



Simulación de Tareas de Recolección en un Sistema Multi-Agente Mediante Webots

Germán Andrés Vargas Torres¹ y Ricardo Andrés Castillo Estepa²

Recibido: Junio 16 de 2013 **Aprobado:** Octubre 15 de 2013

Resumen:

Este artículo presenta el desarrollo de un ambiente virtual orientado a simular tareas de exploración y recolección de objetos en un sistema multi-agente (SMA) compuesto por robots humanoides. Previa caracterización del hardware y software que constituye el sistema real, se hace uso del entorno de desarrollo Webots para crear el ambiente virtual, simplificando la inclusión y distribución de los agentes y objetos a recolectar, al igual que en la implementación de funcionalidades propias de los robots humanoides. Simulando una tarea de exploración y recolección estándar y evaluando la tasa de recolecciones exitosas, el ambiente es usado para determinar el número óptimo de agentes en operación que pueden recolectar un único objeto. Los resultados demostraron que el número óptimo era levemente inferior al número formulado en la hipótesis.

Palabras clave: Control de robots, DARwIn-OP, Robot Bioloid, Robots humanoides, Servomotores, Sistemas multi-agente, Simulación por computador, Webots

Abstract:

This article presents development of a virtual environment to simulate exploration and object gathering tasks in a Multi-Agent System (MAS) composed by humanoid robots. After characterizing the hardware and software constituting the real system, the Webots development environment is used to create the virtual environment, simplifying inclusion and distribution of agents and objects to be collected, as well as in implementation of the humanoid robots' functionalities. By simulating a standard exploration and gathering task and evaluating the rate of successful lifts, the environment is used to determine the optimal number of agents in operation that can collect a single object. Results demonstrated that the optimal number was slightly inferior to the value proposed in the hypothesis.

KeyWords: Bioloid Robot, Computer simulation, DARwIn-OP, Humanoid robots, Multi-agent systems, Robot control, Servomotors, Webots

1 Ingeniero en Mecatrónica. Asistente de Investigación - Grupo de Investigación DAVINCI. Universidad Militar Nueva Granada. Bogotá, Colombia.

2 Ingeniero en Mecatrónica. Magister en Ingeniería Mecánica. Docente de Tiempo Completo e Investigador Principal - Grupo de Investigación DAVINCI. Universidad Militar Nueva Granada. Bogotá, Colombia.



I. INTRODUCCIÓN

El paradigma de los sistemas multi-agente (SMA) según Woolridge (2009) comprende todo sistema computarizado compuesto por múltiples racionales que interactúan entre sí. Para Shoham y Leyton-Brown (2009) el uso de sistemas multi-agente (SMA) en ingeniería busca aprovechar dichas interacciones para que los agentes entablen cooperación al realizar tareas complejas. Cheng, Raja y Lesser (2013) indican que el diseño de un SMA debe contemplar la optimización de una medida de desempeño global gracias a la cooperación entre agentes.

En su mayor grado de abstracción un agente se define como un proceso computacional autónomo, con capacidades de percepción de su entorno y actuación sobre el mismo. Bajo la caracterización de Pérez y Cruz (2007) un agente perteneciente a un SMA también incluye la capacidad de comunicarse con otros agentes para usar técnicas cooperativas tales como colaboración, coordinación y negociación. Con base en dicha definición, se deduce que un robot humanoide es uno de los agentes más complejos que pueden conformar un sistema multi-agente.

El proyecto de investigación “Desarrollo de estrategia con aprendizaje on-line para comando de grupo de robots humanoides utilizando la plataforma robótica DARwIn-OP”, perteneciente al Grupo de Investigación DAVINCI de la Universidad Militar Nueva Granada, posee entre sus objetivos el desarrollo de un ambiente virtual de simulación para evaluar el comportamiento de un sistema multi-agente compuesto por robots humanoides en tareas de exploración y recolección de objetos. La hipótesis a evaluar es la siguiente: para un entorno de trabajo de 1.80 por 1.35 metros el número máximo de agentes que pueden recolectar correctamente un único objeto de 0.3 por 0.3 metros es 3, debido a las restricciones en el espacio de operación.

Dada la complejidad de este tipo de sistemas, se selecciona el entorno de desarrollo Webots para facilitar la creación del ambiente virtual. Como objetivo específico, se requiere implementar los elementos que componen el entorno (creación y distribución de los agentes y objetos a recolectar) y las funcionalidades de los robots humanoides (generación de patrones de marcha y movimientos, visión de máquina y comunicación inalámbrica). La estrategia de investigación consiste en analizar la hipótesis formulada, así como los elementos y funcionalidades mencionados, mediante la planeación y desarrollo de una tarea de exploración y recolección estándar,

así como una serie de criterios que determinan una recolección exitosa. Dicho análisis se realiza de forma experimental: por medio de ejecuciones sucesivas en las cuales se registran los criterios de éxito, se evalúa la tasa de recolecciones exitosas en función del número de agentes en operación.

II. ABSTRACCIÓN DEL SISTEMA MULTI-AGENTE (SMA)

Como paso previo al desarrollo del ambiente virtual de simulación, se requiere caracterizar el sistema multi-agente real a fin de determinar los elementos de hardware que lo conforman, así como la arquitectura de los robots humanoides. La Figura 1 ilustra los componentes de hardware presentes en el sistema: el equipo de agentes comprende cinco robots humanoides Bioloid que interactúan con igual número de mesas de distintas dimensiones distribuidas al interior del entorno o espacio de trabajo; un área rectangular de 1.80 m por 1.35 m (indicada mediante guiones en la figura). Puesto que los robots no poseen cámaras, se requiere incorporar un sexto agente intermediario: un computador ubicado por fuera del espacio de trabajo captura video desde una vista de planta superior del entorno mediante una webcam. Posteriormente procesa los datos obtenidos y transfiere información de posición y orientación a los robots vía comunicación inalámbrica ZigBee.

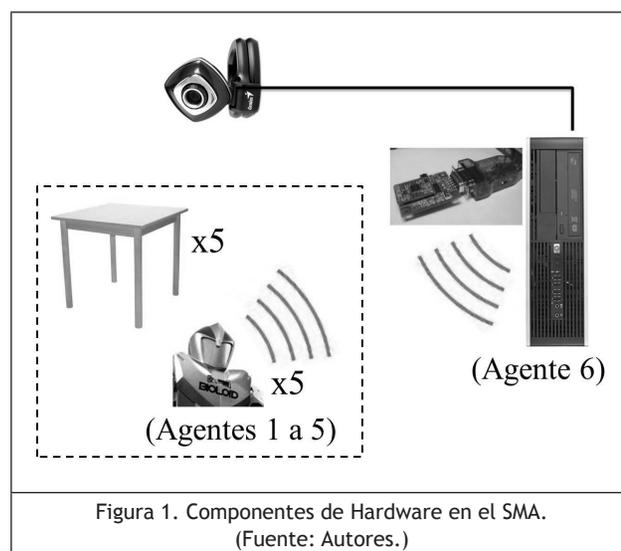
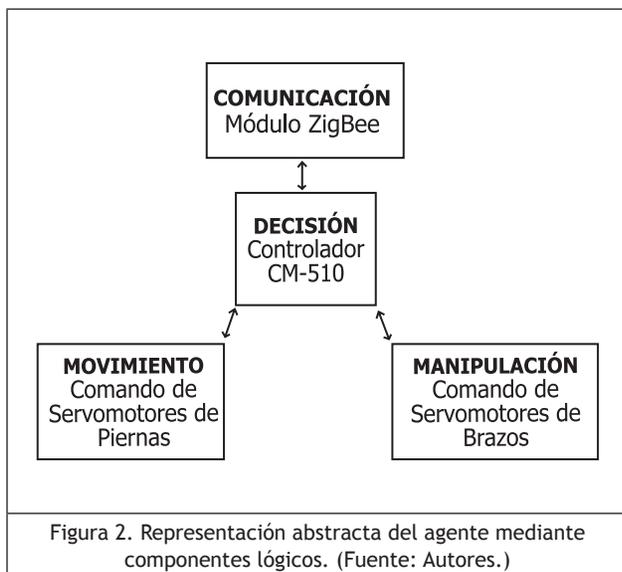


Figura 1. Componentes de Hardware en el SMA.
(Fuente: Autores.)

Se opta por representar los robots humanoides mediante el esquema propuesto en la Figura 2, bajo el cual la arquitectura del agente se abstrae en términos de componentes lógicos e interrelaciones entre los mismos.



Cada bloque reúne múltiples subsistemas presentes en el robot físico según sus funciones. Como ejemplo, el bloque de movimiento agrupa los servomotores que accionan las articulaciones de las piernas del robot y el subsistema de control de bajo nivel de los mismos. Cuando el bloque de decisión envía la orden para ejecutar un determinado patrón de caminata (Ej.: “Marcha hacia adelante”), el bloque de movimiento transforma dicha orden en secuencias de posiciones y velocidades angulares que los servomotores ejecutan para generar el movimiento correcto del robot.

III. CREACIÓN DEL ENTORNO VIRTUAL, AGENTES Y OBJETOS A RECOLECTAR

El entorno de desarrollo Webots se ha popularizado entre la comunidad científica para simplificar la simulación de técnicas de coordinación (Dasgupta, 2012), estrategias de exploración del ambiente (Scott y Changbin, 2009) y distribución de tareas (Sariel, Balch y Erdogan, 2007) en sistemas multi-agente. Fundamentado en la Programación Orientada a Objetos (POO), todo elemento que se agrega en Webots al entorno virtual es un nodo u objeto de una clase preprogramada. Haciendo uso de su capacidad de representación abstracta, una clase avanzada tal como “*robot*” hereda de múltiples clases inferiores tales como “*motor*” o “*sensor*”. Por ende, el desarrollador puede establecer la complejidad de sus algoritmos con base en los subsistemas que componen al robot real.

Al revisar el catálogo de robots (agentes) disponibles, se encontró que Webots no incluye los robots Bioloid en configuración humanoide. Tras evaluar

los modelos cinemáticos del Bioloid (Cerritos, Camarillo, Monsiváis, Castillo, Pérez y Pámanes, 2013) y el robot humanoide DARwIn-OP (Williams II, 2012) y determinar un alto grado de similitud, se opta por hacer uso del robot DARwIn-OP (disponible dentro de Webots) para representar a los Bioloid. Entre otras similitudes se destaca el mismo método de operación y control de sus actuadores, así como el mismo fabricante (Robotis). Se aprovecha una aplicación de DARwIn-OP preexistente en Webots desarrollada por los fabricantes, de la cual se extrae un entorno virtual prediseñado y configurado para dicho robot.

En el caso de las mesas, que son los objetos a recolectar por los robots, se desarrolla un modelo en CAD vía Solidworks que es importado posteriormente a Webots. La ventaja de dicho proceso radica en la generación automática de la geometría de frontera de la mesa, cálculo de la masa total y ubicación del centro de masa en el espacio tridimensional, necesarios para que el motor de física de Webots interprete correctamente las relaciones de contacto entre las mesas y los brazos de los robots.

IV. IMPLEMENTACIÓN EN SOFTWARE DE LOS AGENTES A PARTIR DE SU MODELO ABSTRACTO

Partiendo de la representación abstracta previamente definida en la Figura 1, se procede a identificar e implementar los elementos constituyentes del modelo DARwIn-OP presente en Webots. Para programar y operar los robots con base al paradigma de SMA, dichos elementos deben asociarse a los cuatro bloques de subsistemas sugeridos: movimiento, manipulación, comunicación y decisión.

V. BLOQUES DE MOVIMIENTO Y ACTUACIÓN

El bloque de movimiento es el conjunto de subsistemas más complejo en el robot real, puesto que involucra 12 servomotores presentes en las piernas del Bioloid. A fin de simplificar su operación, los fabricantes desarrollaron una tabla de secuencias para los servomotores, la cual contiene la totalidad de posiciones angulares y tiempos de ejecución que resultan en los patrones de caminata deseados (Ej.: marcha lateral). Webots posee un *Namespace* de administradores que permite mantener dicha facilidad de operación en la simulación: sus funciones se encargan internamente de manejar la ejecución

de la tabla de secuencias, garantizar el equilibrio del robot durante la marcha y regular los procesos de control con base en los ciclos máquina del simulador (la simulación en Webots se comporta como un sistema en tiempo discreto).

Por lo tanto, el uso del bloque de movimiento se reduce a la activación/desactivación de los administradores y el llamado de las secuencias necesarias para desplazar a los robots hacia las mesas. A diferencia del bloque de movimiento, el bloque de actuación corresponde a la operación de los brazos del robot para recoger y descargar las mesas. Tales movimientos se realizan en un solo paso y por tanto se omite la tabla de secuencias: los actuadores se comandan directamente, dado que los robots pueden alzar la mesa con tan solo cambiar una única vez la posición angular en los servomotores de sus hombros.

VI. BLOQUE DE COMUNICACIÓN

La implementación de este bloque se fundamenta en la inclusión de nodos emisores y receptores en cada agente del sistema; esta pareja es equivalente al módulo de comunicación inalámbrica ZigBee presente en el robot Bioloid real. Se define un estándar propio para los mensajes que los agentes intercambian, puesto que los microcontroladores de los robots Bioloid no poseen la capacidad computacional suficiente para manejar los estándares típicamente usados en sistemas multi-agente, tales como el protocolo de comunicación FIPA-ACL, ampliamente divulgado por Bellifemine, Caire y Greenwood (2007). Bajo el estándar propuesto, una trama compuesta por un entero de cuatro dígitos codifica la información a enviar, tal como se establece en la Figura 3.

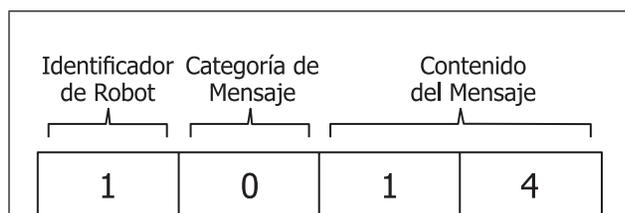


Figura 3. Trama de información según estándar propuesto. (Fuente: Autores.)

El primer dígito identifica cuál es el robot destinatario (1 a n , donde n es el número total de robots recolectores que sean desplegados), el segundo dígito denota el tipo de mensaje ('0' corresponde a un "comando de movimiento") y los dos últimos

dígitos equivalen al contenido del mensaje ('14' indica "girar en sentido antihorario").

VII. BLOQUE DE DECISIÓN

El bloque de decisión corresponde al algoritmo principal que permite a los agentes cumplir las tareas de exploración y recolección de mesas. Su interacción con los demás bloques se estructura en función de la tarea en ejecución y la distancia y orientación del robot respecto a la mesa de interés. Se escoge aplicar el paradigma de la Programación Dirigida por Eventos enunciado por Douglass (2013), bajo el cual el algoritmo ejecuta una serie de instrucciones según el evento correspondiente (Ej.: el evento que activa la tarea de recolección es el posicionamiento adecuado de los robots alrededor de la mesa). El algoritmo implementado para el componente de decisión se presenta en la Figura 4, donde se aprecian los eventos posibles en los bloques condicionales.

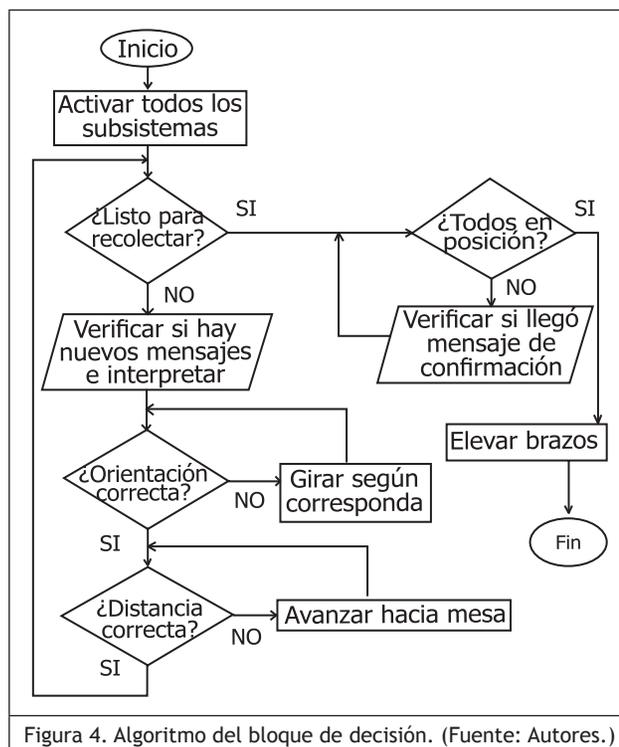


Figura 4. Algoritmo del bloque de decisión. (Fuente: Autores.)

VIII. EVALUACIÓN DE LA HIPÓTESIS DE RECOLECCIÓN

A fin de corroborar la hipótesis del número óptimo de agentes recolectores (tres agentes para un entorno de trabajo de 1.80 por 1.35 metros), se establece una situación de recolección estándar: al iniciar la simulación, todos los robots desplegados deben dirigirse a recolectar la misma mesa, la cual es

seleccionada de forma aleatoria entre tres mesas ubicadas en el entorno. La Figura 5 presenta un ejemplo del inicio de la simulación con cinco agentes (el máximo a evaluar).

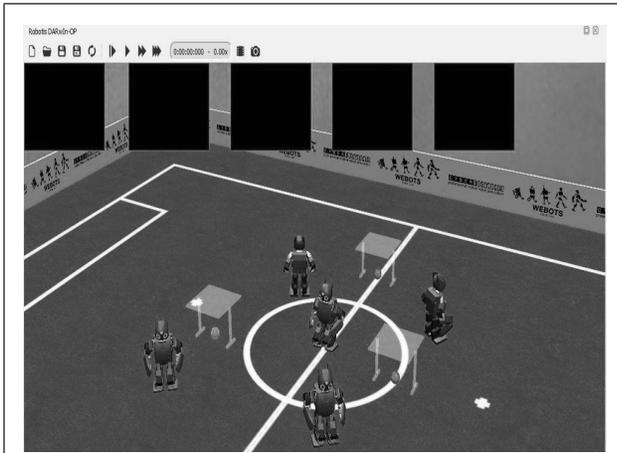


Figura 5. Ejemplo de la situación de recolección estándar con 5 agentes. (Fuente: Autores.)

La técnica de acercamiento consiste en tres etapas. En la primera, el robot rota sobre su eje hasta que la mesa se percibe en el centro de las imágenes obtenidas por su cámara. Acto seguido, inicia un desplazamiento en línea recta, el cual finaliza una vez la distancia percibida por el robot entre sí mismo y la mesa se encuentra en un rango de 0.15 a 0.25 metros.

La recolección de la mesa es considerada exitosa si al terminar se cumplen los siguientes criterios: (a) la mesa no es derribada, (b) ninguno de los agentes se ha caído y (c) ambos brazos de todos los agentes sostienen la mesa. Las pruebas fueron organizadas en cuatro etapas, iniciando con dos agentes recolectores y escalando hasta cinco agentes recolectores. Se realizaron cinco pruebas por cada etapa, para un total de 20 pruebas en la muestra. Los resultados se presentan en la Tabla 1, donde los criterios (a), (b) y (c) se marcan con (✓) si se cumplió o (-) si no se cumplió.

Tabla 1: Resultados para las pruebas de evaluación de la hipótesis según criterios (a), (b) y (c).

Prueba No.	Número de agentes											
	Dos			Tres			Cuatro			Cinco		
	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
1	-	-	-	-	✓	-	-	-	✓	-	-	-
2	✓	✓	-	-	✓	-	-	-	✓	✓	-	-
3	✓	✓	✓	✓	✓	-	-	-	-	-	✓	-
4	✓	✓	-	✓	✓	-	-	-	-	-	-	-
5	✓	-	-	✓	-	-	-	✓	✓	-	✓	-

Al analizar los resultados de la hipótesis formulada (tres agentes recolectores) en la Tabla 1, se encuentra que permite cumplir una tasa media de criterios (siete entre quince, 46.7%). Sin embargo, la mayor tasa de criterios cumplidos exitosamente (ocho entre quince, 53.3%) y la única recolección correcta ocurre cuando se usan solo dos agentes recolectores.

IX. CONCLUSIONES

Los resultados experimentales demuestran que el número óptimo de robots formulado en la hipótesis excede en uno al número óptimo de robots obtenido en las pruebas de validación. La tasa de recolecciones correctas indicó que el número óptimo de recolectores por mesa es tan solo dos; valor que a

su vez es la cantidad mínima de agentes requeridos para levantar dicho objeto del suelo. Aunque más agentes pueden ubicarse simultáneamente alrededor del perímetro de la mesa, su método de acercamiento no se refleja en una mayor tasa de recolecciones correctas.

El entorno de desarrollo Webots demostró ser una herramienta adecuada para simplificar la representación de los elementos de hardware y software que componen el sistema multi-agente (SMA), así como simular las tareas que el SMA realiza. Aunque su uso requiere un análisis minucioso de sus ejemplos de desarrollo; esto es compensado con la versatilidad que brinda para crear situaciones de simulación estándar y la rapidez al modificar los parámetros de operación, elementos vitales para la investigación aquí consignada.

Se recomienda evaluar el comportamiento del sistema multi-agente modificando el método de acercamiento de los robots a la mesa previo a la recolección. Los parámetros clave a analizar que pueden llevar a un aumento en la tasa de recolecciones exitosas son la técnica de detección de las mesas y la secuencia de correcciones de trayectoria (rotaciones) que los robots ejecutan para desplazarse hacia la mesa.

X. BIBLIOGRAFIA

1. Bellifemine, F., Caire, G. y Greenwood, D. (2007). *Developing multi-agent systems with JADE*. Reino Unido: Wiley.
2. Cerritos, J., Camarillo, K., Monsiváis, J., Castillo, G., Pérez, G. y Pámanes, J. (Agosto, 2013). Kinematic Modeling of a Humanoid Soccer-Player: Applied to BIOLOID Premium Type A Robot. *Intelligent Robotics Systems: Inspiring the NEXT Communications in Computer and Information Science*. (volumen 376): pp. 49-63.
3. Cheng, S., Raja, A. y Lesser, V. (Julio, 2013). Multiagent meta-level control for radar coordination. *Web Intelligence and Agent Systems: An International Journal*. (volumen 11), No. 2: pp. 81-105.
4. Dasgupta, P. (Mayo, 2012). *Multi-agent coordination techniques for multi-robot task allocation and multi-robot area coverage*. Trabajo presentado en el 2012 International Conference on Collaboration Technologies and Systems (CTS), Denver, United States of America.
5. Douglass, S. (Abril, 2013). *Autonomous Agents in Complex Event-Driven Software Systems*. Trabajo presentado en el 2013 20th IEEE International Conference and Workshops on the Engineering of Computer Based Systems (ECBS), Phoenix, United States of America.
6. Pérez, A. y Cruz, J. (2007). *Cooperación en sistemas multiagentes*. En: E. González y C. Bustacara, Desarrollo de aplicaciones basadas en sistemas multiagentes (pp. 19-30). Bogotá: Editorial Pontificia Universidad Javeriana.
7. Sariel, S., Balch, T. y Erdogan, N. (Octubre- Noviembre, 2007). *Incremental multi-robot task selection for resource constrained and interrelated tasks*. Trabajo presentado en el IROS 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, United States of America.
8. Scott, A. y Changbin, Y. (Febrero, 2009). *Cooperative multi-agent mapping and exploration in Webots®*. Trabajo presentado en el ICARA 2009 4th International Conference on Autonomous Robots and Agents, Wellington, Nueva Zelanda.
9. Shoham, Y. y Leyton-Brown, K. (2007). *Multiagent systems: algorithmic, game-theoretic, and logical foundations*. New York, United States of America: Cambridge University Press.
10. Williams II, R. (Agosto, 2012). *DARwin-OP Humanoid Robot Kinematics*. Trabajo presentado en el IDETC/CIE 2012 Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Chicago, United States of America.
11. Woolridge, M. (2009). *An introduction to multiagent systems*. 2nd edition. Glasgow, United Kingdom: Wiley.