Investigación en Sistemas Multi-agente y Robótica Cognitiva aplicada a la Robótica Móvil en el dominio de Fútbol de Robots

Alejandro J. García Telma Delladio Mariano Tucat
Nicolás D. Rotstein Diego R. García

Grupo de Robótica Cognitiva
Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA)
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur. Av. Alem 1253, (8000) Bahía Blanca, Argentina
Tel: ++54 291 459 5135 - Fax: ++54 291 459 5136
{ajg,td,mt,ndr,drg}@cs.uns.edu.ar

Resumen

El objetivo principal de este trabajo es presentar las diferentes líneas de investigación que nuestro grupo está desarrollando en inteligencia artificial aplicada a la robótica móvil. En particular, en este trabajo se describen las líneas de investigación que están asociadas al desarrollo de un equipo de fútbol de robots.

Con el desarrollo de estas líneas de investigación el objetivo es doble. En primer lugar se desea aportar nuevas soluciones para los problemas que se presentan en robótica móvil. En segundo lugar se desea utilizar un dominio de aplicación concreto de robots móviles para obtener retro-alimentación y desarrollar mejores formalismos en el área de agentes inteligentes, sistemas multi-agente y robótica cognitiva.

1. Robótica Móvil, Robótica Cognitiva y Sistemas Multi-agente

Disponer de robots móviles con algún grado de inteligencia requiere integrar facilidades de bajo nivel (manejo de sensores, movimientos básicos, evasión de obstáculos, navegación, etc) con habilidades de alto nivel como razonamiento, representación de creencias, planificación, comunicación con otros agentes, interacción, coordinación, etc. El desarrollo de dichas facilidades y habilidades debe tener en cuenta que si el robot móvil opera en un ambiente dinámico, donde además pueden existir otros agentes que interactúan con él, entonces el robot debería ser capaz de razonar con cierto grado de incertidumbre y poder planear sus acciones con la suficiente flexibilidad como para modificar sus planes en ejecución.

Estas habilidades de alto nivel se encuentran actualmente en estudio en dos áreas de inteligencia artificial que han tenido un gran impulso en los últimos años: Robótica Cognitiva y Sistemas Multi-agente (SMA). La investigación en Robótica Cognitiva [16, 12, 8, 5, 3] involucra problemas relacionados con funciones cognitivas de alto nivel, las cuales permiten que robots físicos sean capaces de razonar y desenvolverse en el entorno que los alberga. Algunas de las problemáticas abordadas por el área de Robótica Cognitiva incluyen:

- la necesidad de desenvolverse en dominios dinámicos con conocimiento incompleto acerca de los objetos en el mundo, las acciones disponibles, sus precondiciones, y sus efectos;
- la necesidad de representar grandes espacios de estados y razonar sobre ellos;
- la actualización y revisión del conocimiento disponible en forma eficiente y semánticamente correcta;
- el diseño de arquitecturas de agentes adecuadas que soporten las soluciones a todos esos problemas.

Por esta razón, los trabajos en robótica cognitiva son muy variados y abordan temáticas complejas como:

- representación de conocimiento y razonamiento;
- definición de lenguajes de control de algo nivel para el desarrollo de razonadores;
- la definición y desarrollo de planificadores;
- la definición de formalismos de razonamiento acerca de acciones y cambio;
- revisión de conocimiento y aprendizaje;
- arquitecturas de agentes inteligentes.

Además, si los robots móviles deben desenvolverse en un entorno donde pueden existir otros robots con los cuales tienen que interactuar, entonces surgen otro tipo de problemas. Las soluciones a estos problemas, requieren que estos robots cuenten, no sólo con habilidades cognitivas, sino también con habilidades sociales tales como: cooperación, coordinación, y comunicación entre agentes. Estas habilidades sociales se encuentran actualmente en estudio dentro del área de Sistemas Multi-agente.

Como fue dicho anteriormente, nuestro interés en robótica móvil es doble. Por un lado, uno de lo objetivos centrales es la aplicación de formalismos que han sido desarrollados por nuestro grupo en diferentes áreas de inteligencia artificial, los cuales permitirán obtener las habilidades de alto nivel (cognitivas y sociales) antes mencionadas. Por otro lado, se está utilizando un grupo de robots en un dominio de aplicación específico como lo es un partido de fútbol de robots, con el objetivo de obtener retro-alimentación en temas fundamentales de SMA como coordinación,

interacción, colaboración y comunicación; así como también en temas centrales de Robótica Cognitiva como representación de conocimiento, razonamiento, planificación y aprendizaje.

En la sección 3 se describirán en detalle las líneas de investigación que se están llevando a cabo en nuestro grupo. Estas líneas involucran desarrollos teóricos y prácticos en robótica cognitiva y SMA, y están siendo aplicadas en el desarrollo de un equipo de fútbol de robots el cual se describe en la sección siguiente.

2. El dominio de aplicación

En esta sección se describe brevemente cual es el dominio de aplicación sobre el cual se están aplicando las líneas de investigación que describiremos en la sección siguiente. Se trata de un equipo de fútbol de robots de la modalidad E-league [1], formado por cuatro robots Lego Mindstorm [2]. Para su funcionamiento, se han diseñado e implementado un sistema multi-agente donde cada agente lógico controla un jugador del equipo de robots. A continuación incluimos un resumen del diseño de este sistema, los detalles pueden verse en [10] y [9].

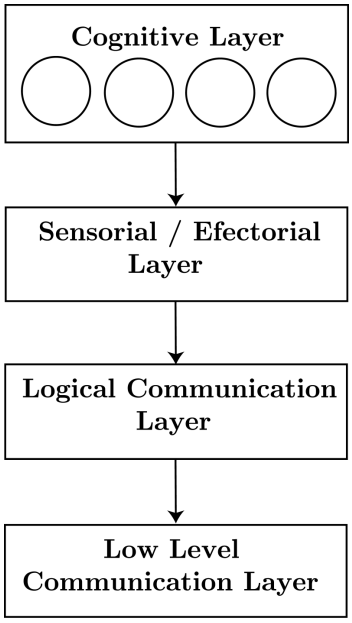

Figura 1: Diseño del Sistema Multi-agente que controla a los robots [9]

El sistema fue desarrollado siguiendo un diseño por capas (Figura 1).

El SMA fue diseñado como muestra la Figura 1 donde cada capa de servicio está asociada a un diferente nivel de abstracción. Cada capa resuelve un conjunto diferente de problemas por medio de los servicios que ofrece. Dichos servicios quedan disponibles a las capas superiores. Por ejemplo, la capa mas baja (low level communication layer) provee los servicios básicos provistos por el hardware de visión y comunicación con los robots. La capa siguiente (logical communication layer) está implementada en C y utiliza los servicios de la anterior y provee

dos primitivas básicas: la función de percepción y la primitiva de comunicación con los robots. La siguiente (sensorial/effectorial) está implementada en Prolog y provee las primitivas básicas para implementar el comportamiento de los robots. La más alta es la que implementa las habilidades cognitivas y sociales.

Como cada capa es modular, modificando las capas inferiores permite adaptar los desarrollo de alto nivel a otros dominios u otros robots. Además, nuevos desarrollos a nivel cognitivo pueden ponerse a prueba simplemente a través de la reformulación de los servicios de la capa cognitiva.

3. Líneas de investigación en curso

En esta sección se describen las siguientes líneas de investigación aplicadas a problemáticas relacionadas con robótica móvil:

- Representación de conocimiento y razonamiento con argumentación rebatible.
- Arquitecturas de Agentes.
- Aprendizaje.
- Interacción y comunicación en Sistemas Multi-agente.
- Planificación.

Estas líneas de investigación se encuentran actualmente en desarrollo en nuestro laboratorio.

3.1. Representación de conocimiento y razonamiento con argumentación rebatible

Como ya se dijo, un robot móvil debería poder desenvolverse en dominios dinámicos con conocimiento incompleto acerca de su entorno. Esto incluye información incompleta acerca de los objetos en el mundo y cierto grado de incertidumbre en los efectos de sus acciones. Por lo tanto, debe proveerse a los robots de un formalismo que les permita razonar sobre información incompleta y cambiante.

En nuestro laboratorio se ha desarrollado un formalismo de razonamiento con argumentación rebatible que permite realizar inferencias sobre conocimiento incompleto y potencialmente contradictorio. Este formalismo, denominado Defeasible Logic Programming (DeLP) o Programación en Lógica Rebatible [11], permite representar conocimiento con reglas rebatibles y estrictas. En DeLP las conclusiones de un agente son sustentadas con *argumentos*. Una conclusión se dice *garantizada* cuando el argumento que la sustenta no posee *derrotadores* (contraargumentos que lo derrotan) o todos sus derrotadores son a su vez derrotados. Dado un argumento que sustenta una conclusión este podría ser atacado por otros argumentos *derrotadores* que lo contradigan. Dichos derrotadores podrán a su vez ser atacados, y así sucesivamente,

generando una secuencia de argumentos llamada *línea de argumentación*. En DeLP el proceso completo considera para cada argumento todos sus posibles derrotadores, lo cual, en lugar de una única línea de argumentación, genera un conjunto de líneas representadas a través de un *árbol de dialéctica*. Un proceso de análisis dialéctico determina, a partir del conocimiento del agente, si la conclusión en cuestión está *garantizada* o no.

La utilización de DeLP para implementar el proceso de razonamiento de un robot nos permite poseer un sistema de razonamiento con las características deseadas y además analizar cómo se comporta DeLP en un dominio de aplicación de robots móviles. En nuestro caso, cada robot representa, como parte de su conocimiento, las acciones de alto nivel que pueden ser ejecutadas, y las razones para creer en el éxito o fracaso de las mismas. Estas razones se apoyan en la información que posee cada agente acerca de las posiciones del resto de los jugadores y la pelota, dentro del campo de juego. Un análisis argumentativo sobre la garantía de la ejecución de las acciones disponibles conforma la base para la etapa final del proceso de razonamiento. Este proceso, enmarcado en una arquitectura BDI (ver Sección 3.2), finaliza con la determinación de una intención que se materializará en una acción física por parte del robot.

3.2. Arquitecturas de Agentes

El objetivo principal de esta línea de investigación es analizar cómo se adecúa al dominio de fútbol de robots una combinación entre una arquitectura de agentes BDI (Belief-Desire-Intention) y un sistema de razonamiento con argumentación rebatible (DeLP). El dominio de aplicación ofrece el desafío de ser altamente dinámico y determina una cierta variedad de acciones, las cuales, de alguna manera, compiten entre sí para determinar cuál será la próxima a ser ejecutada.

La arquitectura BDI que describe la implementación de los agentes de software que controlan a los robots físicos (Figura 2) está conformada por los conjuntos de creencias (B), deseos (D) e intenciones (I) que se detallan a continuación:

- Creencias (*Beliefs*): El conjunto de creencias reúne toda la información que obtiene el agente únicamente a través de su percepción del mundo. A partir de este conjunto el agente puede actuar en consecuencia, ya que tiene su propia representación del entorno que lo rodea. La calidad y el detalle de la información obtenida depende de sus sensores y de la forma en que internaliza el estado del mundo.

- Deseos (*Desires*): Los deseos, como conjunto, representan las actitudes que determinarán el comportamiento del agente. Demuestran la tendencia del agente hacia una forma de actuar determinada y, de alguna manera, las metas que tiene fijadas y que intenta cumplir. Dependiendo de la situación actual, habrá deseos que no podrán cumplirse y deberán ser reprimidos. Esto es imprescindible para que el agente pueda cumplir con la misión que le fue encomendada.

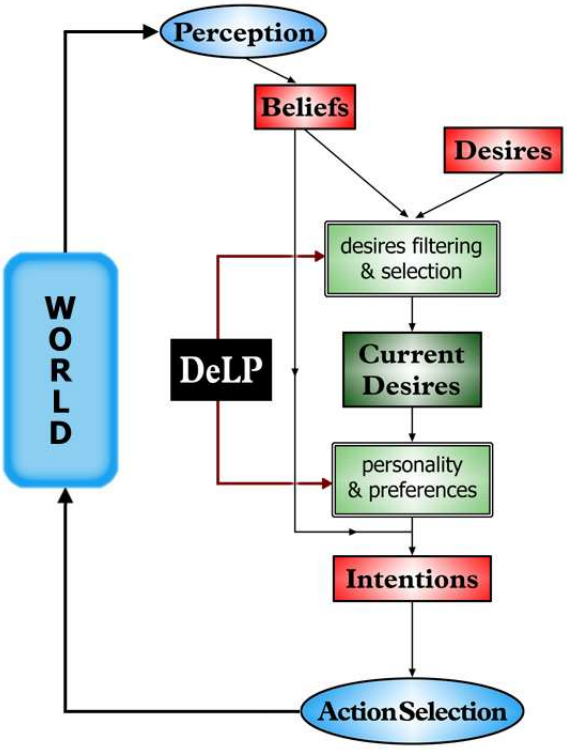

Figura 2: Ciclo BDI utilizando DeLP como módulo de razonamiento

- Intenciones (*Intentions*): Se generan a partir de los deseos y expresan lo que el agente es capaz y quiere hacer en el instante actual. Representan la materialización de los deseos, habiéndolos contrastado previamente con el estado del mundo y del agente en sí. En esta implementación, el conjunto de intenciones será un único elemento del conjunto de deseos y representará la acción de alto nivel que prevalece en ese instante.

Por ejemplo, los conjuntos de Creencias, Deseos e Intenciones de un agente (jugador) podrían ser los siguientes:

```
B =     { arqueroFuera(E), posesión(E) : E ∈ {our,their} } ∪
        { alguienEntre(X,Y) : X, Y ∈ {me,t1,t2,t3,o1,o2,o3,o4} } ∪
        { ganando, empatando, perdiendo } ∪
        { ladoContrario(J), nadieAdelante(J), marcado(J), tienePelota(J) :
            J ∈ {me,t1,t2,t3,o1,o2,o3,o4} }

D =     { patearAlArco(me), puedeLlevarla(me), alinearse } ∪
        { hacerPase(me,T) : T ∈ {t1,t2,t3} } ∪
        { marcarA(O) : O ∈ {o1,o2,o3,o4} }


I ⊆ D, tal que |I| = 1
```

3.3. Aprendizaje

El aprendizaje es una habilidad que está relacionada con la autonomía. Por esa razón, la importancia del estudio de mecanismos de aprendizaje en el desarrollo de agentes autónomos. Dentro del área de Machine Learning, la Programación en Lógica Inductiva se presenta como una propuesta interesante al momento de resolver el problema de aprendizaje de conceptos a partir de ejemplos. Las propuestas de Programación en Lógica Inductiva utilizan formalismos de Programación en Lógica como herramienta de representación de conocimiento y razonamiento, junto con mecanismos de inducción para sintetizar, a partir de ejemplos, programas lógicos que sirven como definición de los conceptos que se desean aprender. La obtención de estas definiciones es una tarea muy costosa y es necesario estudiar mecanismos que permitan resolver esta tarea en forma eficiente. Muchas veces, afrontar una tarea compleja en forma cooperativa puede dar buenos resultados.

Se han realizado estudios considerando la definición de un esquema de aprendizaje grupal [6] como un primer paso para el planteo de soluciones cooperativas al problema del aprendizaje de conceptos a partir de ejemplos dentro del marco de la Programación en Lógica Inductiva. Además, considerando la flexibilidad necesaria para llevar adelante una tarea cooperativa tan compleja, se ha estudiado la utilización de otros paradigmas de programación en lógica como la Programación en Lógica Rebatible [7] como herramienta de representación de conocimiento y razonamiento no monótono a partir de la cual desarrollar aprendizaje inductivo de conceptos. La Programación en Lógica Rebatible como herramienta de razonamiento no monótono, ofrece una mayor flexibilidad al momento de sintetizar definiciones que modelen los conceptos que son objeto de aprendizaje.

En el domino considerado, la habilidad de aprendizaje puede ayudar al mejoramiento del comportamiento individual y grupal de los integrantes del equipo. Cada jugador puede aprender, por ejemplo, nociones de posición y acción las cuales pueden ser modeladas a través de conceptos representados por términos lógicos. La valoración de la correctitud de los conceptos aprendidos puede ser determinada en la dinámica del juego considerando los resultados obtenidos en el transcurso del mismo. A más alto nivel, se pueden plantear tareas de aprendizaje grupal, en donde el conjunto de agentes pueda aprender (aproximar) la estrategia de comportamiento del equipo contrario.

3.4. Interacción y comunicación en Sistemas Multi-agente

La interacción es una característica esencial de un Sistema Multi-Agente. Dicha interacción puede ser llevada a cabo a través del intercambio de mensajes de acuerdo a una política de conversación determinada, o a la ejecución de servicios de acuerdo a los requerimientos de otros agentes. Es por esto que la habilidad de comunicarse y responder a estos requerimientos es una característica que deberían poseer los agentes que forman parte de un SMA.

Un partido de fútbol de robots puede considerarse un SMA donde existe cooperación entre los agentes del mismo equipo y competencia entre agentes de equipos contrarios. En este dominio

de aplicación los agentes (jugadores) necesitan interactuar y comunicarse. Por ejemplo, en casos donde la visión de los robots es parcial, éstos podrían comunicarse para intercambiar la información del entorno que cada uno posee, como ser la ubicación de los demás jugadores. También podrían interactuar para planificar y coordinar jugadas, o simplemente para informar sobre alguna decisión tomada que pudiera afectar el comportamiento del resto del equipo.

Nuestro objetivo es proveer una extensión de la Programación en Lógica que permita implementar comunicación entre agentes sin necesidad de programar elementos de conexión de bajo nivel. En esta línea de trabajo se ha desarrollado una extensión de Prolog [18] que provee un conjunto de primitivas de interacción con las siguientes características:

1. Permite la creación de distintos SMA independientes.
2. Un agente que se une a un SMA, puede comunicarse con los demás integrantes del sistema simplemente conociendo sus nombres. No es necesario especificar información de conexión de bajo nivel como, por ejemplo, el identificador de la máquina y el puerto en el que se encuentra cada agente.
3. Los agentes pueden obtener la lista de los agentes presentes en el SMA.
4. Cada agente puede asociar la llegada de un mensaje de otro agente, a la ejecución de un predicado determinado. Esto permite una programación orientada a eventos.
5. Además, es posible realizar una asociación diferente para cada agente. Es decir, dependiendo de quien envíe el mensaje, se ejecutará el predicado asociado previamente.

3.5. Planificación

En un ambiente dinámico y complejo como un partido de fútbol, resulta muy difícil considerar todas las posibles situaciones que un robot puede enfrentar. Una tarea mas difícil aún es programar su comportamiento para que reaccione adecuadamente a cada una de esas situaciones. Si bien es posible programar un robot de forma general, sus habilidades quedan restringidas a las situaciones previstas por su diseñador, lo cual conspira contra su autonomía.

Dotar al robot con la habilidad de construir y ejecutar planes permite que el mismo pueda enfrentar una variedad de situaciones más amplia y aumentar su autonomía. Más aún, si se desea agregar habilidades al robot, no es necesario volver a programarlo, alcanza con agregar las acciones adecuadas para que sean consideradas al momento de construir un plan.

Por otra parte, la tecnología de planificación actual resulta inadecuada para resolver los tipos de problemas que enfrenta un robot de este tipo. Esto se debe a que los métodos tradicionales de planificación asumen simplificaciones sobre los problemas planificación, que resultan adecuadas para otros dominios y disminuyen la complejidad de los problemas. Las simplificaciones más importantes son:

- El planificador tiene todo el conocimiento relevante que necesita en el momento que comienza a planificar.
- Nada importante ocurre en el mundo, excepto como resultado de sus acciones. Se ve al planificador como un módulo separado, en lugar de estar integrado a la arquitectura cognitiva del robot.

En un ambiente dinámico como un partido de fútbol estas suposiciones pueden fallar debido a que:

- El conocimiento del robot acerca del mundo, por más preciso que sea, será incompleto.
- Las creencias del robot son falibles y a medida que adquiere más información sus creencias pueden cambiar.
- A medida que el mundo cambia, las creencias del robot deben actualizarse.
- El robot interactúa con otros robots que pueden cambiar el estado del mundo.

El propósito de esta línea de investigación es la incorporación y adaptación de técnicas de planificación a la arquitectura de los robots, con el objetivo facilitar el diseño e implementación de su comportamiento y aumentar su autonomía.

3.6. Facilidades de bajo nivel

Además de investigar sobre estos aspectos cognitivos y sociales de alto nivel, en nuestro grupo se han desarrollado algunas facilidades de bajo nivel relacionados con la percepción y acción. Por ejemplo, una tarea elemental que debe realizar un robot al desenvolverse en un medio físico es la de trasladarse desde su posición actual hacia una ubicación deseada. Para lograrlo, debe tener la capacidad de evadir posibles obstáculos, tanto estáticos como dinámicos. Para ello se ha desarrollado un algoritmo de evasión de obstáculos con bajo costo computacional basado en campos de potencial [17]. Otro ejemplo es la minimización de errores que ocurren en la etapa de sensado y en la ejecución de acciones (el resultado real de su ejecución difiere del resultado esperado). Para resolver estos problemas se han propuesto diversas soluciones [13].

4. Conclusiones

En este trabajo se han descripto las diferentes líneas de investigación que se están desarrollando actualmente en nuestro laboratorio en las áreas de robótica cognitiva y SMA, con el objetivo de proveer soluciones a problemas en robótica móvil.

Referencias

[1] http://agents.cs.columbia.edu/eleague/. Official E-League webpage.

[2] http://www.legomindstorms.com. Lego Mindstorms robots and RCX controllers.

[3] Amir, E., and Doyle, P. Adventure games: A challenge for cognitive robotics. *Proceedings of the 3rd International Cognitive Robotics Workshop* (2002).

[4] Bratman, M. E., Israel, D., and Pollack, M. Plans and resource-bounded practical reasoning. In *Philosophy and AI: Essays at the Interface*, R. Cummins and J. L. Pollock, Eds. The MIT Press, Cambridge, Massachusetts, 1991, pp. 1–22.

[5] Castelpietra, C., Iocchi, L., Nardi, D., and Rosati, R. Coordination in multi-agent autonomous cognitive robotic systems. *Proceedings of the 2nd International Cognitive Robotics Workshop, Berlin, Germany* (2000).

[6] Delladio, T. Ilp en aprendizaje grupal. *IV Workshop de Investigadores en Ciencias de la Computación (WICC 2002)* (2002).

[7] Delladio, T. Aprendizaje de conocimiento rebatible. *Anales del IX Congreso Argentino de Ciencias de la Computación (CACiC 2003)* (2003), 566–576.

[8] Dylla, F., Ferrein, A., and Lakemeyer, G. Acting and deliberating using golog in robotic soccer: A hybrid approach. *Proceedings of the 3rd International Cognitive Robotics Workshop* (2002).

[9] García, A. J., Simari, G. I., and Delladio, T. Designing an agent system for controlling a robotic soccer team. In *Proceedings of X Argentine Congress of Computer Science* (2004), p. 227. http://cs.uns.edu.ar/∼ajg/papers/index.htm.

[10] García, A. J., Simari, G. I., Delladio, T., García, D. R., Tucat, M., Rotstein, N. D., Martín, F. A., and Gottifredi, S. Cognitive robotics in a soccer game domain: A proposal for the e-league competition. In *6to Workshop de Investigadores en Ciencias de la Computación (WICC), Universidad Nacional del Comahue* (2004), pp. 289–293. http://cs.uns.edu.ar/∼ajg/papers/index.htm.

[11] García, A. J., and Simari, G. R. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming 4*, 1 (2004), 95–138.

[12] Lesperance, Y., and Ng, H.-K. Integrating planning into reactive high-level robot programs. *Proceedings of the 2nd International Cognitive Robotics Workshop, Berlin, Germany* (2000).

[13] Martín, F., Tucat, M., and García, A. J. Soluciones a problemas de percepción y acción en el dominio de un equipo de fútbol de robots. *Anales del X Congreso Argentino de Ciencias de la Computación (CACiC 2004)* (2004).

[14] Rao, A. S. AgentSpeak(L): BDI agents speak out in a logical computable language. In *Seventh European Workshop on Modelling Autonomous Agents in a Multi-Agent World* (Eindhoven, The Netherlands, 1996), R. van Hoe, Ed.

[15] Rao, A. S., and Georgeff, M. P. BDI-agents: from theory to practice. In *Proceedings of the First International Conference on Multiagent Systems* (San Francisco, 1995).

[16] Reiter, R. *Knowledge in Action Logical Foundations for Specifying and Implementing Dynamical Systems.* The MIT Press, 2001.

[17] Rotstein, N., and García, A. J. Evasión de obstáculos con bajo costo computacional para un equipo de fútbol de robots. *Anales del X Congreso Argentino de Ciencias de la Computación (CACiC 2004)* (2004).

[18] Tucat, M. Primitivas de Interacción para el Desarrollo de Sistemas Multi-agente. Proyecto Final de Ingeniería en Sistemas de Computación. Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, http://cs.uns.edu.ar/∼ajg/interaction_primitives/, mar 2005.