

Integraci'ón de Controladores Difusos en un Robot Aut'onomo

Vicente Matell'an

Lab. de Agentes Inteligentes
Departamento de Ingenier'ia
Universidad Carlos III de Madrid
e-mail: vmate@ing.uc3m.es

Jos'e Manuel Molina

Lab. de Agentes Inteligentes
Departamento de Ingenier'ia
Universidad Carlos III de Madrid
e-mail: molina@ing.uc3m.es

Camino Fern'andez Llamas
CETTICO

Universidad Polit'ecnica de Madrid
e-mail: camino@fi.upm.es

Jos'e Manuel Bl'azquez

Lab. de Agentes Inteligentes
Departamento de Ingenier'ia
Universidad Carlos III de Madrid
e-mail: jblazque@zenon.ing.uc3m.es

Abstract

En este trabajo se propone una metodolog'ia que permite integrar distintos comportamientos difusos de un robot aut'onomo real. Esta integraci'ón se realiza a trav'es de una arquitectura formada por dos niveles. El primer nivel est'a compuesto por diferentes controladores borrosos, cada uno especializado en una tarea distinta. As'i definido, el sistema completo se basa en una aproximaci'ón reactiva, es decir, la decisi'ón de que acciones realizar se toma de forma continua en funci'ón de la informaci'ón que se recibe de los sensores.

El segundo nivel se encarga de decidir cómo generar el comportamiento global a partir de las propuestas de los distintos módulos del nivel inferior. Las decisiones de este módulo se obtendrán mediante razonamiento borroso. Los resultados obtenidos tanto en simulación como en el robot real muestran un comportamiento más eficiente en este sistema que en uno que seleccione la acción básica más prometedora en cada situación.

Robot autónomo, comportamiento difuso, integración, multinivel.

1 Introducci'ón

Desde los primeros pasos de la Inteligencia Artificial se ha abordado el problema de construir robots móviles autónomos capaces de realizar tareas útiles en el mundo real. Sin embargo, las tentativas basadas en los paradigmas clásicos como la abstracción, planificación, búsqueda heurística, etc., como por ejemplo [0], no tuvieron mucho éxito y este campo permaneció en un estado teórico hasta finales de los años 80.

En los últimos años ha tenido lugar un profundo cambio y la mayoría de los robots se construyen ahora en base a un nuevo paradigma: *reacción*, que sostiene que la “inteligencia” de un sistema es el resultado de la interacción del sistema con el entorno. Así, el comportamiento inteligente de un robot autónomo debería surgir como resultado de su interacción con el mundo real a través de sus sensores y actuadores.

La arquitectura que se presenta se basa en dos premisas. La primera es que los robots autónomos deben confiar sus actividades básicas a una aproximación reactiva [0]. Esto quiere decir que el comportamiento global de un robot debe ser el resultado de una fusión no óptima de diferentes reacciones básicas, que denominaremos *comportamientos*. Los comportamientos están diseñados para operar continuamente sobre los actuadores del robot en función de las informaciones proporcionadas por los sensores, con un objetivo concreto, por ejemplo, evitar obstáculos, seguir un objeto, etc.

La segunda premisa es que la lógica difusa [0] ha demostrado ser un entorno robusto para el desarrollo de controladores eficientes en aplicaciones diversas, como por ejemplo [0], [0]. Por tanto, los comportamientos a los que nos hemos referido en los párrafos anteriores pueden construirse como grupos de reglas difusas. Así, en la literatura han aparecido últimamente distintas aproximaciones basadas en controladores borrosos al control de robots, como por ejemplo, [0], [0], [0].

Nuestra propuesta se desarrolla en las cinco secciones siguientes. En la segunda sección se utiliza el razonamiento borroso dentro de este nuevo paradigma para la construcción de robots autónomos. En el tercer apartado se define de forma teórica la arquitectura del sistema propuesto. La cuarta sección es una implementación sobre un robot real de la arquitectura anterior. El siguiente

file=sf-esp.eps,width=8.2cm

Figure 1: Proceso de Fusi'on Borrosa

apartado muestra los resultados obtenidos con dicha implementaci'on y en el 'ultimo apartado se presentan las conclusiones y las futuras l'ineas de trabajo.

2 Robots Reactivos

Se pueden considerar dos aproximaciones reactivas distintas al control de un robot aut'onomo. La primera de ellas consideraria la totalidad de la informaci'on aportada por los sensores para obtener la salida m'as adecuada a la situaci'on en que se encuentra el robot mediante un conjunto de reglas que utilizan las entradas de forma conjunta. Una implementaci'on borrosa del sistema anterior conllevaria la generaci'on de un sistema de reglas muy complejo, tanto en n'umero de reglas (situaciones a considerar) como en complejidad de las mismas, debido a la cantidad de sensores a considerar.

La segunda de las arquitecturas utiliza una descomposici'on por tareas. Es decir, cada comportamiento debe decidir por s'i mismo qu'e hacer en cada instante utilizando un subconjunto reducido de la informaci'on suministrada por los sensores. Una instancia clasica de esta clase de arquitecturas es la *subsumption architecture* [0], [0], donde las salidas de los distintos m'odulos se integran en base a la inhibici'on de aquellas menos relevantes. La *subsumption architecture* utiliza m'aquinas de estado finito para implementar los comportamientos.

La aportaci'on del trabajo que se presenta reside en una aproximaci'on diferente al segundo tipo de arquitectura, bas'andose en la l'ogica difusa. En primer lugar, se han dise nado los comportamientos de forma borrosa, debido a que se considera que las reglas difusas permiten utilizar conceptos imprecisos de modo m'as intuitivo que, por ejemplo, las m'aquinas de estado finito. En segundo lugar, se ha desarrollado un m'etodo difuso para fusionar el conjunto de salidas de los distintos comportamientos, en vez de, por ejemplo, elegir una de ellas mediante inhibiciones.

3 Razonamiento Fuzzy Multinivel

El objetivo de este trabajo es mostrar que un comportamiento complejo en un robot puede emerger de comportamientos simples implementados como conjuntos de reglas difusas. La arquitectura presentada en la figura tendr'a n acciones b'asicas implementadas como controladores fuzzy, FLC (*Fuzzy Logic Controller*) y un m'odulo encargado de decidir c'omo se formar'a la acci'on global a partir de las propuestas por las distintas unidades reactivas.

El primer paso para construir el sistema completo es dise nar cada FLC

file=khepera.ps,width=8.2cm

Figure 2: El mini-robot Khepera

individualmente. Cada uno de ellos deber'a ser capaz de controlar el robot para que consiga la tarea de la que se encarga. Para dise nar cada uno de esos comportamientos se necesitar'a [0]:

- Definir las variables de entrada y salida. Esto es, determinar qu'e fen'omenos se observar'an y cuales son las acciones de control que se pueden tomar.
- Definir la forma en que las observaciones del mundo se expresan como conjuntos difusos.
- Dise nar la base de reglas.
- Determinar la forma en la que las salidas fuzzy se transforman en acciones de control.

Las entradas del sistema son percepciones del mundo, pero no son las mismas para todos los FLC. Cada comportamiento se basa en una visi'on particular del mundo, por ejemplo, el comportamiento *evitar obst'aculos* tiene que ser consciente de las distancias a los objetos que rodean al robot, mientras que un comportamiento como *clasificar objeto* necesitar'a reconocer los bordes del objeto para clasificarlo seg'un las formas predefinidas almacenadas en el robot.

Para terminar el primer paso se prueba cada FLC en un entorno preparado para generar 'unicamente los est'imos que activan uno de los comportamientos. Estos experimentos parciales son necesarios para afinar cada FLC y conseguir que la reacci'on actue de forma adecuada. En esta fase es posible modificar las entradas, las funciones de pertenencia y la base de reglas.

El segundo paso es construir un sistema capaz de encontrar una buena salida para una determinada situaci'on del robot en el mundo. La primera aproximaci'on ser'ia un sistema que eligiese solamente un comportamiento, es decir, la salida de un comportamiento. Este m'odulo de decisi'on ha sido denominado **Sistema 1/n** (elecci'on de 1 sobre n posibles). Por ejemplo:

SI

(Situaci'on **ES** *situaci'on-1*)

ENTONCES

(Comportamiento **ES** *comportamiento-1*)

La dificultad de utilizar esta aproximaci'on radica en establecer los l'imites entre las distintas situaciones percibidas. Una soluci'on podr'ia ser el uso de una definici'on difusa de las situaciones. La representaci'on borrosa de las situaciones permite evaluar el grado de correspondencia entre una situaci'on real y

file=sensores-esp.eps,width=7cm

Figure 3: Disposici'ón f'isica de los sensores

la situaci'ón predefinida. En este tipo de sistemas, la salida es una combinaci'ón de las respuestas predefinidas para cada situaci'ón en funci'ón del grado de correspondencia. Por ejemplo, en [0] se presenta un sistema donde se realiza una integraci'ón fuzzy de acciones predefinidas, no obtenidas mediante razonamiento borroso, para cada una de las situaciones posibles, estas respuestas son del tipo: parar y girar a la izquierda, acelerar y girar a la derecha etc. Este sistema estar'a implementado mediante reglas del tipo:

SI
(Situation-fuzzy-1 **ES** *Alta*)
ENTONCES
(Parar **Y** Girar-izquierda)

En nuestra arquitectura no hay acciones con valores predefinidos porque cada uno de los comportamientos b'asicos obtiene la salida apropiada a cada situaci'ón mediante un razonamiento borroso. Para ello los comportamientos b'asicos se implementan como FLCs. La integraci'ón de las acciones obtenidas de los FLCs se basar'a en el grado de correspondencia borroso entre las situaciones reales y las predefinidas. La integraci'ón se realizar'a mediante reglas del tipo:

SI
situation-fuzzy-1 **ES** *Alta*
ENTONCES
(comportamiento-1 **ES** *Alto*) **Y**
(comportamiento-2 **ES** *Bajo*)
:
SI
(situation-fuzzy-n **ES** *Alta*)
ENTONCES
(comportamiento-1 **ES** *Bajo*) **Y**
(comportamiento-2 **ES** *Muy-alto*)

Este sistema es capaz de obtener una respuesta que corresponde a un estado intermedio de todos los FLCs, obteniendo la importancia de cada comportamiento en la situaci'ón actual. Para mezclar las salidas num'ericas es posible hacer una combinaci'ón lineal. Por tanto, si cada comportamiento tiene una salida v_i y el m'odulo de decisi'ón asigna un grado de adecuaci'ón s_i del comportamiento i a la situaci'ón percibida, la salida fuzzy viene dada por la f'ormula

1, donde los pesos α_i se calculan mediante un proceso de normalizaci3n de los s_i .

$$v_{total} = \sum_{i=1}^n (\alpha_i \times v_i) \quad (1)$$

Para fijar estos grados de adecuaci3n se tiene que utilizar un conjunto de entradas. Estas entradas dependen del objetivo del robot y del n3mero de comportamientos implementados. En particular, para un robot con dos comportamientos, *perseguir un objeto* y *evitar obst3culos*, la entrada del m3dulo de decisi3n ser3a la descripci3n del entorno. En la siguiente secci3n presentamos la descripci3n completa de una implementaci3n real.

4 Una arquitectura real de dos niveles

La arquitectura te3rica presentada en la secci3n anterior se ha desarrollado en un simulador y luego se ha probado, adapt3ndola, en un robot real. Se ha utilizado un simulador del robot real, denominado SRA (Simulador de Robots Aut3nomos), desarrollado en la Universidad Carlos III. Este simulador nos permite dise 1ar un entorno sencillo para probar distintas implementaciones de un comportamiento antes de evaluar las mejores en el robot real.

El robot que se ha utilizado en los experimentos es el mini-robot denominado Khepera [0], desarrollado en el LAMI-EPFL (Suiza). El Khepera, figura , es un mini-robot de 5.5 cm. de di3metro dise 1ado como demostrador tecnol3gico. Tiene ocho sensores de proximidad basados en infra-rojos, que tambi3n sirven como sensores de luz ambiente y dos motores con contadores. Los sensores devuelven un valor num3rico entre 0 y 1023, dependiendo de si se encuentra lejos o cerca del objeto. La distribuci3n de los sensores se puede observar en la figura . As3i mismo, tiene la posibilidad de funcionar de forma aut3noma, o bien, conectado a un ordenador mediante un cable serie.

En las siguientes secciones se describe el sistema en dos pasos. En la primera se aborda cada uno de los comportamientos individualmente. En la segunda, se analizan las distintas posibilidades para la realizaci3n del m3dulo de fusi3n, comenzando con el que denominaremos Sistema 1/n y analizando a continuaci3n el proceso de integraci3n fuzzy.

4.1 Los comportamientos individuales

El primer comportamiento considerado se ha denominado *evitar obst3culos*. Este comportamiento permite al robot deambular por el mundo sin chocar con ning3n obst3culo. El sistema borroso desarrollado, que define este comportamiento, se ha probado en primer lugar sobre el simulador SRA. En la figura se muestra la estructura del controlador difuso que implementa el comportamiento, es decir, las funciones de pertenencia de las variables de entrada, las

file=avoid-esp.eps,width=13cm

Figure 4: FLC “evitar obst’aculos”

tablas que muestran las reglas de control y las funciones de pertenencia de las variables de salida.

En el proceso de *fuzzificaci’on* se pueden observar los valores concretos de la variable *distancia*, su rango abarca de 0 a 1023 que es el rango de valores devueltos por los sensores, este rango se ha dividido en s’olo tres etiquetas debido a la pobre discriminaci’on de los sensores. Las reglas se muestran en las tablas del segundo bloque de la figura, as’i por ejemplo, la regla correspondiente al primer cuadro de la tabla de la izquierda se interpretar como:

SI

(Sensor1 **ES Cerca**) **Y**

(Sensor2 **ES Cerca**) **Y**

(Sensor3 **ES Cerca**)

ENTONCES

(Motor1 **ES Atras**) **Y**

(Motor2 **ES Atras**)

El rango de valores de la variable *velocidad* va de -10 a 10 que es el rango al que hemos limitado la velocidad de los motores. Sobre dicho rango se han definido tres etiquetas, *Atras*, *Stop* y *Delante*, que son suficientes para controlar el robot.

En el caso del comportamiento *evitar obst’aculos*, la variable *distancia* se define con las mismas funciones de pertenencia para los valores de entrada suministrados por los tres sensores. En el caso de que la curva de respuesta de alguno de los sensores fuese distinta, por ejemplo, por deficiencias de fabricaci’on, habr’ia que variar las funciones de pertenencia para ajustarlas al valor real.

A continuaci’on, se ha desarrollado otro comportamiento denominado *seguir objeto* que hace que el robot siga un objeto. Para implementar este instinto primario en simulador se han creado dos sensores que no existen en el robot real y que se han denominado *distancia* y *’angulo*. Dichos sensores devuelven la distancia al objeto a seguir y el *’angulo* que presenta con respecto a la direcci’on del robot. En el robot real se ha utilizado una fuente de luz como objetivo a seguir y se han utilizado los sensores de luz ambiente para determinar la distancia y el *’angulo* con respecto a dicha fuente.

4.2 Integraci’on de comportamientos

Simplemente con los dos comportamientos descritos en el apartado anterior, ya es necesario un m’odulo que integre las respuestas. Este ejemplo sirve tambi’en para ilustrar c’omo los distintos comportamientos tienen distintas entradas.

Así, el encargado de evitar obstáculos tiene como entradas las medidas de los sensores de proximidad, mientras que el encargado de seguir un objeto recibe como entradas el ángulo y la distancia al objeto. Las salidas de ambos es la misma; la velocidad de cada una de las ruedas.

Una primera posibilidad de realizar la fusión de las salidas de ambos comportamientos es elegir la mejor salida para cada situación de entre las propuestas por los comportamientos, siguiendo la filosofía de elegir la salida del comportamiento *seguir objeto* mientras no haya peligro de colisión. A esta aproximación, basada en la *subsumption architecture* de R.A. Brooks, es a la que hemos denominado el *Sistema 1/n*. La segunda posibilidad es realizar una fusión fuzzy como se describe en la sección anterior. A esta segunda arquitectura es a la que denominaremos *Sistema Fuzzy*.

5 Experimentación

Para probar la arquitectura se ha programado el sistema sobre el simulador. Se han implementado ambos comportamientos y se ha procedido a generar un conjunto de escenarios mediante la modificación del entorno: tanto las posiciones de partida como el número y la forma de los objetos. En concreto, se han analizado exhaustivamente doce escenarios que incluyen diferentes situaciones. En cada uno de estos escenarios se ha medido la respuesta de los dos sistemas: $1/n$ y difuso.

Los resultados obtenidos se evalúan a través de la medida del tiempo necesario para alcanzar el objetivo, la distancia recorrida por cada uno de los sistemas y las oscilaciones, medidas como el sumatorio de las variaciones de la dirección. Como ejemplo se presentan las distintas trayectorias, figura , en cuatro escenarios distintos. En la figura puede observarse cómo el sistema borroso sigue trayectorias con menos oscilaciones y por consiguiente la distancia recorrida es menor. El menor número de oscilaciones se debe a la transición suave de un comportamiento a otro en el sistema fuzzy. Además este sistema hace que la trayectoria del robot vaya más próxima al contorno de los obstáculos, lo cual hace que el tiempo y la distancia recorrida para alcanzar la meta sean más cortos en el sistema difuso que en el $1/n$.

En la implementación sobre el robot real se han utilizado todos los sensores agrupándolos según se muestra en la figura . En particular, para el comportamiento *evitar obstáculos* se han utilizado tres agrupaciones de los seis sensores frontales de distancia, que en la figura se denominan como de *proximidad*. Para el comportamiento *seguir objeto*, se han agrupado los ocho sensores de luz ambiente en cuatro: tres frontales y uno trasero, denominados en la figura como de *ambiente*. La integración de las salidas de ambos comportamientos hace que el robot sea capaz de seguir una fuente de luz, evitando los obstáculos que encuentra en su movimiento.

Sobre el robot real los resultados se evalúan en base a los mismos crite-

Figure 5: Comparación de dos sistemas

Escenarios	escenario1		escenario2	
Sistema	Fuzzy	1/n	Fuzzy	1/n
Tiempo (seg.)	205	211	159	160
Distancia (mm.)	508	432	348	339
Oscilaciones	9	17	6	7

Escenarios	escenario3		escenario4	
Sistema	Fuzzy	1/n	Fuzzy	1/n
Tiempo (seg.)	192	195	235	215
Distancia (mm.)	474	409	543	457
Oscilaciones	8	16	8	15

Table 1: Resultados de los experimentos reales.

rios que en el simulador (distancia, tiempo y oscilaciones), obteniéndose los datos a partir de los sensores internos que reflejan el desplazamiento real del robot; odómetros de cada rueda. A partir de los datos obtenidos en las pruebas experimentales realizadas con el Khepera, análogos a los resultados comentados previamente en el simulador, el mejor rendimiento del sistema borroso ha quedado probado, como muestra en la tabla .

6 Conclusiones y trabajos futuros

Se ha presentado una metodología basada en los conceptos de lógica borrosa y en las arquitecturas reactivas, para integrar distintos comportamientos para la navegación de un robot móvil. Partiendo del uso de variables cuantitativas se ha pasado al uso de descripciones lingüísticas similares a las utilizadas en el lenguaje natural. De esta forma se facilita el tratamiento de un problema matemáticamente muy complejo.

Este trabajo presenta una arquitectura teórica, con n niveles y una implementación en un robot real con dos niveles. La evaluación del comportamiento global del robot muestra que el sistema con fusión difusa tiene un comportamiento superior al de aquellos que realizan una fusión tradicional.

Entre los trabajos futuros se puede incluir una primera tarea de prueba de la arquitectura con mayor número de sensores. Con una percepción mayor del entorno, el robot tendrá más información y se espera que muestre un comportamiento más inteligente. Este tipo de extensión sería inabordable en forma de un único controlador fuzzy al incrementarse exponencialmente su complejidad. Con el tipo de arquitectura propuesta sería posible obtener un comportamiento complejo sin un gran incremento de la complejidad del sistema.

El siguiente tipo de prueba consistir'a en aumentar el n'umero de comportamientos. En ese sentido se pueden aadir comportamientos como bordear un obst'aculo o crear mapas. Este segundo grupo de pruebas demostrar'an la habilidad del sistema para abordar tareas m'as complejas.

Por 'ultimo, indicar que el inter'es b'asico de investigaci'on del Grupo de Agentes Inteligentes de la Universidad Carlos III es el desarrollo de modelos cognitivos que permitan la cooperaci'on de agentes, tanto software (asistentes personales, sistemas expertos, etc.) como hardware (robots), en la resoluci'on de problemas complejos. As'i, la arquitectura de comportamientos fuzzy presentada es el nivel reactivo de un modelo multinivel para la cooperaci'on de robots en tareas simples (traslado de objetos, limpieza de suelos, etc), en que se est'a trabajando dentro del grupo.

References

- [1] Badreddin, E., *Fuzzy Relations for Behavior-Fusion of Mobile Robots*. Proceedings of IEEE Int. Conference on Robotics and Automation, 3278-3283, 1994.
- [2] Belmonte-Serrano, M., Sierra, C., Lopez de M'antaras, R., *RENOIR: An Expert System Using Fuzzy Logic for Rheumatology Diagnosis*. International Journal of Intelligent Systems, Vol. 9, 985-1000, 1994.
- [3] Braitenberg, V., *Vehicles. Experiments in Synthetic Psychology*. The MIT Press, Cambridge, Massachusetts, 1984.
- [4] Brooks, R.A., *A Roboust Layered Control System for a Mobile Robot*. IEEE Journal of Robotics and Automation RA-2, 14-23, 1986.
- [5] Brooks, R.A., *Intelligence without representation*. Artificial Intelligence, 47, 139-159, 1991.
- [6] Goodridge, S. G., Luo, R.C. *Fuzzy Behavior Fusion for Reactive Control of an Autonomous Mobile Robot: MARGE* Proceedings of IEEE Int. Conference on Robotics and Automation, 1622-1627, 1994.
- [7] Ishikawa, S. *A method of autonomous mobile robot navigation by using fuzzy control*. Advanced Robotics, Vol. 9, No. 1, pp. 29-52, 1995.
- [8] Mondada, F., Franzi, P.I., *Mobile Robot miniaturisation: A tool for investigation in control algorithms*. Proceedings of the third International Symposium on Experimental Robotics. Kyoto, Japan, 1993.
- [9] Nilsson, N.J. (ed.), *Shakey the robot*. SRI A.I. Center Technical Note 323, April, 1984.

- [10] Pin, F.G., Watanabe, Y, *Driving a Car using Reflexive Fuzzy Behaviors*. Proceedings of the Second International Conference on Fuzzy Systems, pag. 1425-1430, San Francisco, 1993.
- [11] Reignier, P., *Fuzzy logic techniques for mobile robot obstacle avoidance*. Robotics and Autonomous Systems, 12, pag. 143-153, 1994.
- [12] Zadeh, L.A., *Outline of a New Approach to the Analysis of Complex Systems and Decision Processes*. Trans. on SMC, vol 3, 1, January 1973.
- [13] Zimmerman, H.J., *Fuzzy Sets Theory and Applications*. Kluwer, Boston, 1992.