

# Arquitectura Distribuida para un Equipo de Robots Móviles Heterogéneos\*

Patricio Nebot, Daniel Gómez, and Enric Cervera

Laboratorio de Robótica Inteligente  
Universitat Jaume I  
Castellón

**Resumen** En los últimos años se está mostrando un interés cada vez mayor en los sistemas de múltiples robots autónomos para la realización de tareas cooperativas. Dicho interés es lo que nos ha motivado para la implementación de una arquitectura distribuida para un equipo de robots móviles heterogéneos. Dicha arquitectura deberá poseer la capacidad suficiente para permitir al equipo de robots la realización de tareas cooperativas. Este hecho nos ha llevado a la utilización de un sistema multiagente para la implementación de esta arquitectura distribuida.

**Key words:** Robots móviles, heterogeneidad, agentes, cooperación

## 1. Introducción

El trabajo en equipo es una capacidad esencial para múltiples robots móviles. Tener un único robot con múltiples capacidades (visión, láser, manipulador, ordenador) puede ser una pérdida de recursos. Diferentes robots, cada uno con su propia configuración, crean un sistema más flexible, robusto y barato. Además, las tareas a realizar pueden resultar demasiado complejas para un único robot, y teniendo múltiples robots se puede aumentar la efectividad.

Establecer mecanismos de cooperación entre robots implica plantearse un problema de diseño de comportamiento cooperativo: dado un grupo de robots, un entorno y una tarea, como debe llevarse a cabo la cooperación. Dicho problema implica varios retos, destacando entre ellos la definición de la arquitectura del grupo. Los sistemas multiagente son el entorno natural para tales grupos de robots, posibilitando la rápida implementación de potentes arquitecturas para la especificación y ejecución de tareas.

Este artículo describe la implementación y desarrollo de una arquitectura distribuida para la programación y control de un equipo de robots móviles heterogéneos coordinados, capaces de colaborar entre ellos y con personas en la realización de tareas de servicios en entornos cotidianos.

---

\* Proyecto financiado por el Ministerio de Ciencia y Tecnología (DPI2001-3801, HF2001-0112), la Generalitat Valenciana (CTIDIA/2002/195) y la Fundació Caixa-Castelló (P1-1B2001-28).

El resto del artículo se estructura de la siguiente forma: en la sección 2 se hará un repaso a los trabajos relacionados con el presente proyecto; en la sección 3 se abordarán los aspectos técnicos de los robots; en la sección 4 se describirá la arquitectura creada para el proyecto; y por último en la sección 5 se enumerarán las conclusiones obtenidas del presente trabajo.

## 2. Estado del arte

Existen numerosos trabajos donde se muestran arquitecturas para el control y programación de equipos de robots para la realización de tareas cooperativas. Cabe destacar los siguientes:

*ALLIANCE* es una arquitectura orientada a la cooperación de un equipo pequeño-mediano de robots heterogéneos, con escasa comunicación entre ellos[1].

La arquitectura *DPA (Dynamic Physical Agents)*[2] muestra al agente como una entidad modular con tres niveles de abstracción: módulo de control, módulo supervisor y módulo agente.

La arquitectura *ABBA (Architecture for Behaviour Based Agents)*[3] tiene el propósito de diseñar una arquitectura para modelar el comportamiento del robot para que pueda seleccionar comportamientos reactivos de bajo nivel para su supervivencia.

Estos trabajos implementan arquitecturas para que equipos de robots móviles puedan realizar tareas cooperativas. Nuestro trabajo consiste en la implementación de una arquitectura de este tipo, añadiendo la versatilidad y potencia de los sistemas multiagente junto con sistemas distribuidos para la resolución de tareas cooperativas por parte de un grupo de robots heterogéneos.

## 3. Aspectos técnicos de los robots

Para el desarrollo del proyecto, se ha utilizado un equipo consistente en 6 robots Pioneer-2 y un robot Powerbot, todos ellos fabricados por ActivMedia Robotics[4].

Entre los robots existen características diferentes, constituyendo así un grupo heterogéneo. En este grupo varían tanto las dimensiones, como los diferentes accesorios añadidos a los robots.

Todos los robots mantienen el mismo diseño lógico basado en un microcontrolador que transfiere las lecturas de los elementos del robot via RS232 al ordenador donde se ejecutan las aplicaciones. Este ordenador puede estar ubicado en el robot o no (en cuyo caso la comunicación se realiza por radio-módem).

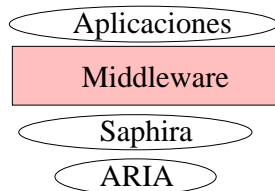
## 4. Implementación de la arquitectura

Nuestro objetivo es el desarrollo de una arquitectura distribuida para la programación y control de un grupo de múltiples robots móviles heterogéneos

capaces de cooperar entre ellos y con personas para la realización de tareas de servicios en entornos cotidianos.

Los robots disponen de una arquitectura de programación con bibliotecas para sus accesorios. Esta arquitectura está formada por una capa inferior, el API ARIA[4], que se encarga de la gestión de las peticiones de los programas a los componentes que forman el robot; y una capa superior formada por el sistema de control Saphira[4,5], que provee unos servicios de un nivel superior. Sin embargo, esta arquitectura está orientada al proceso desde un único controlador, sin definir mecanismos de colaboración mutua.

La tarea a realizar es añadir una capa para compartir los recursos de cada robot entre el grupo. Se trata de desarrollar un nivel intermedio (middleware) entre las funciones del robot (ARIA + Saphira) y las aplicaciones.



**Figura 1.** Diagrama general de la arquitectura.

Para la resolución de la tarea descrita se plantea un enfoque orientado a agentes, mediante el uso de una herramienta de desarrollo de agentes[6].

#### 4.1. Arquitectura de agentes

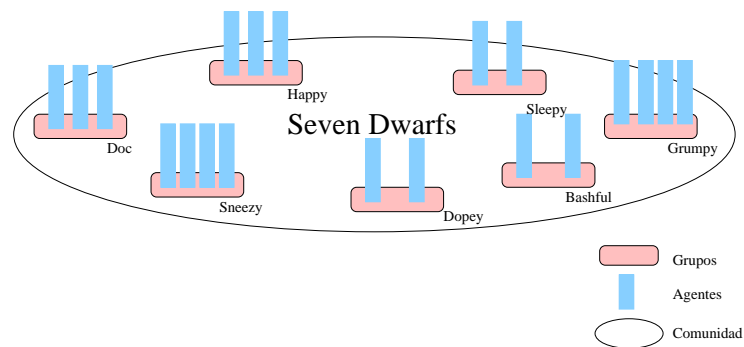
Este enfoque implica la implementación de la capa middleware utilizando un sistema multiagente (SMA). Para dicha implementación se ha utilizado la herramienta de programación de sistemas multiagente MadKit (Multi-Agent Development Kit)[7].

MadKit está basado en un modelo organizacional llamado Aalaadin. A su vez, Aalaadin está basado en tres conceptos: *agente*, *grupo* y *rol*. Además, MadKit implementa otro concepto, la comunidad[8].

Para poder implementar la capa middleware de nuestra arquitectura, se ha creado la comunidad “Seven Dwarfs”. Dentro de esta comunidad hay tantos grupos como robots, como puede apreciarse en la figura 2.

Dentro de cada uno de los grupos, se sitúan los agentes propios de cada robot. Cada agente controla un elemento del robot. Estos agentes son los que se comunican con la capa base (ARIA + Saphira) de la arquitectura global por medio de JNI (Java Native Interface).

Por encima de la capa middleware se encuentra la capa correspondiente a las aplicaciones, que pueden ser implementadas como agentes. Las aplicaciones, para poder acceder a los componentes del robot, deben comunicarse con los agentes



**Figura 2.** Estructura de la comunidad “Seven Dwarfs”.

de la capa middleware, y éstos se comunican con la capa base que controla al robot.

## 5. Conclusiones

En el artículo se ha mostrado como implementar una arquitectura distribuida para un equipo de robots móviles heterogéneos. Esta arquitectura auna las ventajas de la computación distribuida y de los sistemas multiagente, haciendo que sea apropiada para la implementación y ejecución de tareas que requieran de cierta colaboración por parte de los robots del equipo.

Además, al no ser el robot el que conforma un agente, sino que cada componente es un agente específico, se logra que los componentes propios de un robot puedan ser compartidos por todo el equipo, con el consiguiente aumento de la capacidad de prestaciones de los robots.

## Referencias

1. Parker, L.E., ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation. IEEE Transactions on Robotics and Automation, 1998. 14(2): pp. 220-240.
2. Oller, A., de la Rosa, J.Ll., DPA2: Una arquitectura para agentes físicos dinámicos. 1st Workshop on Physical Agents, 2000, pp. 25-43.
3. Jung, D., Zelinsky, A., An Architecture for Distributed Cooperative Planning in a Behaviour-Based Multi-Robot System. Journal of Robots and Autonomous Systems, 99. Vol 26 No.2-3, pp. 149-174.
4. ActivMedia Robotics Inc. <http://www.activmedia.com>
5. Konolige, K., Myers, K., The Saphira Architecture for Autonomous Mobile Robots. Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems, chapter 9, pp 211-242. MIT Press. 1998.
6. The Foundation for Intelligent Physical Agents <http://www.fipa.org>
7. Gutknecht, O., Ferber, J., MadKit Reference page. <http://www.madkit.org>
8. Gutknecht, O., Ferber, J., The MadKit Agent Platform Architecture. LNAI 1887, 2000.