

Formalización de maniobras en robots con múltiples grados de libertad como sistemas multiagente.

José Antonio Martín H., Javier Alonso Ruiz

Instituto de Automática Industrial, C.S.I.C.
Carretera de Campo Real km 0.200 La Poveda Arganda del Rey 28500 Madrid, España
{jmartin, jalonso}@iai.csic.es

Abstract. Presentamos la formalización de maniobras complejas para robots como un sistema multiagente con objetivos globales y locales para cada agente. En este contexto definimos agente como un componente que puede percibir algo de su entorno y reaccionar intentando satisfacer un objetivo. Modelamos un robot humanoide como un sistema multiagente donde cada agente es responsable de optimizar una parte del cuerpo del robot, el agente posee información tanto del estado interno (estado de las articulaciones), como del entorno y actúa fijando unas consignas en los motores propios de su parte del cuerpo. Una maniobra compleja será aquella en la que existan objetivos opuestos en el mismo instante de tiempo y sea necesario coordinar diferentes agentes para alcanzar el objetivo global.

1 Introducción

El problema de coordinar los movimientos de un robot con múltiples grados de libertad se complica a medida que el número de articulaciones y la complejidad de sus maniobras aumentan [7]. Es posible programar explícitamente los movimientos necesarios para que el robot realice una determinada tarea, pero si las tareas se desarrollan en entornos no controlados y/o son de moderada complejidad esta opción se vuelve inviable.

El problema de realizar una maniobra en robots complejos puede también enfocarse como un problema de coordinación multi-objetivo. Con este enfoque se podría utilizar la teoría de juegos e intentar modelar una maniobra compleja como un tipo especial de juego de cooperación, pero aún así se presentan problemas muy complejos por resolver; es por esto que modelamos el problema como un sistema multi-agente, y nos centraremos en el uso de técnicas de aprendizaje por refuerzo para conseguir una buena solución.

Nuestra intención enfocando la tarea de realizar maniobras complejas en robots con múltiples grados de libertad como un problema de coordinación de sistemas multiagente es sugerir y explorar una línea de investigación que permita el desarrollo de nuevas aproximaciones al problema del aprendizaje automático de conductas simples y complejas en robótica.

En este artículo se estudian los diferentes puntos de vista desde los que podemos enfocar la realización de una maniobra compleja en un sistema robótico. Intentamos formalizar el concepto de maniobra como un conjunto de movimientos coordinados orientados a la consecución de un objetivo global por medio de objetivos locales. Puede verse de muchas maneras diferentes; como un problema de aprendizaje automático; como coordinación de un sistema multiagente; o como una clase especial de juego de cooperación donde el óptimo global asegura el óptimo local.

2 Robots humanoides

Para experimentar contamos con un software simulador de robots basado en el modelo *Qrio* de Sony™ [8].

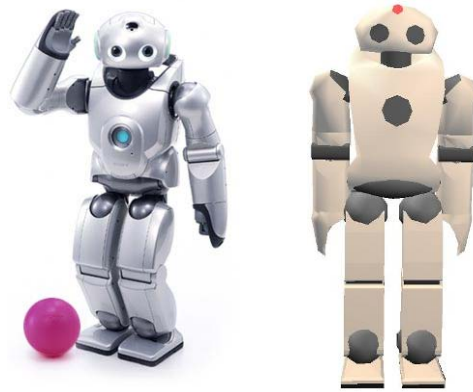


Fig. 1. Robot *Qrio* de Sony™ y robot simulado.

Aunque una simulación no puede sustituir la experimentación con robots reales el software de cyberbotics [5] es excepcionalmente preciso y nos permite utilizar técnicas de aprendizaje que serían inviables con robots reales. Los robots simulados cuentan con:

- 22 servo motores (articulaciones).
- 2 sensores digitales de contacto en cada pie.
- Sensor radial de distancia ubicado en la cabeza.
- Cámara de visión en la parte superior de la cabeza.
- GPS en la espalda con el cual puede medirse la altura en la cual se encuentra el robot, la posición y ya que la matriz que resulta de la lectura del GSP es una matriz rotacional, puede utilizarse también como un inclinómetro extrayendo los ángulos de euler.

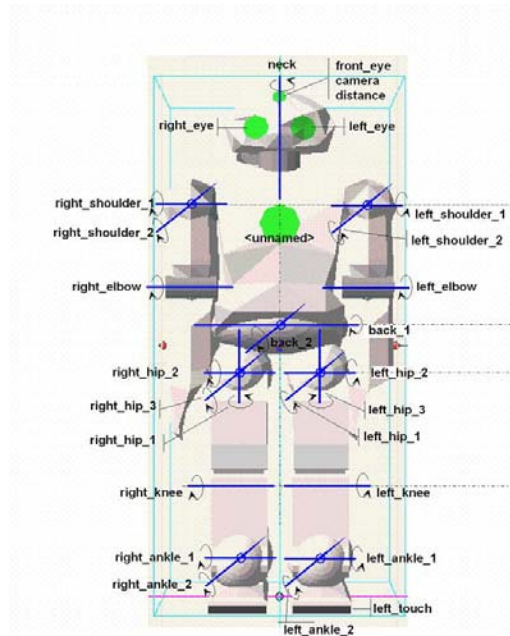


Fig. 2. Esquema con las articulaciones del robot simulado y sus grados de libertad.

Durante el proceso de aprendizaje de la maniobra “ponerse en pie” surgió la idea del modelo multiagente. Al no poder obtener información sobre la presión ejercida sobre las plantas de los pies las maniobras de estabilización resultaban muy difíciles de aprender. Así, a partir del estado instantáneo de cada motor (su ángulo en radianes) creamos sensores virtuales simulando un inclinómetro en cada parte del cuerpo. Cada parte puede ser entonces descrita como un vector, y cada vector puede estar supervisado y optimizado por un agente.

3 Maniobras

Generalmente utilizamos la palabra maniobra para referirnos a un movimiento determinado del cuerpo que se da en un intervalo de tiempo con el fin de satisfacer algún objetivo. En nuestra aproximación el movimiento de un solo motor de forma individual en un instante de tiempo no lo consideramos una maniobra. Definimos maniobra como el movimiento conjunto de todos los motores o actuadores del cuerpo del robot durante un tiempo “suficiente” para lograr un objetivo.

Una maniobra global puede estar compuesta de submaniobras siempre y cuando haya un objetivo para cada subconjunto de movimientos. El problema pasa a ser obtener la optimalidad de la maniobra global compuesta de submaniobras donde es necesario optimizar las submaniobras. Estas submaniobras pueden darse en paralelo,

semiparalelo ó en serie, es decir, que durante intervalos de tiempo hay más de una submaniobra en marcha.

Pasa a ser un problema clásico de optimización multiobjetivo con objetivos contrapuestos, en nuestro caso particular, la optimalidad de algunas inclinaciones de partes del cuerpo se contraponen a la optimalidad de otras durante algunos intervalos de tiempo, lo cual las hace competir y bloquearse entre ellas, este fenómeno produce lo que se llama una descoordinación motora que puede traducirse en bloqueo u oscilaciones.

Por ejemplo el caso de ponerse en pie partiendo de estar tumbado de espaldas.

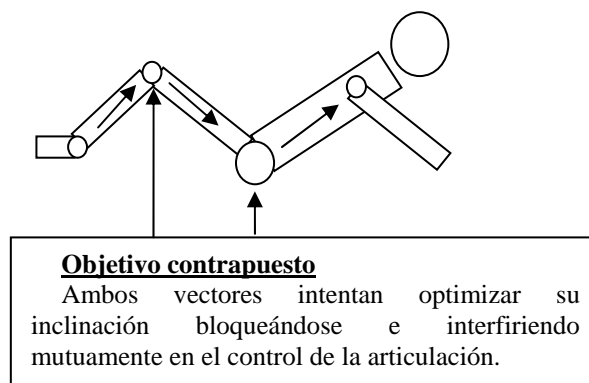


Fig. 3. Esquema de objetivos contrapuestos.

Podemos plantearlo como un problema de coordinación multiagente donde cada parte del cuerpo es un agente que intenta maximizar alguna función local para optimizar una función global y compite por uno o más recursos que son las articulaciones que están directamente relacionadas con él. Entonces el problema es la optimización de una función global mediante la coordinación de un sistema multiagente.

4 Aplicación de Sistemas Multi-Agente (MAS) al problema de la coordinación de maniobras

Los agentes son programas autónomos inteligentes capaces de comunicarse los unos con los otros. Si colaboran para realizar una tarea compleja hablamos de sistemas multi agente cooperativos (MACS). Se define un objetivo global a cumplir y cada agente tiene que colaborar para lograr dicho objetivo global.

En nuestro caso el robot debe realizar una maniobra en un cierto tiempo. Cada parte del cuerpo del robot tiene un vector de orientación asociado. Y el objetivo global se define, entre otras cosas, por una orientación dada en los vectores del robot. Así

tendremos un agente por cada uno de los vectores del robot. Que tendrá que calcular su propia orientación a partir del inclinómetro localizado en la espalda del robot.

La información que reciben todos los agentes del sistema es la misma, y es el estado de todas las articulaciones del sistema y la orientación del inclinómetro.

La manera en la que cada agente actúa sobre las articulaciones del robot determina la dinámica de acción del sistema multiagente. Podemos plantear varias formas de actuar:

1. Cada agente actúa sobre todas las articulaciones intentando maximizar el objetivo global.
2. Cada agente actúa solo sobre sus articulaciones adyacentes intentando maximizar el objetivo global.
3. Cada agente actúa sobre todas las articulaciones intentando maximizar su objetivo local.
4. Cada agente actúa solo sobre sus articulaciones adyacentes intentando maximizar su objetivo local.

En principio, la segunda y la cuarta opciones nos parecen los enfoques más razonables. De esta manera, sobre cada articulación competirían uno o dos agentes, haciendo más sencilla la política de asignación de recursos.

Así tendremos que nuestro modelo de agente puede representarse como sigue:

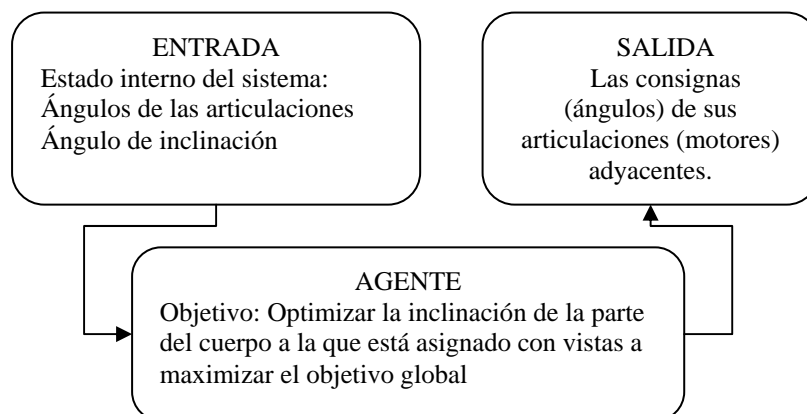


Fig. 4. Modelo de agente: Entrada, Salida y Objetivo

De esta forma el problema es planteado como un sistema multiagente cooperativo donde es necesario aprender la función de coordinación, que es precisamente la que se utilizara para realizar una maniobra en particular.

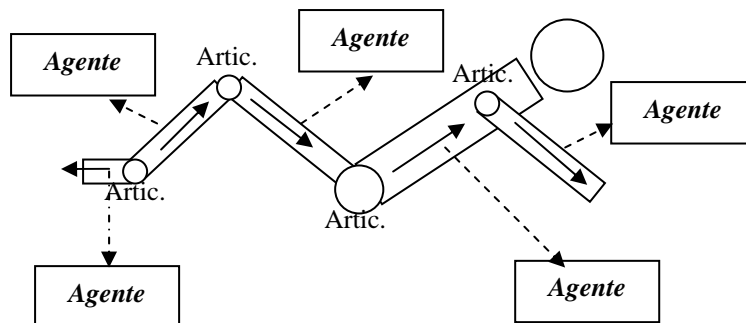


Fig. 5. Cooperación de agentes en la maniobra ponerse en pie.

5 Métodos de aprendizaje

Existen diversas aproximaciones para solucionar este tipo de problemas de aprendizaje, por ejemplo, dentro del aprendizaje por refuerzo clásico, el método Qlearning. Pero este método presenta varios problemas de implantación. Qlearning opera con representaciones discretas del entorno (estados discretos) y su salida es a la vez discreta. Además, la complejidad del algoritmo incrementa exponencialmente con el número de estados en los que el robot puede encontrarse y el número de acciones que puede llevar a cabo [4].

Dado que la entrada al sistema no proviene de una fuente discreta y discretizar la entrada podría introducir sesgos al método, el uso del método Qlearning no es el ideal. Además la salida en robótica suele ser analógica, es decir, se espera una salida continua para ser transmitida como acción hacia un motor.

Por estas razones, hemos elegido métodos de aprendizaje neuronal con entradas y salidas continuas. Entre los métodos neuronales de aplicación directa al presente problema, podemos citar, las redes CMAC (Cerebellar Model Articulation Controller) [1] y el muy conocido Perceptrón Multicapa [6].

Una de las características que quizá definen mejor la capacidad de aprendizaje biológica de orden superior es el aprendizaje por imitación, y no el aprendizaje por ensayo y error, es sabido también que hay evidencias fisiológicas que indican que existen determinadas neuronas cuyo campo receptivo está determinado por las acciones y movimientos corporales de los congéneres, a esta familia de neuronas se la ha llamado neuronas espejo y forman parte fundamental en el aprendizaje por imitación ya que traducen los movimientos corporales observados, en movimientos propios[9] y [2]. Lamentablemente en robótica, se está lejos de poder contar con esta clase de sensor, a pesar de la intensa actividad que se desarrolla en el campo de la percepción artificial. Aunque existe un proyecto en el MIT para intentar de alguna manera producir un tipo de neuronas análogas a las neuronas espejo [3].

Desprendiéndonos, por ahora, de la capacidad natural que poseen los animales superiores para imitar, utilizaremos otro método, no sin intentar buscar alguna forma analógica de imitación. Para lo cual se ha diseñado un estudio de aprendizaje por refuerzo basado en la imitación de movimientos generados por un algoritmo evolutivo donde cada agente tendrá su propia red neuronal independiente del resto de los agentes. Así, podemos perfeccionar maniobras diversas mediante algoritmos evolutivos y aprender la función de coordinación mediante los neuro-agentes.

Esta forma particular de experimentación, resuelve varios problemas a la vez:

1. Tenemos un modelo neuronal continuo del espacio perceptivo y del espacio de acción de los motores que nos permite tener el grado de precisión deseado.
2. Tenemos un sistema de aprendizaje por refuerzo bien conocido y que se sabe que es efectivo (red back-propagation).
3. Tenemos la forma de realizar aprendizaje por imitación.

Hemos obtenido unos buenos resultados iniciales. El robot aprende formas naturales de movimiento, por ejemplo, cuando esta boca abajo en el suelo, aprende a utilizar sus manos para intentar ponerse en pie como si fuera un cuadrúpedo. También aprende, en general, a coordinar sus articulaciones (neuro-agentes) para conseguir maximizar la función de recompensa.

El diseño de esta función de recompensa es la tarea más crítica de todo el proceso, pues no solo determina el punto hasta el que el sistema evolucionará, sino que también fija el modo en el que abordará la tarea. Es decir, al primar la altura o posición de ciertas partes del cuerpo se determina como evolucionará el aprendizaje.

6 Resultados experimentales.

En el experimento que aquí presentamos hemos escogido una serie de técnicas para cada fase del aprendizaje. Se han realizado otros experimentos con diferentes técnicas, pero este presenta buenos resultados y resulta bastante claro.

Se intenta que el robot se levante del suelo. Primero, mediante algoritmos genéticos, se obtiene un conjunto de maniobras que consigan levantar al robot. Estas maniobras son las que los neuro-agentes tratarán de imitar al aprender. Una vez alcanzado un grupo de maniobras para levantarse suficientemente bueno se entrenan las redes de neuronas de los agentes hasta que aprenden las maniobras.

Para reducir el número de agentes solo trabajamos con el lado izquierdo del robot, aplicando un principio de simetría. Así nos quedan 11 agentes (10 motores del lado izquierdo del robot más el servo que da la inclinación al torso) y 11 redes de neuronas.

Se han usado algoritmos genéticos como algoritmo evolutivo. Partimos de una población de 30 individuos (maniobras), donde cada maniobra es un conjunto de 11 genes. Cada gen es una función de onda sinusoidal de la forma:

$$\phi(t) = \text{amplitud} \times \sin(\Omega \times t + \varphi) + \text{const} \quad (1)$$

Donde la amplitud varia entre [-1,1], omega entre [-2π,2π], phi entre [-2π,2π], y const entre [-1,1].

Y la función de Fitness es la siguiente:

$$J = \sqrt{\text{pie_der}^2 + \text{pie_izq}^2 + \text{altura}^2 + \left(\frac{1}{\text{inclinacion} + 0.0001} \right)^2} \quad (2)$$

Una vez evaluados los 30 individuos o maniobras se procede a ordenarlos de mejor a peor, se seleccionan los 10 mejores para la próxima época, se cruzan los 10 mejores para obtener 10 individuos para la próxima época, se mutan los 10 mejores para obtener 10 individuos para la próxima época y se involuciona suavemente toda la población haciendo tender su función a $e^{-\text{sen}(t)}$ donde e tiende a 0. Generándose nuevamente 30 individuos para volver a evaluarlos.

El modelo de las redes de neuronas utilizadas es un perceptrón multicapa con método de entrenamiento back-propagation. La función de activación de las neuronas es la función tangente hiperbólica que va entre [-1,1] de forma que puedan ser más fácilmente modelables con funciones trigonométricas los ángulos positivos y negativos. Cada neuro-agente posee 16 entradas, 22 neuronas en la capa oculta y 1 neurona de acción ó salida.

6.1 Resultados del experimento

En este experimento se ha aprendido la maniobra de ponerse en pie.



Fig. 6. El robot ha aprendido a situar bien los pies como punto de apoyo y ha aprendido a utilizar sus brazos para impulsar el torso y mejorar su inclinación.

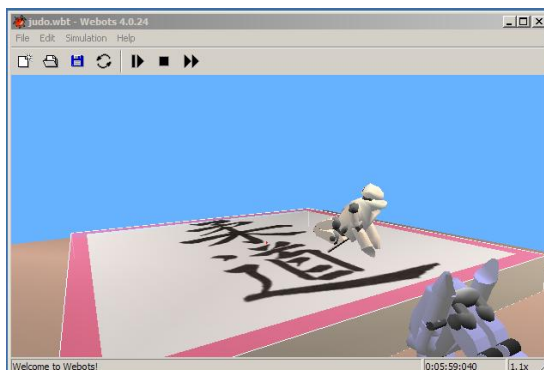


Fig. 7. Intenta enderezar el torso y ganar altura.



Fig. 8. Se lanza impulsado por los brazos y los “músculos abdominales”.



Fig. 9. El Robot se pone en pie.

El Robot finalmente se pone en pie, no obstante, debido a que tiene un momento en dirección al frente bastante alto, caerá de frente al suelo. Esto sucede porque no se ha entrenado para mantener el equilibrio y percibir su momento.

Agradecimientos:

Los autores agradecen su ayuda a los proyectos SIAMI (Carbonar S.A.), financiado por el Centro para el Desarrollo Tecnológico Industrial (CDTI 20040116), Ministerio de Industria, y COPOS, Ministerio de fomento BOE 280, Noviembre 22, Res. 22778. donde se utilizan técnicas semejantes a las aquí descritas.

Bibliografía

1. Albus, J.S. "A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller", in Journal of Dynamic Systems, Measurement, and Control, Volume 97, pp. 220 - 227, 1975.
2. Giacomo Rizzolatti, Laila Craighero, "The Mirror-Neuron System", Annual Review of Neuroscience, July 2004, Vol. 27: 169-192 (doi:10.1146/annurev.neuro.27.070203.144230)
3. Giorgio Metta, Paul Fitzpatrick "Modelling the development of mirror neurons". MIT AI Lab, Humanoid Robotics Group.
4. Martinson, E., Stoychev, A. and Arkin, R. (2001). "Robot Behavioral Selection Using Q-Learning", Technical Report GIT-CC-01-19, College of Computing, Georgia Institute of Technology, Atlanta, GA.
5. Michel, O. / Cyberbotics Ltd - WebotsTM: Professional Mobile Robot Simulation, pp. 39-42, International Journal of Advanced Robotic Systems, Volume 1 Number 1 (2004), ISSN 1729-8806, <http://www.cyberbotics.com>
6. Mikko, L., Jukka, S., Pentti H. and Kimmo K. "Initializing Weights of a Multilayer Perceptron Network by Using the Orthogonal Least Squares Algorithm", Neural Computation, Vol. 7, Issue 5 - September 1995
7. Robin R. Murphy: *Introduction to AI Robotics*. A Bradford Book, MIT Press Cambridge Massachusetts. London, England
8. Sony Corporation, New York City, U.S. <http://www.sony.net/SonyInfo/ORIO/>
9. Yiannis Demiris, "Mirror Neurons, Imitation And The Learning Of Movement Sequences", Proceedings of the 9th International Conference on Neural Information Processing (ICONIP02), Vol1, 2002