



Revista UIS Ingenierías

ISSN: 1657-4583

Universidad Industrial de Santander

Gil, Ángel; Aguilar, José; Rivas, Rafael; Dapena, Eladio  
Coordinación emergente en sistemas multirobots  
Revista UIS Ingenierías, vol. 18, núm. 3, 2019, pp. 75-86  
Universidad Industrial de Santander

DOI: <https://doi.org/10.18273/revuin.v18n3-2019008>

Disponible en: <https://www.redalyc.org/articulo.oa?id=553762534009>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

UNEN  
redalyc.org

Sistema de Información Científica Redalyc  
Red de Revistas Científicas de América Latina y el Caribe, España y Portugal  
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso  
abierto

## Coordinación emergente en sistemas multirobots Emergent coordination in multi-robot systems

Ángel Gil<sup>1,2</sup>, José Aguilar<sup>2,3a</sup>, Rafael Rivas<sup>3b</sup>, Eladio Dapena<sup>3c</sup>

<sup>1</sup>Laboratorio de Prototipos, Universidad Nacional Experimental del Táchira, Venezuela.

Orcid: 0000-0002-4150-5246. Correo electrónico: [agil@unet.edu.ve](mailto:agil@unet.edu.ve)

<sup>2</sup>Tepuy R+D Group, Artificial Intelligence Software Development.

<sup>3</sup>Universidad de los Andes, Venezuela. Orcid: <sup>a</sup> 0000-0003-4194-6882,

<sup>b</sup> 0000-0001-6103-822X, <sup>c</sup> 0000-0002-9135-0967

Correos electrónicos: <sup>a</sup> [aguilar@ula.ve](mailto:aguilar@ula.ve), <sup>b</sup> [rafael@ula.ve](mailto:rafael@ula.ve), <sup>c</sup> [eladio@ula.ve](mailto:eladio@ula.ve)

Recibido: 22 de junio, 2018. Aceptado: 18 de noviembre, 2018. Versión final: 11 de abril, 2019.

### Resumen

En la naturaleza se encuentran sociedades de seres vivos que coordinan sus acciones de forma no centralizada. Por ejemplo, en las colonias de hormigas ocurren procesos emergentes, que combinan las acciones de los individuos en función de un objetivo común. En este trabajo, se describe una capa de gestión que facilita los procesos de coordinación emergente en los sistemas multirobots. Esta capa en particular permite la aparición de la emergencia y la autoorganización en el sistema. En conjunto con las capas de gestión individual y de gestión del conocimiento, manejan los procesos necesarios para el funcionamiento del sistema multirobot.

**Palabras clave:** arquitectura; coordinación; emergencia; auto-organización; sistemas multirobots.

### Abstract

In nature there are societies of living beings that coordinate their actions in a non-centralized way. For example, emergent processes that combine the actions of individuals in the achievement of a common goal occur in ant colonies. This paper describes a management layer that facilitates emerging coordination processes in multi-robot systems, which together with other two layers (one of individual management and another of knowledge management), manage the processes necessary for the operation of the multi-robot system. In particular, this layer allows the appearance of emergency and self-organization in the system.

**Keywords:** architecture; coordination; emergency; self-organization; multi-robot systems.

### 1. Introducción

En los sistemas multirobot se llevan a cabo distintos procesos que van desde la gestión de los individuos que conforman el grupo, hasta la gestión de los procesos colectivos que ocurren en el sistema. Asuntos como los algoritmos de cooperación, las tecnologías de percepción, la emergencia, la autoorganización y las arquitecturas de control, entre otros aspectos relacionados con la naturaleza de estos sistemas, son de particular interés para los investigadores de esta área [1]. La coordinación del

grupo al momento de llevar a cabo las tareas es fundamental.

Al respecto, se han realizado múltiples investigaciones. A continuación, se comentan algunas de ellas. Por ejemplo, en [1] se proponen dos métodos de coordinación para sistemas multirobots en dos casos específicos: la recolección y la formación. En el primer caso, la coordinación se basa en un esquema de enrutamiento; en el segundo, se modifica la propuesta de [2], que se basa en la medición de las posiciones relativas de los vecinos

en la formación y en el seguimiento de líderes, y se incluye un componente para eliminar errores en la formación. En [3], se desarrolla un algoritmo evolutivo para sistemas multiagentes, con el objetivo de generar políticas de control para misiones de exploración dinámica y estocástica. En [4], se propone un modelo de coordinación basado en grafos difusos, aplicable a contextos con incertidumbre en el caso de sistemas compuestos por múltiples vehículos aéreos no tripulados. En [5], se presenta una modificación del algoritmo A\*, para ser utilizado en la coordinación de un sistema multirobot que ejecuta la tarea de transporte: se adiciona un componente de tiempo al algoritmo, que es importante para la predicción de la posición de los vehículos que actúan en el sistema. Finalmente, en [6] se propone una arquitectura de coordinación distribuida para un sistema multirobot, donde los individuos tienen capacidades parcialmente solapadas.

En ese trabajo, se consideran los problemas de autoconfiguración y autooptimización. La configuración del sistema se lleva a cabo mediante descripciones semánticas, lo que permite el emparejamiento semántico de los robots para dar respuesta a las solicitudes de ensamblaje. Esto hace posible seleccionar las mejores configuraciones, minimizando así los costos totales de ensamblaje y los tiempos en que el sistema está fuera de línea. En [7], se propone una solución para la utilización de sistemas multirobots en la resolución del problema del viajero. Puesto que este problema es del tipo NP-Hard, las soluciones tradicionales no son adecuadas para aplicaciones robóticas. En este trabajo, se propone una solución basada en mover-mejorar, en donde un robot se mueve e intenta mejorar su solución mediante la coordinación con sus vecinos. Este proceso se realiza en cuatro etapas: asignación de objetivos, construcción de *tours*, negociación de objetivos conflictivos y mejora de las soluciones. En [8], se presenta una técnica de navegación distribuida para enjambres de robots móviles, donde cada individuo posee un controlador basado en comportamientos básicos, mediante la implementación de un autómata híbrido. La composición paralela de los autómatas individuales permite la coordinación de los individuos para generar un comportamiento de bandada en el grupo.

Por otro lado, en [9] se introduce un algoritmo de planificación priorizado, descentralizado y asíncrono para la planificación de movimientos en un sistema multirobot. Además, se muestra su aplicabilidad como método de coordinación en el contexto del plan para una misión compleja de un equipo de vehículos autónomos aéreos no tripulados. En [10], se presenta un modelo de coordinación para sistemas multirobots basado en comportamientos cooperativos. Para ello, se generan

comportamientos básicos o de bajo nivel, utilizando redes neurales y algoritmos evolutivos. Las pruebas se llevaron a cabo en dos escenarios, y en ambos se utilizaron dos robots. El objetivo de ambos escenarios era alcanzar el acoplamiento de los robots. En el primero, los comportamientos se implementan en solo un robot (activo), mientras el otro robot no los utiliza (inactivo). En el segundo escenario, ambos robots se consideran activos. Lo mejores resultados corresponden al segundo escenario.

En [11], se propone una arquitectura para el control de sistemas multirobots con comportamiento emergente (AMEB), que está estructurada en tres capas: individual, gestión de conocimiento y colectiva. Esta última maneja los procesos relacionados con la coordinación del grupo desde el enfoque emergente. En este trabajo se presenta el diseño de esta capa y algunas consideraciones sobre su implementación en un sistema multirobot, compuesto por robots de propósito general [12].

Este artículo está organizado de la siguiente manera. En la sección 2, se enseñan los conceptos base de un modelo de coordinación emergente. A continuación, en la sección 3, se presenta la arquitectura AMEB. En la sección 4, exponemos nuestra propuesta de coordinación emergente para AMEB, para después presentar un caso de estudio y realizar comparaciones con otros trabajos. Finalmente, la última sección es una síntesis de las conclusiones.

## 2. Modelos de coordinación emergente

La *emergencia* puede definirse como la propiedad de un sistema que presenta comportamientos que no pueden predecirse con anterioridad, pero que son observables en el plano global [13]. Estos comportamientos surgen de la interacción entre los componentes a nivel particular. El comportamiento observado no puede explicarse tomando en consideración únicamente el comportamiento de los individuos.

Por otro lado, se habla de *coordinación emergente* cuando los individuos del sistema tienen comportamientos colectivos sin necesidad de un ente coordinador: ocurre a partir de la interacción entre los individuos [13]. Las sociedades de insectos son un ejemplo de sistemas donde la coordinación emergente aparece de forma natural. Por esta razón, han servido de base para la construcción de modelos artificiales para la resolución de problemas complejos. En estas sociedades, los individuos coordinan sus actividades a través de una forma indirecta de comunicación denominada estigmergia, que está mediada por modificaciones en el medio ambiente [13]. A continuación, se describen algunos de estos sistemas y sus modelos de coordinación emergente.

### 2.1.1. Modelo de colonia de hormigas

Las colonias de hormigas son sociedades naturales muy organizadas. Son capaces de tomar decisiones colectivas sin que exista un líder ni acceso a información centralizada. Las hormigas, sin tener un sentido del conjunto, emplean variables estigméricas para solucionar problemas.: Ejemplos de este comportamiento son: la exploración de entornos, la gestión de los desechos en su colonia y búsqueda de alimento, entre otros. Los comportamientos de las colonias de hormigas se han modelado como sistemas no lineales, donde los agentes actúan de forma estocástica y responden solo a la información de su entorno cercano, siguiendo reglas de decisión simples que están vinculadas a lazos de retroalimentación. En general, esos modelos generan comportamientos robustos que resuelven problemas complejos de forma colectiva y eficiente a partir de lo siguiente [13]:

- a. Generan una variedad de patrones sociales con un limitado número de reglas de comportamiento y de condiciones fisiológicas.
- b. Son multiestacionarios, y los cambios cualitativos discretos emergen, modificando gradualmente sus parámetros.
- c. Usan reglas de decisión simples.
- d. Alteran sus patrones de interacción en función del ambiente, de tal modo que emergen estructuras colectivas.

Las hormigas, al igual que otras especies de insectos, cooperan para recuperar o mover alimentos u objetos. Las hormigas, en particular, son capaces de mover colectivamente grandes objetos, imposibles de mover de forma individual. El proceso comienza cuando una hormiga encuentra un objeto e intenta moverlo sola: si lo logra, lo lleva al nido, si falla, recluta a otras hormigas mediante trazas en el ambiente (feromonas) o contacto directo. Si luego de un tiempo el grupo no logra su objetivo, en algunas especies se recluta a hormigas con características especiales [13, 14]. Para el transporte colectivo, se utiliza un método que consiste en el realineamiento y reposicionamiento, es decir, que, si una hormiga no logra mover un objeto, realinea su posición y vuelve a intentarlo. En ocasiones, es suficiente modificar la posición de la aplicación de la fuerza. Si el realineamiento no basta, se reposiciona. Si después de reposicionarse en varias oportunidades no se logra el objetivo, se recluta a otras hormigas.

El proceso de reclutamiento puede llevarse a cabo de dos formas: la primera, dejando una traza en el aire de la zona

donde se encuentra el objeto, con el fin de atraer a individuos rápidamente. Si el número de hormigas atraídas no es suficiente, se aplica la segunda forma de reclutamiento: las hormigas dejan una traza desde el objeto hasta el nido, con el objetivo de estimular a otros individuos a seguir el rastro. En esta tarea, la coordinación del grupo emerge de la dinámica del sistema [13].

### 2.1.2. Modelo de colmena de abejas

Las colonias de abejas son un buen ejemplo de cómo un grupo se comporta como una entidad supraorganizacional (enjambre), ya que funcionan como una totalidad coherente, donde cada miembro de la colonia contribuye a su mantenimiento. Cada individuo posee comportamientos morfológicos, fisiológicos y especializados que contribuyen al funcionamiento eficiente del sistema al que pertenece. Las colonias de abejas se asemejan a una entidad supraorganizacional, entre otras cosas porque [13]:

- a. La coordinación está basada en la autoorganización y en decisiones locales: en las colonias, las decisiones se basan en estímulos locales, como en el caso de la danza para reclutar abejas para la explotación de fuentes de alimentos. En este comportamiento y otros similares emergen patrones complejos utilizando reglas muy simples, lo que da lugar a una cooperación que parece inteligente.
- b. La coordinación está basada en bucles/lazos de retroalimentación e interacciones no lineales: la capacidad homeostática de las sociedades de abejas no exige un control central, sino que pueden emerger bucles de retroalimentación simples, que son característica de las estructuras autoorganizadas. Por ejemplo, la producción de calor individual de las abejas en la superficie de un *clúster* de abejas.

La construcción de panales en las colonias de abejas es una tarea colectiva en la que los procesos se llevan a cabo sin un coordinador central. Existen dos hipótesis que intentan explicar el fenómeno: la primera, se relaciona con la existencia de plantillas que organizan el panal; la segunda, se fundamenta en la autoorganización. Esta última hipótesis establece que el diseño del panal aparece de forma espontánea a partir de las interacciones dinámicas entre dos procesos relevantes en la colonia: por ejemplo, el depósito y eliminación de las crías, o el depósito del polen y eliminación de la miel. En ambos casos, sin la existencia de un plan que especifique relaciones espaciales. De esta manera, el patrón emerge de las dinámicas que se presentan en la colonia (puesta de huevos, polen y miel, consumo de polen y miel, etc.). Esto es suficiente para organizar el patrón del panal [13].

### 3. Arquitectura para un sistema multirobot con comportamiento emergente

En [11] se presenta una arquitectura para sistemas multirobot con comportamiento emergente. La arquitectura AMEB está estructurada en tres niveles (ver figura 1): individual, colectivo y gestión del conocimiento y aprendizaje.

El nivel individual gestiona los sensores, actuadores y el comportamiento de cada robot del sistema. En este nivel se implementa un componente conductual, que incluye un modelo emocional con la finalidad de dotar de emociones básicas a los individuos del sistema [15]. El módulo de gestión del conocimiento y aprendizaje brinda soporte a los procesos de construcción de conocimiento individual y colectivo del sistema. El nivel colectivo da soporte a los procesos que faciliten la coordinación del equipo de robots, desde el enfoque de la coordinación emergente. En [11, 12] se encuentran más detalles de esta arquitectura. Este trabajo se concentra principalmente en el nivel colectivo.

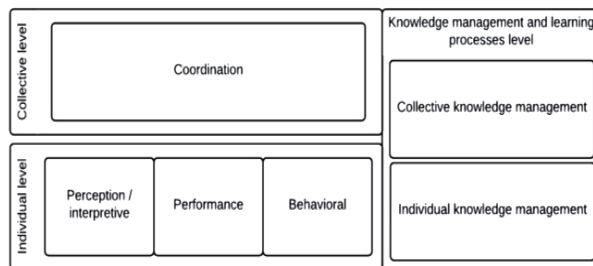


Figura 1. AMEB. Fuente: [11].

### 4. Coordinación emergente en AMEB

La arquitectura AMEB está diseñada para brindar soporte a un sistema multirobot formado por robots de propósito general [12]. El nivel colectivo de AMEB se encarga de gestionar los mecanismos que permiten que los individuos del sistema se coordinen en la ejecución de sus tareas. Los robots del sistema presentan las siguientes características:

- Tienen poco o ningún conocimiento del estado global del sistema ni del objetivo colectivo.
- Tienen poca o ninguna información sobre los otros integrantes del sistema.
- Su proceso de toma de decisiones está regulado por reglas locales. Estas reglas pueden ser el reflejo a una percepción, pueden ser generadas mediante planes

preestablecidos o algoritmos, entre otras formas. Puede darse el caso de que estas opciones se combinen para generar una única regla.

Por otra parte, la emergencia es consecuencia de los estados de los subsistemas a nivel particular y del estado global de sistema. La emergencia puede ocurrir si, además de lo comentado anteriormente, también existe:

- Niveles particular y global: El nivel particular está conformado por los subsistemas que componen el sistema global, en este caso, por los robots/agentes. Cada uno de ellos tiene sus propias capacidades y su propio comportamiento. El comportamiento del sistema global es producto de las acciones de cada individuo a nivel particular.
- Mecanismos de comunicación a nivel particular: Los elementos que componen el sistema deben poder comunicarse entre ellos, ya sea de forma directa o indirecta.
- Memoria colectiva: En la memoria colectiva se almacena información importante de la totalidad del sistema, además de la específica de los subsistemas. Los individuos pueden colocar y extraer información en esa memoria colectiva. La información almacenada en esta memoria está relacionada con un proceso de aprendizaje colectivo.

En la figura 2 se presenta la estructura del módulo colectivo de AMEB, que consta de cuatro capas de gestión centrales: el gestor de interacciones; el de acciones, que posibilita la coordinación emergente; el de mensajes y el de canales de comunicación. Cada individuo representa un microsistema, y las interacciones a nivel particular permiten la aparición de fenómenos a nivel global, que al ser observables pueden considerarse como la aparición de la emergencia en el sistema.

El nivel particular gestiona los estímulos que provienen del entorno y que afectan su comportamiento, como mensajes, marcas en el suelo, obstáculos, etc. Algunos de estos estímulos implican la necesidad de interactuar con el fin resolver alguna situación exterior. Esta necesidad de interacción se eleva al nivel colectivo, que realiza el siguiente proceso:

- El gestor de interacciones recibe la información de cada individuo.
- La información se clasifica según dos casos:

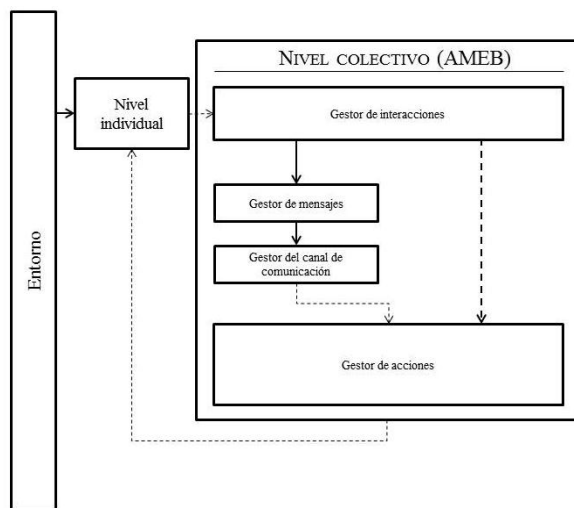


Figura 2. Nivel colectivo AMEB. Fuente: elaboración propia.

- Interacción directa: Ocurre cuando el individuo necesita enviar un mensaje.
- Interacción indirecta: Ocurre cuando el individuo se comunica a través de modificaciones o marcas en el ambiente (estigmergia<sup>1</sup>).

c. Si la interacción es directa, entonces la información se envía al gestor de mensajes, que se encarga de construir el mensaje a enviar de acuerdo con el protocolo de interacción definido previamente y según las capacidades del emisor y del receptor. Se construyen dos tipos de mensajes:

- Mensaje global: Mensaje dirigido a todos los individuos del sistema.
- Mensaje global con información específica: Mensaje que contiene información específica para individuos que cumplan ciertos parámetros. Por ejemplo, para individuos con baja carga de batería o individuos en estado emocional de alegría.

El mensaje se envía al gestor de canal de comunicación, que se encarga de seleccionar el canal configurado en el sistema para el envío de mensajes, por ejemplo, un sistema de memoria compartida tipo pizarra.

d. Si la interacción es indirecta, entonces el gestor de información sensorial del individuo procesa la información recibida y selecciona una respuesta a la acción detectada. Aquella acción es un comportamiento del set de comportamientos del individuo. Para esto, el individuo debe estar en capacidad de reconocer los

cambios que se dan en el entorno y que representan un mensaje.

e. La acción generada se envía al gestor de acciones, que las ensambla con las de otros individuos basado en mecanismos clásicos de coordinación. Este proceso de coordinación puede traer consigo ajustes de acciones a realizar por parte de los agentes, lo que implicaría procesos de emergencia.

#### 4.1. Gestión de interacciones en AMEB

En este nivel, el manejo de las interacciones entre los individuos es un proceso central. Las interacciones pueden ser:

a. Directas: A través del envío de mensajes, que pueden ser de dos tipos:

- Mensajes globales: Son mensajes de difusión amplia, sin información específica para un tipo de receptor en particular, por ejemplo, cuando un robot envía un mensaje indicando que presenta una falla.
- Mensajes con fines específicos: Son mensajes de difusión amplia que, a diferencia de los anteriores, llevan información específica para cierto tipo de receptores. Por ejemplo, cuando un robot encuentra una fuente de recarga y envía un mensaje con información específica para los robots que tienen un bajo nivel de energía.

b. Indirectas: A través de marcas artificiales en el ambiente. Estas marcas pueden ser producto de sensores virtuales, como feromonas artificiales, o pueden encontrarse en el entorno, como huellas en el suelo. Las marcas generan interacción entre los robots cuando son detectadas por más de uno.

El componente gestor de interacciones se encarga de recibir la información que cada robot proporciona, y que representa la necesidad de interactuar. Dicha información puede ser:

a. Un estímulo que recibe el robot a causa de una alteración en el entorno (marca en el suelo, feromona artificial, etc.).

b. Un estímulo causado por un mensaje enviado por otro individuo del equipo.

El estímulo que genera la interacción activa un mecanismo de percepción. El gestor toma en cuenta este mecanismo para la clasificación. El gestor lleva a cabo el siguiente proceso:

<sup>1</sup> Comunicación entre agentes a través del entorno.

- a. Recibe la información del estímulo:  
 <data, sensor\_fuente, Ri>,

donde, data: información del estímulo; sensor\_fuente: sensor que capta la información (visual, mensaje, etc.) y Ri: robot que recibe el estímulo.

- b. Clasifica la información de acuerdo con el sensor fuente.

- c. Envía la información al gestor de acciones.

#### 4.2. Gestor de acciones

Este gestor maneja una serie de reglas de actuación por defecto. La tabla 1 presenta un ejemplo de las reglas establecidas y la acción por defecto que traen asociada. Las reglas dependen directamente de las capacidades del individuo y del estímulo percibido.

Las reglas de la tabla 1 están construidas de acuerdo con las capacidades de los robots objeto de estudio y el ambiente de prueba. Estas mismas reglas podrían generalizarse, basados en ambos aspectos.

El gestor de acciones es responsable de seleccionar los comportamientos que el robot debe ejecutar, tomando como guía las reglas de clasificación de la tabla 1. Estas acciones forman parte del set de comportamientos del robot [15] y están asociadas a sus capacidades de hardware. El robot cuenta con dos clases de comportamientos:

- a. Básicos: avanzar, frenar, detenerse, sentarse, etc.
- b. Específicos, que son la conjunción de dos o más comportamientos básicos: explorar, evadir, perseguir, refugiarse, entre otros.

De acuerdo con el tipo de interacción, el robot ejecutará uno o más comportamientos. La conjunción de los comportamientos individuales de los robots constituye el comportamiento global del sistema. En otras palabras, el ensamblaje de las acciones de los agentes/robots produce una coordinación emergente de la siguiente manera:

El comportamiento del sistema es la suma de los comportamientos de los individuos que lo componen. Este comportamiento es emergente si puede observarse a nivel global. La coordinación se produce cuando: el comportamiento de cada individuo no entra en conflicto con los comportamientos de los otros individuos, se lleva a cabo una tarea que involucra dos o más individuos y a nivel global se observa una acción coordinada. Lo anterior es posible gracias a que:

Tabla 1. Reglas gestor de interacciones

Tipo de estímulo	Condición	Acción
Marcas en el suelo	Si representa una zona de carga (Regla 1)	Enviar un mensaje de difusión específico Si su estado actual indica la necesidad de recarga, permanecer en la zona
	Si representa una zona segura (Regla 2)	Si su estado actual indica la necesidad de refugio, permanecer en la zona Enviar un mensaje de difusión general
	Si representa un límite (Regla 3)	Detenerse Planificar la próxima acción
Detección de obstáculos	Si es un objeto fijo (Regla 4)	Detenerse Planificar la próxima acción
	Si es un objeto móvil (Regla 5)	Planificar la próxima acción
Detección de feromona artificial	Si la concentración de feromona es baja (Regla 6)	Depositar feromona Detenerse, no seguir el rastro
	Si la concentración de feromona es alta (Regla 7)	Depositar feromona Seguir el rastro
Mensaje recibido	Si el mensaje es global (Regla 8)	Analizar el mensaje Planificar la próxima acción
	Si el mensaje contiene información específica (Regla 9)	Revisar el estado interno para decidir si debe ser atendido Si es atendido, planificar la próxima acción Si el mensaje no debe ser atendido, ignorar y seguir con la acción que se estaba ejecutando

Fuente: elaboración propia.

- a. Los comportamientos reactivos tienen prioridad sobre los cognitivos e imitativos, esto con el objetivo de garantizar la supervivencia del robot en el entorno.
- b. Cada robot toma decisiones locales que buscan salvaguardar su integridad y llevar a cabo su objetivo inmediato: desplazarse, cambiar de dirección, etc.
- c. Los robots interactúan con el entorno y con otros individuos, dependiendo del estímulo que reciban.
- d. La interacción, sea directa o indirecta, puede generar un comportamiento social en el individuo. Este comportamiento puede traducirse en la necesidad de coordinación, lo que desemboca en una actuación colectiva.

En específico, cada individuo ejecuta un conjunto  $C_{Rn} = \{c_1, c_2, c_3, \dots, c_n\}$  de comportamientos en un intervalo de tiempo  $[t_0, t_n]$ , en donde los objetivos de algunos pueden coincidir. Es entonces que la coordinación emergente se produce:

$$C_{\text{colectivo}} = C_{SR0} \cup C_{SR1} \cup \dots \cup C_{SRn}$$

donde:

$$CS_{R0} \subseteq C_{R0}; CS_{R1} \subseteq C_{R1}; \dots; CS_{Rn} \subseteq C_{Rn}$$

El conjunto de comportamientos  $C_{\text{colectivo}}$  en un intervalo de tiempo  $[t_0, t_n]$  representa el comportamiento global del sistema.

### 4.3. Gestor de mensajes

Cuando la interacción es directa, este componente recibe la información del gestor de interacciones y se encarga de construir el mensaje que será enviado. Como antes se ha dicho, el mensaje puede ser de dos tipos:

- a. Difusión global: Mensajes que van dirigidos a todos los robots activos en el sistema.
- b. Difusión específica: Mensajes que van dirigidos a individuos específicos en un determinado momento.

El mensaje contiene información que el emisor quiere transmitir sobre su entorno más inmediato, sobre su estado particular y sobre el estado l del receptor objetivo, si aplica. El mensaje de forma general es una N-tupla de este tipo:

$\langle I, S_f, E_e, E_r \rangle$ , donde:

- I: Información a transmitir.
- $S_f$ : Sensor fuente.
- $E_e$ : Estado del emisor.
- $E_r$ : Estado del receptor objetivo.

Si el valor de  $E_r$  es nulo, el mensaje está dirigido a todos los individuos del grupo (difusión global); en caso contrario, va dirigido a individuos con esos estados (difusión específica).

Este mensaje es enviado al gestor de canales de comunicación.

### 4.4. Gestor de canales de comunicación

Los individuos que conforman el sistema pueden contar con distintos canales de comunicación según su configuración. Esto hace que los protocolos para el envío de mensajes sean cambiantes. Este componente permite seleccionar el canal de comunicación de acuerdo con la configuración de los robots/agentes, además de construir el mensaje de acuerdo con el protocolo de los dispositivos.

## 5. Caso de estudio

En esta sección se proponen casos de estudio, con el objetivo de mostrar el proceso de coordinación emergente que hace posible la arquitectura propuesta.

Las pruebas se llevaron a cabo en una plataforma de robots construidos con LEGO® MindStorms NXT, que tenían la capacidad de detectar objetos al frente y contaban con tracción diferencial. Además, se emplearon robots virtuales que funcionaban en una plataforma multi-agente [29], con soporte de visión artificial para el funcionamiento en conjunto de ambos tipos de robots. El entorno está formado por bloques que forman paredes y obstáculos.

### a. Transporte de un objeto

El transporte de un objeto es una tarea muy común en varias especies de insectos. Normalmente, las hormigas transportan objetos de gran tamaño. Para ello, reclutan a otras hormigas y, en conjunto, coordinan una formación alrededor de los objetos que permita trasladarlos [13]. Los investigadores han estudiado esta actividad de transporte para evaluar las capacidades emergentes en los sistemas multirobot [16, 17, 18]. En esta tarea en particular, ocurre el siguiente proceso:

- a. La hormiga intenta mover un objeto.
- b. Si el intento es exitoso, la hormiga lo lleva a la colonia.
- c. Si no es exitoso, la hormiga intenta reclutar a otras hormigas, bien sea por marcas en el suelo (interacción indirecta), bien sea por contacto directo (interacción directa).
- d. Cuando el transporte se va a realizar de forma cooperativa, el método puede variar dependiendo del objeto a transportar. Por ejemplo, pueden reposicionarse alrededor del objeto o descomponerlo.

La caracterización del proceso emergente se da de la siguiente manera:

- a. Objetivo del sistema global: Transportar un objeto.
- b. Nivel global: Sistema multirobot.
- c. Nivel particular: Robots de propósito general; objeto.
- d. Retroalimentación entre los niveles global y particular: Objeto que cambia de posición.
- e. Memoria colectiva: Actualización del estado interno de cada robot; posición del objeto; estímulo recibido y acción realizada.
- f. Reglas locales de decisión: Cada robot decide la próxima acción a ejecutar.

Se asume un entorno con 5 robots, un objeto y obstáculos representados por marcas en el suelo (Ver figura 3).

Gestor de Interacciones:

En este caso de estudio, el módulo colectivo gestiona las interacciones que se presentan en la tabla 2. Estas



interacciones se generan a partir de la actuación de cada individuo en el sistema.

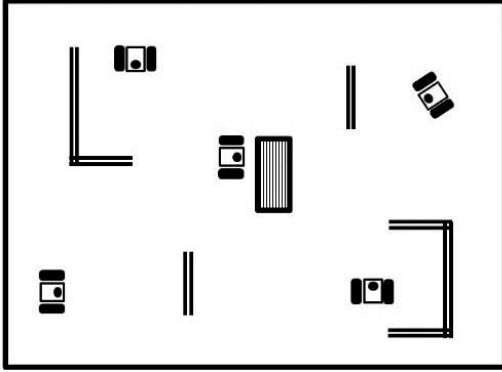


Figura 3. Escenario de transporte de un objeto.  
Fuente: elaboración propia.

Tabla 2. Interacciones, caso de transporte de un objeto

Interacción	Tipo
Detección de marca en el suelo (obstáculo)	Interacción indirecta
Detección de objeto (objeto a mover)	Interacción indirecta
Detección de objeto móvil (robot)*	Interacción directa
Envío/recepción de mensaje	Interacción directa

\*La detección de un objeto móvil (robot) para este caso es considerado como una interacción directa, de acuerdo con el modelo biológico en el que se inspira.

Fuente: elaboración propia.

Gestor de acciones:

Cada robot explora el entorno durante un tiempo  $t$ , donde su estado interno (estado emocional, comportamiento, nivel de batería) se actualiza de acuerdo con su interacción con el entorno. Como se muestra en la tabla 2, cada robot ejecuta el siguiente proceso durante la ejecución del sistema de acuerdo con el estímulo recibido:

1. Encontrar una marca en el suelo (se activa el gestor de interacciones): En este caso, las marcas representan límites. Por lo tanto, el robot debe decidir qué hacer. Según su estado interno, el robot puede (se activa el gestor de acciones):

- Detenerse y girar en otra dirección.
- Detenerse y retroceder.
- Detenerse y esperar algún estímulo que le permita tomar una decisión.

2. Detectar el objeto (se activa el gestor interacciones): En este caso, según su estado interno o emocional, el robot puede:

- Intentar mover el objeto.

- Detenerse y esperar.

2.1 Si el intento de mover el objeto es exitoso, el robot tratará de llevar el objeto hasta otro punto del entorno. Esta acción provoca una retroalimentación con el nivel global, donde se actualiza la posición  $p$  del objeto.

2.2 Si el intento de mover el objeto falla, el robot envía un mensaje global (se activan los gestores de mensajes y de canal de comunicación) para informar sobre la incapacidad de mover el objeto. Como consecuencia, el robot debe esperar que otro más llegue durante un umbral de tiempo  $u$ . Durante esta maniobra, los robots pueden ejecutar las siguientes acciones:

2.2.1 Coincidir con uno de los robots que esperan en el sitio del objeto: interacción directa. Ir a punto 3 (activando el gestor de acciones).

2.2.2 Posicionarse directamente sobre el objeto e intentar el punto 2 (activando el gestor de acciones).

3. Detectar un objeto móvil: Al detectar otro robot en el entorno, el primero debe ejecutar una maniobra de evasión (activando los gestores de interacciones y de acciones).

Cada robot lleva a cabo el proceso recién descrito. Puede observarse que existen toma de decisiones locales, actualización de la memoria compartida al actualizar el estado actual de cada robot y una retroalimentación entre el nivel particular y el global, que se da por el logro del objetivo. El sistema funciona durante un tiempo finito  $t$  o mientras existan robots activos.

En la ejecución de este caso de estudio intervienen todos los niveles de la arquitectura: el gestor de interacciones se encarga de clasificar la necesidad de interactuar, que en este caso se presenta al detectar una marca o un objeto. Si se trata del segundo escenario, los gestores de mensajes y de canal de comunicación se activan para construir el mensaje a enviar a los otros individuos. En ambas situaciones, el gestor de acciones se encarga de enviar al nivel individual la acción que el individuo debe ejecutar. En todo momento, los componentes emocionales individuales, de actuación y de percepción están activos, para gestionar el funcionamiento del individuo.

### 5.1. Búsqueda de rutas

Cuando las hormigas encuentran la ruta hacia una fuente de alimento, depositan feromonas con el fin de que otras hormigas de la colonia puedan seguir el camino [13, 19]. El presente caso de estudio utiliza un sistema multirobot

para encontrar rutas óptimas, libres de obstáculos en entornos no estructurados (figura 4).

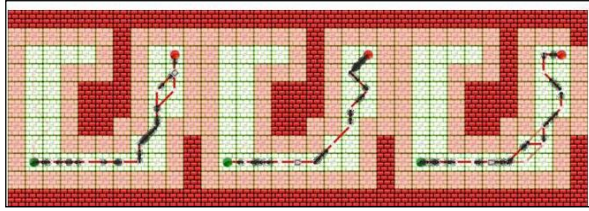


Figura 4. Escenario búsqueda de rutas. Fuente: elaboración propia.

Se identifican los siguientes componentes:

- Objetivo del sistema global: Encontrar una ruta libre de obstáculos.
- Nivel global: Sistema multirobot heterogéneo.
- Nivel particular: Robots físicos y virtuales de propósito general.
- Retroalimentación entre los niveles global y particular: Cuando el robot encuentra un punto objetivo en el entorno.
- Memoria colectiva: Feromona depositada.
- Reglas locales de decisión: Cada robot calcula la velocidad y la dirección de su desplazamiento en función de la feromona detectada y de los obstáculos en el camino.

**Gestor de Interacciones:** En este caso, se gestionan las interacciones que se presentan en la tabla 3.

Tabla 3. Interacciones, caso de búsqueda de rutas

Interacción	Tipo
Detección/depósito de feromona artificial	Interacción indirecta
Detección de punto de llegada	Interacción indirecta
Detección de objeto móvil (robot)	Interacción directa
Detección de objeto fijo	Interacción indirecta

Fuente: elaboración propia.

El nivel individual gestiona los procesos de cada uno de los robots durante la exploración del entorno en busca del punto de llegada. Por otro lado, cada robot deposita feromonas artificiales a lo largo del trayecto, apoyado en el componente de actuación del nivel individual. Si los individuos se topan con un obstáculo (nivel individual y gestor de interacciones), deben ejecutar un comportamiento de evasión, activando para ello el gestor de acciones. Además, si se encuentran con otro robot en la ruta, entonces se aplica el algoritmo ORCA [20] que, a través de la información compartida (memoria colectiva) sobre la posición y la velocidad de los robots, genera un comportamiento para evitar la colisión. En este caso particular, por una parte, la cooperación emerge de

las interacciones de los individuos, por otra parte, el gestor de interacciones se pone en funcionamiento con las interacciones indirectas que producen los obstáculos u otros robots en la ruta, mediante la detección de feromonas artificiales.

## 5.2. Comparación con trabajos previos

En esta sección se contrasta AMEB con investigaciones similares, para lo cual se establecen los siguientes criterios de comparación:

- Coordinación emergente (CE): Este criterio establece si la coordinación se implementa bajo este enfoque.
- Diversidad (D): Tipo de sistema multirobot.
- Adaptabilidad (A): Grado de adaptabilidad a distintas tareas.
- Aprendizaje (Ap): Implementación de mecanismos de aprendizaje.
- Componente emocional (E): Implementación o no de emociones en los individuos.

Tabla 4. Comparación con trabajos previos

Arquitectura	Criterios de comparación				
	CE	E	A	D	Ap
AMEB	Sí	Sí	Sí	Heterogéneo	Sí
SWARM-BOT [21]	Sí	No	Sí	Heterogéneo	No
Swarm Robot System [22]	Sí	No	Sí	Homogéneo	Sí
ACROMOVI [23]	No	No	No	Heterogéneo	No
Arquitectura basada en comportamiento [24]	No	No	No	Híbrido	Sí
Arquitectura de control para sistemas multirobot [25]	No	No	No	Heterogéneo	No
Arquitectura híbrida [26]	No	No	Sí	Homogéneo	No
Arquitectura híbrida basada en ROS [27]	No	No	Sí	Heterogéneo	No
Arquitectura orientada a servicio [28]	No	No	Sí	Heterogéneo	No

Fuente: elaboración propia.

Como se muestra en la tabla 4, AMEB implementa la gestión de los procesos básicos de un robot de propósito general dentro de un sistema multirobot. También, permite la construcción de sistemas heterogéneos, a la par que gestiona procesos de aprendizaje. Además, implementa emociones artificiales en este tipo de robots, lo que regula su comportamiento. Los mecanismos de cooperación no se centralizan, ni se establecen *a priori*, sino que emergen de la interacción de los individuos. En otros trabajos, la gestión de estos procesos no se implementa simultáneamente; la cooperación emergente puede aparecer, pero en sistemas no regulados por emociones o sin mecanismos de aprendizaje colectivo. Finalmente, la coordinación centralizada se presenta en algunos de estos trabajos.

## 6. Conclusiones

La arquitectura AMEB organiza los procesos del sistema multirobot para propiciar la aparición de comportamientos emergentes. Esta arquitectura presenta las características propias de los sistemas emergentes: es distribuida, gestiona la toma de decisiones locales, existen procesos de aprendizaje colectivo desde lo individual y presenta espacios de memoria compartidos. Por otro lado, los componentes de la arquitectura se aplican de forma distribuida, con el objeto de gestionar los procesos propios del funcionamiento de cada individuo del sistema y los que emergen de su interacción. El nivel particular gestiona el comportamiento de cada individuo: sus procesos de percepción, actuación y su componente afectivo. Este nivel se relaciona estrechamente con el nivel colectivo, donde se gestionan las interacciones ocurren entre los individuos y de cada uno con el ambiente. El nivel de gestión de conocimiento permite construir el conocimiento a partir de las experiencias individuales.

La gestión de las interacciones entre los individuos del sistema, sean directas o indirectas, facilita la coordinación natural o emergente. Esta arquitectura también brinda soporte a ambos tipos de interacción, facilitando la configuración de estas de acuerdo con las características de los individuos que conforman el sistema.

La arquitectura propone la implementación de un componente afectivo, que permite dotar de emociones a los individuos del sistema. La emoción modifica el comportamiento del individuo, por lo que sus capacidades sensoriales, de actuación y de toma de decisiones se ven afectadas por el estado emocional activo en un instante  $t$ . Esto hace que un mismo comportamiento sea ejecutado de distintas formas por el individuo, lo que contribuye al comportamiento emergente del sistema.

Como trabajos futuros, se propone ampliar el espectro emocional de AMEB, para pasar de un modelo unidimensional a un modelo bidimensional que incluya nuevas emociones. Esto permitirá al robot regular comportamientos de mayor complejidad y mejorar su proceso de toma de decisiones. Adicionalmente, se propone la implementación de un mecanismo de reconocimiento que permita a los robots reconocer el estado emocional de los otros, con el fin de llevar a cabo tareas complejas más eficientemente. La posibilidad de reconocer las emociones permitiría, por ejemplo, reclutar robots con un estado emocional específico para una tarea determinada, siempre que aquel estado de ánimo

contribuya particularmente a la realización de la tarea que otros.

## Agradecimientos

Proyecto parcialmente financiado por el Decanato de Investigación de la Universidad Nacional Experimental del Táchira, Venezuela, bajo el código 01-004-2013.

## Referencias

- [1] S. Zhang, Z. Lin, and G. Yan, "Local coordination and experiments," in *12th International Conference on Control, Automation, Robotics and Vision, ICARCV*, 2012, pp. 913–918. doi: 10.1109/ICARCV.2012.6485279
- [2] W. Ding, G. Yan, and Z. Lin, "Collective motions and formations under pursuit strategies on directed acyclic graphs," *Automatica*, vol. 46, no. 1, pp. 174–181, 2010, doi: 10.1016/j.automatica.2009.10.025.
- [3] M. Colby, J. J. Chung, and K. Tumer, "Implicit adaptive multi-robot coordination in dynamic environments," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5168–5173. doi: 10.1109/IROS.2015.7354105
- [4] H. Ahmadzadeh and E. Masehian, "Fuzzy coordination graphs and their application in multi-robot coordination under uncertainty," in *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, 2014, pp. 345–350. doi: 10.1109/ICRoM.2014.6990925
- [5] J. Santos, P. Costa, L. F. Rocha, A. P. Moreira, and G. Veiga, "Time enhanced A\*: Towards the development of a new approach for Multi-Robot Coordination," in *2015 IEEE International Conference on Industrial Technology (ICIT)*, 2015, pp. 3314–3319. doi: 10.1109/ICIT.2015.7125589
- [6] M. Lujak and A. Fernández, "Distributed multi-robot coordination combining semantics and real-time scheduling," in *2015 10th Iberian Conference on Information Systems and Technologies (CISTI)*, 2015, pp. 1–6. doi: 10.1109/CISTI.2015.7170599
- [7] O. Cheikhrouhou, A. Koubâa, and H. Bennaceur, "Move and improve: A distributed multi-robot coordination approach for multiple depots multiple travelling salesmen problem," in *2014 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2014, pp. 28–35. doi: 10.1109/ICARSC.2014.6849758

- [8] F. J. Mendiburu, M. R. A. Morais, and A. M. N. Lima, "Behavior coordination in multi-robot systems," in *2016 IEEE International Conference on Automatica (ICA-ACCA)*, 2016, pp. 1–7. doi: 10.1109/ICA-ACCA.2016.7778506
- [9] M. Cáp, P. Novák, M. Selecký, J. Faigl, and J. Vokffnek, "Asynchronous decentralized prioritized planning for coordination in multi-robot system," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3822–3829. doi: 10.1109/IROS.2013.6696903
- [10] Y. Quiñonez, I. Tostado, and O. Sánchez, "Coordination Model for Multi-robot Systems Based on Cooperative Behaviors," in *2013 12th Mexican International Conference on Artificial Intelligence*, 2013, pp. 33–37. doi: 10.1109/MICAI.2013.46
- [11] A. Gil, J. Aguilar, R. Rivas, E. Dapena, and K. Hernandez, "Architecture for Multi-robot Systems with Emergent Behavior," in *International Conference on Artificial Intelligence (ICAI)*, 2015.
- [12] A. Gil, J. Aguilar, R. Rivas, E. Dapena, and K. Hernandez, "Diseño de una plataforma multi-robot de propósito general," in *Congreso Internacional de Métodos Numéricos en Ingeniería y Ciencias Aplicadas. Simulación y aplicaciones recientes para ciencia y tecnología*, 2016.
- [13] J. Aguilar, "Introducción a los Sistemas Emergentes," Universidad de Los Andes, Mérida, Venezuela, 2014.
- [14] C. R. Kube and E. Bonabeau, "Cooperative transport by ants and robots," *Rob. Auton. Syst.*, vol. 30, no. 1, pp. 85–101, 2000, doi: 10.1016/S0921-8890(99)00066-4
- [15] Á. E. Gil Pérez, J. L. Aguilar Castro, R. D. J. Rivas Estrada, and E. Dapena González, "Módulo Conductual Inmerso En Una Arquitectura De Control Para Sistemas Multi-Robots," *Ing. al Día*, vol. 2, no. 1, pp. 40–57, Jan. 2015.
- [16] Y. Tohyama and H. Igarashi, "Cooperative transportation by multi-robots with selecting leader," in *35th Annual Conference of IEEE Industrial Electronics*, 2009, pp. 4179–4184. doi: 10.1109/IECON.2009.5415079
- [17] S. Moon, D. Kwak, and H. J. Kim, "Cooperative control of differential wheeled mobile robots for box pushing problem," in *2012 12th International Conference on Control, Automation and Systems*, 2012, pp. 140–144.
- [18] A. Ghosh, A. Ghosh, A. Konar, and R. Janarthanan, "Multi-robot cooperative box-pushing problem using multi-objective Particle Swarm Optimization technique," in *2012 World Congress on Information and Communication Technologies*, 2012, pp. 272–277. doi: 10.1109/WICT.2012.6409087
- [19] M. Dorigo, V. Maniezzo, and A. Coloni, "Ant system: optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 26, no. 1, pp. 29–41, 1996, doi: 10.1109/3477.484436
- [20] H. Cheng, Q. Zhu, Z. Liu, T. Xu, and L. Lin, "Decentralized navigation of multiple agents based on ORCA and model predictive control," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 3446–3451. doi: 10.1109/IROS.2017.8206184
- [21] E. Sahin *et al.*, "SWARM-BOT: pattern formation in a swarm of self-assembling mobile robots," in *IEEE International Conference on Systems, Man and Cybernetics*, 2002, vol. 4. doi: 10.1109/ICSMC.2002.1173259
- [22] S. Sang-Wook, Y. Hyun-Chang, and S. Kwee-Bo, "Behavior learning and evolution of swarm robot system for cooperative behavior," in *2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2009, pp. 673–678. doi: 10.1109/AIM.2009.5229933
- [23] P. Nebot and E. Cervera, "La arquitectura Acromovi: Una arquitectura para tareas cooperativas de robots móviles," in *Una Perspectiva de la Inteligencia Artificial en su 50 Aniversario*, Albacete: Gráficas Quintanilla, 2006, pp. 365–376.
- [24] F. de la Rosa and M. E. Jimenez, "Simulation of Multi-robot Architectures in Mobile Robotics," in *2009 Electronics, Robotics and Automotive Mechanics Conference (CERMA)*, 2009, pp. 199–203. doi: 10.1109/CERMA.2009.57.
- [25] J. H. Posselius *et al.*, "Control architecture for multi-robot system," U.S. Patent No. 9,527,211, 27-Dec-2016.
- [26] J. Stephan, J. Fink, V. Kumar, and A. Ribeiro, "Hybrid architecture for communication-aware multi-robot systems," in *2016 IEEE International Conference*

on *Robotics and Automation (ICRA)*, 2016, pp. 5269–5276. doi: 10.1109/ICRA.2016.7487737

[27] C. Hu, C. Hu, D. He, and Q. Gu, “A new ROS-based hybrid architecture for heterogeneous multi-robot systems,” in *The 27th Chinese Control and Decision Conference (2015 CCDC)*, 2015, pp. 4721–4726. doi: 10.1109/CCDC.2015.7162759

[28] Y. Cai, Z. Tang, Y. Ding, and B. Qian, “Theory and application of multi-robot service-oriented architecture,” *IEEE/CAA J. Autom. Sin.*, vol. 3, no. 1, pp. 15–25, 2016, doi: 10.1109/JAS.2016.7373758.

[29] J. Aguilar, M. Cerrada, G. Mousalli, F. Rivas, and F. Hidrobo, “A Multiagent Model for Intelligent Distributed Control Systems BT - Knowledge-Based Intelligent Information and Engineering Systems,” in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2005, pp. 191–197.