

# Generando un agente robótico autónomo a partir de la evolución de sub-agentes simples cooperativos

Ricardo A. Téllez  
Grup de Recerca en Enginyeria del Coneixement (GREC)  
Technical University of Catalonia. Rambla de l'Exposició s/n  
08800 Vilanova i la Geltrú. Spain  
r\_tellez@ouroboros.org

Cecilio Angulo  
Grup de Recerca en Enginyeria del Coneixement (GREC)  
Technical University of Catalonia, Rambla de l'Exposició s/n  
08800 Vilanova i la Geltrú. Spain  
cecilio.angulo@upc.es

## Resumen

Se mostrará cómo generar un agente robótico autónomo a partir de la unión de diversos sub-agentes simples implementados mediante redes neuronales sencillas que aprenden a cooperar por medio de neuro-evolución. En ningún momento se especificará cómo deben cooperar los sub-agentes sino que serán ellos mismos los que decidirán cómo hacerlo. Este tipo de implementación presenta diversas ventajas frente a sistemas centralizados, entre las que se incluyen mayor rapidez en el aprendizaje, mayor generalización ante nuevos eventos y mayor resistencia a sensores ruidosos o parcialmente defectuosos.

**Palabras Clave:** agentes, sub-agentes, cooperación, redes neuronales, neuro-evolución

## 1. Introducción

La idea de dividir un agente robótico en diversos agentes que cooperan entre sí no es nueva. Existen numerosos trabajos en la literatura en los que se toma esta aproximación a la hora de crear un agente autónomo [1][14][16]. Usualmente, en estos trabajos se divide al agente robótico en varios sub-agentes especializados en una tarea concreta (planificador, agente reactivo, coordinador, etc...) dando lugar a los llamados sistemas multi-agente (MAS). En los sistemas MAS todos los agentes y la manera en que se coordinan suelen estar programados de antemano por el diseñador del sistema, de forma adecuada a la tarea a realizar.

Nuestra propuesta, aun tomando como punto de partida la división del agente robótico en un conjunto de sub-agentes que cooperan, realiza tal división de una manera diferente. Para ello se forzará hasta el máximo nivel posible la idea expuesta por Minsky en [12], según la cual un agente inteligente podría estar compuesto por una multitud de pequeños sub-agentes simples sin inteligencia cuya única razón de ser sería la de realizar bien una tarea simple, pero que combinados de la manera correcta para cooperar entre ellos, darían lugar a un *ente* global inteligente. Por ello, siguiendo dicha filosofía, dividiremos a nuestro agente robótico en pequeños sub-agentes formados por controladores muy simples y no pre-programados. La forma de llevar el concepto de agente físico hasta su nivel más simple se traducirá en la definición de cada uno de estos sub-agentes, de forma que estén a cargo de tan sólo uno de los sensores o actuadores que conforman el agente robótico, aunque al mismo tiempo se encuentren en comunicación directa con el resto. La idea es que los sub-agentes aprendan por ellos mismos a cooperar para conseguir que el agente físico global (el robot) realice la tarea que le haya sido encomendada. Debe hacerse notar que esta aproximación lleva al límite la idea de la necesidad de un cuerpo (embodiment) para la obtención de inteligencia [10][11].

Como puede observarse, en esta aproximación al problema del control de un agente robótico, la división del agente global en sub-agentes se realiza a nivel muy inferior al típico de los sistemas MAS, proporcionándonos un *grano de sub-agente* mucho más pequeño, fino y versátil, en comparación con los estándares en la literatura..

Otra de las peculiaridades de esta aproximación es que en ningún momento se codifica de antemano qué es lo que los sub-agentes deben hacer para conseguir su objetivo, es más, ni siquiera se especifica el *objetivo* particular de cada sub-agente, sino que se indica cual debe ser el comportamiento global del agente robótico [3][4]. Esta importante

característica se logra de nuevo gracias al bajo nivel de necesidad computacional asociado a cada sub-agente. De forma implícita el sistema presupone que debe existir una conexión reactiva entre sensores y actuadores, y la información generada por estos elementos físicos es traducida, en función de las necesidades del conjunto de sub-agentes y del grado de cumplimiento de la tarea general, por cada uno de los sub-agentes. Tan sólo se les dan a los sub-agentes los medios necesarios para que puedan realizar la tarea asignada, buscando ellos mismos la mejor manera de hacerlo. Ninguno de los sub-agentes está a cargo de ninguna actividad especial, sino que su implicación en el proceso general es reconocida por ellos mismos durante el proceso de aprendizaje. El proceso de aprendizaje se realiza con un método no supervisado de neuro-evolución denominado Enforced SubPopulations (ESP) especialmente diseñado para este tipo de controladores [2].

El objetivo de los experimentos realizados es el de demostrar que un controlador neuronal distribuido basado en sub-agentes muy simples es posible y que además se comporta más inteligentemente que un controlador neuronal centralizado. Una vez demostrado, y dada la facilidad de implementación y reducida capacidad computacional requerida por nuestra arquitectura, harían de ésta una perfecta candidata para todos aquellos sistemas en los que se necesitase un control inteligente de múltiples sensores y actuadores (como por ejemplo el movimiento de una mano robótica, o el control de una casa domótica).

## 2. Descripción del sistema multi-agente

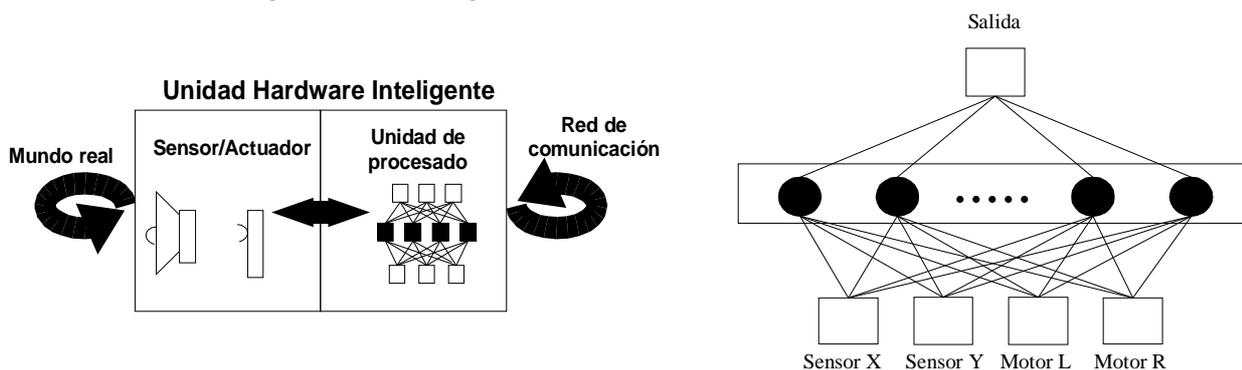
Para la realización de los experimentos utilizamos un robot simple, simulado en un ordenador bajo el entorno matemático Scilab. Todos los resultados presentados en este artículo corresponden a una simulación (utilizando como referencia el procedimiento expuesto en [8]) dado que el robot físico real aún no ha sido terminado. A continuación se describen cada una de las partes simuladas.

### 2.1. El robot

En nuestro caso hemos utilizado un robot compuesto por dos ruedas motrices (actuadores) y por dos sensores de infrarrojos. El robot está formado por una plataforma rectangular. En las dos esquinas inferiores se encuentran cada una de las ruedas. En la esquina superior izquierda se encuentran los dos sensores de infrarrojos, uno apuntando hacia delante y otro apuntando hacia la izquierda, midiendo de esta manera las distancias entre el robot y un objeto enfrente de él o entre el robot y un objeto a su izquierda. Por último, el robot dispone de una rueda delantera utilizada para tener estabilidad pero que no es motriz (por lo tanto, sin motor).

### 2.2. Los sub-agentes

Definimos una unidad básica y genérica que utilizaremos para dividir al agente global. A esta unidad la llamaremos elemento hardware inteligente y está formada por un sensor o actuador y por un pequeño microcontrolador que implementa una red neuronal artificial de capacidad reducida. Cada una de estas unidades del robot es lo que consideraremos un sub-agente del robot (Figura 1a).



Figuras 1a y 1b: la figura 1a muestra la estructura del elemento hardware inteligente y la figura 1b muestra la red neuronal implementada en cada sub-agente

Por cada sensor y actuador de los que disponga el agente robótico, se creará una de estas unidades. A nivel conceptual, mientras que la parte del sensor/actuador conecta al sub-agente (y por ende al agente global) con el mundo exterior, la unidad de proceso realiza el procesamiento de la información recibida/enviada y conecta al sub-agente con el mundo interior, es decir, con el resto de sub-agentes. El sensor/actuador es el dispositivo físico que realiza la acción de sentir o actuar desde o sobre el mundo exterior. El elemento de procesamiento es, en nuestro caso, una red neuronal feedforward con capa oculta, con cuatro neuronas de entrada y una de salida (ver figura 1b). Las neuronas de entrada contienen las salidas de las redes neuronales de los otros sub-agentes y la neurona de salida proporciona el valor final procesado,

que, según se trate de un sensor o un actuador, indicará el valor medido o la acción que deberá realizar el actuador. Dicho resultado es enviado también al resto de sub-agentes para que lo tengan en cuenta en sus futuros procesados.

Se trata pues, desde el punto de vista de agentes, de elementos muy sencillos de procesado con una zona actuación limitada, pero muy bien definida. Desde la óptica del control de procesos, se estaría hablando del uso de sensores inteligentes y actuadores con control. Sin embargo, en ambos casos, la aproximación es novedosa, puesto que el control global se obtiene por medio de la colaboración de elementos neuronales muy simples y que aprenden en conjunto. Esta sencillez en la conexión entre el mundo físico y la capacidad de actuación para cumplir una tarea global de procesamiento es su principal baza. Además, pondrá de relieve cómo una estructura de control distribuida o colaborativa, desde la perspectiva de control de procesos, o un sistema multi-agente son capaces de alcanzar tareas de forma más inteligente y eficaz que los tradicionales esquemas unificados o los esquemas jerárquicos.

Llegados a este punto, es necesario indicar la especial consideración que requieren los sub-agentes sensores. Mientras que se podría pensar que su inclusión es redundante y poco útil dado que la tarea básica que realizan los sensores es captar una variable del mundo real y enviarla al resto de sub-agentes, los experimentos realizados demostraron que el incluir todo el elemento hardware inteligente en los sensores, incluida la unidad de procesado, proporcionaba diversas ventajas frente a pequeñas inconsistencias o errores de los mismos. En realidad, la unidad de procesado asociada al sensor será capaz de aprender a procesar la señal del sensor según las circunstancias que marque el entorno.

### 2.3.El algoritmo de aprendizaje

Para que los sub-agentes sean capaces de realizar lo que tienen que realizar es necesario que previamente lo aprendan, ya que como se ha mencionado, nuestro agente robótico no tendrá ningún tipo de condicionamiento pre-programado. El entrenamiento de las redes se realiza por medio de un algoritmo de aprendizaje no supervisado llamado Enforced SubPopulations (ESP) [7][9]. Este algoritmo está especialmente diseñado para evolucionar al mismo tiempo diversas redes neuronales con un objetivo común y ha sido probado con éxito en otros problemas en los que era necesario obtener un controlador distribuido o cooperación entre agentes [5][6].

## 3. Resultados

Para la realización de los experimentos hemos utilizado la simulación del robot descrito en la sección anterior bajo el entorno matemático Scilab. Todos los resultados presentados en este artículo corresponden a una simulación dado que el robot físico real aún no ha sido completado. A continuación se describen los resultados obtenidos. No se detallarán los valores exactos obtenidos sino que se mostrarán los resultados finales. Para una descripción más detallada de los experimentos y resultados consultar [15].

### 3.1.El sistema multi-agente obtenido

Utilizando el sistema multi-agente descrito, se procedió a evolucionar todas y cada una de las redes neuronales de los sub-agentes, con el objetivo de que el agente robótico fuese capaz de encontrar un objeto rectangular situado en su espacio y comenzase a orbitar alrededor del mismo (figura 2). Para ello, y dado que el método de entrenamiento es no supervisado, fue necesario definir una función de evaluación (fitness) que controlase la evolución simultánea de todas las redes neuronales. La función definida premiaba al robot cuando éste se encontraba moviéndose y detectando al objeto a su izquierda, a la vez que no detectaba nada enfrente del robot (moviéndose a lo largo del objeto).

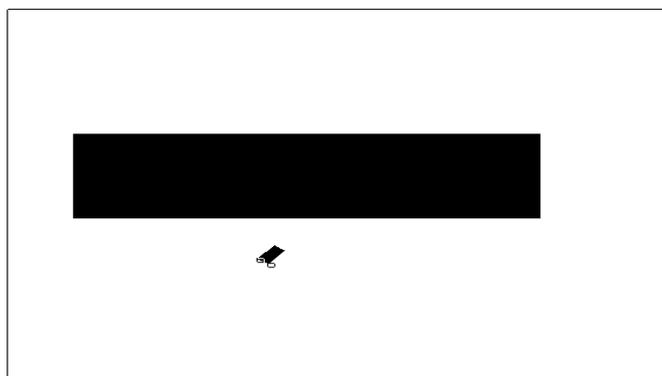


Figura 2: Muestra de la pantalla de simulación. Esta sirve para comprobar que el robot realiza correctamente la tarea asignada

El resultado fue un conjunto de redes neuronales capaces de *controlar* al robot de la forma deseada. Efectivamente, el

robot era capaz de buscar el objeto y tras encontrarlo comenzar a orbitar a su alrededor.

Resultados similares han sido presentados en [2] donde el autor generaba un neuro-controlador para el dominio presa-depredador. En este trabajo, el autor comparaba el comportamiento de 3 agentes depredadores sobre un agente presa, obtenido mediante el uso de un controlador por agente o un controlador global para todos. El trabajo muestra como el controlador global aprende mas lentamente y peor que el controlador basado en varios agentes.

Para comprobar que las mismas conclusiones eran extrapolables al caso del control de un agente robótico, realizamos también la comparación con un controlador neuronal central para nuestro robot. En este caso, el robot estaba formado por un único agente controlado por una red neuronal como la de la figura 3, controlando todos y cada uno de los sensores y actuadores del agente robótico.

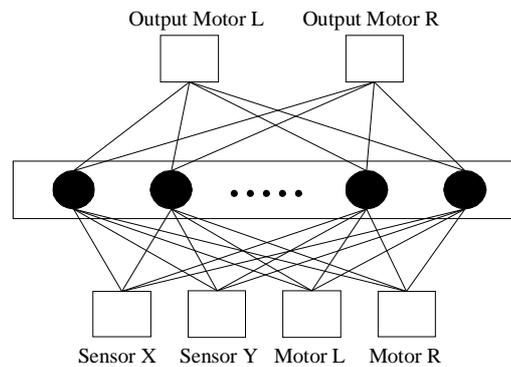


Figura 3: Neuro-controlador global. Un único agente, gobernado por esta red neuronal, controla al robot.

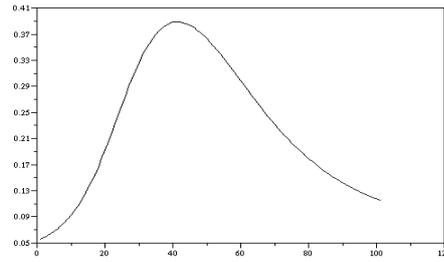
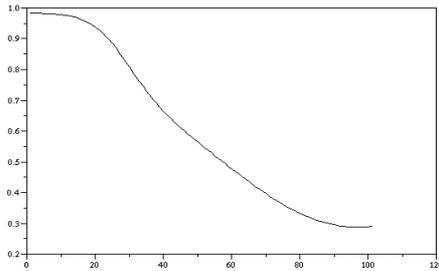
Se comparó el número de generaciones necesarias para obtener un comportamiento aceptable de rotación alrededor del objeto central. El resultado fue que el robot formado por pequeños agentes aprendía aproximadamente en la mitad de generaciones que el robot formado por un único agente. Además el robot multi-agente era capaz de afinar más en la búsqueda de la solución, llegando a obtener siempre una mayor puntuación en su función de evaluación que el robot mono-agente.

### 3.2.Los sub-agentes sensores

A pesar del éxito mostrado en el anterior apartado, podría discutirse la necesidad de utilizar una red neuronal en elementos tan pasivos como son los sensores. En principio, no está claro que el coste computacional de entrenar las redes neuronales de los sensores merezca la pena, cuando lo único que deberían hacer estos agentes es captar el valor del mundo real y pasarlo a los sub-agentes de los actuadores. Para demostrar que los sub-agentes de los sensores son realmente necesarios realizamos dos experimentos nuevos.

En el primero, evolucionamos un controlador multi-agente como el anterior, pero en el cual los sub-agentes de los sensores no existían. De esta forma, los valores obtenidos por los sensores eran pasados directamente hacia las dos únicas redes neuronales de las que se componía el agente robótico. El resultado fue que, aunque el robot seguía siendo capaz de presentar el comportamiento requerido, el fitness máximo obtenido era ligeramente menor que el del caso de 4 sub-agentes, demostrando de esta forma que los sub-agentes en los sensores realizan de alguna manera un trabajo que ayuda a obtener un controlador más inteligente. No obstante, no puede afirmarse categóricamente que dicha mejora valga el coste computacional añadido al tener 4 sub-agentes.

En el segundo experimento simulamos un mal funcionamiento de uno de los sensores infrarrojos y volvimos a generar los 4 sub-agentes. Aunque el comportamiento global del agente robótico variaba bastante, aún era capaz de ir orbitando alrededor del objeto rectangular, por lo que el sistema fue capaz de obtener un comportamiento adecuado a las circunstancias. Entonces comparamos las funciones aprendidas por el sub-agente del sensor defectuoso, cuando funcionaba correctamente y cuando funcionaba de forma incorrecta. Los resultados, mostrados en las figuras 4a y 4b, muestran como la función que aprende el mismo sub-agente depende realmente del comportamiento de su sensor asociado y de la situación en la que se encuentra



Figuras 4a y 4b: funciones aprendidas por el sensor de las X, (sólo se muestra la dependencia respecto al valor X captado), en el caso del sensor correcto (a) y en el caso del sensor defectuoso (b).

## 4. Conclusiones

Con los experimentos realizados hemos demostrado que es posible obtener un agente inteligente a partir de la unión de otros sub-agentes más simples, sin la necesidad de especificar qué es lo que debe hacer cada uno de ellos. Cuando se compara el agente obtenido de esta manera con el agente obtenido al evolucionar una única red neuronal compleja, se observa que el primero aprende más rápido y mejor que el segundo. No obstante, dado que las diferencias entre ambos fueron pequeñas y que los dos tipos fueron capaces de mostrar el comportamiento requerido, no parece existir una justificación sólida para seleccionar una de las arquitecturas frente a la otra. Sin embargo, parece lógico pensar que la arquitectura multi-agente escalaría mejor en un robot con múltiples sensores y actuadores, al mismo tiempo que sería más robusta frente a errores o mal funcionamiento defectuoso de sus elementos. La demostración de estos supuestos es un trabajo que se encuentra actualmente en desarrollo.

Además, los resultados obtenidos en la sección 3.2 muestran que es beneficioso utilizar sub-agentes en los sensores a pesar del coste computacional que ello supone. En caso de que un sensor esté defectuoso, sólo la red neuronal que extraiga la máxima información posible del sensor evolucionará, mejorando de esta manera el comportamiento general del robot, dadas las circunstancias.

Como último comentario, y en este sentido están evolucionando nuestros trabajos futuros, estos resultados nos sugieren que tal vez la mejor manera de obtener un comportamiento inteligente con un agente físico, sería especificando tan sólo una arquitectura capaz de capturar el nivel de inteligencia requerido para la tarea, pero sin especificar cómo debe ser esa inteligencia. Es decir, proporcionar las estructuras hardware y software necesarias para que el agente pueda aprehender cómo funciona el mundo exterior y cómo debe actuar entonces para conseguir su objetivo. La forma en cómo actuaría entonces el agente dependería de las estructuras proporcionadas y del entorno en el que aprendería a usarlas.

## Agradecimientos

Este trabajo ha sido parcialmente financiado a través del proyecto CCYT “Arquitectura Cooperativa de Computación Ubicua basada en elementos hardware inteligentes en el entorno de una vivienda Asistida” (ACCUA TIC2003-09179-C02-01).

## Referencias

- [1] B. Innocenti, B. López y J. Salvi, *Multi-agent System Architecture with Planning for a Mobile Robot*. Próxima publicación.
- [2] C. Han Yong. y R. Miikkulainen, *Cooperative Coevolution of Multi-Agent Systems*. Technical Report AI01-287, Department of Computer Sciences, University of Texas at Austin, 1999
- [3] D. Floreano y F. Mondada, *Evolution of Homing Navigation in a Real Mobile Robot*, IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26:396—407, 1996
- [4] D. Floreano y J. Urzelai, *Evolution and Learning in Autonomous Robotic Agents*, In Bio-Inspired Computing Systems, D. Mange and M. Tomassini, 191-36, 1997
- [5] D. Moriarty y R. Miikkulainen, *Evolving Obstacle Avoidance Behavior in a Robot Arm*. In From Animals to Animals: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior, 1996
- [6] F. Gómez y R. Miikkulainen, *Incremental Evolution of Complex General Behavior*. The University of Texas

at Austin, Technical Report AI96-248,1996

[7] F. Gómez y R. Miikkulainen, *Solving Non-Markovian Control Tasks with Neuroevolution*. Proceedings of the IJCAI99

[8] J. Rasmussen, *A general robot simulator for evolutionary robotics*. Master's Thesis in Computer Science, University of Copenhagen,2000

[9] K. O. Stanley y R. Miikkulainen, *Efficient Reinforcement Learning through Evolving Neural Network Topologies*. Machine learning, 22:11-32, 1996

[10] L. Steels. *Emergent Functionality in Robotic Agents through On-Line Evolution*, in Brooks R.A., Maes, P.(eds.): Artificial Life IV, Proc. of the Fourth Int. Workshop on the Synthesis and Simulation of Living Systems, MIT Press, 1994

[11] L. Steels y P. Vogt, *Grounding adaptative language games in robotic agents*, European Conference on Artificial Life, 1997

[12] M. Minsky, *The Society of Mind*. Touchstone Books,1988

[13] R. A. Brooks, *A Robust Layered Control System for a Mobile Robot*. In IEEE Journal of Robotics and Automation, RA-2, April 1986

[14] R. C. Arkin, *Behavior-Based Robotics*. The MIT Press,1998

[15] R. Téllez y C. Angulo, *Neuro-evolved Agent-based Cooperative Controller for a Behavior-based Autonomous Robot*, en 1a Jornada de Recerca en Automatica, Visio i Robotica, 2004

[16] T. Balch, *Learning roles: behavioral diversity in robot teams*, 1997