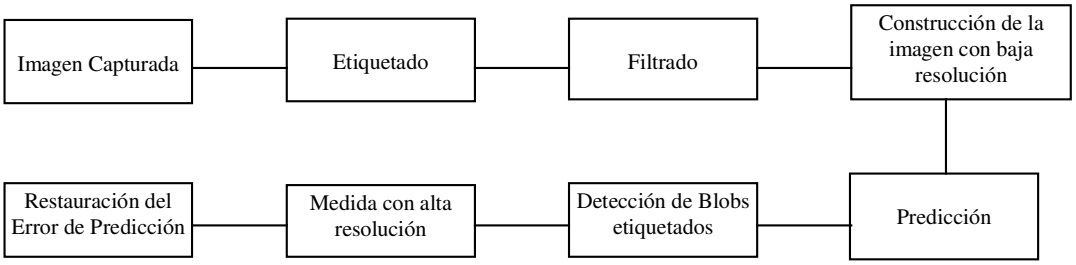
- Bloque Cooperativo : El nivel agente es la conexión entre el agente en sí (robot autónomo) y la colectividad. Por lo tanto, contiene acciones comunicativas como pedir o informar acerca de las decisiones de los demás. Este bloque está basado en la programación orientada a agentes, y utiliza el conjunto de primitivas de comunicación definidas en el AGENT0 [5].

Con esta estructura, el proceso de decisión tiene en cuenta tanto las decisiones colectivas como las individuales, y se basa en la comunicación entre agentes y el intercambio de decisiones entre los niveles supervisor y agente.

De esta manera, las decisiones colectivas se pueden basar en reglas estratégicas, mientras que las individuales se basan en índices de coste subjetivos calculados a partir de propiedades de la trayectoria a realizar. Éste índice se calcula en el nivel supervisor y, al estar contenido en el bloque privado, contiene diferentes parámetros para cada agente, es decir, el comportamiento de cada agente está incluido en el algoritmo de planificación de trayectorias.

7. SISTEMA DE VISIÓN

El sistema de visión se compone de una cámara color, de un procesador de color específico y de una tarjeta de adquisición de imágenes. La cámara de coloca a 2 metros de altura por encima del campo de juego y el sistema en su conjunto segmenta y sigue los robots móviles y la pelota según la siguiente secuencia :



Cada robot está equipado con dos bandas superiores de color : una consiste en el color del equipo y la otra es su color específico. La búsqueda del centro de gravedad de las regiones permite al sistema conocer la posición y la orientación de cada robot.

La paso de segmentación se realiza mediante un procesador específico que efectúa la conversión de RGB al espacio HSI en tiempo real. Este procesador utiliza las propiedades discriminatorias de los atributos de los tres colores para segmentar las regiones de la escena: Hue, Saturación e Intensidad. Seleccionando diferentes valores de estos atributos para cada objeto permite evitar los problemas de correspondencia que aparecen en los entornos multi-tracking. El resultado del modelo de segmentación es una imagen de 768x576 con diversas regiones etiquetadas en un fondo negro. El tiempo de ciclo del módulo de segmentación es de 40 ms, es decir, tiempo de vídeo en el CCIR estándar.

El segundo paso consiste en el cálculo de posiciones y orientaciones de los objetos usando la imagen segmentada. En primer lugar se utiliza un filtro de la mediana en las zonas etiquetadas para eliminar el ruido. Después el cálculo se basa en un proceso de multiresolución : se construye una segunda imagen tomando sólo uno de cada cuatro píxeles de la imagen filtrada y se buscan las zonas etiquetadas. Posteriormente se usa la imagen de alta resolución para calcular la posición más exactamente.

Hay un vector de estado que contiene información sobre las posiciones de robot de forma que en cada tiempo de muestreo se realiza una predicción basada en un modelo regresivo polinomial 2D a partir de la imagen de baja resolución. Dicha predicción se utiliza para facilitar la búsqueda de los objetos con la ayuda de un factor de error que varía en cada muestreo en función del error de la predicción anterior. El tiempo de ciclo de este módulo de seguimiento varía según la situación de los robots : el tiempo normal oscila entorno a los 150 ms.

Al final de cada tiempo de muestreo, el sistema de visión envía el vector de estado hacia el host de decisión mediante conexión TCP/IP.

8. CONCLUSIONES Y TRABAJO FUTURO.

Aunque el uso de agentes en el diseño de sistemas de control es posible, las herramientas de desarrollo están todavía en etapas iniciales para recabar la atención de los ingenieros de control y matemáticos aplicados. Buenos resultados en análisis e implementación de estas ideas aplicadas a los robots de fútbol hacen de sea bonito y claro el análisis (pero no el diseño, que toma mucho más tiempo que los enfoques tradicionales, aunque entonces la implementación es, más o menos, automática) no da ánimos para continuar la investigación. Esta comunicación se ha limitado a una descripción de los componentes del desarrollo de la plataforma de robots en el juego de fútbol.

El trabajo futuro se centrará pues en introducción de control predictivo en el nivel de control así como en la actividad de supervisión y control. También, el lenguaje PLACA será incluido para comparar como puede mejorar nuestro trabajo actual en términos de persistencia y representación del análisis, después de la mejora de la comunicación comportamiento (agente) y supervisión. La esperanza es que aumente el uso de agentes en aplicaciones simples de control automático con desarrollos parciales de la futura librería de agentes desarrollada en este entorno.

Agradecimientos

Este trabajo ha sido parcialmente cofinanciado por el TAP96-1114-C03-03 *Plataformas Integradas de CAD de Supervisión y Metodologías* de CICYT, y la red temática 1996XT 00023 *Tele-ensenyament i Recerca en Automàtica-IITAP*.

Referencias

[1] N.Clavel, R .Zapata, F.Sevilla. Non-Holonomic Local Navigación por means de Time and Speed Dependent Artificial Potencial Fields. IFAC Intelligent Autonomous Vehicles. Southampton, UK, 1993.

[2] O.Khatib, 1985. Real Time Obstacle Avoidance for manipulators and Mobile Robots. IEEE St. Louis (USA), 25-26 March 1985.

[3] Micro-Robot Futbol Tournament, 1997. KAIST, Taejon, Korea, June 1 - 5, 1997.

[4] MATLAB (ver. 4.2c.1) y SIMULINK (ver. 1.3c) are trademarks of Mathworks Inc.

[5] Y.Shoham. 1990. Agent-Oriented Programming. Technical Repport STAN-CS-1335-90, Computer Science Dep., Stanford Univ., Stanford, CA.

[6] S.R. Thomas. 1992. A logic for representing action, belief, capability, and intention. Stanford Working Document. Stanford, CA.

[7] de la Rosa J. L., Oller A., Vehí J., y Puyol J., "Futbol Team based on Agent-Oriented Programming", Robotics y Autonomous sistemas. Ed. Elsevier. In Pres.