

# **Diseño en Implementación de un equipo de robots autónomos que toman decisiones en tiempo real: Fútbol Robótico -Componente Inteligente.**

César Villarroel S.<sup>1</sup>, Carlos Calderón G.<sup>1</sup>, Rommel Carrillo C.<sup>1</sup>, Boris Vintimilla B.<sup>2</sup>

## **RESUMEN**

El presente trabajo resume los resultados obtenidos del componente inteligente <sup>3</sup> desarrollado sobre el proyecto de fútbol robótico, el cual comprende el diseño e implementación de un conjunto de algoritmos y técnicas de control e inteligencia artificial que permiten a los robots de un equipo cumplir satisfactoriamente el objetivo principal de anotar suficientes goles para ganar un partido de fútbol.

La motivación para desarrollar este proyecto fue que el fútbol robótico ha demostrado ser una plataforma excelente para el estudio y desarrollo de los *Sistemas Multiagente*, subcampo de la *Inteligencia Artificial Distribuida*, la cual tiene por propósito construir sistemas que involucran varias entidades (agentes o robots) que tienen un mismo objetivo y se desenvuelven en un ambiente complejo y dinámico. Representa además una buena oportunidad para desarrollar otros temas relacionados con el fútbol robótico puesto que éste es una área multidisciplinaria.

## **1. INTRODUCCION.**

Las competiciones de fútbol con robots fueron creadas básicamente para estimular la investigación y el desarrollo en los campos de la robótica y la inteligencia artificial, además de las áreas de visión por computador, procesamiento de imágenes, electrónica e informática por mencionar unas cuantas [1] [12].

Los competidores, estudiantes y científicos de universidades, deben diseñar robots equipados con sistemas de visión que detectan la pelota y que distinguen entre los jugadores de su equipo y los del equipo contrario. Cada robot posee inteligencia y actúa individual o colectivamente de acuerdo a la situación del juego, siempre con el objetivo de meter gol al equipo oponente.

---

<sup>1</sup> *Tesistas de Ingeniería en Computación, FIEC, ESPOL*

<sup>2</sup> *Ph.D. en Ingeniería Industrial, FIEC, ESPOL*

<sup>3</sup> *Proyecto: CICYT 1111-36-01 ESPOL.*

Básicamente la plataforma de trabajo sobre la cual se implementan las técnicas para cada equipo de fútbol de robots consta de: una cámara de video, una tarjeta de digitalización de imágenes, un computador, un transmisor-receptor de radiofrecuencia (RF), 3 micro-robots por equipo, un sistema de iluminación y un tablero de 1.3 mts. x 1.5 mts. que hace las veces de campo de juego, tal como se muestra en la figura 1 [16]:

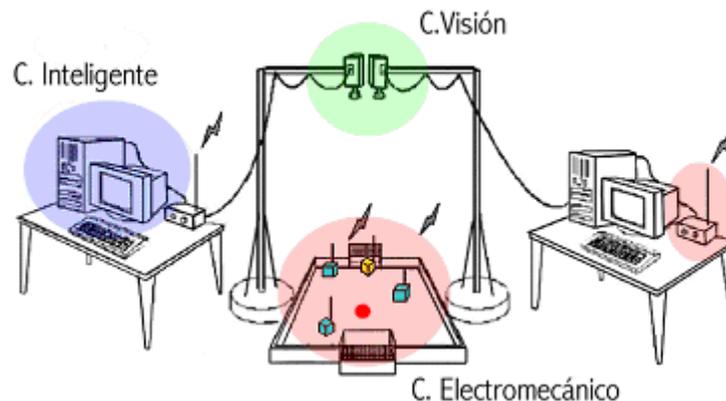


figura 1. Descripción gráfica de los componentes de fútbol robótico.

El proyecto en su forma global es complejo como para tratar de enfrentarlo en un solo problema. Por ello, se ha dividido en tres componentes que cumplen funciones específicas dentro del sistema. Estos tres componentes son: componente de visión, componente inteligente y componente electromecánico.

En este trabajo se describe el diseño e implementación del componente inteligente definido como el conjunto de algoritmos y técnicas de control e inteligencia artificial que permiten a robots móviles cumplir satisfactoriamente el objetivo principal de anotar suficientes goles para ganar un partido [17].

## 2. SISTEMAS MULTIAGENTE APLICADOS AL FÚTBOL ROBÓTICO.

Un juego de fútbol robótico es un ejemplo de un escenario dinámico donde múltiples agentes independientes interactúan en un mismo dominio denominado sistema multiagente [17]. Un estudio detallado sobre la clasificación de sistemas multiagente y los diversos tipos de sistemas de fútbol robótico son descritos en [1][10][13]. Basados en este estudio, hemos situado a nuestro sistema multiagente dentro de los *agentes homogéneos reactivos* (donde todos poseen las mismas características físicas, reciben las mismas entradas, son capaces de realizar las mismas acciones y actúan dependiendo de las posiciones de los compañeros y de la pelota) *no comunicativos* (puesto que no existe comunicación entre ellos)

resulta más conveniente aplicar un método de navegación autónomo reactivo para controlar los movimientos del robot.

### 3. IMPLEMENTACIÓN DEL PRIMER ALGORITMO PARA PATEAR LA PELOTA

El objetivo al patear el balón, es dirigir el robot hacia la posición de la pelota, para lo cual se requiere calcular las velocidades para cada rueda, y el ángulo de dirección del robot para orientarlo hacia la pelota. Para esto, primero se define un punto  $(target.x, target.y)$  hacia donde el robot debe ir, luego se orienta el robot en dirección al arco, y finalmente se envía al robot hacia la pelota con la velocidad máxima.

La figura 2 muestra gráficamente el desarrollo de este algoritmo inicial propuesto para ir a la pelota con el propósito enviarla a una dirección establecida.

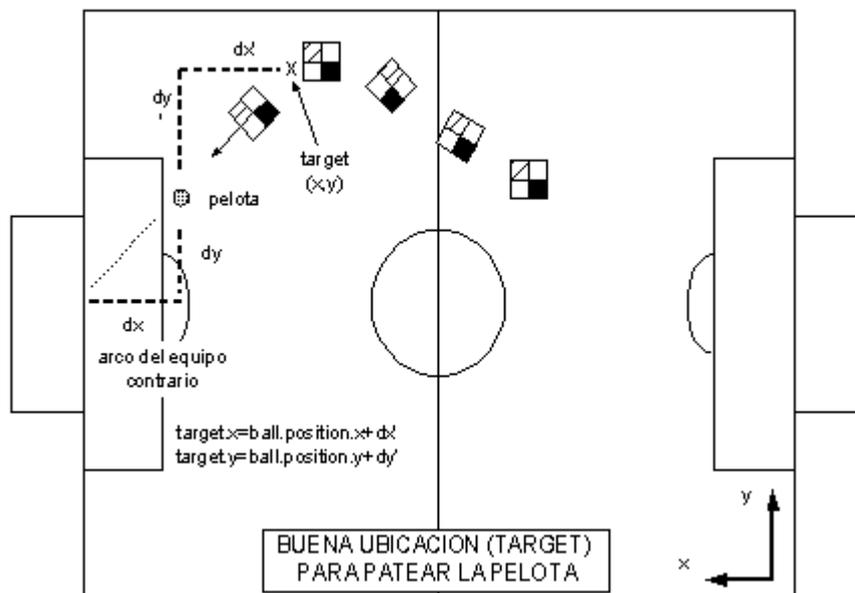


figura 2.- Algoritmo para que un robot patee la pelota hacia el arco mediante la definición de un objetivo  $(targetx, targety)$  [1].

### 4. ALGORITMO PARA EVITAR OBSTÁCULOS MEDIANTE LA GENERACIÓN DE OBJETIVOS INTERMEDIOS.

Los robots tratan de evitar colisiones planificando un camino entre los obstáculos. Este algoritmo trabaja calculando objetivos intermedios cuando existen obstáculos enfrente del jugador, los cuales pueden ser compañeros o adversarios. Básicamente el robot se dirige hacia una posición objetivo y al encontrar un

obstáculo en su camino calcula un objetivo intermedio, se dirige hacia él, para luego continuar su camino hacia el objetivo original.

La importancia de este algoritmo radica en que los robots pueden realizar sus funciones con mayor eficiencia en el momento de efectuar una jugada en conjunto y además evitar averías en los robots, debido a los continuos choques entre ellos. Una descripción grafica de la implementación del algoritmo puede ser vista en la figura 3.

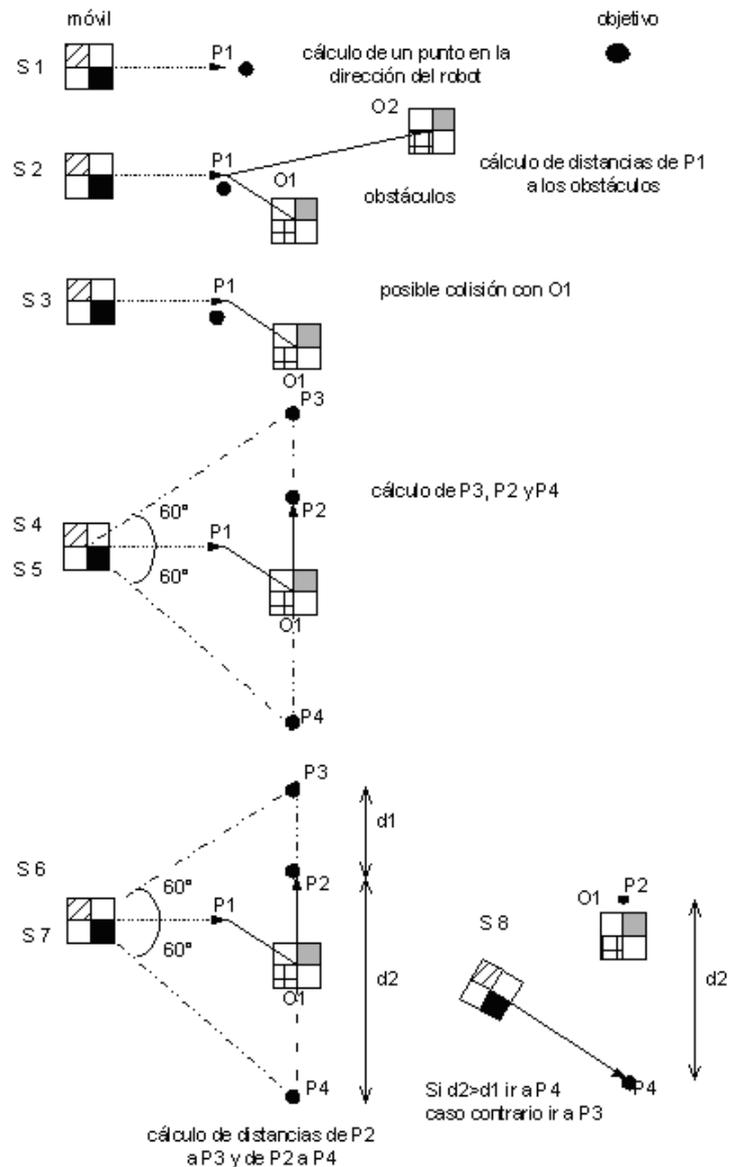


Figura 3.- Algoritmo implementado para evitar obstáculos mediante la generación de objetivos intermedios [1].

La descripción del conjunto de estados es la siguiente:

S0: Se predice un punto P1 a una distancia definida en la dirección del robot (móvil).

S1: Se calcula la distancia de cada obstáculo (otros robots) a este punto.

S2: Si alguna de estas distancias es menor que un umbral, entonces habrá colisión.

S3: Se identifica la posición y orientación del obstáculo.

S4: Se calculan dos puntos, el primero P3 sumando  $60^\circ$  al ángulo del robot y el segundo P4 restando  $60^\circ$  al ángulo del robot a una distancia de 10 unidades.

S5: Se predice un punto P2 en la dirección del obstáculo.

S 6: Se obtienen las distancias  $d_1$  y  $d_2$ ; entre P2 y P3 y entre P2 y P4 respectivamente.

S7: La distancia más grande,  $d_1$  ó  $d_2$ , determinará cual punto, P3 ó P4, será la posición hacia la que debe ir el robot para evitar la colisión.

S8: Continuar al estado S0 y así sucesivamente hasta que el robot llegue al objetivo final.

Este algoritmo de evasión de obstáculos inicialmente fue probado en partidos realizados en simulación, posteriormente se mejoró la implementación de este comportamiento para un robot móvil real aplicando el método de *campos univectoriales*.

## **5. ALGORITMO PARA SALIR DE LOS BORDES DEL CAMPO DE JUEGO.**

El objetivo de este algoritmo es evitar que el robot quede atrapado en los bordes de la cancha o dentro de los arcos. Este algoritmo funciona haciendo girar al robot manteniendo uno de los motores con velocidad cero, y el otro diferente de cero, hasta sobrepasar el ángulo de orientación con el cual el robot puede quedar atascado en el borde del tablero. Cabe señalar que este algoritmo está diseñado para ser ejecutado en el menor tiempo posible, con la finalidad de que ninguno de los robots oponentes tomen ventaja de esta situación.

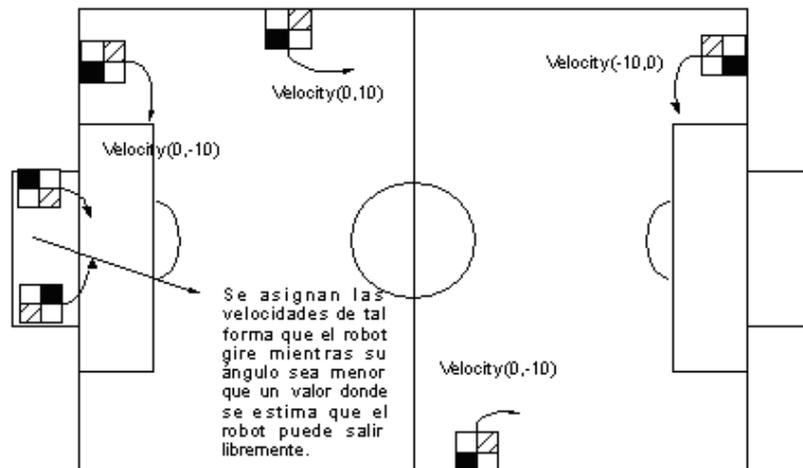


Figura 4.- Algoritmo para evitar un borde cancha y borde arco [1].

La figura 4 muestra la descripción gráfica del desarrollo de este algoritmo, el cual fue probado tanto en los partidos de SIMUROSOT (*Simulated Robot Soccer Tournament*) como en los de MIROSOT.

## 6. ALGORITMO PARA MOVER EL ARQUERO DE ACUERDO A LA POSICIÓN DE LA PELOTA.

Este algoritmo tiene como objetivo principal permitir al arquero bloquear el paso de la pelota a su arco. Para impedir que la pelota ingrese al arco, el robot arquero deberá realizar movimientos constantes en la dirección del eje  $y$  (figura 5 y figura 6). Para esto, la orientación de este robot siempre será de  $90^\circ \pm 1^\circ$ , donde el frente del robot está hacia el borde superior del tablero de juego.

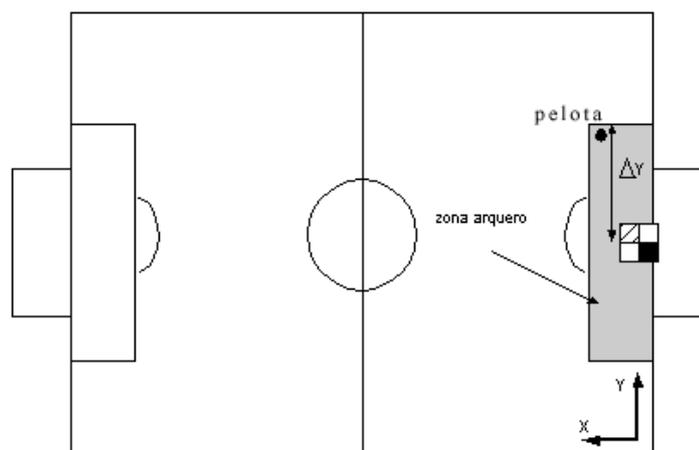


Figura 5. Algoritmo del arquero de acuerdo a la posición de la pelota

Cabe mencionar que el proceso anterior es ejecutado siempre y cuando la pelota esté dentro de la zona de arquero. De otro modo, el robot siempre tiende a posicionarse en el centro del arco si la pelota no está cerca de la portería. Para definir la habilidad de bloquear la pelota cuando ésta se dirige hacia la portería, se incluyó una función de predicción tanto para el arquero como para los defensas con el fin de estimar la trayectoria del balón. De este modo, la posición estimada de la pelota era usada en forma óptima para ubicar al robot arquero en diferentes posiciones, para esto se asumen tres situaciones distintas, descritas en detalle en la sección 5.2.4 de [1], mostrando gráficamente en la figura 6 donde la imagen a) muestra la ubicación inicial del robot dentro del tablero de juego y las otras tres las posiciones que deberá asumir el robot en cada situación.

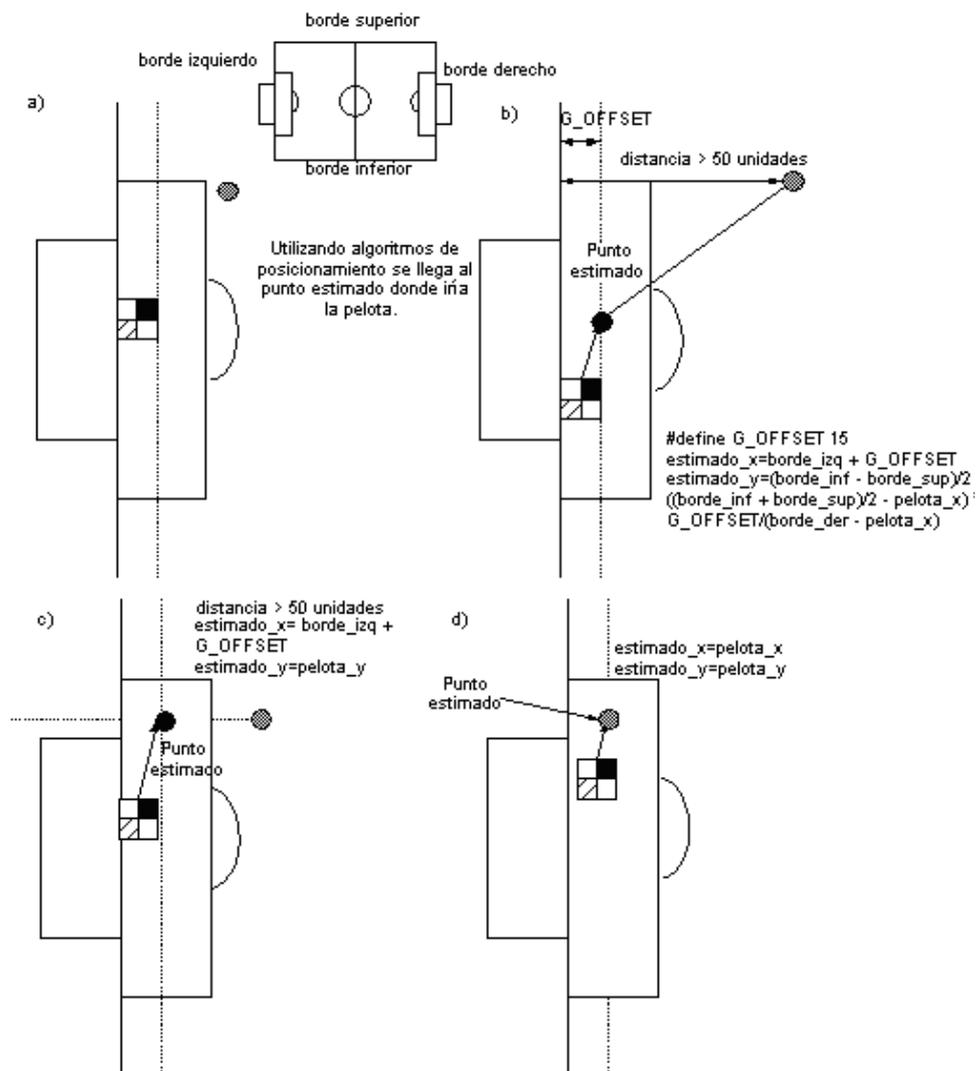


Figura 6. Algoritmo para estimar la trayectoria de la pelota

## **7. IMPLEMENTACIÓN DE LA ESTRATEGIA DE JUEGO PARA UN EQUIPO DE CINCO ROBOTS USANDO ESTRUCTURAS DE DECISIÓN.**

Para la implementación de este algoritmo, considerando un equipo de 5 robots (competición *Middle League MIROSOT* de la *FIRA*), se definieron: modos de juego (defensa, normal y ataque), zonas de juego (arquero, defensa 1, defensa 2, atacante 3 y atacante 4), posiciones pasivas y roles de cada jugador.

Inicialmente la estrategia del equipo se desarrolló mediante algoritmos jerárquicos IF-THEN-ELSE similares a un árbol de decisión, que obtiene información de una lista enlazada de dos nodos que guarda el estado actual y anterior de cada robot. Los últimos nodos del árbol representan comportamientos básicos (seguir la pelota, por ejemplo) y son guardados en la lista.

Una parte importante en la estrategia de juego es definir las zonas para cada jugador, esto es la zona de arquero, la zona para los defensas y la zona para los atacantes. La decisión del comportamiento que tomará un jugador en un momento del partido la realiza un módulo de selección, basándose en la posición de cada robot en la cancha y el rol (arquero, atacante o defensa) que tiene asignado cada uno de estos.

## **8. CONTROL DE MOVIMIENTO DE LOS MICRO-ROBOTS DE LA MIROSOT.**

Puesto que la categoría de MIROSOT (*Micro-Robots World Cup Soccer Tournament*) usa robots reales móviles con ruedas que deben trasladarse desde un punto a otro en el campo de juego de la forma más eficiente y rápida, es necesario aplicar diversas técnicas de control que tomen en consideración las restricciones cinemáticas existentes en los robots cuando estos ejecutan sus movimientos, el desarrollo de esta investigación es presentada en [1] y [18].

La figura 7 muestra el modelo de la cinemática del robot.

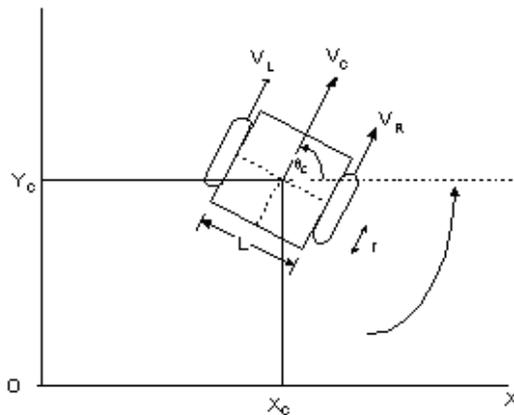


figura 7 Modelo de la cinemática del robot móvil [9]

donde  $r$  es el radio de las ruedas,  $L$  es el ancho del robot,  $V_R$  es la velocidad del motor derecho,  $V_L$  es la velocidad del motor izquierdo,  $V_C$  es la velocidad promedio de los dos motores,  $\theta_c$  es el ángulo del robot respecto a las coordenadas absolutas  $x-y$ , y  $(X_c, Y_c)$  es el centro de masa del robot.

De esta forma para ubicar al robot en un punto basta con asignar adecuadamente el valor de las velocidades  $V_L$  y  $V_R$ . El gráfico ilustra la ecuación y las variables de control para la asignación de las velocidades.

## 9. CAMPOS POTENCIALES UNIVECTORIALES USADOS PARA EVITAR OBSTÁCULOS Y PARA IR HACIA LA PELOTA.

Un conocido algoritmo de planificación en línea es el algoritmo de *funciones potenciales*, el cual se caracteriza por que genera diferentes tipos de vectores de tal forma que el robot es orientado en la dirección de esos vectores. Estos vectores son generados continuamente permitiendo controlar a los robots en tiempo real. Los vectores generados por el algoritmo de funciones potenciales pueden ser de al menos dos tipos: uno de atracción hacia el objetivo y otro de repulsión hacia los obstáculos sean estos adversarios o compañeros de equipo, e inclusive en algunos casos la pelota. Por tanto, la planificación consistirá en una adecuada combinación de ambos tipos de vectores [2][3][4][5].

Para facilitar la implementación del algoritmo, se asume que la magnitud de los vectores en cada momento tiene el valor de la unidad [14], es por ello que estos campos son llamados *campos univectoriales*.

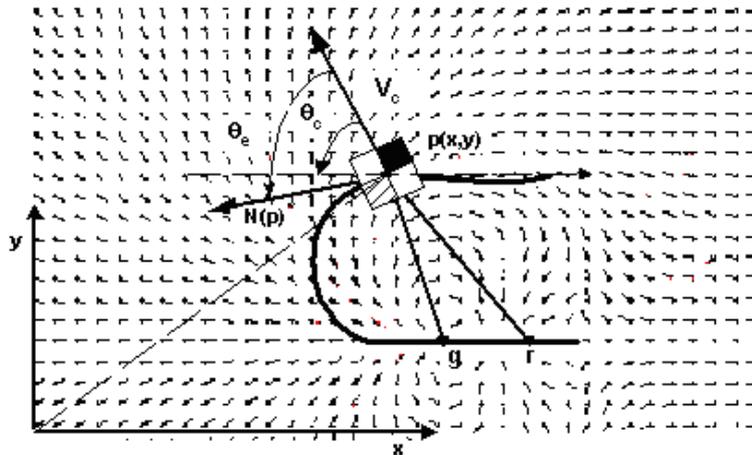


Figura 8.- Campo univectorial para una posición g deseada.

En la figura 8 las pequeñas flechas representan la dirección del campo potencial en un determinado punto. Las líneas negra gruesa representa trayectoria que puede seguir el robot.

Inicialmente la dirección del robot en una posición  $p$  sigue la dirección del *campo univectorial de atracción*  $N(p)$  hacia el objetivo. Sin embargo, cuando el robot se encuentra en presencia de un obstáculo, una parte de la dirección de este *campo univectorial*  $N(p)$  se modifica, definiendo de esta forma, un *nuevo campo potencial de repulsión*  $R(p)$  alrededor del obstáculo.

Para generar  $R(p)$  en la posición  $p$  del robot se define un radio  $R_o$  desde el centro del obstáculo hacia un poco más allá del borde del círculo englobante del obstáculo (círculo de línea roja), figura 9. De esta forma, la dimensión de  $R_o$  tiene una holgura suficiente para que el robot no colisione con el obstáculo. El *nuevo campo univectorial*  $R(p)$  se orienta en dirección tangencial al borde del círculo de radio  $R_o$ , y una línea  $L$  desde la posición  $p$  del robot hacia la posición del centro del obstáculo forma un triángulo rectángulo.

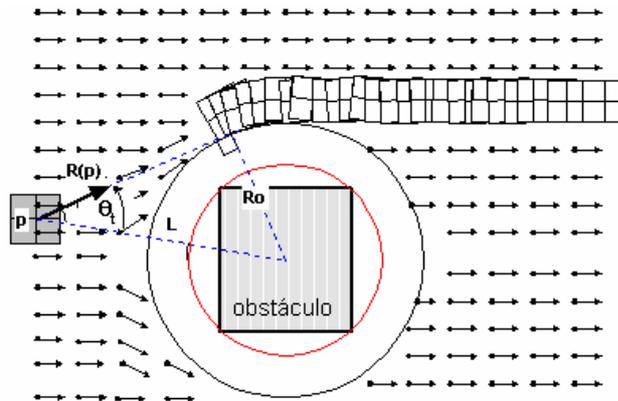


Figura 9 Generación del campo univectorial de repulsión para evitar un obstáculo

Para propósitos de la implementación del algoritmo de generación de  $R(p)$  se define un anillo de grosor  $Mo$ , el cual es aproximadamente igual al ancho del robot, alrededor de  $Ro$ , tal como se muestra en la Figura 10 (anillo sombreado de rojo). Este anillo define la zona del nuevo campo univectorial  $R(p)$  que permite evitar el obstáculo, una vez fuera de este anillo el robot usa el *campo univectorial de atracción*  $N(p)$  para ir hacia el objetivo.

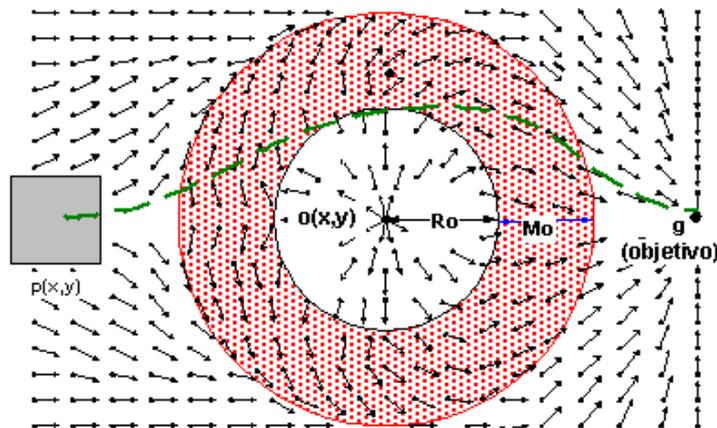


Figura 10 Zona del campo univectorial  $R(p)$  que permite evitar un obstáculo

En esta zona definida por  $Mo$  el valor del campo modifica los valores de las velocidades  $V_L$  y  $V_R$  correspondientes a los motores izquierdo y derecho del robot respectivamente.

Para determinar cuando se debe aplicar *el campo univectorial de repulsión*  $R(p)$  para evitar obstáculos se ha propuesto considerar lo siguiente:

Una modificación implementada a este algoritmo permite especificar las condiciones para realizar este comportamiento, utilizando como parámetros la distancia del obstáculo a la recta formada entre el objetivo y el robot, el radio del círculo formado por el obstáculo, las distancias al obstáculo y al objetivo, así como también los valores absolutos de los ángulos entre el objetivo-robot, obstáculo-robot, ángulo-robot. Para mantener un mejor control del robot móvil durante la ejecución del algoritmo, la velocidad lineal del robot se reduce en presencia del obstáculo.

Para probar los algoritmos de control basados en campos univectoriales (potenciales) se utilizaron robots de 7.5 cm x 7.5 cm de la categoría MIROSOT. Las velocidades de los dos motores se deben acotar en un rango [-50,50] para mantener un mejor control cuando el robot esta en presencia de obstáculos.

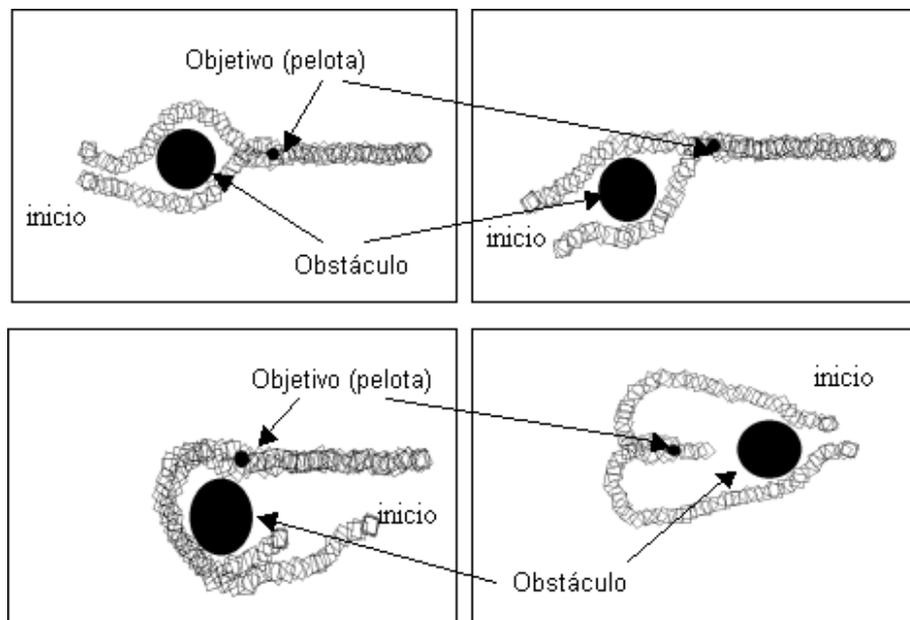


Figura.11. Comportamiento del algoritmo de evitar obstáculos.

Los valores de restricción utilizados para la velocidad máxima lineal y velocidad máxima de giro fueron  $V_m = 150 \text{ cm/s}$ ,  $R_m = 500 \text{ rad cm/s}$  respectivamente. La figura 11 muestra varias trayectorias ideales que podría tomar el robot con la finalidad de evitar chocar con los obstáculos.

La figura 12 presenta el resultado de aplicar el algoritmo de evitar obstáculos sobre una situación de juego en un sistema MIROSOT de 3 vs. 3. En esta figura se aprecia que la trayectoria que sigue el robot para llegar a la pelota sigue hacia la parte superior del tablero de juego en lugar de la parte inferior. Esto se debió a que el robot estaba inicialmente orientado hacia esa dirección.

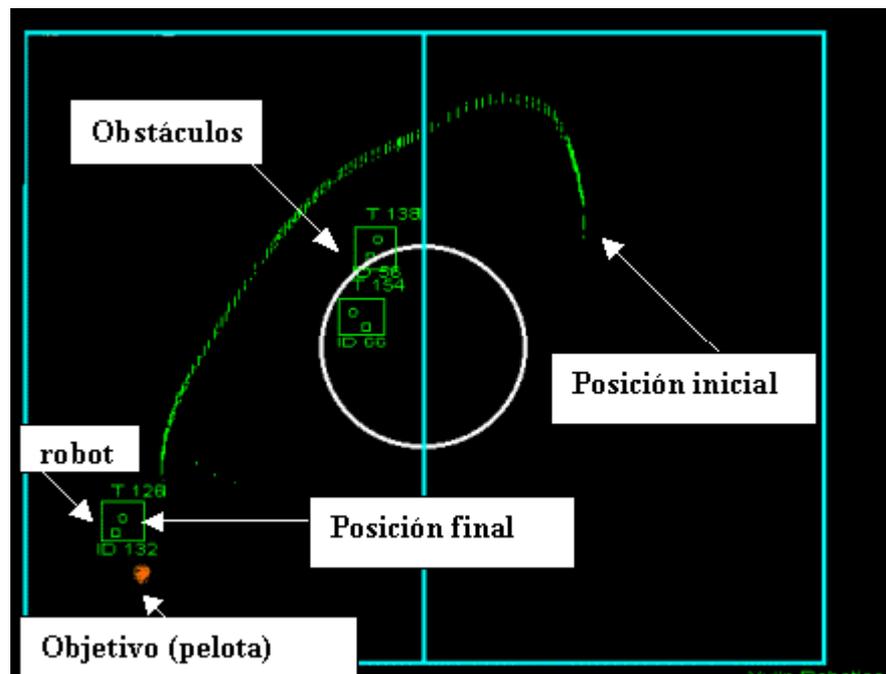


Figura 12. Resultado de la aplicación del algoritmo de evitar obstáculos sobre el sistema MIROSOT de 3 vs. 3

Con relación a la aplicación de la función de predicción de la posición futura de la pelota, es importante configurar adecuadamente los colores de los robots y de la pelota en el sistema de visión, tal que no se detecten posiciones falsas (inexistentes) de la pelota.

## 10. APRENDIZAJE POR REFUERZO APLICADO A FÚTBOL ROBÓTICO

Es relativamente sencillo construir un rol para cada robot (atacante, defensa, arquero) y que cada uno lo lleve a cabo individualmente durante un partido. El problema radica en hacer que los robots coordinen sus movimientos y cooperen todos para cumplir con el objetivo del juego, que es ganar el partido. Nosotros hemos afrontado este problema usando una técnica de aprendizaje por refuerzo llamada *aprendizaje Q*, debido a que es un método muy utilizado en este tipo de aplicaciones [7][8].

Para la construcción de una estrategia para un equipo de tres robots (competición *Small League MIROSOT* de la *FIRA*), visto como un sistema multiagente, se propuso aplicar “aprendizaje Q modular”, aquí cada agente incluye un módulo de aprendizaje cuyas acciones a realizar son decididas por un módulo que hace las veces de mediador. El módulo mediador elige el rol que tendrá cada robot y las acciones que ejecutarán en un determinado momento del juego, esto es retribuido con una recompensa, cuyo valor se incrementa conforme se ejecuta cada acción.

Para la construcción de este algoritmo, de acuerdo a las situaciones de juego (estados) se consideró a 2 agentes (robots) como uno solo, llamado *agente acoplado*, con el objetivo de resolver en gran parte el problema de gasto en espacio de memoria, reduciendo el número de estados y acciones que un agente individual necesitaría para aprender a realizar las mejores acciones en un partido de fútbol [11].

Debido a que no es conveniente describir todas las situaciones de un juego de fútbol robótico solamente con algunos enunciados de condición, debido a su naturaleza dinámica y compleja, resulta factible emplear técnicas como *aprendizaje por refuerzo*, ya que nos permite enseñar al agente una conducta a través de iteraciones de pruebas y errores [14].

Otras de las ventajas que presenta el aprendizaje Q es que al aplicarlo no se necesita de un modelo o patrón de juego que deba seguir el agente, las decisiones a tomar pueden aprenderse directamente a partir de la retroalimentación de recompensas obtenidas a lo largo del juego.

La figura 13 muestra un esquema de la arquitectura de aprendizaje Q modular.

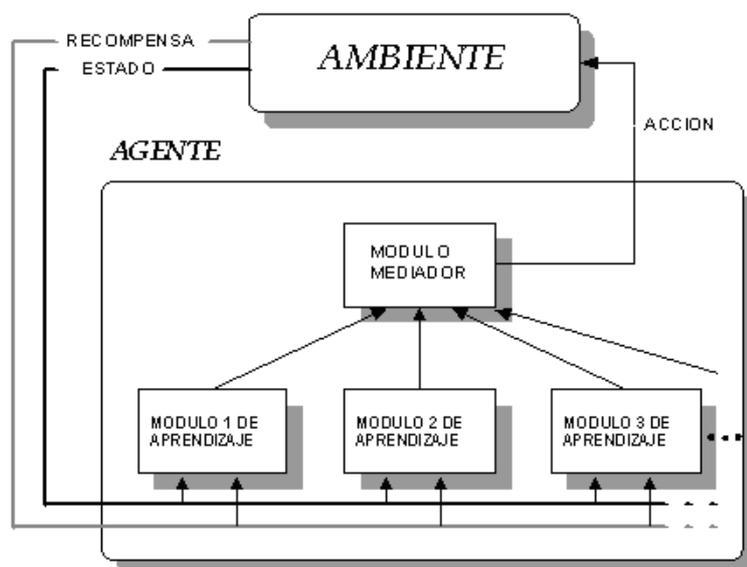


figura 13. Arquitectura de Aprendizaje – Q modular.

En el aprendizaje Q modular cada agente incluye un módulo de aprendizaje Q, donde se concentra el aprendizaje de un solo agente y no el de otros agentes. Para completar la meta global de aprendizaje multiagente, un módulo mediador, realiza la decisión final seleccionando la acción más conveniente, basándose en los

valores Q recibidos desde cada uno de los módulos de aprendizaje. En [11], cuando el módulo mediador hace esta elección considerando el más alto valor Q recibido desde cada uno de los módulos de aprendizaje, es llamada la estrategia de la cantidad más grande.

## **11. RESULTADOS DE LAS TÉCNICAS DE MOVIMIENTO UTILIZADAS POR EL EQUIPO FUROEC EN LAS COMPETICIONES DE LA FIRA 2002 WORLD CUP.**

Nuestro equipo de robots representó al Ecuador en las competiciones mundiales 2002 de fútbol robótico realizado en el mes de mayo en Corea y bajo la organización de la *Federación Internacional de Fútbol Robótico Asociado (FIRA)* [6]. Las técnicas implementadas permiten a los robots: atacantes, defensas y arquero, poseer habilidades para patear, salir de los bordes de la cancha y rechazar la pelota. Estas técnicas se basan en algoritmos que actúan dependiendo de la posición de la pelota dentro del campo de juego. Todas las técnicas implementadas fueron aplicadas tanto en las competiciones de SIMUROSOT (*Simulated World Cup Soccer Tournament*) como en las competiciones de fútbol robótico con robots reales MIROSOT, ambas en la categoría de 5 vs. 5 robots.

Los algoritmos utilizados para las competiciones fueron los siguientes: un algoritmo y sus diagramas de estado para patear la pelota, un algoritmo para evitar obstáculos móviles (definiendo puntos intermedios al objetivo), un algoritmo sencillo para no quedar atrapado en los bordes del campo de juego, un algoritmo para definir los movimientos de un arquero dependiendo de la posición de la pelota, un algoritmo que utilizan los defensas para rechazar la pelota, se desarrollaron estrategias para atacante y defensa, finalmente, se desarrolló la estrategia general del equipo, definiendo zonas de juego, posiciones pasivas y roles para cada jugador [15].

## **12. CONCLUSIONES**

En el fútbol robótico cada equipo debe tener una estrategia de juego que le permita obtener el mayor número de goles sobre su equipo oponente. Esta estrategia de juego incluye darle a los robots un posicionamiento, una formación, un rol específico, etc.

La utilización de los algoritmos de campos potenciales, mejoró significativamente el control de movimiento del robot móvil con ruedas en un ambiente de trabajo tan dinámico y complejo como es el fútbol robótico.

El método de planificación en línea de campos univectoriales para agentes reactivos se puede aplicar en ambientes donde se obtenga información de las

posiciones absolutas y el tamaño relativo de cada obstáculo, para que el robot pueda llegar a su objetivo final usando varios campos univectoriales que lo lleven a su destino con la orientación y posición adecuados.

Mediante la aplicación del aprendizaje Q modular, se consiguió mejorar la falta de coordinación entre dos robots, puesto que permite el intercambio de roles entre un robot defensa y un robot atacante. Con esto se reduce la posibilidad de bloqueo mutuo entre los dos robots, y aumenta la posibilidad de que el ataque sea ejecutado por el robot que se encuentra en la posición más idónea para realizar esta función.

Futuras implementaciones considerarán la posibilidad de seleccionar la acción que el agente acoplado deba realizar diferentes situaciones del juego, como en situaciones de mutuo bloqueo o bloqueo de parte de los robots oponentes.

Diseñar estrategias de juego para un equipo de robots puede ser muy complicado, por ello es recomendable comenzar con algoritmos sencillos y probarlos en ambientes simulados, luego una vez que estos hayan sido suficientemente probados implementarlos en ambientes reales. Una plataforma de prueba de los algoritmos es el simulador de fútbol robótico desarrollado por la FIRA.

Se ha logrado publicar los resultados del proyecto en diferentes congresos y eventos, internacionales y nacionales, además de pertenecer al organismo FIRA (Federación Internacional de Fútbol – Robot Asociados) para participar en los campeonatos y eventos organizados por el mismo.

Con el desarrollo de este proyecto se quiere incentivar la elaboración y ejecución de proyectos de investigación en el campo de Visión por Computador, Inteligencia Artificial, Robótica, Procesamiento de Imágenes entre otras áreas.

### **13. BIBLIOGRAFÍA.**

- [1] Villarroel César, Calderón Carlos, Carrillo Rommel. “Diseño e Implementación de un equipo de robots autónomos que toman decisiones en tiempo real: Fútbol Robótico - Componente Inteligente”, Tesis, Escuela Superior Politécnica del Litoral, Guayaquil, 2003.
- [2] Alvarez Juan Carlos, "Planificación del Movimiento de Vehículos Autónomos basada en Sensores", (Tesis Doctoral Universidad de Oviedo 1998)
- [3] Arkin Ronald C., “Integrating Behavioral, perceptual and World Knowledge in Reactive Navigation”, Robotics and Autonomous Systems, vol 6, pp. 105-122, 1990.
- [4] Borenstein J., Y. Koren. “Real-time obstacle avoidance for fast mobile robots”, IEEE, Trans. Systems Man and Cybernetics, pp-. 1179-1187, 1989.

- [5] Brooks Rodney A., "A Robust Layered Control System for a Mobile Robot", IEEE Journal on Robotics and Automation, vol, RA-2, no 1, March 1986.
- [6] Federation of International Robot-soccer Association (FIRA). Homepage: <http://www.fira.net>.
- [7] Itsuki Noda, Hitoshi Matsubara and Kazuo Hiraki, "Learning Cooperative Behavior in Multi-agent Environment", Electrotechnical Laboratory, Tsukuba, Ibaraki 305, Japan, August, 1996.
- [8] Kaelbling L.P., M. Littman L., Moore A. W., "Reinforcement Learning: A survey", Journal of Artificial Intelligence Research, vol 4, pp 237-285, 1996.
- [9] Kim Jong Hwan, Kim Yong-Jae. "Univector Field Navigation Method for Fast Mobile Robots in Dynamic Environment", 2002 FIRA Robot World Congress Proc., pp. 165-170. Seoul, 2002.
- [10] Kim, J.-H., Shim, H.-S., Kim, H.-S., Jung, M.-J., Choi, I.-H., and Kim, J.-O. "Cooperative multi-agent system and its real time application to robot soccer." Proc. IEEE Int. Conf. on Robotics and Automation, Albuquerque, New Mexico, Vol. 1, (April 1997), pp 638-643.
- [11] Kui Hong Park , Jong Kwan Kim, "Modular Q- learning based Action Selection for Narosot", Proc. 2002 FIRA Robot World Congress, Seoul, Korea, May 2002, pp. 222-226.
- [12] Robot World Cup Initiative (RoboCup). Homepage: <http://www.robocup.org>.
- [13] Stone P., "Layered Learning in Multiagent Systems: Robot Soccer", School of Computer Science Cargenie Mellon University, Pittsburgh 1998.
- [14] Sutton R. S., Barto A. G., "Reinforcement Learning: An Introduction", Bradford Brooks/MIT Press, 1998.
- [15] Villarroel C., Carrillo R., Calderón C., "Técnicas de movimiento utilizadas por el equipo FUROEC en las competiciones de la FIRA 2002 World Cup", EspolCiencia 2002, Escuela Superior Politécnica del Litoral, Guayaquil, Ecuador, Octubre 2002.
- [16] Vintimilla B., Monsalve A. C., Villarroel S. C., Calderón G. C., Carrillo C. R., Salamea V., Paillacho C. D., Chicala J., Atiencia L., Jacho R., "Sistemas multiagentes aplicados al futbol robotico: problematica existente", IV Jornadas Iberoamericanas de Robotica, Seminario 3, Panama, 27-31 Enero, 2003
- [17] Vintimilla B.X., Villarroel C., Calderón C., Carrillo R., "Survey on Learning in Multi-Agent Systems", 2002 FIRA Robot World Congress, Seoul, Korea, May 2002, pg. 305-308. ISBN: 89-86522-47-0-93560 \* Este artículo fue seleccionado para ser publicado en el Libro de *Intelligent Robots: Vision, Learning and Interaction*, Sección: *Learning for Navigation and Control*, Editorial: KAIST Press, año 2003.
- [18] Vintimilla B.X., Villarroel C., Calderón C., Carrillo R., "Univector Fields for Motion Control in Mirosoft Robots ", submitedo en el 2003 FIRA Robot World Congress, Austria.