

CONTROL DISTRIBUIDO APLICADO A PLATAFORMAS ROBÓTICAS CON  
ARQUITECTURAS DE MULTIPROCESADORES

HEYSON BÁEZ RAMOS

PONTIFICIA UNIVERSIDAD JAVERIANA  
FACULTAD DE INGENIERÍA ELECTRÓNICA  
MAESTRÍA EN INGENIERÍA ELECTRÓNICA

BOGOTÁ

2012

CONTROL DISTRIBUIDO APLICADO A PLATAFORMAS ROBÓTICAS CON  
ARQUITECTURAS DE MULTIPROCESADORES

HEYSON BÁEZ RAMOS

Tesis de Maestría

Director

Enrique González, Ph.D.

Profesor Tiempo Completo Pontificia Universidad Javeriana

PONTIFICIA UNIVERSIDAD JAVERIANA  
FACULTAD DE INGENIERÍA ELECTRÓNICA  
MAESTRÍA EN INGENIERÍA ELECTRÓNICA

BOGOTÁ

2012

## AGRADECIMIENTOS.

En primer lugar, a Dios que me dio sabiduría y fuerzas para seguir luchando por mis metas y por ayudar a todos los que me rodean.

Quiero agradecer a mi director de tesis, Enrique González por haberme dado la oportunidad de trabajar con él, por todo lo que aprendí a su lado, por su constante guía, por la paciencia que me tuvo, por los consejos oportunos y acertados, por la confianza depositada en mi, gracias y mil gracias.

Al ingeniero Carlos parra por ayudarme con sus conocimientos y visión de la investigación. A Juan Manuel Calderón y todo el grupo por abrirme las puertas de su amistad y apoyo.

A mis amigos por animarme a continuar luchando sin importar las dificultades presentes, por ser las grandes personas que son y haberme brindado su apoyo. Por último, a todas las personas que de una forma u otra me colaboraron.

Finalmente, mi más sincero agradecimiento a mis padres, Rosalino Báez y Gloria Ramos por todo el amor que siempre me han brindado, por su apoyo incondicional y sobre todo por su ejemplo de superación. Gracias por el amor, los valores y la educación que me han brindado, los quiero con todo mi corazón.

*A Dios y a mi Familia, muchas gracias.*

# CONTENIDO

<b>INDICE DE FIGURAS</b> .....	<b>6</b>
<b>INDICE DE TABLAS</b> .....	<b>7</b>
<b>1. INTRODUCCIÓN</b> .....	<b>8</b>
<b>2. MARCO CONCEPTUAL</b> .....	<b>11</b>
2.1 Motivación e Importancia de la Utilización de un Control Distribuido y Aplicado a Plataformas Robóticas _____	11
2.2 Estado del Arte del Control Distribuido _____	12
2.3 Términos Utilizados en el Desarrollo de este Trabajo _____	14
2.4 Potenciales Aplicaciones _____	15
<b>3. MODELO DE LA ARQUITECTURA GENERAL</b> .....	<b>17</b>
3.1 Plataforma Robótica _____	19
3.1.1 Estructura Física _____	19
3.1.2 Metas Generales _____	21
3.1.3 Unión de la Plataforma Robótica Entre la Arquitectura Física y las Metas _____	24
3.2 Multiprocesadores _____	24
3.3 Esquema De Interacción _____	26
3.4 Forma de Conexión de la Arquitectura Propuesta _____	27
3.5 Arquitectura de los Agentes _____	27
3.6 Arquitectura De Los Contenedores _____	28
3.7 Propagación _____	29
3.8 Atenuación _____	30
3.9 Conflictos _____	30
3.10 El Agente _____	31
3.10.1 Arquitectura _____	31
3.10.2 Características De Los Agentes _____	32
<b>4. MODELO</b> .....	<b>33</b>
4.1. Modelo de Propagación por Medio de Regiones Circulares _____	34
4.2. Modelo de Propagación por Medio de una Región Circular y la Distancia Mínima de Cercanía _____	39

<b>4.3. Modelo de Propagación por Medio de Regiones Circulares para la Plataforma AMARANTA.</b>	<b>42</b>
<b>4.4. Modelo de Propagación por Medio de Regiones Circulares con Distancia Mínima para la Plataforma AMARANTA.</b>	<b>44</b>
<b>5. IMPLICACIONES Y PRUEBAS</b>	<b>45</b>
<b>5.1 Propagación de Agentes por Medio de Regiones Circulares</b>	<b>45</b>
5.1.1 Brazo Planar de Dos Grados de Libertad	45
5.1.2 Brazo Planar de Tres Grados de Libertad	46
5.1.3 AMARANTA	47
<b>5.2 Propagación de Agentes por Medio de Regiones Circulares y la Mínima Distancia de Desplazamiento</b>	<b>48</b>
5.2.1 Brazo Planar de Dos Grados de Libertad	48
5.2.2 Brazo Planar de Tres Grados de Libertad	49
5.2.3 AMARANTA	50
<b>5.3 Implementación Del Modelo Final De Análisis</b>	<b>50</b>
5.3.1 Plataforma Robótica	51
5.3.2 Estructura Simulación	52
5.3.3 Análisis Comparativo Experimental con los Dos Métodos Propuestos	52
<b>6. CONCLUSIONES</b>	<b>59</b>
<b>BIBLIOGRAFÍA</b>	<b>62</b>

## INDICE DE FIGURAS

<i>Figura 1 Arquitectura Global Propuesta.....</i>	<i>18</i>
<i>Figura 2 Distribución Implementada en un Brazo Planar de Dos Grados de Libertad.....</i>	<i>21</i>
<i>Figura 3 Estructura de las Metas Generales, Metas Particulares y Agentes.....</i>	<i>23</i>
<i>Figura 4 Arquitectura de Multiprocesadores .....</i>	<i>25</i>
<i>Figura 5 Procesadores y su Relación en la Arquitectura Global.....</i>	<i>26</i>
<i>Figura 6 Arquitectura de Interconexión Distribuida de los Contenedores .....</i>	<i>29</i>
<i>Figura 7 Arquitectura Propuesta de los Agentes.....</i>	<i>31</i>
<i>Figura 8 Propagación de los Dos Agentes del Brazo Planar de Dos Grados de Libertad.....</i>	<i>33</i>
<i>Figura 9 Cálculo de la Posición Actual del Contenedor y Consecuentemente la Distancia con el Objetivo Particular.....</i>	<i>35</i>
<i>Figura 10 a). Posiciones Alcanzables por el Contenedor del Brazo b) Circunferencias en Función del Contenedor Brazo y de los Datos Propagados.....</i>	<i>36</i>
<i>Figura 11 Circunferencia y Meta Particular del Agente que Controla al Contenedor del Brazo .....</i>	<i>38</i>
<i>Figura 12 (a) El Agente del Brazo Realiza una Circunferencia del Brazo y (b) Determina su Propia Meta Particular en Función de la Distancia más Corta a la Circunferencia.....</i>	<i>41</i>
<i>Figura 13 Comportamiento de un Agente para Determinar la Posición Máxima Alcanzable para Obtener el Evento Motivador.....</i>	<i>43</i>
<i>Figura 14 Comportamiento de los Agentes por Propagaciones Determinando las Posiciones Máximas Alcanzables para cada Agente de la Plataforma.....</i>	<i>43</i>
<i>Figura 15 (a) Plataforma Simulada AMARANTA en Dos Dimensiones y (b) Recorrido de la Plataforma Simulada en MATLAB.....</i>	<i>44</i>
<i>Figura 16 Circunferencias y Puntos de Intersección en el Brazo Planar de Dos Grados de Libertad.....</i>	<i>46</i>
<i>Figura 17 Resultados en Matlab del Movimiento Resultante de la Propagación por Medio de Regiones Circulares y su Respectiva Simulación.....</i>	<i>47</i>
<i>Figura 18 Comportamiento de un Agente Para Determinar la Posición Máxima Alcanzable Para Obtener el Evento Motivador.....</i>	<i>47</i>
<i>Figura 19 Cálculo de las Distancias del Brazo y el Antebrazo.....</i>	<i>48</i>
<i>Figura 20 Obtención de la Meta General por Medio de una Cooperación entre Agentes .....</i>	<i>49</i>
<i>Figura 21 Resultados en Matlab del Movimiento Resultante de la Propagación por Parámetros de Distancia y su Respectiva Simulación .....</i>	<i>49</i>
<i>Figura 22 Plataforma Simulada AMARANTA en dos Dimensiones Punto Final (Color Verde) .....</i>	<i>50</i>
<i>Figura 23 Modelo Obtenido en Simmechaniks con Centros de Gravedad y Características Físicas de la Plataforma Robótica .....</i>	<i>51</i>
<i>Figura 24 Modelo Obtenido en Virtual-Real para la Interacción del Modelo en 3D.....</i>	<i>52</i>
<i>Figura 25 Comparación del Gasto de Energía entre el Método por Distancias Cortas y el Método por Circunferencias.....</i>	<i>54</i>
<i>Figura 26 Comparación del Sobreimpulso Máximo entre el Método por Distancias Cortas y el Método por Circunferencias.....</i>	<i>55</i>
<i>Figura 27 Comparación del Error de Estado Estacionario Entre el Método por Distancias Cortas y el Método por Circunferencias.....</i>	<i>56</i>
<i>Figura 28 Comportamiento de la Plataforma Variando el Punto Inicial y Dejando un Punto Fijo de Referencia .....</i>	<i>57</i>
<i>Figura 29 Comportamiento de la Plataforma Variando la Meta Final y Dejando un Punto Fijo de Inicio.....</i>	<i>58</i>

## INDICE DE TABLAS

<i>Tabla 1 Características Relevantes de los Sistemas Distribuidos.....</i>	<i>11</i>
<i>Tabla 2 Términos Utilizados y su Definicion .....</i>	<i>14</i>
<i>Tabla 3 Aplicaciones y Ejemplos del Control Distribuido .....</i>	<i>15</i>
<i>Tabla 4 Pruebas Obtenidas del Método de Distancias Cortas (Dificultad Baja y Media).....</i>	<i>53</i>
<i>Tabla 5 Pruebas Obtenidas Del Método de Distancias Cortas (Dificultad Alta y Media).....</i>	<i>53</i>
<i>Tabla 6 Pruebas Obtenidas del Método por Circunferencias (Dificultad Baja y Media).....</i>	<i>53</i>
<i>Tabla 7 Pruebas Obtenidas del Método por Circunferencias (Dificultad Alta y Media).....</i>	<i>54</i>
<i>Tabla 8 Promedio del Tiempo de Respuesta de Cada uno de los Dos Métodos.....</i>	<i>55</i>
<i>Tabla 9 Puntos Iniciales de la Plataforma .....</i>	<i>57</i>
<i>Tabla 10 Puntos Finales de la Plataforma.....</i>	<i>57</i>

## 1. INTRODUCCIÓN

A continuación se presenta una breve introducción del tema a desarrollar, además, se exponen algunas ventajas de desarrollar un modelo de control distribuido para plataformas robóticas con multiprocesamiento, junto a los objetivos propuestos necesarios para diseñar, implementar y validar un modelo de control distribuido para plataformas robóticas con multiprocesamiento y finalmente se describe la organización presente a esta tesis.

En la actualidad los sistemas robóticos se han perfeccionado de una forma exponencial, tanto para las distintas aplicaciones, como también para los requerimientos de la sociedad. Causando que los sistemas internos del robot requieran utilizar: una mayor cantidad de procesos, un aumento en la cantidad de sensores, mejores velocidades de comunicación y el perfeccionamiento de la manipulación de su entorno.

En consecuencia a lo anterior, se producen inconvenientes en los sistemas robóticos como son: las saturaciones en los procesos, las pérdidas de información de sus estados, las latencias en las comunicaciones y las respuestas de control desfasadas a los eventos en su medio, como ejemplos, se referencia a Petrina [PETR2010] y Jarvis [JARV2008]. Como si fuera poco, en estos sistemas robóticos se incrementó la utilización de múltiples procesadores o Control Distribuido en un mismo robot, generando mayor complejidad en la implementación y manipulación de los sistemas requeridos, esto se puede observar en Christian Schack [SCHA2010] y Butler [ZACK2003].

Para solucionar estos inconvenientes se requiere que exista un enfoque integrado y sistemático, que aborde simultáneamente los problemas de cooperación eficaz entre los sistemas integrados de control; Que conlleven a generar una arquitectura de multiprocesamiento, donde los procesadores implementados (en los robots) sean dedicados y controlen partes específicas del sistema robótico, pero en conjunto, estos procesadores controlen todo el sistema con un control distribuido. Por lo tanto, como objetivo general se plantea diseñar e implementar un modelo de control distribuido basado en sistemas Multi-agentes para plataformas robóticas con arquitectura de multiprocesadores y validarlo en una plataforma de simulación.

Inicialmente, para dar una idea base del pensamiento que se quiere recrear con esta investigación, Minsky [MINS1988], expone una teoría que relaciona



a la mente y los agentes<sup>1</sup>. La importancia de visualizar a una mente como una sociedad de agentes, es que los diferentes agentes pueden basarse en diferentes tipos de procesos, con diferentes propósitos, diferentes formas de representación del conocimiento y diferentes métodos para producir los resultados. ¿Qué truco de magia nos hace inteligentes? El truco, es que no hay truco. El poder de la inteligencia se deriva de nuestra gran diversidad, no de un principio único, perfecto<sup>2</sup> [MINS1988].

Con este principio de procesos individuales conocidos como agentes, se da una pauta para utilizar los múltiples procesamientos en un sistema robótico. Cada procesamiento, es un agente que funciona independientemente y en paralelo a los demás agente. Cada agente descentraliza las funciones y el control del sistema, con lo cual, como primer objetivo específico esta el diseño del modelo de control distribuido para múltiples procesadores soportado en la cooperación entre agentes racionales.

Cabe notar, que las ventajas de la implementación de los sistemas distribuidos con multiprocesamiento son: razones económicas, debido a que estos sistemas tienen en potencia una proporción precio-desempeño; Velocidad, debido a que puede tener mayor poder de cómputo; confiabilidad, debido a que al distribuir la carga de trabajo en muchos procesadores, la falla de un circuito descompondrá máximo a un procesador y el resto seguirá intacto; crecimiento por incrementos, es decir, si se necesita añadir poder de cómputo podrían añadirse más procesadores al sistema; flexibilidad, debido a se difunde la carga de tareas entre los procesadores disponibles en la forma más eficaz en cuanto a los costos<sup>3</sup>.

Además, la combinación de procesos e interacción de robots en múltiples entornos de trabajo, busca metodologías o pautas para que la interacción y la cooperación fuesen posibles, por lo tanto, como segundo objetivo específico es el implementar el modelo propuesto sobre una plataforma de simulación y probarlo en un prototipo básico de estudio. El desarrollo está basado en software, con sus múltiples simulaciones de las plataformas desarrolladas y su posterior validación. Comprobando que él método propuesto puede ser implementado en un trabajo futuro en una plataforma real que requiera obtener las múltiples ventajas de los sistemas distribuidos. Además, las arquitecturas de control requieren la distribución de la carga computacional entre muchos procesadores que deben funcionar en tiempo

---

<sup>1</sup> Se expone que la mente humana y cualquier otro sistema natural evolucionado, es una vasta sociedad de procesos individuales simples conocidos como agentes y en conjunto, son las entidades que proporcionan los pensamientos fundamentales que construyen a las mentes humanas.

<sup>2</sup> MINSKY, Marvin. The Society of Mind, Simon and Schuster, New York. Simon and Schuster, Nueva York. March 15, 1988. ISBN 0-671-65713-5 15-03 de 1988. Pg. 21.

<sup>3</sup> HILDA Castillo; Sistemas Operativos Distribuidos; MC BUAP, 1990. Pg. 2.

real<sup>4</sup>. Por lo tanto, como tercer objetivo particular se corrobora el modelo por medio de un prototipo simulado, para el robot AMARANTA [CARL2009], generando un análisis del diseño conceptual en las diferentes variables presentes en el robot AMARANTA [CARL2009]. Este objetivo verificará el método propuesto implementado en el robot AMARANTA, su funcionalidad y aplicabilidad a otras múltiples plataformas distribuidas.

Lo primero que se hizo fue recolectar información sobre las diferentes aplicaciones y estudios de control distribuido y aplicado en plataformas robóticas. Basado en estos datos, se propuso un modelo de brazo robótico de tres grados de libertad. Con este brazo robótico se observó el comportamiento obtenido con la implementación del método propuesto del sistema distribuido aplicando circunferencias propagadas entre los agentes de los multiprocesadores de la plataforma. Además, se caracterizó y analizó el comportamiento obtenido con el método de propagación propuesto. En donde, como último objetivo se evalúa el modelo con algún tipo de coordinación de movimiento entre las patas del robot AMARANTA [CARL2009], y observar la factibilidad de realizar estas tareas y lograr al menos realizar un tipo de tarea propuesta. Finalmente se realiza un primer acercamiento a la implementación por medio del prototipo simulado junto a su animación.

El presente trabajo se encuentra organizado en seis capítulos. En el Capítulo 2, se presenta la definición y una descripción detallada de los elementos relacionados que definen al método implementado. Se hace especial énfasis en la aplicación de sistemas multiagentes para su implementación en el control distribuido. Además, en el capítulo 3, se presenta el modelo de la arquitectura propuesta para el control distribuido de una plataforma con multiprocesadores. A continuación, en el capítulo 4 se presentará el modelo propuesto por medio de circunferencias y por circunferencias con la distancia más corta. En el Capítulo 5, se presenta las implicaciones y pruebas obtenidas. En el Capítulo 6, se presentan las conclusiones del trabajo. Por último, en el capítulo 9, se enuncia la bibliografía.

---

<sup>4</sup> Arthur, C., SANDERSON. Dynamic Analysis and Distributed Control of the Tetrobot Modular Reconfigurable Robotic System, Autonomous Robots, Department Of Mechanical Engineering, Aeronautical Engineering, Rensselaer Polytechnic Institute, Troy, 2001. Pg. 7.

## 2. MARCO CONCEPTUAL

En este capítulo se presentan los fundamentos teóricos y experimentales necesarios para el modelamiento, implementación y caracterización del método de control distribuido utilizando multiprocesamiento. En la sección 2.1 se presenta la motivación e importancia de la utilización de un control distribuido aplicado a plataformas robóticas, en la sección 2.2 se presenta la definición y el estado del arte del control distribuido, posteriormente, en la sección 2.3, se describen los términos utilizados en este trabajo para determinar las principales características del proyecto y aplicarlo al método de control distribuido. Finalmente, en la sección 2.4, se hace un resumen de sus potenciales aplicaciones.

### 2.1 Motivación e Importancia de la Utilización de un Control Distribuido y Aplicado a Plataformas Robóticas

Una de las características más importantes en el mundo contemporáneo es el multiprocesamiento en los sistemas de todo tipo<sup>5</sup>. Entre estos multiprocesamientos se encuentran las múltiples tareas y procesos que una plataforma debe realizar ante su entorno de trabajo. Por lo tanto, una de las problemáticas relevantes en las plataformas con múltiples tareas es el aumento de procesamientos internos, retardos, respuestas tardías, conflictos, etc. Con lo anterior, la motivación para solventar la implementación de múltiples tareas en las plataformas robóticas es desarrollar, implementar y aplicar un método propuesto de control distribuido aplicando agentes e intercomunicados por medio de propagación entre los vecinos (agentes) directos que lo requieran. Además, se fundamenta en la distribución de tareas sin requerir un análisis global de la cinemática de la plataforma, además, los análisis de la plataforma deben ser tareas locales en cada uno de los agentes de la plataforma en cuestión. Según Coulouris las características relevantes de los sistemas distribuidos se exponen en la siguiente tabla 1.

Tabla 1 Características Relevantes de los Sistemas Distribuidos

<b>Compartir Recursos</b>	La plataforma de manera útil comparte entre cada una de las partes del sistema distribuido información y recursos. Estos procesos son dedicados y están físicamente encapsulados en uno de los procesadores. Específicamente el procesador que requiera los recursos o la información que tiene otro de los procesadores en un sistema distribuido sólo son enviados y utilizados por el procesador que genero la
---------------------------	---

<sup>5</sup> Tomado de César Hervás. Tema 1. Introducción A Los Sistemas Distribuidos. Ampliación Sistemas Operativos. Curso 2004/2005 [CESA2004].

	petición limitando el procesamiento de otros procesadores del sistema en tareas que no le son de su interés.
<b>Redundancia</b>	Una arquitectura con un sistema distribuido tiene redundancia, debido, a que si se presenta un fallo en una parte del sistema las demás partes siguen funcionando de manera correcta y funcional. Además, con la redundancia si se presentan fallos en algún procesador del sistema robótico, otro agente de un procesador que se encuentra funcionando puede tomar el control o dominio de la parte afectada para lograr su funcionamiento.
<b>Escalabilidad</b>	La plataforma puede ser escalable si se presentan nuevos requerimientos de recursos o de tareas especializadas. Al generar estos requerimientos se pueden asignar otros procesadores al sistema robótico, generando que los procesadores ya existentes no se modifiquen y el procesador asignado para la nueva tarea sea especializado. Además, la escalabilidad en los sistemas distribuidos también es llamada sistema abierto <sup>6</sup> debido a que el sistema puede ser extendido de diversas maneras en hardware (añadir periféricos, memoria o interfaces de comunicación, etc.).
<b>Tomas de decisiones locales</b>	Los sistemas distribuidos tienen la facultad de que si se presenta un problema en una de las áreas de desarrollo, una parte del procesamiento que tiene el control de esa zona toma las decisiones independientes para luego ejecutarlas. Esto con el fin, de evitar la sobresaturación y latencias de información que se presenta en un procesador central si es dedicado a la toma de decisiones. Además, se produce la independencia y autonomía entre cada uno de los agentes junto a sus propias acciones para cumplir con las tareas del sistema robótico <sup>7</sup> .
<b>Distribución de tareas</b>	En los sistemas distribuidos se presenta normalmente la distribución de tareas. En donde, una tarea general de la plataforma robótica pueda segmentarse en tareas más sencillas a desarrollar en cada uno de los procesadores dedicados [FERN2006]. Además, estas tareas segmentadas son realizadas independientemente por cada procesador, lo cual, genera una acción más ágil y con menor tiempo de respuesta.
<b>Bajo ancho de banda</b>	Debido a que cada procesador puede realizar múltiples tareas locales, la comunicación entre procesadores es mínima y por lo tanto, el ancho de banda es inferior a un ancho de banda requerido por sistemas centralizados. Además, son mínimas las pérdidas o saturación de información.
<b>Velocidad de respuesta</b>	En un sistema distribuido la velocidad de respuesta es mayor sin importar la cantidad de tareas se puedan presentar. Debido, a que cada procesador es independiente y realiza sus tareas sin esperar respuestas o permisos de otra parte de la plataforma.

## 2.2 Estado del Arte del Control Distribuido

"Un sistema distribuido es una colección de ordenadores independientes que se muestra a sus usuarios como un único sistema coherente<sup>8</sup>". En donde, el término de sistema distribuido ha venido utilizándose para denominar indistintamente a diferentes clases de sistemas informáticos y de procesamiento, en los que la potencia de tratamiento de la información se encuentra repartida en el espacio y que por diferentes razones han aparecido como una alternativa a aquellos otros sistemas, en los que la potencia de tratamiento de la información se encuentra concentrada en un único elemento, procesador o computadora<sup>9</sup>.

<sup>6</sup> Tomado de FALGUERAS, Benet Campderrich, página 261

<sup>7</sup> Tomado de <http://www.augcyl.org/?q=glol-intro-sistemas-distribuidos>.

<sup>8</sup> Tomado de Alberto Lafuente Mikel Larrea, Introducción a los sistemas distribuidos Dpto. ATC, UPV/EHU Tanenbaum.

<sup>9</sup> Tomado de arquitectura de los sistemas distribuidos pág. 19, [ALAB1992].

El control distribuido tiene múltiples aplicaciones y desarrollos en diferentes áreas del conocimiento. Lo cual, comenzó fundamentado en la implementación de sistemas de software requeridos en las aplicaciones computarizadas y el internet. En el área de la robótica se desarrollo para solventar las necesidades de las múltiples tareas y la descentralización de las plataformas, fundamentado en la teoría de [MINS1988], de la bioinspiración y el emular a un cerebro humano.

En donde, algunas publicaciones que podemos nombrar sobre el tema de estudio se encuentran Christian Ott [CHRI2008], que describe el control de un robots por medio de subsistemas. Otra propuesta, es la que presenta Zachary Kuliz, [ZACH2008], donde habla de un Marco de Control Distribuido (DCF) que proporciona amplio apoyo a los sistemas robóticos, junto a una arquitectura de coordinación y gestión. Además aporta, un modelo de simulación que da soporte al hardware y un alto nivel de independencia de la plataforma.

Otros escrito definen varios métodos para realizar una distribución de tareas y la respectiva coordinación entre plataformas. Por lo tanto, un método que es de suma importancia y se utilizará en este trabajo es del método de propagación, en donde, podemos nombrar a Kyoung-Hwan [KYOU2008], que presenta un nuevo método de localización cooperativo basado en propagación de restricciones. En donde, los múltiples robots móviles comparten información de posición con los demás. Pero no soluciona la problemática de sincronismos requeridos para el buen funcionamiento de la plataforma robótica

En el caso de los convencionales métodos probabilísticos, tales como EKF, dan la estimación más probable en lugar de garantizar los valores deterministas o límites para la posición real de los robots. Este método PC (control probabilístico) da un valor limite que contiene la posición real de la plataforma y de la localización de un conjunto de robots con ruedas<sup>10</sup>. La fusión de los datos y la aplicación de CP pero con problemas de asincronismo.

Finalmente en el área de la propagación podemos nombrar a María Jiménez, [MARI2006], que presenta el estudio sobre la propagación para movimientos simples y la coordinación de Multi-robots modulares basado en mecanismos de oscilación para movimientos simples y de evasión colectivo. Estos mecanismos de oscilación dependen de las circunstancias del entorno y de

---

<sup>10</sup> Trabajo presentado y soportado por the Dual-Use Technology Program of DAPA/DUTC and MIC/IITA [06-DU-LC-01, Development of Multi-Purpose Dog-Horse Robot based on the Network. K. Jo is with the BK21 Mechatronics Group at Chungnam National University, Daejeon, Korea (e-mail: neoview@ cnu.ac.kr).

los conflictos presentes. Obteniendo, resultados satisfactorios y validados utilizando Multi-robot Mini-trans.

Finalmente, se puede afirmar con los anteriores artículos y los demás artículos expuestos en el anteproyecto: que faltan estudios sobre la implementación de un control distribuido con multiprocesadores para plataformas robóticas. Es decir, la falta de un método o arquitectura que se debe seguir para implementar correctamente un control distribuido. Además, se tiene la falta de modelos de propagación como método aplicable a plataformas robóticas distribuidas. Y finalmente, el control distribuido presenta problemáticas reales en la sincronización de los diferentes procesadores y la segmentación de metas a desarrollar, entre otras.

Por lo anterior, en este trabajo se desea realizar un modelo de la arquitectura general para múltiples plataformas, además, se desea realizar un modelo aplicando propagación con multiagentes para la interacción de los múltiples procesadores de una forma autónoma y funcional. Por otra parte, se quiere presentar una solución para solventar la problemática del asincronismo, como es la atenuación en la plataforma robótica (expuesta en el documento) y finalmente presentar por medio del modelo propuesto una forma de segmentar las metas generales en metas particulares de fácil desarrollo en el procesamiento de la plataforma robótica.

### 2.3 Términos Utilizados en el Desarrollo de este Trabajo

Para la implementación y desarrollo de esta Tesis se toma como base la utilización de los sistemas multiagentes en cada uno de los procesos y desarrollos del control distribuido. Por lo tanto, se presentará algunos términos junto a su definición utilizada en este documento tabla 2:

Tabla 2 Términos Utilizados y su Definición

<b>Agente:</b> Un agente es un sistema físico o virtual (software), capaz de elegir autónomamente una acción más adecuada para cumplir sus tareas, Además, un agente es la unidad abstracta racional fundamental de los sistemas multiagentes. Siendo el ente que obtiene información del entorno que lo rodea por medio de entradas o sensores para producir cambios externos en su ambiente [DAVI2003]. Finalmente, la distribución de los agentes se puede encontrar en un mismo medio de procesamiento (procesador) o en múltiples medios de procesamiento (multiprocesadores).
<b>Comportamientos:</b> Los agentes están compuestos por un conjunto de comportamientos concurrentes <sup>11</sup> que pueden ser idénticos o diferentes entre cada uno de los agentes. Estos comportamientos dependen de las características de las tareas a resolver y de las limitantes que pueda tener el mismo agente en cuestión. Además, se debe tener en cuenta que todos los agentes tienen un mismo nivel de jerárquico entre los agentes.
<b>Evento motivador:</b> Los eventos se definen como acciones externas al robot que desencadenan la

<sup>11</sup> González, Enrique; Ávila, Jamir; "BESA" Bogotá Colombia. 2003.

ejecución de los comportamientos de cada agente. Estos agentes perciben a los eventos y realizan las acciones acordes o coherentes que son pertinentes por el evento generado. Estos eventos, son fundamentales para la realización de las metas particulares y generales y de las acciones de comunicación y propagación a los demás agentes vecinos.
<b>Conflictos:</b> Los conflictos son las limitantes por la utilización de los recursos limitados, tanto de la plataforma como de su área de trabajo.
<b>Metas generales:</b> Las metas generales son las tareas relevantes de la plataforma. Es decir, la tarea totalitaria que se presenta en la plataforma robótica en general.
<b>Metas particulares o locales:</b> Las metas particulares son particiones específicas de la meta general; estas particiones dependen de la cantidad de agentes y de las sub-tareas que pueden realizar un agente. La meta general es descompuesta en metas particulares pero la resolución de las metas particulares cumple con todos los requerimientos presentes en la meta general.
<b>Sistema Multi-tarea:</b> Un sistema multitarea permite que varios procesos sean ejecutados en forma paralela <sup>12</sup> ; realizando estas tareas en uno o más procesadores. Además, al realizar las multitareas permite que se puedan desarrollar estas tareas simultáneamente en tiempo real.
<b>Propagación:</b> Para resolver las problemáticas de interacción entre agentes del mismo nivel se tomo como método los propagadores de información básica entre los agentes vecinos directos. La propagación entre agentes busca determinar los datos más apropiados para que el agente que los reciba pueda desarrollar su acciones e informe a otros agentes datos relevantes todo con el fin de cumplir con las acciones y tareas que requieren cooperación entre agentes.
<b>Atenuación o influencia:</b> La atenuación es la cual se proporciona una variable existente entre el agente y sus metas. Además, la atenuación es una variable numérica o lumínica que proporciona un valor de información obtenida en el entorno de trabajo y requerida en la propagación que envía cada uno de los agentes del sistema robótica para que los demás agentes puedan realizar las correspondientes acciones para cumplir con sus propias metas.
<b>Sistema Multi-agentes (SMA):</b> Un sistema multiagentes es un sistema computacional con características autónomas, capaz de ejecutar diferentes tareas locales dentro de un entorno determinado. Las acciones que realiza el sistema multiagentes están definidas por un conjunto de instrucciones que pueden ser ejecutadas o no [DAVI2005]. Los sistemas multiagentes son arquitecturas de software que se interconectan entre los demás agentes del sistema. Finalmente, estos agentes también realizan múltiples acciones locales y la interacción de otros agentes cercanos a su medio para lograr las tareas que se requieran de apoyo.
<b>Control Distribuido:</b> El control distribuido es el método propuesto para la interacción de múltiples agentes y contenedores para que una plataforma robótica pueda realizar todas sus acciones y tareas requeridas.
<b>Punto de Referencia o Punto Atractor:</b> El punto atractor es un punto de referencia que la plataforma debe buscar y es atraído hacia este punto de referencia. Cabe notar, que un punto atractor puede ser una meta general que la plataforma debe alcanzar.

## 2.4 Potenciales Aplicaciones

Las aplicaciones del método de control distribuido son bastas en múltiples ciencias del conocimiento, pero en este documento se presentan las potenciales aplicaciones del método desarrollado en esta tesis específicamente en la robótica y sus áreas afines, estas aplicaciones se exponen en la siguiente tabla 3.

Tabla 3 Aplicaciones y Ejemplos del Control Distribuido

Áreas de Aplicación	Descripción de la Aplicación	Autores ejemplo	Descripción del Ejemplo
Robótica Móvil	En la robótica móvil puede ser solventada por medio de multiprocesadores para	Arndt Lüder [ARND2007]	Aplica un control distribuido a la robótica Humanoide;
		Gordon Wyeth	Habla del desplazamiento de los pies

<sup>12</sup>Tomado de Magin Florez. Besa micro-edition, arquitectura de sistemas multiagentes para microcontroladores.

	realizar múltiples tareas simultáneamente.	[GORD2004]	de una plataforma bípeda con control distribuido.
		Thomas [TIGL1994]	Expone sobre los sistemas Multiagentes y habla del estudio de la interacción entre robots para realizar múltiples tareas globales.
<b>Robótica fija</b>	Este método de control distribuido, se puede aplicar según los requerimientos de tareas de la plataforma.	Stephen Derby y David Brown [STEP2003]	Presentan una propuesta para un robot fijo basado en un control distribuido para un circuito de pistas (track) cerrado.
<b>Robótica Móvil de Enjambre</b>	Se requiere tener un nuevo enfoque para la coordinación de un gran número de robots relativamente simples. Este enfoque es bioinspirado en los animales de enjambre como muchos insectos sociales. Se caracterizan por que generan un funcionamiento distribuido, robusto, flexible y escalable de los robots en desarrollo.	Mohd Ahmad [MOHD2002] y Anders Lyhne Christensen [ANDE2008]	Expone un sistema Multi-robot y de sus posibles auto-ensamblajes o conexiones físicas entre los robots individuales,
		Manish Kumarse [MANI2010] y Andrew Howard [ANDR2006]	Presentan unos experimentos llevados a cabo como parte del programa de SDR DARPA <sup>13</sup> (Software de Robótica Distribuido DARPA). En donde, el principal reto para este programa es desarrollar un sistema capaz de desplegar 80 robots en un edificio sin explorar.
		Levent Bayindir [BAYI2007] y Terry Huntsberger [TERR2004]	Presenta la posibilidad de implementar un control distribuido tanto de Multi-robots como de unidades de robots que puedan aportar un nuevo enfoque a las investigaciones de la NASA <sup>14</sup> .
<b>Robótica Reconfigurable</b>	En los Robots Reconfigurables se requiere de plataformas que puedan adaptarse a diferentes tareas o ambientes de reorganización con sus configuraciones mecánicas posibles.	Arthur C. Sanderson [ARTH2001]	Presenta un sistema robótico reconfigurable que puede adaptarse a diferentes tareas o ambientes de reorganización.
		Esben Hallundbrek [ESBE2003]	Describe una serie de experimentos encaminados a explorar el uso artificial de elementos para la evolución automática.
		Kazuya Suzuki [KAZU2009]	Expone que la naturaleza debe ser auto-ensamblada en un sistema robótico para las necesidades actuales.
<b>Simuladores</b>	Un elemento que se debe estudiar es la simulación de sistemas robóticos. Que tiene como finalidad, desarrollar estudios más precisos y representaciones cercanas de los comportamientos que se pueden presentar en el entorno de los sistemas robóticos reales.	Cao Qixi [QIXI2007]	Describe un Sistema De Simulación De Doble Brazo En Un Robot Móvil;
		con Myriam Arias Ruiz [MYRI2008]	Presenta la arquitectura y comportamiento de los Agentes para fútbol robótico
		Shu Da [WANG2009]	Presenta un sistema Multi-agente de simulación basado en el algoritmo de aprendizaje por refuerzo.

<sup>13</sup> Tomado de: <http://www.darpa.mil/>

<sup>14</sup> Tomado de: <http://www.nasa.gov/>



### 3. MODELO DE LA ARQUITECTURA GENERAL

En este capítulo, se presenta un modelo propuesto propio general, basado en múltiples estudios, análisis y desarrollos realizados en esta investigación. Teniendo como fin, que el modelo de control distribuido con múltiples procesadores y multitareas pueda ser implementado funcionalmente en diferentes plataformas robóticas.

Con lo anterior, inicialmente se realiza una analogía y estudio de tres tipos diferentes de plataformas robóticas para realizar el método. Para este análisis, se estudian tres casos de plataformas robóticas que son los robots ratones, los robots bípedos y los robots de múltiples patas. Estos casos de análisis están fundamentados en tres artículos [TERR2004], [GORD2004] y [ANTH2002], que se comparan para integrar bajo una sola visión los elementos analizados en los diferentes casos de estudio. Finalmente, se realiza una analogía con miras a identificar un conjunto de elementos comunes y la interacción entre ellos. Con el fin, de obtener una arquitectura general propuesta basada en agentes para el control de múltiples sistemas robóticos con multiprocesadores (Ver ANEXO 1.).

A continuación, se presentará un modelo global del sistema propuesto y que será explicado gradualmente en el transcurso de este texto. Esta descripción inicial dará una breve introducción a los componentes propuestos en el modelo para el desarrollo e implementación de un control distribuido para plataformas robóticas. Con el fin, de mostrar la relevancia del modelo en la implementación de los sistemas con multiprocesadores y multiagentes.

#### *Descripción del Modelo*

En función al modelo propuesto, la arquitectura del modelo global Fig. 1, tiene como eje principal a la plataforma robótica. Esta plataforma robótica se compone por dos conjuntos importantes, que son: las metas generales y la estructura física del robot.

Las metas generales se componen de una o varias metas particulares. Cada meta particular se realiza gradualmente en búsqueda de la obtención de las metas generales del sistema robótico. Estas metas particulares están directamente relacionadas con los agentes, en donde, los agentes tienen como función desarrollar las acciones requeridas para cumplir con las metas particulares. Finalmente, la distribución de los agentes se puede encontrar en

un mismo medio de procesamiento (procesador) o en múltiples medios de procesamiento (multiprocesadores).

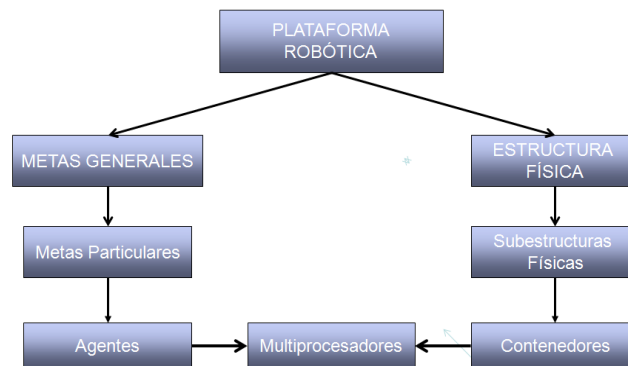


Figura 1 Arquitectura Global Propuesta

Además, en la arquitectura general como en una arquitectura específica los agentes se identifican por las metas que realizan, en donde, si las metas son idénticas un agente puede ser duplicado (reusable) para utilizarlo en varias zonas estructurales de una misma plataforma robótica.

Cabe notar, que los agentes propuestos son homogéneos en su arquitectura, es decir, se pretende desarrollar un agente básico general, que tiene como finalidad, facilitar su implementación en múltiples arquitecturas robóticas. De igual forma, los agentes implementados pueden ser agentes heterogéneos, siempre y cuando su heterogeneidad se fundamente en sus metas y funciones. La homogeneidad y la heterogeneidad tienen como finalidad que el agente sea funcional, fácil de implementar y que cumpla con las metas sin importar la arquitectura del robot.

Por otra parte, la estructura física se compone de múltiples subestructuras que tienen características o funciones particulares. Las subestructuras están directamente ligadas a los contenedores. Con lo cual, un contenedor realiza la unión entre una subestructura y unas metas particulares que están relacionadas entre sí. Además, los contenedores pueden distribuirse en un mismo medio de procesamiento (procesador) o en múltiples medios de procesamiento (multiprocesadores).

Por lo tanto, los contenedores interaccionan con uno o varios agentes. En donde, cada agente realiza unas acciones de control de una subestructura física que está ligada a la meta particular, teniendo como fin, que cada agente cumpla con las metas particulares para la obtención de la meta general del robot.

Finalmente, la estructura física y los agentes con sus metas particulares del sistema robótico tienen una aproximación modular, esto quiere decir, que los

contenedores y los agentes se distribuyen modularmente en función de los requerimientos, tanto de las metas particulares, como de la manipulación de las subestructuras físicas del robot. A continuación, se presentará la explicación de cada una de las partes del modelo propuesto, con la finalidad de comprender mejor la arquitectural que se desea desarrollar.

### 3.1 Plataforma Robótica

La plataforma Robótica es el conjunto de elementos eléctricos, mecánicos, de sensado, electromecánicos, como también, elementos que generan las acciones, comportamientos, funciones y/o metas del sistema robótico. Por lo tanto, se propone en este estudio que la plataforma robótica este compuesta por dos conjuntos particulares para la arquitectura global. Estos dos conjuntos particulares son: la estructura física del robot y las metas generales del sistema.

#### 3.1.1 Estructura Física

La estructura física se compone por múltiples elementos que realizan las acciones mecánicas, eléctricas y sensoriales requeridas para el funcionamiento del robot. Entre los elementos que componen la estructura física de un sistema robótico están: los sistemas eléctricos, los sistemas mecánicos, los sistemas sensoriales y los sistemas electromecánicos, entre otros, que en su conjunto son la plataforma robótica (cuerpo, brazos o patas). Además, la estructura física es fundamental en una plataforma robótica para conocer las limitantes físicas, los comportamientos y las posibles metas que se pueden desarrollar.

Cabe notar, que la estructura física tiene una estrecha relación con las metas generales y particulares a desarrollar por el sistema robótico. Debido, a que estas metas se desarrollan por medio de la estructura física. Como también, la estructura física en base a los sistemas sensoriales aporta información del ambiente externo, tanto de sus movimientos como de la interacción con su medio.

Finalmente, el modelo propuesto está orientado a una implementación o aproximación modular. Esto quiere decir, que la estructura física se puede descomponer en múltiples subestructuras particulares, con el fin, de simplificar y facilitar el funcionamiento de la plataforma robótica. Además, esta simplificación en subestructuras agrupadas por medio de contenedores generan una distribución de metas y consecuentemente, requieren de una implementación de un control distribuido.

✓ Subestructura Física

Las subestructuras son la descomposición de la estructura física general. En donde, las subestructuras se definen como las particiones simples de la estructura física, con el fin, de estructurar un modelo de control distribuido que pueda desarrollar múltiples metas particulares al mismo tiempo. Estas subestructuras particulares agrupan elementos mecánicos, eléctricos y sensoriales de una parte del robot. Asimismo, las subestructuras están estrechamente ligadas con los contenedores. En donde, estos contenedores agrupan a una o a más subestructuras según la distribución implementada.

Igualmente, estas subestructuras tienen como característica que la cantidad de elementos físicos de una subestructura es menor a la cantidad de elementos de la estructura física en su totalidad y en donde las acciones realizadas tienen menor complejidad. Las subestructuras están orientadas a una implementación modular. Esto quiere decir, que las subestructuras se pueden distribuir en el sistema robótico en función de los requerimientos, tanto de las metas generales, como de la complejidad de la estructura física del robot.

Por lo demás, la distribución de las subestructuras no se asigna de cualquier forma. Estas subestructuras se distribuyen en relación a las metas particulares que interactúan con los procesos y acciones que la plataforma robótica debe realizar.

❖ Contenedor

Los contenedores tienen como función agrupar una o más subestructuras específicas con el fin de controlar una parte de la estructura global. En donde, el agrupamiento de las subestructuras físicas está relacionado con las metas particulares que se requiere en la plataforma robótica. Por lo tanto, un contenedor se encuentra orientado a la estructura física pero estrechamente ligado a las metas particulares requeridas por la plataforma.

Asimismo, un contenedor es indivisible en el medio físico pero tiene una estructura fija que agrupa, controla y relaciona las subestructuras que contiene. Por lo demás, el contenedor ordena y produce una interacción entre las subestructuras y las metas particulares directamente asociadas, con el fin, de poder realizar las metas particulares.

Por lo tanto, internamente en un procesador podrán existir uno o múltiples contenedores. Estos contenedores serán entendidos como el

medio de interacción local de los agentes (metas particulares) y de las subestructuras físicas directamente relacionadas.

Además, en un contenedor puede existir una interacción entre uno o más agentes, que serán encargados de unas metas particulares específicas que manipulen a las subestructuras agrupadas en el contenedor en cuestión.

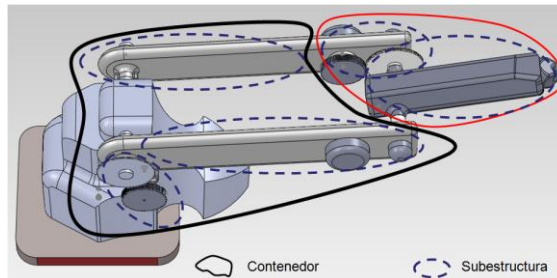


Figura 2 Distribución Implementada en un Brazo Planar de Dos Grados de Libertad

En conclusión, la plataforma robótica tiene como uno de sus ejes fundamentales a la estructura física. Esta estructura física se descompone por subestructuras particulares. Además, las subestructuras se controlan y están estrechamente ligadas por los contenedores, como ejemplo, se expone la distribución propuesta en un brazo planar en la Fig. 2.

### 3.1.2 Metas Generales

Las metas generales se definen como las metas más importantes y complejas que el sistema robótico debe realizar. En donde, las metas generales se fundamentan en las características físicas o funcionales, en las metas requeridas y en la elevada complejidad de los procesos o acciones realizados por el sistema robótico.

Además, las metas generales también se pueden definir como el conjunto de acciones u objetivos particulares, en su búsqueda de cumplir con las metas asignadas para el sistema robótico. Por lo tanto, estas metas generales se pueden descomponer en metas particulares.

#### ✓ Metas Particulares

Las metas particulares son la descomposición de una meta general. En donde, las metas particulares se definen como las particiones simples de las metas generales, con el fin de simplificar y facilitar el cumplimiento de la meta general. Además, estas metas particulares se caracterizan por que los procesos a realizar son de mínima complejidad y que las acciones realizadas se desarrollan por una parte específica del sistema

robótico. Es decir, si las metas particulares o simples gradualmente se pueden cumplir, esto conllevará a obtener la meta general requerida para el sistema robótico.

Por otra parte, las metas particulares están orientadas a una implementación modular. Esto quiere decir, que las metas particulares se distribuyen en el sistema robótico en función de los requerimientos, tanto de las metas generales, como de la manipulación de la estructura física del robot.

En conclusión, el sistema robótico al desarrollar las metas particulares simples puede cumplir en su totalidad con la meta general compleja. Estas metas particulares tienen un nivel de procesamiento mínimo y tienen pocas acciones a desarrollar en proporción a las acciones que se desarrollan de la meta general. Además, la distribución de las metas particulares no se asigna de cualquier forma, estas metas particulares se distribuyen en los agentes que interactúan directamente con los contenedores.

#### ❖ Agentes

Inicialmente se debe definir el concepto de agente, en donde el agente es un sistema físico o virtual (software), capaz de elegir autónomamente e independientemente la acción más adecuada para cumplir sus metas, según, la percepción de su entorno de trabajo, los recursos existentes y sus habilidades. Estos agentes tienen como función agrupar una o más metas particulares para la selección y el control de las metas a realizar. Este agrupamiento de las metas particulares está directamente relacionado con los contenedores y las subestructuras de la plataforma robótica. Por lo tanto, un agente se encuentra orientado a la estructura lógica (metas particulares).

Conjuntamente, el agente está estrechamente ligado a los contenedores que tienen el control de las subestructuras físicas, con el fin, de relacionar las metas particulares con las subestructuras físicas directamente conectadas. Asimismo, los agentes serán encargados de unas metas particulares específicas que manipulen a las subestructuras agrupadas en un contenedor.

Por otra parte, la relación entre Multi-agentes se basa en la cantidad de agentes que cooperan entre sí para obtener una meta particular. Teniendo en cuenta, que la meta particular obtenida es requerida para llegar a una meta general, cumpliendo con las necesidades globales del sistema robótico.

Finalmente, la plataforma robótica tiene como uno de los ejes fundamentales a las metas generales Fig. 3. Estas metas generales por su elevada complejidad se descomponen en metas particulares, facilitando y simplificando el desarrollo de las metas generales requeridas. Además, las metas particulares están directamente relacionadas con múltiples agentes que desarrollan acciones para el cumplimiento de las metas particulares. Esta descomposición de las metas generales produce un control distribuido de las acciones y de las metas en la plataforma robótica.

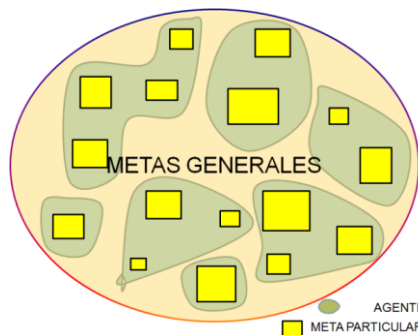


Figura 3 Estructura de las Metas Generales, Metas Particulares y Agentes

Los eventos y la percepción de los eventos requieren de unas características fundamentales para la funcionalidad y desarrollo de la plataforma a implementar en el modelo propuesto. Siendo los eventos y la percepción fundamentales para la obtención de las metas generales propuestas.

## EVENTOS MOTIVADORES

Los eventos motivadores son acciones ocurridas externas al robot que generan una reacción o meta general acorde o coherente del sistema robótico. Por lo tanto, el evento es una ocurrencia motivadora en el agente para cumplir sus metas requeridas. Los posibles eventos externos motivadores que afectan a un sistema robótico son:

- Sistema de sensado  
Estos sistemas de sensado son eventos que el robot requiere para realizar alguna acción relacionada a su correspondiente meta. Entre los sistemas de sensado están, los sistemas de visión, sensores de choque, sensores de proximidad, etc.
- Comandos entre usuario-maquina  
Estos comandos son eventos que el robot recibe para realizar alguna acción relacionada a una meta general. Entre los posibles

comandos están, comandos de movimiento, comandos de posicionamiento entre otros.

- Interacción entre otro robot  
Esta interacción son eventos que el robot percibe realizando alguna acción relacionada a una meta específica.

## **PERCEPCIÓN DEL EVENTO**

Para el modelo propuesto la percepción del evento se fundamenta en ¿qué agente recibe los eventos ocurridos externamente?, en donde, se puede definir que la percepción está directamente relacionado con el agente directo que recibe el evento presente. Es decir, si se presenta un evento el agente que lo percibe y propaga es el agente directamente afectado.

### **3.1.3 Unión de la Plataforma Robótica Entre la Arquitectura Física y las Metas**

La unión de la plataforma robótica está relacionada con la unión de la arquitectura física y de las metas generales. En donde, esta unión e interconexión se encuentra estrechamente ligada a los contenedores de la parte física y a los agentes de las metas particulares. La interconexión entre los contenedores y los agentes tiene como finalidad cumplir las metas requeridas por la arquitectura robótica.

La interacción y unión entre los contenedores y los agentes se implementa directamente por medio de los procesadores. En donde, pueden existir uno o más procesadores implementados en una arquitectura robótica. Por lo tanto, se define que los procesadores son el medio interno de interacción que debe existir entre los contenedores de la parte física y los agentes de la parte lógica de la arquitectura robótica. Además, la comunicación entre agentes y entre procesadores se basa en un modelo ideal y funcional. Con lo cual, la comunicación no genera un conflicto entre los agentes de la arquitectura robótica.

## **3.2 Multiprocesadores**

Para describir el modelo que se quiere implementar, se tiene que tener en cuenta que para cualquier plataforma robótica con múltiples procesadores se podría aplicar la distribución de su control, mejorando el funcionamiento del sistema. En donde, para distribuir el control se propone en este texto



desarrollar un modelo de control distribuido basado en la implementación de Multi-Agentes. Teniendo en cuenta, que en los procesadores los agentes serán las entidades de procesamiento, control y toma de decisión requerida.

Igualmente, las metas particulares y la estructura física de una plataforma robótica, pueden estar interconectadas a un procesador particular Fig. 4. También, los procesadores no están colocados o asignados de forma aleatoria, si no que se distribuyen en relación a los contenedores implementados en la estructura física del robot de una forma estratégica y funcional.

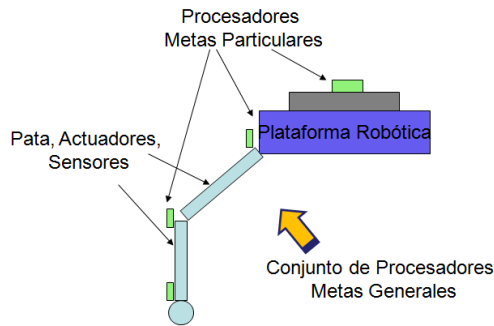


Figura 4 Arquitectura de Multiprocesadores

Finalmente, el conjunto de procesadores presentan una naturaleza asincrónica y realizarán el procesamiento de los agentes encargados de la estructura física y de las metas generales. Además, cada uno de los procesadores se encuentra operando al mismo tiempo en forma paralela y concurrente, lo cual, implica compartir recursos, canales de comunicación y memoria en el sistema del procesador. Entre estos procesos realizados por los Multi-procesadores se pueden nombrar:

- ✓ El comportamiento global o control general del sistema  
Que se basa en la interacción entre los contenedores y los agentes internos del procesador. Partiendo de algún proceso particular y obteniendo una reacción del robot en su totalidad. Estas reacciones son generadas por los eventos para obtener una meta particular.
- ✓ Toma de decisiones  
Que son las acciones acordadas por los Multi-agentes que se encuentran en los multiprocesadores. Estas decisiones de los Multi-agentes pueden ser particulares o sociales según las características requeridas por el sistema robótico.
- ✓ Desarrollo de múltiples metas  
Debido a que la plataforma robótica tiene una arquitectura física y unas metas generales, los multiprocesadores deben ser modulares, desarrollando unas acciones según las características físicas de los

contenedores internos. Y consecuentemente, de las características particulares de cada agente que desarrolla una meta particular.

Por lo tanto, la coordinación se caracteriza por utilizar la propagación por medio de la atenuación como el método para sincronizar los procesos y metas de cada agente de la plataforma robótica. Finalmente los procesadores desarrollan el agrupamiento de la estructura lógica (agentes) y la estructura física generando la interacción entre los dos conjuntos particulares que compone a una estructura robótica que se observa en la Fig. 5.

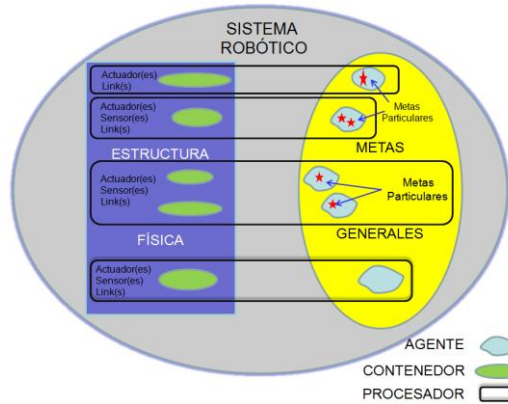


Figura 5 Procesadores y su Relación en la Arquitectura Global

### 3.3 Esquema De Interacción

Los agentes deben tener una interacción asociada con los demás agentes, con el fin, de que cada agente coopere y coordine las metas requeridas por el sistema robótico. Por lo tanto, la interacción está relacionada inicialmente con la estructura o arquitectura física del robot (estructura Fija o variable). En donde, cada agente debe estar en interacción con su agente vecino directo, luego, los agentes vecinos interaccionan con otros agentes vecinos directos consecutivamente.

Además, se debe tener en cuenta que la estructura de la plataforma puede ser fija o variable (Multi-Robots o Robot con procesadores). En donde, los vecinos directos son los agentes que se encuentran a una menor proximidad (cercanos) o con una inter-localidad física cercana. Por lo tanto, se debe analizar la estructura física (fija o variable) del sistema robótico, para determinar los vecinos directos de cada agente, esta interacción entre agentes busca desarrollar una arquitectura funcional de agentes interconectados directamente entre ellos (punto a punto). Además, con una arquitectura punto a punto se puede proponer un método de coordinación entre agentes basada en la propagación.

Dentro del sistema robótico, los agentes deben tener una comunicación para interactuar y cooperar, con el fin, de cumplir con las metas particulares que están inaccesibles para un solo agente. Estos agentes deben por su parte tener autonomía y facultades de toma de decisión. En base de estas decisiones se generará el comportamiento del robot según la situación existente. La elección del comportamiento es determinada por algún evento ocurrido en el entorno de trabajo del sistema robótico y finalizado con la meta particular que se debe cumplir. Igualmente, en un contenedor existirá una interacción entre uno o más agentes, que serán encargados de las metas particulares específicas que manipulen a las subestructuras agrupadas en el contenedor en cuestión.

### 3.4 Forma de Conexión de la Arquitectura Propuesta

A continuación, en base a la interacción debe existir una forma de conexión entre los diferentes elementos de la plataforma robótica. En donde, se explicará la propuesta arquitectural de los Multi-agentes en general y de los contenedores. Este modelo tiene como objetivo facilitar la intercomunicación, control y las interacciones relacionadas en las conexiones de la arquitectura robótica propuesta. En donde, cada agente y/o contenedor debe estar en interacción con su agente vecino o con un contenedor directamente relacionado. La conexión de la arquitectura propuesta se describirá en base a las arquitecturas de los agentes y de los contenedores.

### 3.5 Arquitectura de los Agentes

Como punto de partida, se describirá la arquitectura de control distribuido propuesta con sus agentes y se comparará con otras arquitecturas de control. Mostrando algunas características favorables y desfavorables de las arquitecturas relacionadas con este caso de estudio.

Inicialmente se propone desarrollar una arquitectura de interconexión distribuida e igualitaria entre los agentes. En donde, los agentes tienen un mismo nivel de elección y autonomía. Esta arquitectura igualitaria se puede definir como la no existencia de agentes maestros que impongan la meta particular o un dominio con respecto a los agentes vecinos del sistema. Además, este mismo nivel genera que los agentes lleguen a un acuerdo funcional y coordinado para cumplir con las metas generales del sistema robótico. En una arquitectura distribuida e igualitaria, los agentes desarrollan las acciones o metas particulares durante un mismo instante en forma

paralela e independiente. Estos agentes en la arquitectura del modelo propuesto, tienen conexiones que les permiten comunicarse directamente con los agentes vecinos en forma online, es decir, tener mecanismos para obtener la información de los agentes que prestan servicios requeridos en algún momento dado.

Cada agente en su búsqueda de la meta particular requiere tener un conocimiento de su entorno de trabajo, con el fin, de evitar los posibles conflictos. Teniendo como conflicto a las colisiones o problemáticas presentes en uno o varios agentes, limitando la obtención del evento. Por lo tanto, se debe tener en cuenta, que el entorno de trabajo del agente incluye la posición de los vecinos, las características de las metas y las limitantes de su funcionamiento o de su medio de acción.

Además, en un sistema distribuido igualitario los agentes pueden ser redundantes, en donde, si se presentan fallas de procesamiento los agentes se pueden sustituir o suplir. Los puntos de fallos son puntos particulares de una pequeña zona, variando en forma mínima el comportamiento global del sistema.

En cambio, en una arquitectura centralizada se producen limitantes para el desarrollo de múltiples acciones durante un mismo instante (denominado cuello de botella) generando retardos y funcionamientos secuenciales. El conocimiento del entorno de trabajo del robot tiene un procesamiento elevado, limitando otros procesos requeridos. Finalmente, al tener una falla el sistema robótico deja de funcionar en su totalidad (punto único de fallo) y no puede realizar otras metas requeridas.

Por otra parte, en una arquitectura jerárquica, existe un agente totalitario que controla a los demás agentes presentando retardos en la funcionalidad del sistema robótico. Además, las comunicaciones se desarrollan por medio de un servidor en el agente dominante, presentando dificultades en la comunicación, como los retardos y procesamientos elevados. El conocimiento del entorno de trabajo del robot solo lo controla el agente dominante. Finalmente, se pueden sustituir o suplir los fallos ocurridos, siempre y cuando, el fallo no se produzca en el agente dominante (Punto crítico de fallo) “en consecuencia, las arquitecturas jerárquicas y centralizadas no se aplican en el modelo propuesto”.

### 3.6 Arquitectura De Los Contenedores

La arquitectura de los contenedores busca tener un control distribuido en toda la plataforma robótica. Definiendo como control distribuido al método

propuesto para la interacción de múltiples agentes y contenedores Fig. 6. Estos contenedores podrán controlar y manipular diferentes partes del sistema robótico en forma paralela, pero al unir cada una de las partes puntuales de control, se desarrolle un control general del robot sin que existan errores o conflictos en el mismo.

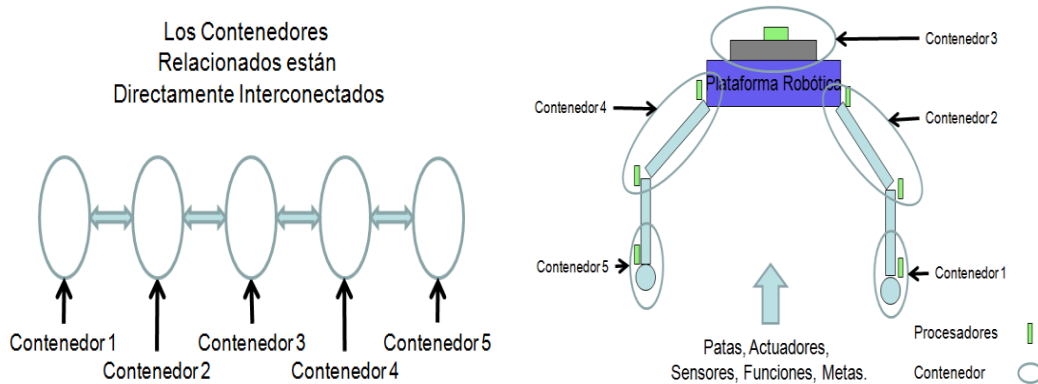


Figura 6 Arquitectura de Interconexión Distribuida de los Contenedores

En un sistema distribuido igualitario los contenedores pueden ser redundantes, en donde, si se presentan fallas de una subestructura (contenedor) se pueden sustituir o suplir, además, los puntos de fallos son puntos particulares de una pequeña zona, variando en forma mínima el comportamiento global de la estructura física del sistema.

Estos contenedores en la arquitectura del modelo propuesto, deben tener conexiones que les permiten intercomunicarse directamente con los contenedores vecinos o directamente relacionados en forma online, es decir, tener mecanismos para obtener la información de los contenedores directos.

### 3.7 Propagación

Cada agente debe tener un sistema de propagación, que se define como la reacción producida en el agente por un evento relevante que se comunica a los demás agentes. Luego, los agentes vecinos que recibieron la propagación de ese evento, podrán ignorar el evento o podrán si es el caso propagar sobre el evento a sus propios vecinos. Esta información que se encuentra fluyendo entre los agentes vecinos consecutivamente, corresponde a una propagación del evento ocurrido. Esta propagación tiene como finalidad, generar una información para los agentes pertinentes relacionados, cumpliendo con la meta asociada al evento.

Por otra parte, la propagación tiene como objetivo que los sistemas Multi-agentes tengan una coordinación. Esta coordinación, se define como la forma de articular las acciones individuales de cada uno de los agentes logrando un resultado final coherente (obtención de la meta general).

Por lo demás, la propagación también deberá fluir, si es el caso, por cada uno de las conexiones de los contenedores, en donde, la propagación fluirá entre los contenedores directamente relacionados enviando una información de propagación, en donde aquellos agentes que reciben esta propagación tomen decisiones de acción y soluciones en función a la cooperación del agente que propaga la información.

### 3.8 Atenuación

La información propagada se compone de datos pertinentes sobre la importancia del evento ocurrido, denominada como la atenuación. Es decir, la atenuación se relaciona con el evento (importancia e intensidad de obtener la meta particular) en donde, se genera una variable que puede ser numérica, con un valor proporcional a la importancia del evento ocurrido. Esta información del evento, indica los datos de la importancia o cercanía a la meta relacionada con el evento.

### 3.9 Conflictos

Los conflictos son las problemáticas por la utilización de los recursos compartidos entre los agentes y que estos recursos están limitados. En donde, los agentes pueden generar un conflicto por la posición en un espacio del ambiente ocupado o utilizado por otro agente. En donde, se propone de tener una lista de restricciones y reglas para evitar los conflictos entre sus vecinos. Estas listas de restricciones y reglas son las limitaciones y las distancias entre cada uno de los link o del sistema robótico.

Estas limitaciones deben ser conocidas para la obtención de una meta particular. Por otra parte, si esta solución presente no obtiene buenos resultados, se debe implementar una comunicación entre agentes. Con el fin de la interacción y de la resolución de los conflictos entre estos agentes.

### 3.10 El Agente

Cada agente debe tener una arquitectura homogénea en sus componentes internos. En donde el agente se fundamenta de las metas particulares que debe realizar y en la búsqueda de llegar a solucionar las diferentes problemáticas presentes en una arquitectura robótica.

Por otra parte, se puede presentar en esta propuesta, que al producirse el hallazgo del evento, el agente que lo encontró puede pasar eventualmente a un nivel de mando específico sin requerir un nivel de procesamiento más elevado entre los demás agentes de la arquitectura robótica.

#### 3.10.1 Arquitectura

En la arquitectura de los agentes Fig. 7. deben existir unos elementos o características que se encuentran relacionados para la funcionalidad del agente. Teniendo en cuenta que las metas pueden ser diferentes y en consecuencia los agentes son en su estructura lógica heterogéneos. Entre estas características que se incorporan en la arquitectura están: la comunicación, identificación y/o reconocimiento, manejo de eventos, elección de toma de decisión (metas), desarrollo de la propagación, restricciones, reglas y control de los actuadores.

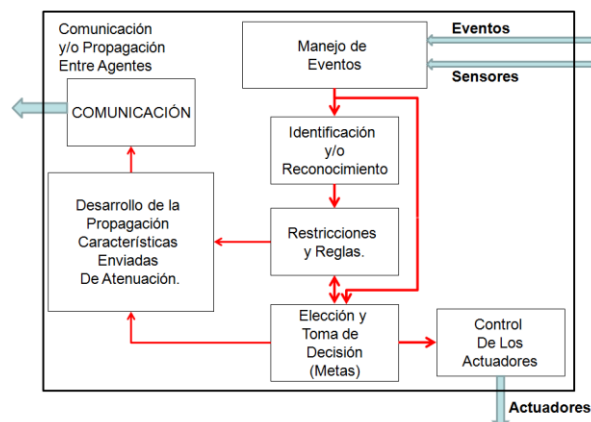


Figura 7 Arquitectura Propuesta de los Agentes

Estas características de la arquitectura del agente se fundamentan, en buscar soluciones para las diferentes problemáticas que se pueden presentar en un modelo de control distribuido. Además, estas características tienen como eje principal la obtención de las metas particulares y generales. Dado que los agentes cooperativos comparten los mismos objetivos, tienen

capacidad de acción y de respuesta al cambio de su entorno de trabajo externo, se comunican e interactúan entre otros agentes o incluso entran en conflictos. Se explicarán a continuación algunas características requeridas por los agentes.

- ✓ Identificación y/o Reconocimiento  
Los agentes deben tener métodos de identificación y/o reconocimiento de sus propios vecinos. Con el fin, de poder intercambiar información entre agentes vecinos, para la cooperación en la obtención de la meta general.
- ✓ Comunicación  
Al presentarse posibles conflictos en un sistema robótico es requerida una comunicación. En donde, esta resolución de conflictos se definen como un conjunto de técnicas que permiten manejar las diferencias que pueden tener los agentes por el acceso a recursos compartidos limitados.
- ✓ Manejo De eventos y Toma de Decisión  
Estos Agente deben estar interconectados a los sistemas sensoriales y a los estados del agente, con el fin, de tener una capacidad de acción y de respuesta al cambio de su entorno de trabajo exterior autónomamente. Como también, los Agentes deben tener algunos mecanismos relacionados entre cada agente para tomar decisiones inteligentes.
- ✓ Restricciones y Reglas  
Cada Agente debe tener una lista de restricciones y reglas a cumplir, con el fin, de que el agente pueda ser autónomo y libre en sus actos pero sin producir conflictos entre sus vecinos. Estas listas de restricciones son las limitaciones del sistema robótico. En donde, este sistema robótico queda bloqueado por aspectos del ambiente o de sus propias limitaciones mecánicas.

### 3.10.2 Características De Los Agentes

Las características esperadas por estos agentes son similares a las características generales de la robótica en donde los agentes son:

- ✓ Reprogramables
- ✓ Manipulador
- ✓ Multifuncional
- ✓ Repetitividad
- ✓ Toma De Decisión



## 4. MODELO

Para desarrollar e implementar el modelo propuesto con un sistema de múltiples agentes sin un conocimiento previo se pueden presentar dificultades o problemáticas desconocidas. En consecuencia a lo anterior, se determinó aplicar los conocimientos adquiridos (del sistema robótico básico) en un sistema robótico de múltiples agentes (Brazo planar de múltiples grados de libertad) y consecuentemente en un sistema robótico complejo AMARANTA [CARL2009], reduciendo las posibles dificultades y mejorando la funcionalidad en la implementación del modelo propuesto con Multi-agentes en un control distribuido.

Para la explicación del modelo propuesto se elige presentarlo por medio de la plataforma robótica básica (brazo planar de dos grados de libertad). Donde, esta plataforma consta de dos ejes de movimiento: hombro y codo. Con dos agentes, el primer agente del Antebrazo es el agente dominante que requiere llegar a las posiciones asignadas en las metas generales y el segundo agente es el agente del Brazo colaborativo que apoya los movimientos del Antebrazo para cumplir las metas generales propuestas. Para un mayor conocimiento de la arquitectura implementada para la plataforma básica (brazo planar de dos grados de libertad) esta plataforma su arquitectura y sus características de implementación se encuentran en el anexo 2.

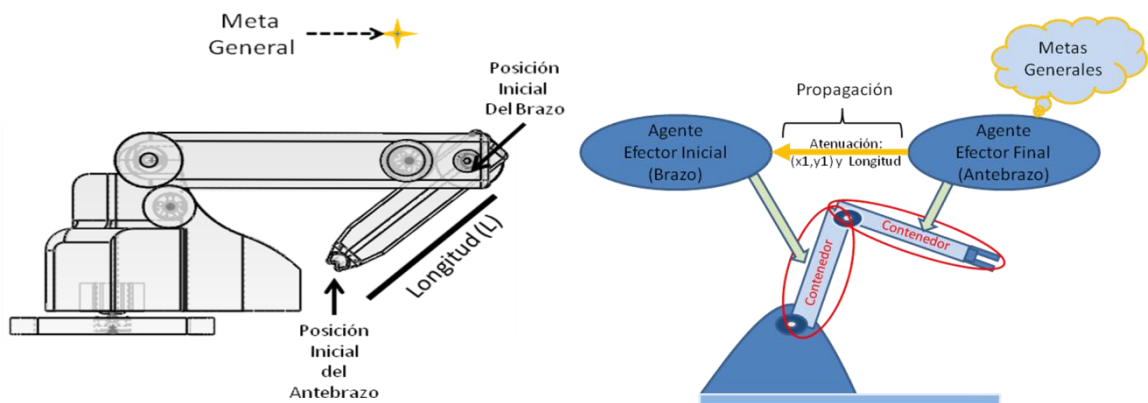


Figura 8 Propagación de los Dos Agentes del Brazo Planar de Dos Grados de Libertad

En el Brazo planar de dos grados de libertad las metas generales son conocidas por el agente que controla al contenedor del efector final (Antebrazo). Este agente del efector final realiza las acciones pertinentes de su propio contenedor, como también envía una propagación a su vecino

directo (Agente del Brazo). Esta propagación tiene como finalidad, aportar información al agente que controla al contenedor del Brazo ignorando el evento o si es el caso, realizando las acciones relacionadas al evento que le fue informado y obteniendo una cooperación para una meta general Fig. 8. Además, se tienen en cuenta que no son datos jerárquicos de control o datos de la cinemática, sino datos que tienen una perspectiva distribuida, autónoma, independiente y cooperativa entre los agentes que interactúan en la plataforma robótica.

#### 4.1. Modelo de Propagación por Medio de Regiones Circulares

Cada agente debe autónomamente determinar la posición actual de su propio contenedor basado en la estructura física del robot Fig. 9. Por lo tanto, para determinar las posiciones de cada contenedor se utilizan las EC. 1. y EC. 2. Además, los agentes al determinar la posición actual y conocer la meta particular pueden determinar la distancia a la meta particular, que fundamentalmente es el radio de la ecuación de la circunferencia EC. 3. Con lo cual, se puede establecer un desplazamiento, velocidad, o aceleración requerido por el actuador de un contenedor particular para alcanzar la meta particular requerida.

$$\begin{array}{ll} \text{EC. 1.} & \text{Seno (alfa)} = \text{PosY\_Inicial} / \text{Hipotenusa} \\ & \text{PosY\_Inicial} = \text{Hipotenusa} * \text{Seno (alfa)} \\ \text{EC. 2.} & \text{Coseno (alfa)} = \text{PosX\_Inicial} / \text{Hipotenusa} \\ & \text{PosX\_Inicial} = \text{Hipotenusa} * \text{Coseno (alfa)} \end{array}$$

En donde:

- Alfa: Angulo actual que se encuentra el contenedor.
- PosX\_Inicial: incógnita de la posición actual en X, en donde se encuentra el contenedor.
- PosY\_Inicial: incógnita de la posición actual en Y, en donde se encuentra el contenedor.
- Hipotenusa: Longitud de la extremidad o del contenedor.

$$\text{EC. 3} \quad (x_2 - x_1)^2 + (y_2 - y_1)^2 = \text{Radio}^2$$

Luego, desde el agente Antebrazo se envía una propagación hacia el agente Brazo. Esta nueva propagación aporta la información relacionada con una coordenada de la meta particular y con la distancia de alcance de su link (del agente del Antebrazo) Fig. 8.

Este agente del Brazo por medio de un código fuente determina su propia circunferencia (Circunferencia Base). En donde, para determinar

la ecuación de la Circunferencia Base se requiere del radio del círculo y del punto céntrico EC. 4.

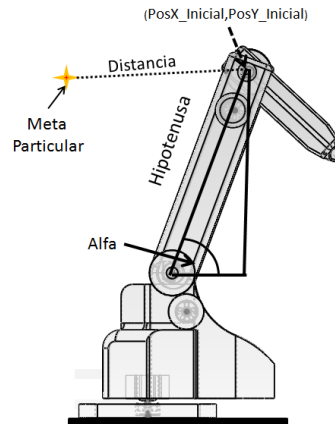


Figura 9 Cálculo de la Posición Actual del Contenedor y Consecuentemente la Distancia con el Objetivo Particular

$$\text{EC. 4} \quad (x - \text{PuntoXcentro})^2 + (y - \text{PuntoYcentro})^2 = \text{Radio}^2$$

Por lo tanto, en este caso particular el agente del Brazo asigna al radio el valor de la longitud del contenedor del Brazo ( $L_{\text{Brazo}}$ ) y el punto céntrico se determina como el punto inicial fijo de referencia (0,0). Por lo que se obtiene una ecuación representativa del Círculo Base del Brazo EC. 5.

$$\text{EC. 5.} \quad (X)^2 + (Y)^2 = L_{\text{Brazo}}^2$$

También se puede representar esta ecuación del Círculo Base como EC. 6.

$$\text{EC. 6.} \quad X^2 + Y^2 - L_{\text{Brazo}}^2 = 0$$

Además, se pueden obtener los múltiples puntos del Círculo Base por medio de la EC. 7. En donde, los cálculos para obtener los puntos de la circunferencia están en función de los valores angulares ( $teta$ ), representando los puntos por medio de las coordenadas en  $x$  y en  $y$ .

$$\text{EC. 7.} \quad x = L_{\text{Brazo}} * \cos(teta);$$

$$y = L_{\text{Brazo}} * \sin(teta);$$

El desarrollo implementado para la obtención de la Circunferencia Base tiene como finalidad establecer las posibles posiciones o metas particulares que el agente del Brazo puede alcanzar. Por medio de un

código en Matlab ver anexo 3 <sup>15</sup> <sup>16</sup> se puede obtener la circunferencia base desarrollada por el agente del Brazo, este agente del Brazo puede determinar qué cualquier punto de la circunferencia es una meta particular (posición a la que debe trasladarse) cooperando con el agente del Antebrazo. Por lo tanto, el agente del Brazo al tener conocimiento de los puntos de la circunferencia base determina que estos puntos en cuestión son posiciones físicamente alcanzables Fig. 10 (a) Posiciones Alcanzables Por El Contenedor Del Brazo.

Por otra parte, al recibir la propagación el agente del Brazo realiza unos cálculos propios para determinar la meta particular a realizar. En este caso, el agente del Brazo al conocer en la atenuación la longitud del link (L\_Atenuacion) y la posición (PuntoX\_Atenuacion, PuntoY\_Atenuacion) se puede utilizar la EC. 8. Para obtener la circunferencia de propagación.

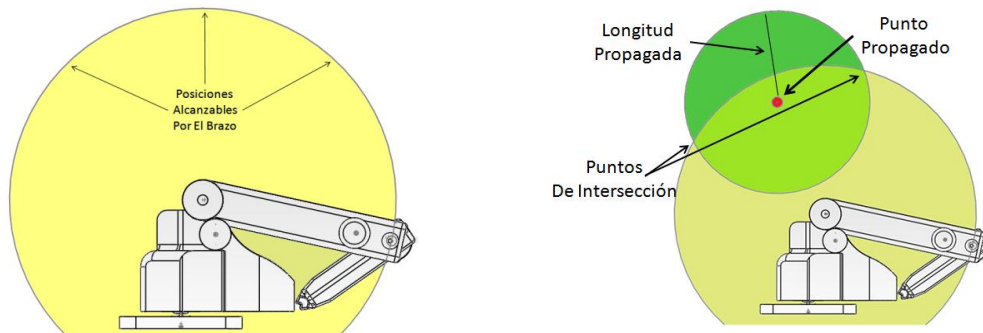


Figura 10 a). Posiciones Alcanzables por el Contenedor del Brazo b) Circunferencias en Función del Contenedor Brazo y de los Datos Propagados

**EC. 8.**  $(x - \text{PuntoX\_Atenuacion})^2 + (y - \text{PuntoY\_Atenuacion})^2 = L\_Atenuacion^2$

También se puede representar la ecuación del círculo anterior despejando los paréntesis EC. 9.

**EC. 9.**  $x^2 + y^2 + D * x + E * y + F = 0$

Donde:

$D = - 2 * \text{PuntoX\_Atenuacion}$

$E = - 2 * \text{PuntoY\_Atenuacion}$

$F = (\text{PuntoX\_Atenuacion})^2 + (\text{PuntoY\_Atenuacion})^2 - L\_Atenuacion^2$

Por otra parte, el agente del Brazo realiza los cálculos para obtener los puntos de la circunferencia en función de los valores angulares (Teta),

<sup>15</sup> Anexo 1. Diagrama de flujo correspondiente al proceso de la obtención de la Circunferencia Base.

<sup>16</sup> Anexo 2. Código fuente correspondiente al proceso de la obtención de la Circunferencia Base.

representando los puntos por medio de las coordenadas en x y y. Estos cálculos se fundamentan por medio de la siguiente ecuación EC. 10.

$$\begin{aligned} \text{EC. 10.} \quad x &= L\_Atenuacion * \cos(\text{teta}) + \text{PuntoX\_Atenuacion}; \\ y &= L\_Atenuacion * \sin(\text{teta}) + \text{PuntoY\_Atenuacion}; \end{aligned}$$

Igualmente, el agente del Brazo por medio de un código fuente y de la EC. 10. determina los puntos de la circunferencia en función de los datos propagados y en consecuencia las posibles metas particulares que el agente del Brazo debe realizar. Por medio de un código en Matlab<sup>17 18</sup> el agente del Brazo al desarrollar las circunferencias, puede determinar por medio de unos cálculos los puntos de intersección entre las circunferencias en estudio. En donde, estos puntos de intersección determinan las posibles metas que el agente del Brazo puede realizar para la cooperación Fig. 10 (b).

Para determinar los puntos de intersección entre las dos circunferencias, inicialmente se deben utilizar la EC. 6. y la EC. 9. que representan a las dos circunferencias en estudio. Con estas dos ecuaciones se realiza una igualdad. Obteniendo una nueva ecuación EC. 11. representativa a una recta que entrecruza a los puntos de intersección de las dos circunferencias.

$$\text{EC. 11.} \quad D * x + E * y + F = 0$$

Donde:

$$D = -2 * \text{PuntoX\_Atenuacion}$$

$$E = -2 * \text{PuntoY\_Atenuacion}$$

$$F = (\text{PuntoX\_Atenuacion})^2 + (\text{PuntoY\_Atenuacion})^2 - (L\_Atenuacion)^2 - (L\_Brazo)^2$$

Con la ecuación de la recta EC. 11. Se despeja x que es una de las variables para obtener una nueva ecuación EC. 12. Esta ecuación 12. Se reemplaza en una de las ecuaciones de las circunferencias en estudio, obteniendo una nueva ecuación cuadrática EC. 13 ver Anexo 3<sup>19</sup>.

$$\text{EC. 12.} \quad x = -F/D - y * E/D$$

$$\text{EC. 13.} \quad (-F/D - y * E/D)^2 + Y^2 - L\_Brazo^2 = 0$$

<sup>17</sup> Anexo 3. Código fuente correspondiente al proceso de la obtención de la circunferencia en función de los datos propagados.

<sup>18</sup> Anexo 4. Diagrama de flujo correspondiente al proceso de la obtención de la Circunferencia en función de los datos propagados.

<sup>19</sup> Anexo 4. Diagrama de flujo correspondiente al proceso de la obtención de la Circunferencia en función de los datos propagados.

Donde:

$$\begin{aligned} D &= -2 * \text{PuntoX\_Atenuacion} \\ E &= -2 * \text{PuntoY\_Atenuacion} \\ F &= -((\text{PuntoX\_Atenuacion})^2 + (\text{PuntoY\_Atenuacion})^2 - \\ &\quad (\text{L\_Atenuacion})^2 - (\text{L\_Brazo})^2) \end{aligned}$$

Si se resuelve la ecuación cuadrática EC. 13. podemos obtener dos soluciones que representan a los dos puntos que interceptan a estos dos círculos. En consecuencia, esta propuesta se fundamenta en la obtención de metas particulares de cada uno de los agentes y la sumatoria de estas metas particulares obtienen la meta general del Brazo planar. En donde, el agente del Brazo al desarrollar estas anteriores ecuaciones puede determinar la meta particular autónoma y cooperar con el agente del Antebrazo anexo 2<sup>20</sup>.

A continuación se representa el modelo con un ejemplo, en donde, cada uno de los contenedores (contenedor del Brazo y contenedor del Antebrazo) se encuentra en una posición inicial. El contenedor del Antebrazo por medio de su agente determina la meta general realizando una propagación para el contenedor del Brazo. Teniendo en cuenta que los agentes deben conocer sus posiciones iniciales en su entorno.

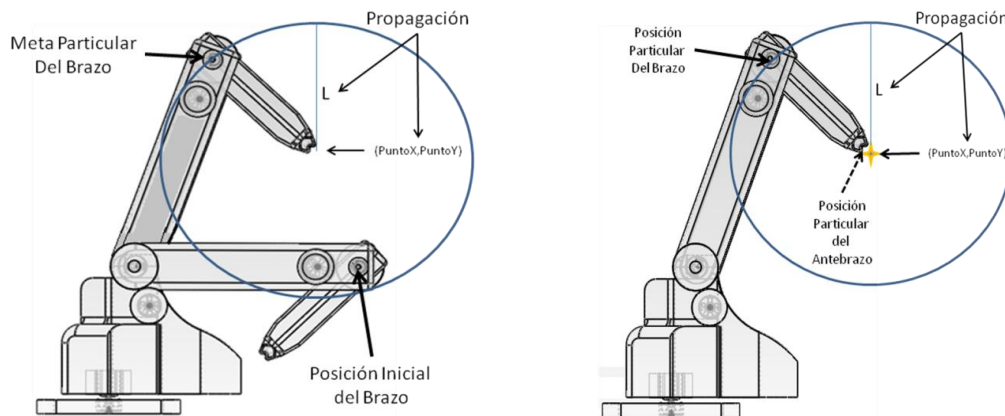


Figura 11 Circunferencia y Meta Particular del Agente que Controla al Contenedor del Brazo

Luego de la propagación recibida por el agente del Brazo, este agente realiza los cálculos anteriormente explicados, con el fin de determinar autónomamente y desarrollar las acciones correspondientes al cumplimiento de la meta particular elegida. Finalmente, al obtener las

<sup>20</sup> Programas desarrollados por medio de Matlab junto a su simulación.

metas particulares de los agentes se puede obtener totalmente la meta general requerida<sup>21</sup> Fig. 11.

#### 4.2. Modelo de Propagación por Medio de una Región Circular y la Distancia Mínima de Cercanía

Este modelo de propagación es un modelo complementario del modelo por circunferencias que también se desarrolló como otro método especializado propuesto. Este método por parámetros de distancia se fundamenta en que el agente del antebrazo conoce la meta general y busca una posición o meta particular que se encuentre a una distancia menor con respecto a su recorrido. En donde, por medio de la propagación se envían datos de atenuación que están relacionadas con una coordenada de la meta particular y con la distancia de alcance de su link (del agente del Antebrazo). Por lo tanto, los agentes al determinar la posición actual pueden realizar unos cálculos que determinen las metas particulares a desarrollar. Inicialmente, el agente del Antebrazo al conocer la meta general, determina la meta particular y realiza las acciones correspondientes a la activación de su actuador para el acercamiento a su propia meta particular.

Como breve explicación, el agente del brazo debe determinar la posición de su base que se define como la posición fija de inicio (0,0). Para luego establecer la distancia que tiene su extremidad en relación con los datos propagados EC.3. En donde, el agente del brazo para desarrollar y realizar las ecuaciones que requiere. Debe tener un conocimiento de las siguientes variables que a continuación se definirán para facilitar su comprensión:

- L\_Brazo: longitud de la extremidad del brazo.
- PosX\_Base: posición en X de la base del brazo.
- PosY\_Base: posición en Y de la base del brazo.
- L\_Atenuacion: longitud propagada por el agente del antebrazo.
- PosX\_ Atenuacion: posición en X propagada por el agente del antebrazo.
- PosY\_ Atenuacion: posición en Y propagada por el agente del antebrazo.

Con lo anterior, El agente del brazo al conocer las variables expuestas define inicialmente dos ecuaciones EC. 15. y EC. 16.

$$\text{EC. 15. } (\text{PuntoX\_Atenuacion}-x1)^2 + (\text{PuntoY\_Atenuacion}-y1)^2 = L\_Atenuacion^2$$

---

<sup>21</sup> Imágenes obtenidas de simulación y animación simultánea en Matlab junto a Virtual Real Simmechaniks.

$$\text{EC. 16. } (\text{PuntoX\_Base}-x1)^2 + (\text{PuntoY\_Base}+y1)^2 = L\_Brazo^2$$

Con estas ecuaciones se observa que el agente del brazo tiene dos ecuaciones y dos incógnitas que son  $x1$  y  $y1$ . En consecuencia, por medio de la resolución de estas dos ecuaciones se puede determinar los múltiples valores de  $x1$  y  $y1$  que satisfagan a las dos ecuaciones anteriores. Por lo tanto, estas dos incógnitas determinan las posiciones que el agente del brazo puede elegir como su meta particular.

Esta activación de su actuador está directamente relacionada con la distancia actual del contenedor y la meta particular que desea alcanzar. La distancia entre los dos puntos en estudio se obtiene por medio de la EC. 17. Simultáneamente, el agente del antebrazo propaga los datos requerido para que el agente del brazo pueda cooperar en función de la obtención de la meta general. En donde, la propagación recibida por el agente del brazo está relacionada con la longitud (Long\_Antebrazo) y la meta particular (Punto\_XAnt\_Meta, Punto\_YAnt\_Meta) del agente del antebrazo que propaga esta información.

$$\text{EC. 17. } \text{Distancia} = ((x2-x1)^2 + (y2-y1)^2)^{1/2}$$

En donde, el agente del brazo inicialmente determinar su posición actual de su extremo más lejano a la base de la plataforma robótica, por medio de las EC. 1. Y EC. 2. Luego al recibir la propagación realiza una circunferencia del antebrazo que relaciona los puntos en donde debe encontrarse el Antebrazo para lograr la meta particular que está intentando alcanzar Fig. 12 a).

Luego determina un punto de la circunferencia al que debe desplazarse con el menor esfuerzo posible, es decir, determina el punto con la menor distancia a recorrer o más cercano a su posición actual, este punto más cercano es el punto elegido como meta particular que debe realizar el agente del brazo Fig. 12 b). Para desarrollar la circunferencia del antebrazo el agente del brazo utiliza la EC. 3. y la EC. 4.

Además, para determinar el punto de la circunferencia que se encuentra a una distancia más corta, el agente del brazo utiliza dos ecuaciones que son: la ecuación de los datos propagados por el antebrazo EC 18. Y la ecuación de la recta en función de la posición actual del brazo (Pos\_XBrazo\_Actual, Pos\_YBrazo\_Actual) y la meta particular propagada (Punto\_XAnt\_Meta, Punto\_YAnt\_Meta) EC.19.

Con estas dos ecuaciones se pueden igualar y obtener los puntos de intersección, en donde, se presentan dos soluciones, es decir, dos puntos de intersección entre el círculo y la recta. A estos dos puntos se les calcula su distancia en relación al punto actual del brazo y estas distancias obtenidas se



comparan para determinar la distancia mínima, en donde, el punto con la distancia menor se elige como meta particular del brazo.

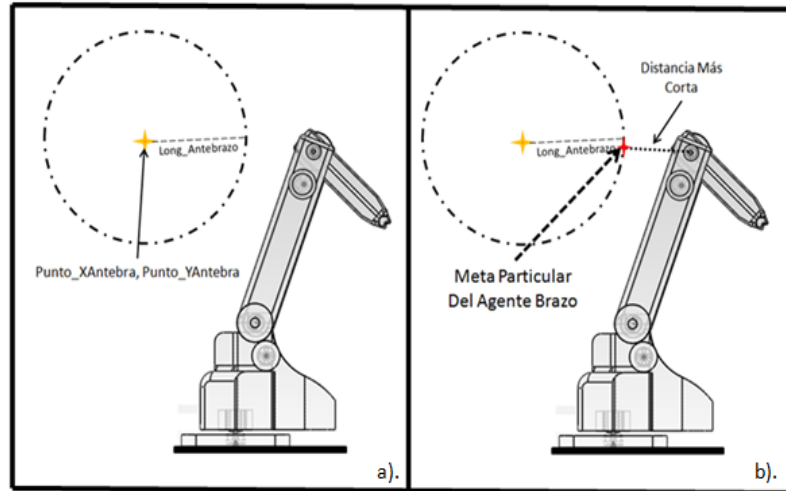


Figura 12 (a) El Agente del Brazo Realiza una Circunferencia del Brazo y (b) Determina su Propia Meta Particular en Función de la Distancia más Corta a la Circunferencia

$$\text{EC. 18. } (x - \text{Punto\_XAnt\_Meta})^2 + (y - \text{Punto\_YAnt\_Meta})^2 = \text{Long\_Antebrazo}^2$$

$$\text{EC.19. } (\text{Punto\_YAnt\_Meta} - \text{Pos\_YBrazo\_Actual}) = m (\text{Punto\_XAnt\_Meta} - \text{Pos\_XBrazo\_Actual})$$

Este modelo de propagación puede ser implementado, pero tiene una limitante en relación al contenedor del brazo debido a que este contenedor está limitado a un movimiento circular por su base y consecuentemente la elección de un punto con la distancia más corta puede ser un punto no alcanzable pero la plataforma se encontrará lo más cerca posible a la meta general.

Para la implementación del modelo propuesto en la plataforma AMARANTA se toma de manera idéntica al modelo de propagación desarrollado anteriormente. Pero se debe tener en cuenta la forma que se propaga la información en este caso de un robot móvil con múltiples grados de libertad. En la propagación, uno de los agentes conoce el punto atractor o de referencia final, este agente realiza algunas acciones y propaga una nueva información relevante al punto de referencia final y el ángulo de posicionamiento final.

Esta información propagada genera nuevas acciones de los demás agentes en forma colaborativa para alcanzar el punto de referencia propagado. Por otra parte, si el evento presente está relacionado a la estabilidad de la plataforma, la propagación se desarrolla entre agentes vecinos directos para determinar las condiciones de cada agente que cumplan con la estabilidad de la plataforma y evitar caídas o movimientos erráticos.

Por otra parte, la atenuación está relacionada directamente con la sincronización entre patas de la plataforma. Esta sincronización es fundamental para la estabilidad y coordinación entre patas de la plataforma. Además, cada agente debe tener restricciones del comportamiento para evitar la simultaneidad de movimientos y la cantidad de patas que deben encontrarse apoyadas en el suelo. Esta sincronización determina en qué momento cada pata puede moverse o interactuar simultáneamente con otra pata de la plataforma robótica.

#### 4.3. Modelo de Propagación por Medio de Regiones Circulares para la Plataforma AMARANTA.

Para la implementación de una interacción entre agentes en la plataforma AMARANTA fue realizado por medio de regiones circulares. Donde, por medio de la capa superior los agentes conocen la posición general de la plataforma y consecuentemente determina su posición en el entorno local EC. 24. Y EC. 25.

$$\begin{aligned}
 \text{EC 24} \quad & \text{Pos\_Y\_BasePata1}=(\text{LongCentro}*\sin(\text{AranaAng}+\text{Alfa1}))+\text{AranaPos}(2); \\
 & \text{Pos\_X\_BasePata1}=(\text{LongCentro}*\cos(\text{AranaAng}+\text{Alfa1}))+\text{AranaPos}(1); \\
 & \text{Pos\_Y\_ExtPata1}=(\text{LongExtPata1}*\sin(\text{AngHombroPata1}+\text{AranaAng}+\text{Alfa1}))+ \\
 & \quad \text{Pos\_Y\_BasePata1}; \\
 \text{EC 25} \quad & \text{Pos\_X\_ExtPata1}=(\text{LongExtPata1}*\cos(\text{AngHombroPata1}+\text{AranaAng}+\text{Alfa1}))+ \\
 & \quad \text{Pos\_X\_BasePata1};
 \end{aligned}$$

Luego, el agente que conoce la meta general determina el rango que puede su propia subestructura contenida desplazarse; teniendo en cuenta que el agente tiene un conocimiento de la estructura física que controla y las dimensiones de los links propios, sin tener en cuenta elementos externos físicos de su propio contenedor. Finalmente determina por medio de regiones circulares la posición alcanzable a la que se desplazará para intentar llegar al evento motivador Fig. 13. Luego, este agente propaga una información para que los agentes vecinos puedan determinar su tarea particular alcanzable.

Por lo tanto, cada contenedor por medio de su respectivo agente conocerá su propia posición en referencia a la posición general de la plataforma. Con lo cual, un agente desarrolla cálculos geométricos para determinar la posición particular que debe realizar intentando alcanzar el evento motivador (Meta General) de esta pata en particular. Otra pata vecina al conocer la meta particular que le fue propagada realiza por medio de cálculos geométricos "propios" la obtención de la meta particular que debe realizar para ayudar a la pata que le propago el evento motivador.

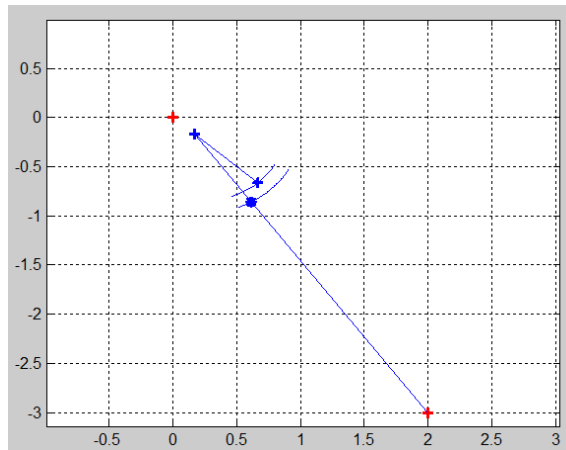


Figura 13 Comportamiento de un Agente para Determinar la Posición Máxima Alcanzable para Obtener el Evento Motivador

Consecuentemente, esta pata propaga información a sus vecinas directas que realizan nuevos cálculos para determinar las metas particulares que deben realizar para ayudar a la pata vecina que propago la información, así esta propagación se modifica por cada agente y se distribuye por toda la plataforma robótica Fig. 14.

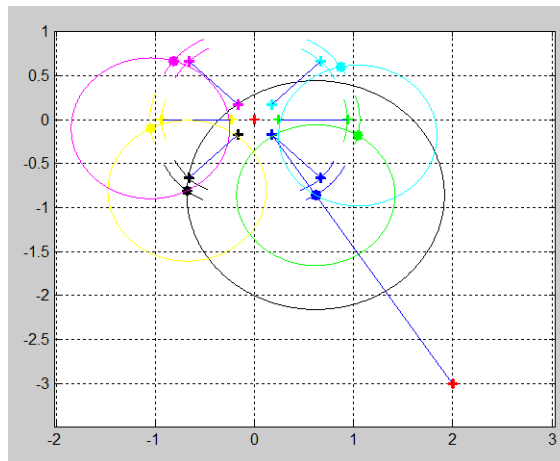


Figura 14 Comportamiento de los Agentes por Propagaciones Determinando las Posiciones Máximas Alcanzables para cada Agente de la Plataforma

Con lo anterior, cada una de las patas vecinas puede determinar metas particulares sin conocer la meta general. Estas metas particulares al ser cumplidas acercan a la pata que conoce el evento motivador a la meta general y finalmente logran alcanzar el evento motivador de la plataforma, sin importar la cantidad de extremidades que pueda tener la plataforma y el sistema puede ser incrementada tanto en elementos físicos como en las metas a desarrollar.

#### 4.4. Modelo de Propagación por Medio de Regiones Circulares con Distancia Mínima para la Plataforma AMARANTA.

Este modelo geométrico básico que determine la posición alcanzable por cada una de las patas en forma independiente. Cabe notar, que para realizar el modelo geométrico se debe tener en cuenta que cada agente de la plataforma conoce la posición local de la plataforma (Por una capa superior) y consecuentemente determina la posición particular de la subestructura física en el espacio de trabajo, como también conoce las características mecánicas y de alcance del sistema físico que se encuentra inmerso en el contenedor que tiene el agente en su control.

En las siguientes figuras Fig. 15 (a), y Fig. 15 (b) se observa por medio de Matlab las posiciones de las metas particulares de cada uno de los agentes deciden proceder para su movimiento. En donde, cada pata que esta controlada por un agente independiente realiza sus acciones y se desplazan cumpliendo con una coordinación para alcanzar las metas particulares y consecuentemente la meta general.

Se debe tener en cuenta que el color verde es la meta general y los puntos negros y azules, son las metas particulares que debe realizar cada uno de los agentes. En secuencia al lograr cumplir con cada una de las metas particulares gradualmente se van acercando a la meta general.

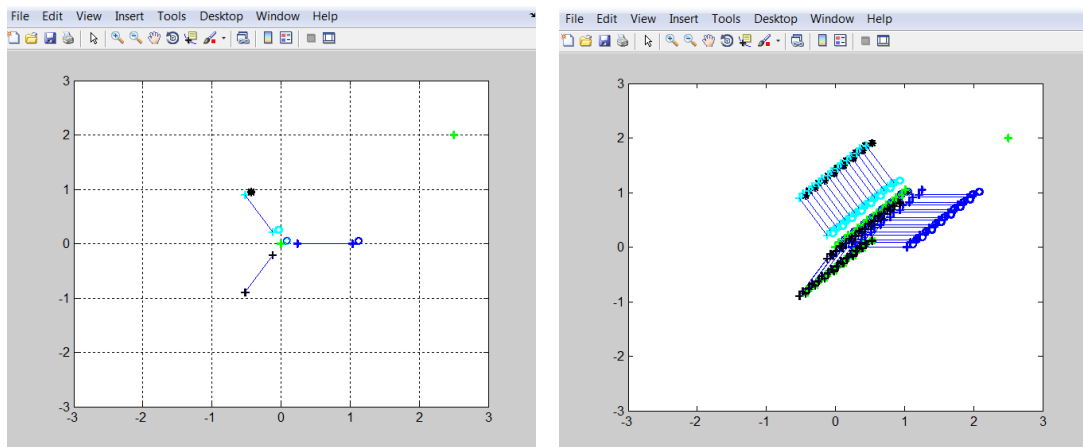


Figura 15 (a) Plataforma Simulada AMARANTA en Dos Dimensiones y (b) Recorrido de la Plataforma Simulada en MATLAB

## 5. IMPLICACIONES Y PRUEBAS

En el siguiente capítulo se presentarán las implicaciones y pruebas realizadas del modelo propuesto de control distribuido. Estas pruebas tienen como finalidad: verificar la viabilidad de la propagación entre agentes, comprender las interacciones de los agentes y sus procesos. Igualmente, se presentarán algunos datos obtenidos por simulación por medio de la plataforma de programación Matlab y Solidwork.

### 5.1 Propagación de Agentes por Medio de Regiones Circulares

A continuación, se presentarán algunos ejemplos desarrollados para el modelo propuesto. En donde, se presentará un ejemplo por cada plataforma desarrollada en esta investigación<sup>22</sup> (brazo planar de dos grados de libertad, brazo planar de múltiples grados de libertad y AMARANTA). Junto al análisis comparativo experimental de los dos métodos propuestos y un análisis de sensibilidad.

#### 5.1.1 Brazo Planar de Dos Grados de Libertad

El agente del Antebrazo al conocer la meta general realiza unas acciones propias para alcanzar su meta particular y envía la propagación requerida para informar a su vecino directo sobre el evento sucedido. En este caso particular, la propagación enviada se fundamenta en la longitud del antebrazo ( $L_{Atenuacion}$ ) y un punto ( $PuntoX_{Atenuacion}$ ,  $PuntoY_{Atenuacion}$ ) relacionado con la meta particular del agente del Antebrazo:

$L_{Atenuacion}=556.670$   
 $PuntoX_{Atenuacion}=1200$   
 $PuntoY_{Atenuacion}=100$

*(Valores asignados en Matlab en relación al modelo implementado en Solidwork y en Animation).*

El agente del brazo al obtener la atenuación con la información propagada realiza inicialmente un cálculo de las posiciones propias alcanzables por medio de la EC. 7. En relación a la longitud (Radio) de su efector. Esto con el fin, de obtener los puntos factibles por el contenedor del agente del brazo.

---

<sup>22</sup> Si se desea observar otras pruebas realizadas para el modelo propuesto se encuentran anexas en el documento final.

$$\text{Radio} = L_{\text{Brazo}} = 946.886$$

Luego el agente del brazo determina los puntos de intersección. Estos puntos de intersección establecen las dos posibles metas particulares a las que debería desplazarse el contenedor del agente del brazo Fig. 16. En donde, el programa de Matlab muestra los puntos de intersección (En Color Rojo) y las regiones circulares para este caso particular en un brazo planar de dos grados de libertad. Los Puntos de intersección obtenidos son:

$$\begin{aligned} x_1 &= 807.4085 \\ y_1 &= 494.6561 \\ x_2 &= 878.1459 \\ y_2 &= -354.1932 \end{aligned}$$

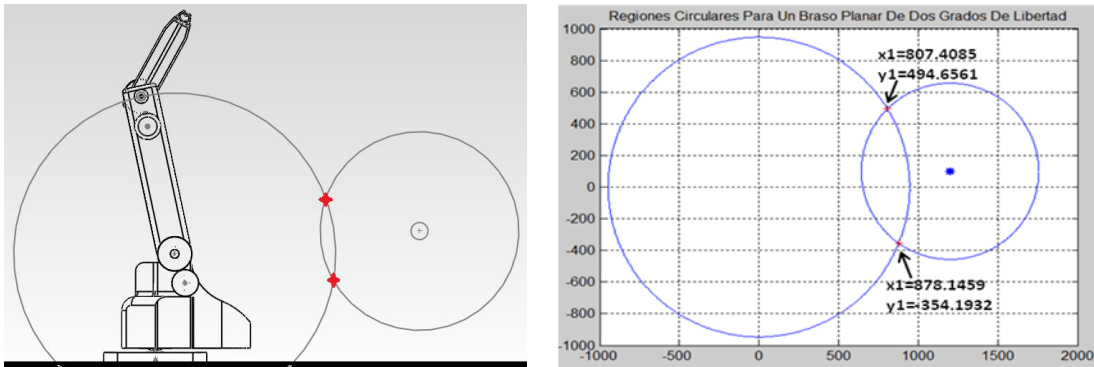


Figura 16 Circunferencias y Puntos de Intersección en el Brazo Planar de Dos Grados de Libertad

A continuación, el agente del brazo y el agente del Antebrazo realizan sus acciones, con el fin de desplazarse a sus propias metas particulares cumpliendo con la meta general.

### 5.1.2 Brazo Planar de Tres Grados de Libertad

Por medio de un código en Matlab se desarrollaron unos agentes que interactúan entre sí, estos agentes determinan independientemente sus metas particulares y realizan un comportamiento netamente individual. En donde, cada agente solo conoce una información limitada de su entorno, el ángulo de inclinación de su contenedor y la información propagada por sus vecinos directos Fig. 17.

Teniendo en cuenta que, de color azul muestran las acciones y movimientos desarrolladas por el link del brazo; de color rojo se muestran las acciones y movimientos desarrolladas por el link del antebrazo; de color verde se muestran las acciones y movimientos desarrolladas por el link de la mano.

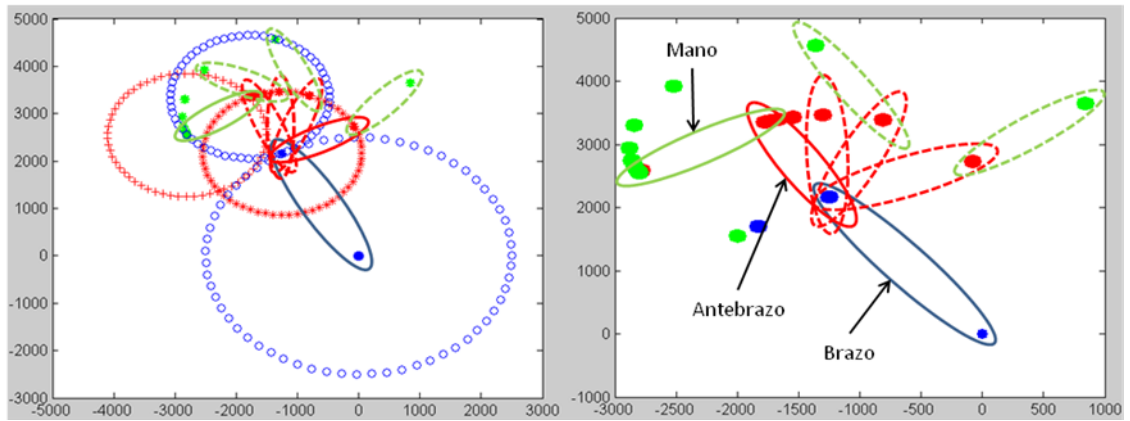


Figura 17 Resultados en Matlab del Movimiento Resultante de la Propagación por Medio de Regiones Circulares y su Respectiva Simulación

### 5.1.3 AMARANTA

Determina por medio de regiones circulares la posición alcanzable a la que se desplazará para intentar llegar al evento motivador Fig. 18. Luego, este agente propaga una información para que los agentes vecinos puedan determinar su meta particular alcanzable.

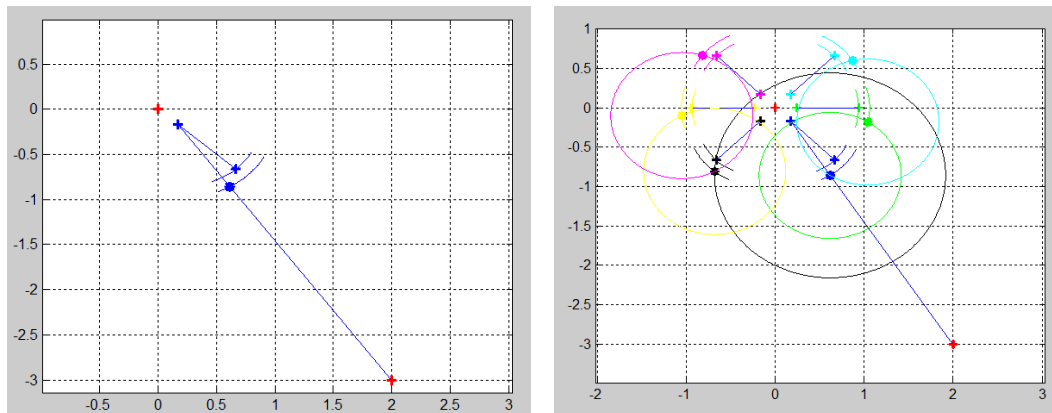


Figura 18 Comportamiento de un Agente Para Determinar la Posición Máxima Alcanzable Para Obtener el Evento Motivador

Los agentes vecinos que recibieron la propagación por medio de sus propios conocimientos calculan la meta particular a realizar. Esta meta particular depende de sus propios alcances y determina el punto de movimiento posible para esta subestructura. Consecutivamente se generan nuevas propagaciones, que los demás agentes faltantes por conocer el evento motivador reciben esta información propagada y calculan sus metas particulares Fig. 18.

## 5.2 Propagación de Agentes por Medio de Regiones Circulares y la Mínima Distancia de Desplazamiento

A continuación, se presentarán algunos ejemplos del modelo por medio de las plataformas en estudio. Para poder realizar una comparación de los dos modelos propuestos y su viabilidad en la implementación en una plataforma real.

### 5.2.1 Brazo Planar de Dos Grados de Libertad

En este caso particular, la propagación enviada se fundamenta en la longitud del antebrazo ( $L_{Atenuacion}$ ) y un punto ( $PuntoX_{Atenuacion}$ ,  $PuntoY_{Atenuacion}$ ) relacionado con la meta particular del agente del Antebrazo:

$$\begin{aligned} L_{Atenuacion} &= 556.670 \\ PuntoX_{Atenuacion} &= -50 \\ PuntoY_{Atenuacion} &= 500 \end{aligned}$$

*(Valores asignados en Matlab en relación al modelo implementado en Solidwork y en Animation).*

Luego, el agente del Brazo desarrolla el cálculo para determinar las posibles posiciones particulares a las que el contenedor del Brazo debe posicionarse. Estas posibles posiciones establecen la meta particular a las que debería desplazarse el contenedor del brazo. Los agentes calculan las distancias y determinan la meta particular más cercana para su realización Fig. 19.

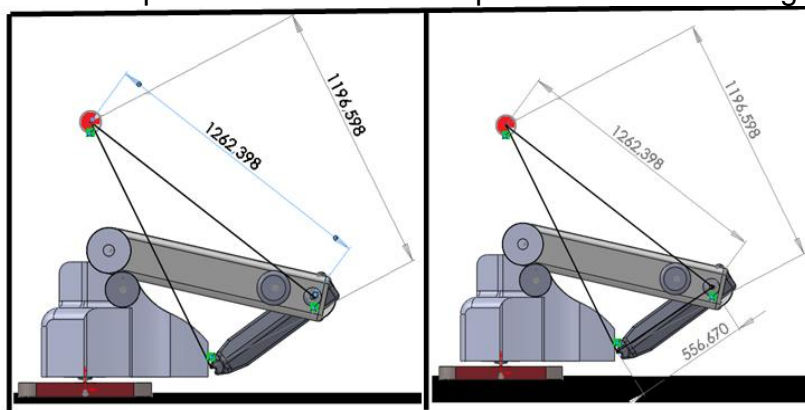


Figura 19 Cálculo de las Distancias del Brazo y el Antebrazo

A continuación los agentes realizan las acciones pertinentes por cada contenedor y consecuentemente avanzan, para lograr las metas particulares de cada uno de los contenedores hasta llegar a la meta general Fig. 20.



Cabe notar que cada agente debe conocer su propia posición para determinar estas distancias. Además, cada agente debe determinar autónomamente su propia posición por medio de recibir una propagación con la información de la posición de su vecino directo.

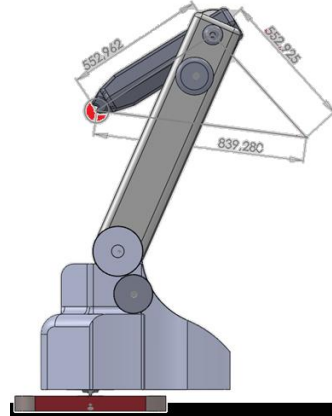


Figura 20 Obtención de la Meta General por Medio de una Cooperación entre Agentes

### 5.2.2 Brazo Planar de Tres Grados de Libertad

Por medio de un código en Matlab se desarrollo el modelo expuesto junto a tres agentes que interactúan entre sí, estos agentes determinan independientemente sus metas particulares y realizan un comportamiento netamente individual. En donde, cada agente solo conoce una información limitada de su entorno, el ángulo de inclinación de su contenedor y la información propagada por sus vecinos directos.

En la Fig. 21 se pueden observar las acciones de cada uno de los links de la plataforma robótica. En donde, de color verde se representa el link del Brazo; de color rojo representa el link del Antebrazo y de color azul representa el link de la Mano. Los puntos junto a su color representan algunas posiciones de movimientos de cada uno de los agentes de los links para alcanzar sus metas particulares.

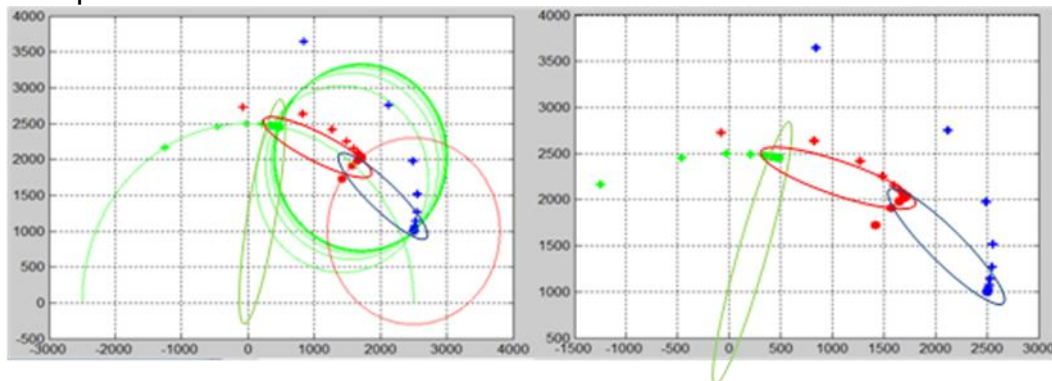


Figura 21 Resultados en Matlab del Movimiento Resultante de la Propagación por Parámetros de Distancia y su Respectiva Simulación

### 5.2.3 AMARANTA

La plataforma Amaranta para su desplazamiento general requiere determinar en forma distribuida cada una de las posiciones de desplazamiento particulares de cada pata del robot.

En las siguientes Fig. 22, se muestra las simulaciones por medio de Matlab del recorrido de la plataforma con tres patas en dirección a la meta general requerida. Cada pata está controlada por un agente independiente que realiza sus acciones y se desplazan cumpliendo con una coordinación para alcanzar las metas particulares y consecuentemente la meta general. En donde, se muestra en este método los puntos de secuencia de cada uno de los agentes determina como meta particular y que gradualmente al realizarse se van acercando a la meta general que es la posición final alcanzable de la plataforma.

Se debe Tener en cuenta que el color verde es la meta general y los puntos negros son las metas particulares que debe realizar cada uno de los agentes y que en secuencia al lograr cumplir con cada una de las metas particulares gradualmente se van acercando a la meta general.

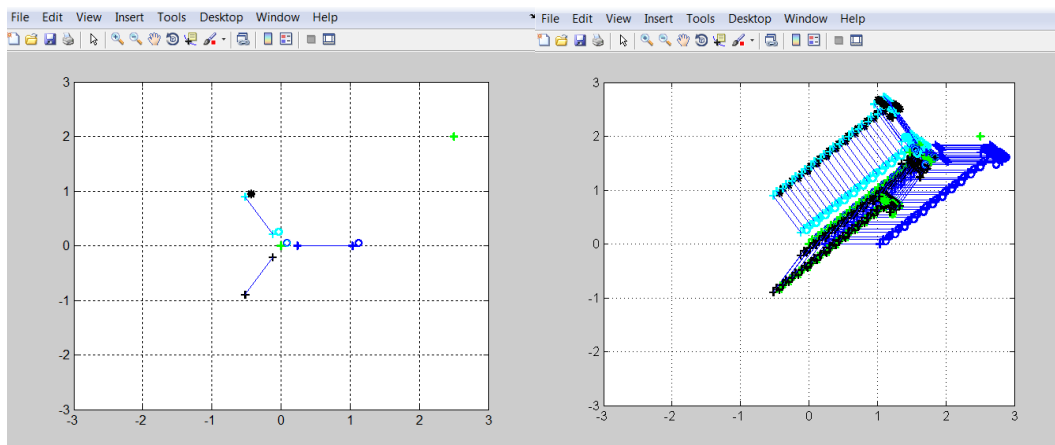


Figura 22 Plataforma Simulada AMARANTA en dos Dimensiones Punto Final (Color Verde)

### 5.3 Implementación Del Modelo Final De Análisis

El modelo implementado se realizó por medio de Matlab, Simulink y Animation que en este caso se desarrolló el programa de los multiagentes y se interconectó con una plataforma virtual para determinar su comportamiento. En este caso, el desarrollo fue gradual y se realizó una

plataforma virtual de 6 patas. Teniendo en cuenta que los multiagentes son homogéneos y las características de los contenedores son idénticas se tomo la decisión que el análisis sea con tres patas. Para profundizar en la plataforma AMARANTA, la arquitectura propuesta y su implementación en Simulink y Animation (Ver ANEXO 4).

Para realizar las pruebas de los modelos finales se realizaron múltiples pasos de diseño e implementación. Con el fin, de obtener resultados lo más cercanos a la realidad. En donde, se encuentra: anexo el desarrollo de la plataforma robótica para interconectar esta plataforma con el modelo propuesto en Simulink; anexo la estructura de simulación realizada en Matlab-Simulink (ANEXO 6); anexo la estructura de programación realizada de control distribuido e implementada en Matlab (ANEXO 5); y finalmente, anexo del modelo básico de control realizado para cada uno de los actuadores de las plataformas robóticas (ANEXO 5).

### 5.3.1 Plataforma Robótica

Las plataformas mostradas en este escrito se desarrollaron por medio de Solidwork<sup>23</sup>, Virtual-Real<sup>24</sup> y Simmechaniks<sup>25</sup>. Con la finalidad, de obtener resultados del modelo con características mecánicas y de movimientos rotacionales reales entre cada uno de los links representados en el sistema. En la Fig. 23 se presenta el modelo del brazo planar por medio de Simmechaniks, en donde, se representan las características físicas relevantes de la plataforma robótica. En la Fig. 24, se presenta el modelo obtenido en Virtual-Real basado en el modelo obtenido en Solidwork, para la conexión del modelo en Simulink y realizar la simulación junto a su animación en 3D.

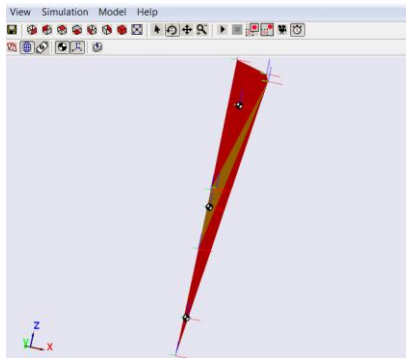


Figura 23 Modelo Obtenido en Simmechaniks con Centros de Gravedad y Características Físicas de la Plataforma Robótica

<sup>23</sup> Software de diseño mecánico [www.solidworks.com/](http://www.solidworks.com/)

<sup>24</sup> Software de diseño en 3D anexo a Simulink Animation <http://www.mathworks.com/products/3d-animation/>

<sup>25</sup> Software de diseño y simulación de elementos mecánicos con variables físicas en estudio.

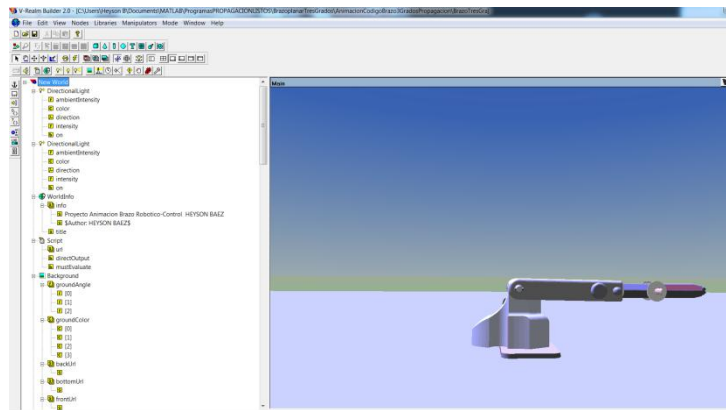


Figura 24 Modelo Obtenido en Virtual-Real para la Interacción del Modelo en 3D

### 5.3.2 Estructura Simulación

Una de las estructuras de simulación implementadas se encuentra en el ANEXO 7. En donde, esta estructura está fundamentada en el software de Matlab y Simulink, en esta estructura de simulación se representan los bloques de fricción, gravedad, programación, interconexión con el modelo virtual, entre otros aspectos, para observar el comportamiento de la plataforma robótica<sup>26</sup>.

### 5.3.3 Análisis Comparativo Experimental con los Dos Métodos Propuestos

Para el brazo planar de tres grados de libertad se implementaron dos métodos para el desarrollo de las acciones entre los agentes del prototipo robótico y consecuentemente cada uno de estos métodos produjo un comportamiento particular de la plataforma. Por lo tanto, se desarrollará un análisis que determine la funcionalidad, exactitud y energía gastada por cada uno de los métodos desarrollados.

Finalmente este análisis entre los dos métodos se desarrollará en función de una tabla comparativa que exponga los resultados de la funcionalidad de la plataforma y que aporte múltiples conclusiones de cada uno de los métodos desarrollados para su perfeccionamiento. Teniendo en cuenta que se mostrarán las tablas más relevantes del estudio y otras tablas se podrán observar el ANEXO 8.

En función de los datos obtenidos se procede a la caracterización de la respuesta temporal de la plataforma robótica. Esta caracterización del comportamiento de la plataforma se fundamenta en el análisis de los datos gráficos obtenidos de los dos métodos comparados. Para analizar los datos

<sup>26</sup> Se anexa los videos obtenidos de los comportamientos de la plataforma por medio de los diferentes métodos de control distribuido aplicando multiagentes y propagación.

gráficos se toman unas características de funcionamiento de la plataforma robótica. En donde, las características elegidas para su comparación son: energía, tiempo de oscilación, máximo sobre pico, tiempo de pico, tiempo de establecimiento y error final. Ver ANEXO 8 en donde se da una breve explicación de cada uno de los parámetros a calcular:

Tabla 4 Pruebas Obtenidas del Método de Distancias Cortas (Dificultad Baja y Media)

METODO DE DISTANCIAS CORTAS						
Exigencia o Complejidad			Dificultad Baja			Dificultad Media
ESCENARIOS			1 Escenario	2 Escenario	3 Escenario	4 Escenario
Pos. Inicial			5100-0	5100-0	5100-0	5100-0
Pos. Final			4410-2500	2050-2838	400-270	-4417-2550
Energía	E	J	1,29E-04	1,18E-04	3,21E-04	6,96E-04
Tiempo de Crecimiento	Tr	S	0,84	1,14	0,29	5
Máximo Sobrepico	Mp	%	32,92	4,79	37,4	0,0334
Tiempo de Pico	Tp	S	1,665	1,8	0,56	5,1
Tiempo de Establecimiento	te	S	14,2	2,68	4,68	5,3
error final	e	%	0	0,0135	0,048	0,28

Tabla 5 Pruebas Obtenidas Del Método de Distancias Cortas (Dificultad Alta y Media)

METODO DE DISTANCIAS CORTAS							
Exigencia o Complejidad			Dificultad Alta			Dificultad Media	
ESCENARIOS			1 Escenario	2 Escenario	3 Escenario	4 Escenario	5 Escenario
Pos. Inicial			5100-0	5100-0	5100-0	5100-0	5100-0
Pos. Final			-4700-100	-3000-2000	-4500--500	0-2250	-2140-1523
Energía	E	J	1,43E-03	7,41E-04	6,60E-04	2,38E-04	2,49E-04
Tiempo de Crecimiento	tr	S	4	3,7	3,2	2,42	2,84
Máximo Sobrepico	Mp	%	0	0	0	0,46	0
Tiempo de Pico	tp	S	0	0	0	2,6	0
Tiempo de Establecimiento	te	S	4,32	4	4	2,96	2,94
error final	e	%	0,072	0,014	0,0052	0,028	0,0195

Tabla 6 Pruebas Obtenidas del Método por Circunferencias (Dificultad Baja y Media)

METODO POR CIRCUNFERENCIAS					
Exigencia o Complejidad			Dificultad Baja		Dificultad Media
ESCENARIOS			1 Escenario	2 Escenario	3 Escenario
Pos. Inicial			5100-0	5100-0	5100-0
Pos. Final			4410-2500	2050-2838	400-270
					5100-0 -2140- 1523

<b>Energía</b>	<b>E</b>	J	6,36E-05	1,85E-04	3,66E-04	7,40E-04
<b>Tiempo de Crecimiento</b>	<b>tr</b>	S	0,88	0,93	0,55	3,52
<b>Máximo Sobrepico</b>	<b>Mp</b>	%	21,33	14	7,35	4,41
<b>Tiempo de Pico</b>	<b>tp</b>	S	1,44	1,18	3,31	4,05
<b>Tiempo de Establecimiento</b>	<b>te</b>	S	6	3,86	20	10
<b>error final</b>	<b>e</b>	%	3,17	0,13	3,23	2,45

Tabla 7 Pruebas Obtenidas del Método por Circunferencias (Dificultad Alta y Media)

METODO POR CIRCUNFERENCIAS							
Exigencia o Complejidad	Dificultad Alta			Dificultad Media			
ESCENARIOS			5 Escenario	6 Escenario	7 Escenario	8 Escenario	9 Escenario
Pos. Inicial Pos. Final			5100-0 -4700-100	5100-0 -3000-2000	5100-0 -4500--500	5100-0 -4417-2550	5100-0 0-2250
<b>Energía</b>	<b>E</b>	J	No Controlado	9,28E-04	No Controlado	1,10E-03	Inf
<b>Tiempo de Crecimiento</b>	<b>tr</b>	S	No Controlado	2,58	No Controlado	4,3	2
<b>Máximo Sobrepico</b>	<b>Mp</b>	%	No Controlado	4,04	No Controlado	0	Inf
<b>Tiempo de Pico</b>	<b>tp</b>	S	No Controlado	3,5	No Controlado	0	Inf
<b>Tiempo de Establecimiento</b>	<b>te</b>	S	No Controlado	8,35	No Controlado	5	Inf
<b>error final</b>	<b>e</b>	%	No Controlado	0,042	No Controlado	0,5	Inf

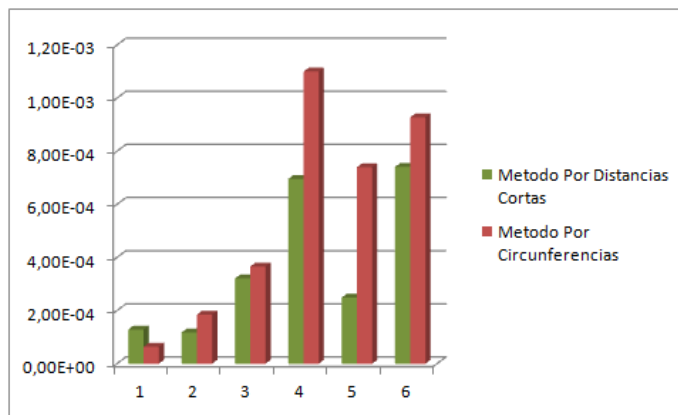


Figura 25 Comparación del Gasto de Energía entre el Método por Distancias Cortas y el Método por Circunferencias

Consecuentemente, se realiza una comparación de los dos métodos. La Fig. 25 se observa el comportamiento del gasto de energía que realiza cada uno de los dos métodos, en donde, el método por distancias cortas en la mayoría de pruebas gasto menos energía en relación al método por circunferencias. Concluyendo que el método por circunferencias requiere mayor energía en la

mayoría de los casos. Esto quiere decir, que requiere una mayor cantidad de movimientos para lograr llegar a la meta general deseada.

Por otra parte, se realiza la comparación de los dos métodos en relación al tiempo de crecimiento. Observando que este tiempo de establecimiento es muy similar a en los dos modelos pero en promedio el método por circunferencias es un poco más rápido que el método por distancias cortas tabla 8.

Tabla 8 Promedio del Tiempo de Respuesta de Cada uno de los Dos Métodos

	Promedio tiempo de Crecimiento
<b>Método Por Distancias Cortas</b>	2,32 segundos
<b>Método Por Circunferencias</b>	2,1 segundos

En la Fig. 26 se realiza la comparación de los dos métodos en relación al porcentaje del sobreimpulso máximo. En este caso, se observa que el sobreimpulso en el método de distancias cortas es mayor al del método por circunferencias pero en metas de dificultad alta el método por cercanía no tiene sobreimpulso tanto que en el método de cercanías no puede llegar a esas metas específicamente o si presenta un sobreimpulso.

Por otra parte, el tiempo de oscilación y el máximo sobre pico en la mayoría de casos presenta menores valores el método de circunferencias. Generando que los tiempos de establecimiento fueran más rápidos y acarreado un menor porcentaje del error final en el método por cercanías.

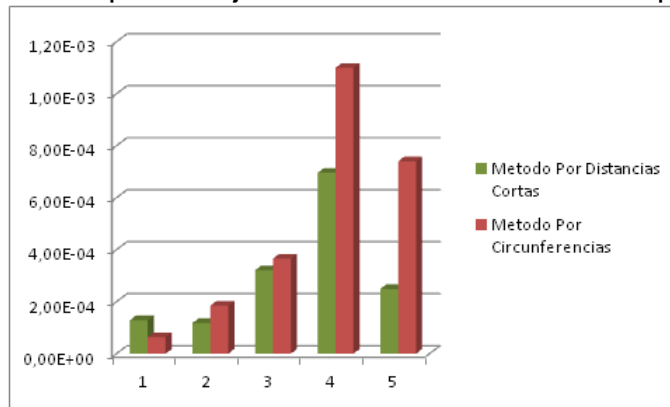


Figura 26 Comparación del Sobreimpulso Máximo entre el Método por Distancias Cortas y el Método por Circunferencias

Finalmente, en la Fig. 27 se compara el error de cada modelo propuesto, determinando que el error del método por cercanías cortas es mucho menor, logrando en su mayoría de casos llegar a la meta sin dificultades.

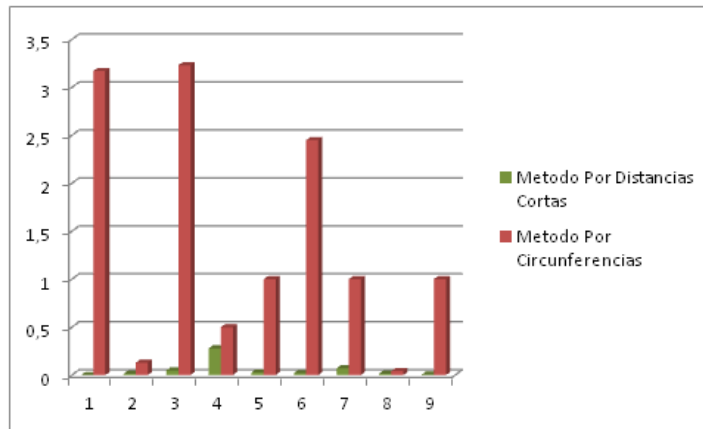


Figura 27 Comparación del Error de Estado Estacionario Entre el Método por Distancias Cortas y el Método por Circunferencias

Como conclusión, para metas de alta dificultad el método por circunferencias no es muy favorable por su inestabilidad. Además, el modelo de circunferencia por cercanía mínima tiene mayor velocidad de estabilización en relación al modelo por circunferencias. Además, el movimiento por cercanías tiene mayor exactitud, como también, su tiempo de oscilación es menor. Esto quiere decir, que el método que utilizará para este proyecto y para la validación del modelo es el método por cercanías.

#### ✓ Análisis De Sensibilidad

El análisis de sensibilidad es una de las partes fundamentales en el comportamiento de la plataforma y su toma de decisiones, puesto que permite determinar cuándo una solución sigue siendo óptima, dados algunos cambios de la posición inicial o de la posición final requerida. Por lo anterior, un objetivo fundamental del análisis de sensibilidad es identificar los cambios repentinos en el comportamiento de la plataforma y sus acciones distribuidas tanto en su control como entre los Multiagentes. Además, establecer intervalos en los cuales el comportamiento sigue siendo estable.

Además, la variación en el punto inicial o en la meta se analizarán independientemente, es decir, se analiza la sensibilidad del comportamiento debido a la modificación de un parámetro que en este caso es la posición inicial, asumiendo que los demás parámetros permanecen sin ser alterados. Esto es importante para determinar que el análisis de la sensibilidad es de forma estática y no dinámica, pues solo contempla el cambio de un parámetro a la vez y no el de varios parámetros.

En este caso se fija un punto final que es:



$$X=-3500, Y= 1000$$

Luego se establecen nueve puntos de inicio de la plataforma:

Tabla 9 Puntos Iniciales de la Plataforma

Punto Inicial (X=3750;Y=3100)	Punto Inicial (X=3650;Y=3000)	Punto Inicial (X=3600;Y=3000)	Punto Inicial (X=3550;Y=2950)
Punto Inicial (X=3500;Y=2900)	Punto Inicial (X=3400;Y=2800)	Punto Inicial (X=3300;Y=2700)	

En donde, por medio de la figura 28 se observa que el comportamiento de la plataforma no varía aunque existan cambios graduales en el punto inicial. Además, el tiempo de respuesta, tiempo de establecimiento y error de estado estacionario entre otros no varía en su totalidad.

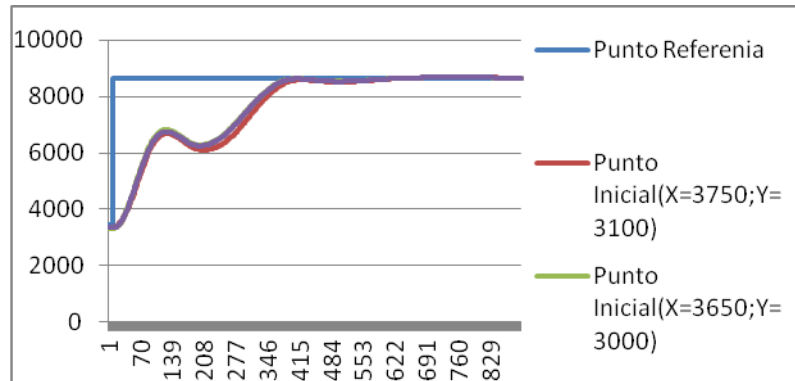


Figura 28 Comportamiento de la Plataforma Variando el Punto Inicial y Dejando un Punto Fijo de Referencia

En este caso se fija el punto inicial y se variará el punto de meta con lo que se podrá determinar si existen cambios en el comportamiento o variaciones en la movilidad de la plataforma.

Se fija un punto inicial que es:

$$X=3600, Y= 3000$$

Luego se establecen varios Metas de la plataforma:

Tabla 10 Puntos Finales de la Plataforma

Punto Final (-3500;1000)	Punto Final (-3400;900)	Punto Final (-3300;1100)	Punto Final (-3200;1000)
-----------------------------	----------------------------	-----------------------------	-----------------------------

En donde, por medio de la Figura 29 se observa que el comportamiento de la plataforma no varía aunque existan cambios graduales en la meta a

completar. Además, el tiempo de respuesta, tiempo de establecimiento y error de estado estacionario entre otros no varía en su totalidad.

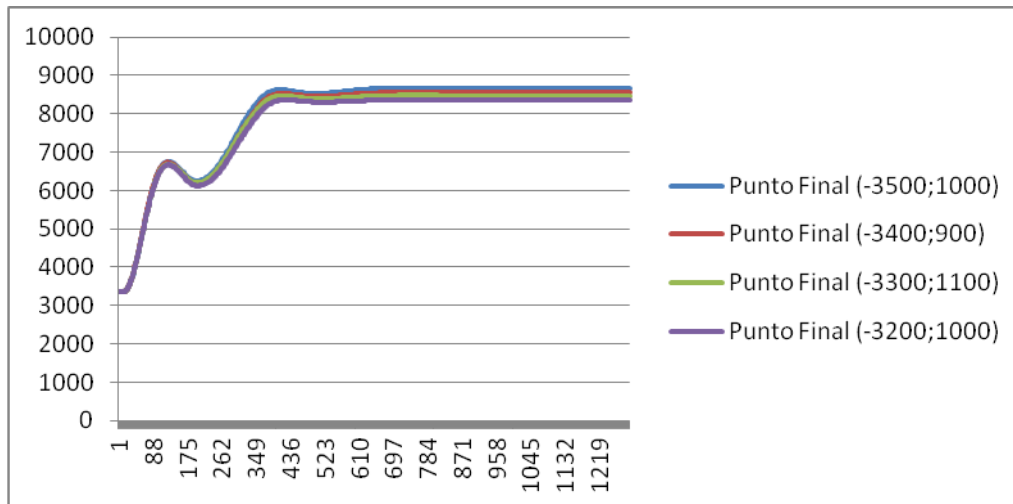


Figura 29 Comportamiento de la Plataforma Variando la Meta Final y Dejando un Punto Fijo de Inicio

Esto quiere decir, que el comportamiento entre agentes y en general el comportamiento de la plataforma no es sensible a los posibles cambios que se puedan presentar en la plataforma. Por lo tanto esta implementación es positiva, tanto en funcionamiento como en la estabilidad del comportamiento de la plataforma y se puede demostrar que la plataforma es determinista por qué no varía en sus acciones ni en sus respuestas.

Finalmente, el modelo por circunferencia se validó en el Brazo Planar De Tres Grados De Libertad y en AMARANTA. Mostrando algunos datos presentes en las plataformas y por medio de las simulaciones y animaciones. En el ANEXO se expone el comportamiento de las plataformas; como también la funcionalidad del modelo propuesto para estas plataformas. Finalmente, se anexan las simulaciones y animaciones desarrolladas en este estudio Ver ANEXO 6.

## 6. CONCLUSIONES

Se obtuvo pautas para caracterizar modelos robóticos modulares, reconfigurables dinámicamente, tolerantes a fallos y con una configuración válida de ensamblaje.

Se expuso que la propagación es el método propuesto por este estudio para solventar las dificultades presentes en la comunicación y coordinación de un sistema robótico distribuido y asincrónico.

Se propuso un nuevo método de cinemática basado en circunferencias para la distribución de metas particulares en un sistema distribuido de multiprocesadores y multiagentes independientes.

Se presentaron nuevas propuestas de distribución de metas particulares para un sistema distribuido con su respectivo análisis y comparación. En donde, se pudo corroborar la funcionalidad del método de circunferencias para la distribución de las metas en la plataforma robótica.

Se logró realizar movimientos coordinados de las diferentes plataformas desarrolladas y simuladas en esta investigación.

Se comprendió que un control distribuido es robusto, escalable y flexible en los múltiples prototipos implementados con Multi-agentes.

Se analizó que los sistemas robóticos distribuidos pueden presentar colisiones con elementos del entorno, lo cual, requiere que exista una secuencia de reglas para eliminar o limitar estas colisiones producidas.

Se manifestó que las arquitecturas de control distribuido requieren la distribución de la carga computacional entre muchos procesadores que funcionan en tiempo real.

Se demostró que un sistema de control distribuido es suficientemente robusto para llevar a cabo de forma autónoma metas relativamente complejas realizando metas particulares que conllevan a la meta general.

Los resultados demuestran que el control distribuido usa menos recursos del sistema, debido a la segmentación de las metas complejas teniendo una ejecución positiva y satisface las peticiones de los múltiples estudios robóticos implementados.

Se aprendió que la distribución de metas generales en metas particulares produce mayor velocidad tanto en tiempo de procesamiento, como en el cumplimiento de las metas generales.

Se expuso la posibilidad de implementar en múltiples plataformas robóticas el modelo propuesto en esta investigación. Debido, a que el modelo funciono correctamente en una plataforma fija (brazo planar) y en una plataforma móvil (AMARANTA).

Se mostró que la implementación de múltiples procesadores requiere la asignación de agentes que desarrollen metas o comportamientos internos en el sistema, independientes entre sí.

Se conoció que cada agente puede ser redundante, flexible, reconfigurable y autónomo, pero con unas características fundamentales como son: los métodos de comunicación, las reglas de cooperación, las reglas de coordinación y las reglas de resolución de conflictos.

Se concluyo que la interacción de los agentes debe tener un conocimiento con el medio que lo rodea y un conocimiento de sus vecinos (agentes) más cercanos.

Se expuso que para la resolución de conflictos, los agentes del control distribuido deben tener una comunicación directa entre vecinos (punto a punto).

Se tiene en cuenta que para la implementación del modelo que los contenedores, los métodos de comunicación y la heterogeneidad de los agentes, depende de los requerimientos del sistema robótico en general.

Se comprendió que los agente deben tener un evento motivador que interactúe para el desarrollo de una meta, tanto individual como de cooperación.

Se concluyo que la comunicación es un elemento importante para la cooperación, de cualquier sistema robótico y por lo tanto debe existir.

Se realizo una comparación de un sistema centralizado básico con el modelo propuesto distribuido logrando demostrar que el modelo propuesto es funcional y puede realizar acciones coordinadas para obtener las metas generales.

Se comprendió que la interacción entre los agentes busca una arquitectura de comunicación (punto a punto). En donde, esta arquitectura punto a punto al implementarse genera un método de coordinación entre agentes basado en la propagación de eventos ocurridos que afectan a la plataforma robótica.

Se debe tener en cuenta sin importar la estructura fija o variable, que los vecinos directos son los agentes que se encuentran a una menor proximidad (cercanos) o con su inter-localidad física entre los agentes.

Se dio a conocer que el control distribuido para multiprocesadores es una herramienta para solventar elevadas dificultades presentes en la robótica actual generando la posibilidad de escalabilidad y disminución de procesamiento en múltiples plataformas.

## BIBLIOGRAFÍA

- [ANTH2002].** M. Anthony, LEWIS and George, A. BEKEY. Gait Adaptation in a Quadruped Robot, Autonomous Robots, Department of Computer Science, University of Southern California, Los Angeles, Manufactured in The Netherlands, 2002.
- [ARND2007].** Arndt, Lüder y Jörn, Peschke. Incremental design of distributed control systems using GAIA-UML. with the Center Distributed Systems at Institut for Ergonomics, Manufacturing Systems, and Automation at Otto-von-Guericke University Magdeburg, Universitätsplatz. Magdeburg, Germany. 2007.
- [ARTH2001].** Woo, Ho, LEE and Arthur, C., SANDERSON. Dynamic Analysis and Distributed Control of the Tetrobot Modular Reconfigurable Robotic System, Autonomous Robots, Department Of Mechanical Engineering, Aeronautical Engineering, Rensselaer Polytechnic Institute, Troy, 2001.
- [BAYI2007].** Levent, BAYINDIR and Erol, SAHIN. A Review of Studies in Swarm Robotics, Turk J Elec Engin, Kovan Research Lab., Dept. of Computer, Eng. Middle East Technical University, Ankara-TURKEY, VOL.15, 2007.
- [CAPE2003].** J. V. CAPELLA; A. BONASTRE and R. ORS. An Advanced and Distributed Control Architecture Based on Intelligent Agents and Neural Networks, IEEE, International Workshop on Intelligent, Polytechnic University of Valencia, Camino' de Vera, September 8-10, 2003,
- [CARL2009].** Carlos, SANTACRUZ; Snaider, CARRILLO; Diego, BOTERO; Carlos, PARRA; Alejandro, FORERO; Alvaro, HILARIÓN and Martha MANRIQUE. AMARANTA: Modular Platform for a Mine Hunting Robot, Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS-CNRS), France y Pontificia Universidad Javeriana, Bogotá, Colombia, 2009.
- [CHRI2008].** Christian, OTT and Yoshihiko, NAKAMURA. Employing wave variables for coordinated control of robots with distributed control architecture, IEEE, International Conference on Robotics and Automation Pasadena, Department of Mechano-Informatics, University of Tokyo, May 19-23, 2008.
- [CHUN2008].** Chun, WANG; Hamada, GHENNIWA and Weiming, SHEN. Real time distributed shop floor scheduling using an agent-based service-oriented architecture, Department of Electrical and Computer Engineering, the University of Western Ontario, Canada, May 01, 2008.
- [ENRI2005].** Enrique, GONZÁLEZ; Jamir, AVILA and César, BUSTACARA; BESA: Behavior-oriented, Event-driven and Social-based Agent Framework, Pontificia Universidad Javeriana. Facultad de Ingeniería Grupos de Investigación SIDRE-SIRP. Bogotá, Colombia, 2005.
- [ENRI2007].** Enrique, GONZÁLEZ y César, BUSTACARA. Desarrollo de aplicaciones basadas en sistemas multiagentes, Pontificia Universidad Javeriana, Bogotá, Colombia, Febrero 2007.

- [ESBE2003].** Esben, HALLUNDBREK and Henrik, HAUTOP. Evolving Control for Modular Robotic Units, IEEE, international Symposium on Computational Intelligence in Robotics and Automation, University of Southern Denmark, July 16-20, 2004.
- [GALD2008].** J. GALDUN, L. TAKÁ, J. LIGUS, J.M. THIRIET and J. SARNOVSKÝ. Distributed Control Systems Reliability: Consideration of Multi-agent Behavior, IEEE, Department of Cybernetics and Artificial Intelligence, Košice, Slovakia, 2008.
- [GORD2004].** Gordon, WYETH and Damien, KEE. Distributed Control of Gait for a Humanoid Robot, School of Information Technology and Electrical Engineering University of Queensland, St. Lucia, Queensland, Australia. 2004.
- [JARV2008].** Ray, JARVIS. Intelligent Robotics: Past, Present And Future, Technomathematics, Director, Intelligent Robotics Research Centre, Monash University, Australia, Vol. 5, No. 3, pp 23 – 35, 2008.
- [JUAN2007].** Juan, A. ESCALERA; Manuel FERRE; Rafael, ARACIL and José, BACA. ROBMAT: Teleoperation of a Modular Robot for Collaborative Manipulation, Springer - Verlag Berlin Heidelberg, Dpto. Automática, Ing. Electrónica e Industrial, Universidad Politecnica de Madrid, 2007.
- [KHOS1992].** Khosrow C. HADAVI. A Hybrid Architecture for Distributed Autonomous Control Agents, May 01, 1992.
- [MANE2005].** Manuel, VELASCO; Pau, MARTÍ and Manel, FRIGOLA. Bandwidth Management for Distributed Control of Highly Articulated Robots, IEEE, International Conference on Robotics and Automation, Barcelona, Spain, April, 2005.
- [MARC2007].** Marco, MAMEI and Franco, ZAMBONELLI. Programming Modular Robots with the TOTA Middleware, ESOA, Springer – Verlag, Dipartimento di Scienze e Metodi dell'Ingegneria, University of Modena and Reggio Emilia, Berlin Heidelberg, 2007.
- [MARK2006].** Mark, MAIMONE; Andrew, JOHNSON; Yang, CHENG; Reg, WILLSON and Larry, MATTHIES. Autonomous Navigation Results from the Mars Exploration Rover (MER) Mission, Springer - Verlag, Experimental Robotics, Jet Propulsion Laboratory, California Institute of Technology Oak Grove Drive, Pasadena, USA, 2006.
- [MICH2005].** Michele, FOLGHERAITER and Giuseppina, GINI. MaximumOne: An Anthropomorphic Arm with Bio-inspired Control System, DEI Department, Politecnico di Milano, piazza Leonardo da Vinci, Italy, 2005.
- [MINS1988].** Minsky, Marvin. The Society of Mind, Simon and Schuster, Nueva York. 15 de marzo 1988.
- [PETR2010].** A. M. PETRINA. Advances in Robotics, Allerton Press, Scientific and Technical Information Institute, Russian Academy of Sciences, Moscow, Received December 24, 2010.
- [PING1994].** Kai, JIN; Ping, LIANG and Gerardo, BENI. Stability of Synchronized Distributed Control of Discrete Swarm Structures, IEEE, 1994.

- [QIXI2007].** Cao, QIXIN; Zhang, ZHEN and GU, JIAJUN. A Distributed Control and Simulation System for Dual Arm Mobile Robot, IEEE, International Symposium on Computational Intelligence in Robotics and Automation Jacksonville, FL, USA, June 20-23, 2007.
- [ROBE2002].** Robert, W. BRENNAN; Martyn, FLETCHER and Douglas, H. NORRIE. An Agent-Based Approach to Reconfiguration of Real-Time Distributed Control Systems, IEEE, transactions on robotics and automation, vol. 18, no. 4, august, 2002.
- [SANT2009].** Carlos Santacruz, «AMARANTA: Modular Platform for a Mine Hunting Robot,» *IEEE*, 2009.
- [SCHA2010].** Christian, SCHACK; Wolfgang, HEENES y Rolf HOFFMANN. Multiprocessor Architectures Specialized for Multi-Agent Simulation, Technische Universit at Darmstadt, FB Informatik, FG Rechnerarchitektur, Darmstadt, Germany, 2010.
- [SHEN2006].** Wei-Min, SHEN; Maks, KRIVOKON and Harris, CHIU, Multimode locomotion via SuperBot reconfigurable robots, Springer Science, Information Sciences Institute and Computer Science Department, University of Southern California, Published online: 22 April, 2006.
- [SHIG2003].** Shigeo, HIROSE and Edwardo, F. FUKUSHIMA. Snakes and Strings: New Robotic Components for Rescue Operations, Tokyo Institute of Technology, Okayama, Japan, 2003.
- [STEP2003].** Stephen, J. DERBY and David, BROWN. Distributed control of a track based multi-head robot, *Industrial Robot: An International Journal*. Volume 30, Number 3, 2003.
- [TERR2004].** Terry L, HUNTSBERGER; Ashitey, TREBI-OLLENNU; Hrand, AGHAZARIAN and Paul, S, SCHENKER. Distributed control of multi-robot systems engaged in tightly coupled tasks autonomous robots, Kluwer academic publishers, Manufactured in the Netherlands, 2004.
- [THOM2008].** Thomas, SCHMICKL; Ronald, THENIUS; Christoph, MOESLINGER and Gerald, RADSPIELER. Get in touch: cooperative decision making based on robot-to-robot collisions, *Auton Agent Multi-Agent Syst*, Springer Science, 2008.
- [WILL2003].** William, BLUETHMANN; Robert, AMBROSE and Myron, DIFTLER. Robonaut: A Robot Designed to Work with Humans in Space, *Autonomous Robots*, This work is sponsored by NASA and the Mobile Autonomous Robot Software (MARS) program in the DARPA Information Processing Technology Office (IPTO), Manufactured in The Netherlands. 2003.
- [YASU2006].** G, YASUDA. Implementation of Distributed Control Architecture for Robotic Manufacturing Systems using Petri Nets, *IMACS Multiconference on Computational Engineering in Systems Applications (CESA)*, Nagasaki Institute of Applied Science, Beijing, China, October 4-6, 2006.



**[ZACK2003].** Zack, BUTLER; Robert, FITCH and Daniela, RUS. Experiments in Distributed Control for Modular Robots, Springer-Verlag, Department of Computer Science Dartmouth College, Hanover, USA, 2003.

**[ZACH2008].** Zachary, KULIS; Vikram, MANIKONDA; Babak, AZIMI-SADJADI and Priya, RANJAN. The Distributed Control Framework: A Software Infrastructure for Agent-based Distributed Control and Robotics, American Control Conference Westin Seattle Hotel, Seattle, Washington, USA, June 11-13, 2008.