ARQUITECTURA DE UN CONTROLADOR HÍBRIDO PARA NAVEGACIÓN ROBÓTICA EN AMBIENTES PARCIALMENTE CONOCIDOS

Oscar Goñi José A. Fernández León* Nelson Acosta

Instituto INTIA - Dept. Computación y Sistemas
Facultad de Ciencias Exactas - Universidad Nacional del Centro de la Provincia de Buenos Aires
*CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina
(7000) Tandil - Buenos Aires - Argentina
{ oegoni, jleon, nacosta }@ exa.unicen.edu.ar

RESUMEN

En el presente trabajo se describe un sistema híbrido de navegación robótica. La denominación de híbrido se refiere a que se incorporan aspectos de control reactivo y deliberativo simultáneamente. Para el diseño del control reactivo se utilizó una metodología recientemente nueva basada en los sistemas biológicos llamada Robótica Evolutiva. Esta permite generar neuro-controladores para navegación reactivos de manera automática. Por otra parte se diseñó un sistema deliberativo el cual, implícitamente, se le indica al robot un objetivo físico a ser alcanzado (mojón). Un objetivo en particular es diferenciado del resto utilizando un sistema de visión el cual reconoce parches de colores. Las pruebas del sistema fueron llevadas a cabo en simulación, cuando fue posible, y luego sobre diversos robots reales. En las experiencias se pudo comprobar la eficacia del sistema reactivo y algunas deficiencias en el deliberativo.

Palabras claves: Robótica Evolutiva, mecanismos de coordinación, sistemas híbridos.

1-INTRODUCCIÓN

La Robótica Evolutiva es una nueva metodología para la creación automática de robots autónomos [3]. Inspirado en el principio Darwiniano de la reproducción selectiva del más apto es posible ver a los robots como organismos artificiales que pueden desarrollar sus propias habilidades sin la intervención humana. Por otra parte, basado principalmente en aspectos biológicos, son utilizadas redes neuronales, algoritmos genéticos, sistemas dinámicos, e ingeniería computacional para llevar a cabo el proceso de control robótico. Algunas de las características presentes en los robots desarrollados con dicha metodología, al igual que los sistemas biológicos simples, involucran características de robustez, simplicidad, flexibilidad y modularidad [3].

En los últimos años, en la mayoría de los experimentos en Robótica Evolutiva son utilizados sistemas de control robotico mediante redes neuronales artificiales [1, 3, 4, 5]. La arquitectura de dichas redes incluyen desde conexiones recurrentes, las cuales permiten contemplar aspectos de dinámica temporal [6, 7], hasta arquitecturas simples del tipo Feed-Fordward [3]. Estudios sobre la combinación de algoritmos evolutivos y aprendizaje senso-motor puede ser encontrados en [8, 9].

Sin embargo, uno de los principales desafíos que afronta la robótica basada en comportamientos es el diseño de controladores que implementen comportamientos complejos. Estudios previos [11, 13] son algunos de los trabajos que plantean formas de coordinación para lograr dichos comportamientos. Mientras que el primero de tales trabajos propone una estructura de coordinación de comportamientos fija y estipulada por el programador, el segundo propone un enfoque evolutivo mediante evolución artificial para dicha coordinación. Por otro lado [12] plantea usar redes neuronales con pesos fijos para lograr comportamientos complejos. A modo general, es posible decir que [11] y [12] dependen de gran medida que las prioridades asignadas a cada comportamiento sea la correcta, responsabilizando al programador de dicha coordinación. En cambio [13] propone un esquema automático para coordinación de comportamientos.

En el presente trabajo se describe el diseño de un sistema de navegación robótico considerando en su parte reactiva un enfoque que adopta las características más deseadas de los esquemas mencionados como son robustez, simplicidad, flexibilidad y modularidad. Siguiendo el mismo lineamiento que para lograr los comportamientos simples, se utiliza la evolución artificial para lograr coordinación de comportamientos simples mediante el uso de redes neuronales para tal fin.

En el proyecto aquí descrito, en general, se involucra control reactivo basado en el sensado del entorno para realizar la tarea de navegación y control deliberativo mediante visión para resolver el problema de determinar el camino a seguir.

En la siguiente sección se describen con más profundidad los objetivos del trabajo.

2- OBJETIVOS

A lo largo del proyecto que a continuación se describe, se plantea como principal finalidad estudiar e implementar una arquitectura estándar para navegación robótica automatizando, en lo posible, el desarrollo de

la misma. Un trabajo de similares características es el realizado en [14] el cual propone un enfoque de coordinación multiagente para lograr navegación compartiendo recursos y a la vez compitiendo por ellos. Sin embargo éste trabajo plantea una arquitectura para control de navegación para una plataforma robótica específica y no para la navegación de robots en general.

Para éste proyecto en particular, entre las características deseadas del control de navegación se destacan la robustez (funcionar en condiciones adversas) y adaptación al entorno. El proyecto aquí presentado se divide en etapas bien definidas. La primera se trata del *diseño reactivo del control a realizar sobre el robot*, el cual es el encargado de guiar al robot por lugares no conocidos mediante el uso de sensores. La segunda etapa presupone *la identificación del entorno* por medio de visión artificial, la cual controla el *sistema deliberativo*, siendo éste el desarrollo a considerar en la última etapa.

Cabe destacar que en la segunda etapa, el sistema reconoce mojones de colores aunque el diseño más avanzado soportará identificación de objetos o códigos de barras como se propone en [14]. En las siguientes secciones se describen las diferentes etapas, así como los resultados parciales del proyecto.

3 - PRIMERA ETAPA: Diseño del módulo reactivo

Una de las ventajas que posee la robótica basada en comportamientos es que no necesita representación del mundo real para ejecutar una tarea [10]. Por esa razón, y por otras no mencionadas en este artículo, es que se implementa en esta etapa un módulo que usa sólo información sensorial para realizar una tarea.

Para la implementación de un controlador reactivo se utilizaron redes neuronales. Un controlador neuronal o neuro controlador posee varias neuronas en una capa de entrada, las cuales son alimentadas por los valores de los sensores. Estas neuronas se encuentran totalmente conectadas a una capa intermedia de neuronas, las cuales a su vez se encuentran de igual forma conectadas a las neuronas de salida. Estas últimas entregan valores a los actuadores del robot.

La idea de aplicar Redes Neuronales artificiales para control robótico ha sido aplicada en diferentes trabajos [18,19], aunque los pesos de dichas redes están ajustados por el programador y su desempeño depende de la experiencia previa del mismo. Es por eso que la Robótica Evolutiva [3] propone someter a los pesos de la red a un proceso evolutivo. Un proceso evolutivo consta de varias etapas donde existe una población inicial aleatoria. Cada uno de los individuos de la población, es probado en el entorno y su desempeño es cuantificado. Los mejores individuos de la población pasarán a la siguiente generación en forma de clones o mutaciones de ellos. Este proceso se realiza tantas veces como el programador crea conveniente o bien cuando se haya alcanzado un valor determinado de desempeño.

Mediante la aplicación de un mecanismo de evolución artificial, se modelan controladores para tres comportamientos sencillos. Estos son, *fototaxis* (ir hacia la luz), *seguimiento de paredes*, y *evasión de obstáculos*. Cada uno de los comportamientos es logrado eligiendo individuos de una población que representan potenciales neuro controladores a gobernar las acciones del robot en relación a la tarea para la cual se los evoluciona. Una vez logrado el rendimiento deseado, las redes son congeladas, o sea que no sufrirán ningún tipo de cambio en sus pesos sinápticos. La próxima tarea es lograr un comportamiento complejo, es decir, coordinar los comportamientos simples obtenidos. Siguiendo con la metodología de evolucionar los pesos de las redes, una red de coordinación es creada y sometida al mismo proceso evolutivo. En este caso, las entradas a la red de coordinación pertenecen a los valores de activación de las neuronas de salida de los comportamientos simples. Las neuronas de salida en cambio siguen describiendo el valor de los actuadores al igual que en los comportamientos simples. Una tarea no trivial es hallar la métrica que permita comparar el desempeño de los individuos, dado que la función objetivo de un comportamiento suele deferir de las otras. Como métrica preliminar se utilizó la suma de los desempeños de los comportamientos simples obteniéndose buenos resultados. La evolución de los neuro controladores tanto de los comportamientos simples como la coordinación fue llevada a cabo siguiendo los parámetros que se muestran en la tabla 1. Una vez probados los controladores en un entorno simulado simple, se pasó a experimentar con un simulador más realista y finalmente en robots reales (veáse [13] para mayor detalle).

Configuración de la Red Neuronal Artificial (RNA)	
Tipo	Feed Forward
Neuronas de entrada	8
Neuronas en capa oculta	6
Neuronas de salida	2
Configuración de la evolución	
Generaciones	300
Corridas	10
Iteraciones (Pasos / Corrida)	200
Individuos	30
Operadores Genéticos	Mutación: 5%, Selección: 50%, Elitismo: 5 primeros

Tabla 1 **– Configuración para la evolución de comportamientos simples y coordinación**

4 - SEGUNDA ETAPA: Diseño del Sistema de Visión

Esta parte del proyecto aún se encuentra poco desarrollada. Sin embargo se ha implementado un sistema de detección de mojones o parches de colores que ha mostrado resultados aceptables en ambientes semi controlados. Si bien el procesamiento de la imagen captada es sencillo, dado que se estaba en una etapa de

experimentación, fue suficiente para identificar a un mojón a varios metros de distancia. La responsabilidad del módulo de visión es informar la posición relativa de un mojón determinado. Esto implica que cualquiera que sea el criterio de identificación no modifica la integridad del sistema. Sin embargo, en una próxima etapa del robot hacia su objetivo, éste sería capaz de identificar mojones con códigos de barras, formas u objetos.

5 - TERCER ETAPA: Diseño del Sistema deliberativo

La responsabilidad de este módulo es guiar al robot por un camino conocido. Para esto, el sistema posee un grafo con la disposición topológica de los mojones señalados anteriormente. De esta manera, el sistema deliberativo guiará al robot por el camino más corto o bien por el menos costoso. Otra responsabilidad de este módulo es realizar correcciones en el grafo. Esta situación se da por ejemplo cuando el robot está dirigiéndose a un mojón y se le antepone un obstáculo fijo, por lo que el camino que era considerado se ve obstaculizado y debe ser recordado en un futuro. En este momento se le cede el control al módulo reactivo. Una vez llegado a otro mojón, se vuelve a tomar el control y se replantea el recorrido desde ese mojón. En *la figura 1* se muestra la arquitectura completa del sistema de navegación.

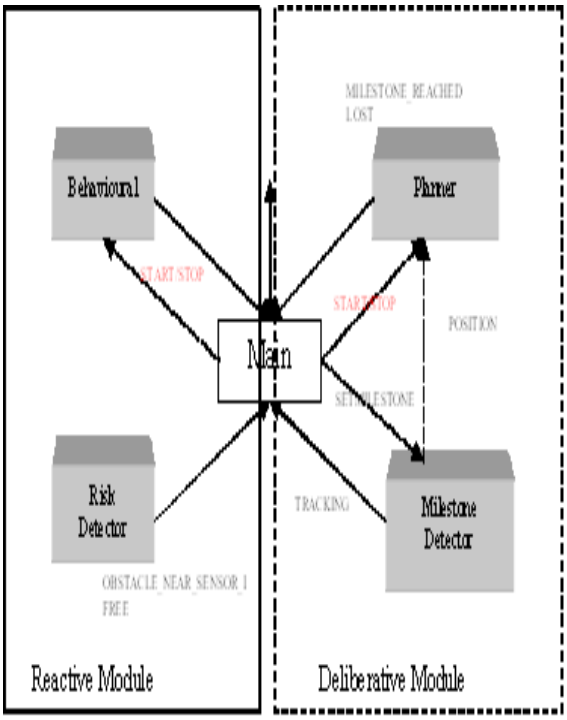

Figura 1 **– Arquitectura actual del Controlador**

6 - EXPERIENCIAS Y RESULTADOS PARCIALES

Varias de las experiencias se realizaron o se esperan realizar por separado etapa por etapa, para luego considerar el conjunto de ellas a fin de ajustarlas a la finalidad deseada: navegación autónoma. Para ello, cada uno de los comportamientos fueron evaluados por separado para validar su correcto funcionamiento. Luego, una vez logrado todos los comportamientos pretendidos, se realizaron pruebas, en un principio, con varios simuladores, entre ellos Yaks [15]. Posteriormente se probó en el robot Huerken Kellu [17] obteniendo similares resultados a los de la simulación. La etapa en la que se desarrolla el sistema de visión, requiere otro tipo de pruebas, por ejemplo, probar con distintos tipos de cámaras, con distinta iluminación y distintos colores de mojones. La tercer etapa de pruebas consiste en comprobar el funcionamiento del control deliberativo. Al igual que en el sistema de visión, las pruebas fueron realizadas fuera del mundo de la simulación. Esto se debe a que no se contaba con un simulador que contara con dicha funcionalidad. Por tal razón fue que se le adicionó una cámara a un robot más pequeño, de manera tal que pueda ser controlado en un entorno reducido, altamente controlado y libre de obstáculos. El robot en cuestión es del tipo Victo de Yujin [16].

Una vez verificado que era posible guiar al robot por el entorno, se realizó una prueba mixta donde los valores suministrados a la parte reactiva provenían del simulador Yaks, mientras que la imagen era captada por una cámara real y los comandos de salida eran transmitidos a un robot Victo. El robot fue colocado en un rincón de la arena, tanto en el simulador como en el entorno real, procurando dejar el robot simulado en el mismo ángulo que el real. Una vez iniciado el control tanto el robot simulado como el real comenzaron a navegar. El seguimiento de paredes coincidió aunque en el primer obstáculo evadido, hubo que reubicar el robot ya que los ángulos de desviación final no coincidían.

Finalmente, se probó el sistema de navegación con el robot Huerken Kellu. Para esta prueba el ambiente fue menos controlado ya que se encontraba expuesto a obstáculos de formas no homogéneas, obstáculos móviles, variaciones de luz, etc. Si bien el procesamiento se realizó en un nodo externo, los datos fueron intercambiados por medio de una red inalámbrica. Esta configuración limita la velocidad de actualización de los movimientos del robot ya que se depende de la velocidad máxima de transmisión de datos por la red. Por otro lado, la potencia de cálculo depende solo de procesamiento del nodo externo, lo que posibilitó hacer procesamiento de imágenes en tiempo real y actualización de las redes sin inconvenientes. Vale aclarar que en todas las pruebas se utilizó el mismo sistema de navegación.

7 - CONCLUSIONES Y TRABAJO FUTURO

En el presente proyecto se plantea una arquitectura estándar para navegación robótica. Las partes de la

misma fueron generadas por etapas. La primera etapa consistía en la generación automática, en la medida de lo posible, de un controlador reactivo mediante evolución de redes neuronales. A pesar de contar con una estructura de red neuronal sencilla, se obtuvieron los resultados esperados tanto en los diversos entornos simulados como en el robot real. Por otro lado, en el robot real se pudo observar que los neuro controladores tuvieron un desempeño aceptable aún frente a los ruidos causados por el ambiente real.

El uso de un nodo de procesamiento externo implica demoras en la transmisión de los datos, lo que causó reacciones tardías en ciertas ocasiones. El costo asumido para subsanar el error es la disminución de la velocidad de actualización de movimientos del robot. Como trabajo futuro se pretende programar al robot con la arquitectura y solo actualizar ciertos parámetros, como por ejemplo, los pesos de las redes que controlan algún comportamiento. Otra de las partes de la arquitectura que se debe mejorar es el sistema de visión. A pesar de que este tuvo un correcto funcionamiento, la cantidad de colores que es reconocida es limitada. Como se mencionó anteriormente, seria conveniente identificar a los mismos por códigos de barras o mejor aún de objetos. En un futuro un poco más distante, la incorporación de un sistema de visión 3D con soporte de Hardware adicional, seria conveniente para una correcta identificación y modelamiento más real del entorno

Por otro lado, se está trabajando para que el sistema de visión forme parte del módulo reactivo y mediante evolución ajustar comportamientos. De ésta manera el módulo deliberativo estaría encargado solamente de armar y modificar planes de ruta, así como recordar experiencias pasadas.

Otro de los trabajos a desarrollar en el futuro, implican la verificación del sistema presentado en otros ambientes no controlados, tales como ambientes externos.

8 – REFERENCIAS

[1] Urzelai J. and Floreano D.. *Evolutionary Robotics: Coping with Environmental Change.* Proceedings of the Genetic and Evolutionary Computation Conference. 2000.

[2] Di Paolo, E. A. *Evolving spike-timing dependent plasticity for robot control*. EPSRC/BBSRC International Workshop: Biologically-inspired Robotics, The Legacy of W. Grey Walter, WGW'2002. HP Labs, Bristol, 2002.

[3] Nolfi S. and Floreano, D. *Evolutionary Robotics: The biology, Intelligence, and Technology of Self-Organizing Machines*. MA: MIT Press/Bradford Books. 2000.

[4] Tuci E., Harvey I., and Quinn M. *Evolving integrated controllers for autonomous learning robots using dynamic neural networks*. Proceedings of The Seventh International Conference on the Simulation of Adaptive Behavior (SAB'02), 4-9 August 2002, Edinburgh, UK

[5] Di Paolo, E. A. *Evolving robust robots using homeostatic oscilators*. Cognitive Science Research Paper 548, COGS, University of Sussex. 2002.

[6] Blynel J. and Floreano D. *Levels of Dynamics and Adaptive Behavior in Evolutionary Neural Controllers*. In B. Hallam, D. Floreano, J. Hallam, G. Hayes, and J.-A. Meyer, editors. Recompilation de papers. 2002.

[7] Beer, R. D. and Gallagher, J. C. *Evolving dynamical neural networks for adaptive behaviour. Adaptive Behavior*, 1(1):91-122. 1992.

[8] Floreano D. and. Urzelai J. I. *Evolution and learning in autonomous robots*. In D. Mange and M. Tomassini, editors, Bio-Inspired Computing Systems. PPUR, Lausanne, 1998.

[9] S. Nolfi and D. Floreano. *Learning and Evolution. Autonomous Robots*, 7(1): forth-coming, 1999.

[10] Arkin, R. C. *Behavior Based Robotics*, MIT Press, 1998

[11] Brooks, R. A Robust Layered Control System for a Mobile Robot IEEE Journal of robotics and Automation Vol. RA-2 No.1, pp 14-23, 1986

[12] Payton, D "Do whatever Works: A robust approach to falult-tolerant Autonomous Control, MIT Press

[13] Fernández León J., Acosta, G. and Mayosky, M. *An Experimental Study of Scaling Up in Mobile Robot Motion Control using Evolutionary Neuro-Controllers, IEEE* 2006

[14] Font D. *A multiagent Approach to Qualitative Navigation in Robotics*, Universidad Politéctnica de Cataluña, 2005.

[15] Karlson, J. YAKS - Yet Another Khepera Simulator. URL: http://r2d2.ida.his.se/. 2002.

Yujin Robotics. Victo Robot. 2006 http://www.edrobot.com/english/product/ysra.asp

17 Corán D., Sorroche, G., Huerken Kellu: Diseño e implementación de un robot móvil autónomo. UNCPBA, 2003

18 Frank L., *Neural Network Control of Robot Manipulators*, IEEE Expert: Intelligent Systems and Their Applications, v.11 n.3, p.64-75, June 1996

19 .Buessler J. L, Urban J. P., *Modular neural architectures for robotics, Biologically inspired robot behavior engineering*, Physica-Verlag GmbH, Heidelberg, Germany, 2003