

# Integrando capacidades de alto nivel en la toma de decisiones de agentes inteligentes

Marcelo Errecalde<sup>†</sup>, Guillermo Aguirre<sup>†</sup>, Edgardo Ferretti<sup>†</sup>, Federico Schlesinger

<sup>†</sup>Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)<sup>1</sup>  
Departamento de Informática  
Universidad Nacional de San Luis  
Ejército de los Andes 950 - Local 106  
(D5700HHW) - San Luis - Argentina  
Tel: (02652) 420823 / Fax: (02652) 430224  
e-mail: {merreca, gaguirre, ferretti}@unsl.edu.ar, fedest@gmail.com

## Resumen

**Contexto:** Este artículo describe, en forma resumida, los trabajos de investigación y desarrollo que se están llevando a cabo en la línea “Agentes y Sistemas Inteligentes” del LIDIC, en el área de agentes cognitivos. A diferencia de otras líneas de investigación del LIDIC, que se centran en problemas de optimización y el uso de enfoques de *soft computing*, nuestra línea se centra en la formalización, diseño y desarrollo de agentes computacionales con capacidades cognitivas de alto nivel, y en especial en enfoques basados en lógica, razonamiento argumentativo y teoría de decisión cualitativa. En este sentido, gran parte de los desarrollos realizados en estos temas se han realizado en forma conjunta con investigadores del LIDIA<sup>2</sup> que trabajan en temáticas afines.

Los enfoques utilizados en nuestra línea de trabajo, buscan dotar a los agentes inteligentes con las capacidades necesarias para enfrentar problemas complejos de la vida real, que involucran ambientes dinámicos, inciertos, limitados en recursos y parcialmente observables. Incluyen el uso de arquitecturas BDI, la integración con Web Services, formalismos de comunicación de alto

nivel (speech acts) y la toma de decisiones mono y multi-atributo cualitativa. Estos trabajos se basan, mayoritariamente, en procesos de razonamiento argumentativo que permiten enfrentar de manera elegante, situaciones con información incompleta y contradictoria, como por ejemplo, las que se presentan en distintas aplicaciones de robótica y en la comunicación de agentes de software.

## 1. Introducción

La capacidad de los *agentes inteligentes* [26] de comportarse de manera *flexible* (*reactiva, proactiva y social*), los han convertido en herramientas muy interesantes para abordar problemas complejos del mundo real. Estas habilidades, suelen ser fundamentales en el tipo de ambientes en que estamos interesados, donde se debe tratar con información incierta, incompleta, contradictoria y cambiante y donde, además, se debe considerar un contexto con recursos computacionales limitados.

Proveer con estas capacidades a un agente no es una tarea sencilla. Ciertos enfoques, enfatizan las capacidades *reactivas* [8] de los agentes basadas en la información disponible en el ambiente, pero este enfoque suele exhibir serias falencias en ambientes parcialmente observables. En otros casos, la toma de decisiones del agente se basa en enfoques más formales como la teoría de decisión clásica [5], aunque no es claro en estos casos como tratar con el cambio o la incorporación de nuevos criterios en las preferencias del agente, considerando la naturaleza estática de la función de utilidad determinada en la etapa de diseño del agente [14].

Si bien es difícil encontrar soluciones integrales a todos los problemas que debe enfrentar un agente inteligente, se pueden identificar ciertos aspectos que jugarán un rol principal en el buen funcionamiento de cualquier agente exitoso. En primer lugar, la *información relevante* que éste tenga disponible respecto al problema bajo consideración, es un factor importante a la hora de

<sup>1</sup>Las investigaciones realizadas en el LIDIC son financiadas por la Universidad Nacional de San Luis y por la Agencia Nacional de Promoción Científica y Tecnológica.

<sup>2</sup>Laboratorio de Investigación y Desarrollo en Inteligencia Artificial. Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina.

tomar una decisión “correcta”. En consecuencia, un aspecto fundamental es ampliar las capacidades perceptivas del agente para obtener esta información. Por ejemplo, en el caso de los agentes de software, ésto puede involucrar el acceso a diversas fuentes de información o la comunicación con otros agentes para intercambiar información que pueda ser útil para resolver su problema.

El acceso a la información relevante, no es condición suficiente para garantizar su uso efectivo por parte del agente. Esta información podrá ser incompleta y contradictoria, y la decisión de cuando adoptarla como nueva creencia justificada, podrá involucrar alguna forma de *razonamiento orientado a las creencias* (o *razonamiento teórico* según [26]), que permita contrastar la nueva información disponible con las creencias previas del agente.

Los pasos previos, sirven de soporte para el proceso esencial de un agente, la *toma de decisiones*, encargado de determinar cual de las acciones disponibles, es la “mejor” acción a realizar en esas circunstancias. Este proceso podrá incluir la ponderación de distintos caminos alternativos de acción, la determinación de cuándo una línea de acción ya no tiene sentido de ser continuada o bien, compatibilizar los distintos criterios que deben ser considerados respecto a las alternativas disponibles. Estas actividades, involucran en general alguna forma de *razonamiento orientado a la acción*, conocido popularmente como *razonamiento práctico* [26].

En nuestra línea de investigación, se atacan estos 3 aspectos fundamentales referidos al diseño de un agente inteligente: 1) el acceso a la *información relevante*, 2) el *razonamiento teórico* con información incompleta y contradictoria y 3) el *razonamiento práctico* necesario para la toma de decisiones en ambientes complejos con recursos computacionales acotados. Respecto al primer punto, en la sección 2.1.1 se describirán brevemente nuestros trabajos para proveer de información a los agentes de software mediante el uso de Web services. La sección 2.1.2 por su parte, explicará nuestra propuesta para aquellos casos en que la información es obtenida mediante la comunicación con otros agentes utilizando *speech acts*. El mecanismo utilizado para razonar con información incompleta y contradictoria es brevemente introducido en 2.2.1. Por último, los aspectos vinculados con el razonamiento práctico basado en BDI son presentados en 2.3.1, y la toma de decisiones basada en múltiples criterios se des-

cribe en 2.3.2. El trabajo concluye con una breve descripción de los resultados esperados y obtenidos en esta línea de investigación (sección 3) y una síntesis de los trabajos de tesis realizados en estos temas (sección 4).

## 2. Líneas de Investigación y desarrollo

A continuación, se describen las ideas generales de los temas que se están investigando en nuestra línea, de acuerdo a los criterios especificados en la sección previa.

### 2.1. Accediendo a información desde agentes de software

#### 2.1.1. Agentes y Web services

Las fuentes de información a las que puede recurrir un agente son numerosas y variadas. Entre estas fuentes están los *Web services* (WS) ya que responden a un comportamiento bien definido, ideal para ser utilizado por agentes computacionales actuando como clientes. De esta manera, se combinan dos dominios de aplicación diferentes; por un lado los WS que pueden invocar los servicios de algunos agentes y vice versa, los agentes pueden invocar a los WS.

Intuitivamente, y según la definición provista por WIKIPEDIA, un *servicio web* (en inglés *Web service*) “es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.”

La plataforma JADE [4] provee una buena interconectividad entre ambos dominios (agentes y WS), mediante el WSIG (Web Service Integration Getway). Esto se consigue de una manera casi transparente para el programador de agentes ya que en primer lugar, la interacción se logra mediante el uso de solamente dos clases de mensajes ACL: de información y de requerimiento. En segundo lugar, se pueden utilizar los directorios establecidos como repositorios, a fin de facilitar acciones como registrarse, desregistrarse y modificaciones. Finalmente, quizás el aspecto más interesante, es la posibilidad de presentar

los WS como si fuesen facilidades brindadas por agentes convencionales y, en sentido contrario, los servicios de los agentes como si fuesen WS. Esto se consigue mediante la facilidad de *páginas amarillas* de JADE, que permite que los agentes puedan requerir los servicios de un agente o WS sin notar la diferencia entre ambos.

Más allá de utilizar distintos WS para proveer de información a los agentes, nuestros trabajos también incluyen mejoras en la *identificación* de los WS relevantes. A fin de reducir los inconvenientes que surgen en la especificación de los servicios debido a la falta de un estándar, nuestra línea está trabajando en el uso de *búsquedas por similitud* que permiten acceder a los servicios, de acuerdo a ciertas *características generales* especificadas por los agentes.

### 2.1.2. Comunicación entre agentes

Uno de los temas de interés en las actividades de investigación y desarrollo de nuestra línea, es el uso de los actos del habla (speech acts [3]) en la comunicación entre agentes. Desde las primeras consideraciones de los actos del habla en el procesamiento del lenguaje natural [9], se observó la conveniencia de considerar a estos actos como acciones que forman parte de planes. ¿Por qué una persona elige comunicar una idea en lugar de otra? Para encontrar las razones por las que se genera una frase, es mucho más simple tomarla como una acción antes que intentar encontrar esas razones a partir del análisis de las palabras que componen una frase. La comunicación es una actividad netamente social que los agentes usan como parte de un plan para conseguir sus metas. Cada acto comunicativo tiene su razón de ser, por eso se puede llegar a comprender mejor un enunciado al considerarlo como un operador que modifica el estado actual del mundo.

Durante su accionar, los agentes deben escoger el acto del habla más adecuado para cada situación, lo que es una tarea compleja y puede llegar a requerir analizar distintas variables y contextos. En este sentido, para que los agentes puedan proceder de una manera aceptable, investigamos el uso de un formalismo de argumentación rebatible [23] para la elección de las próximas acciones o actos del habla a ser aplicados. Este formalismo tiene en cuenta las condiciones necesarias para aplicar un operador y los cambios producidos [1].

## 2.2. Razonando con información incompleta y contradictoria

### 2.2.1. Programación en Lógica Rebatible

La argumentación es un modelo de razonamiento basado en la construcción y evaluación de argumentos que interactúan entre sí. Estos argumentos tienen como fin, soportar, explicar o atacar enunciados que pueden ser creencias, opiniones, decisiones, etc. Existen numerosas propuestas de sistemas de argumentación abstractos y concretos [10], donde el sistema abstracto propuesto por Dung [15], ha sido un trabajo seminal sobre el cual se han desarrollado otros sistemas argumentativos. De los sistemas de argumentación concretos, nosotros utilizamos para la representación de conocimiento y razonamiento a DeLP (*Defeasible Logic Programming*) [23], un formalismo de argumentación rebatible basado en programación lógica. Este formalismo ha sido exitosamente integrado en aplicaciones del mundo real tales como búsqueda inteligente en la web [11, 13], clustering [24], procesamiento del lenguaje natural [12] y knowledge management [7], entre otras.

En particular, para realizar los trabajos de investigación y desarrollo, en la línea se usa DeLP-Server [22], una implementación del intérprete de DeLP que provee un servicio de razonamiento argumentativo para agentes deliberativos. Este servicio se ejecuta como una aplicación stand-alone que interactúa con los agentes deliberativos usando el protocolo TCP/IP. De esta manera, los agentes son vistos como clientes que envían consultas al servidor (DeLP-Server) el cual, luego de evaluarlas, retorna las respuestas a los clientes. Un cliente puede conectarse a múltiples DeLP-Servers y un DeLP-Server puede recibir consultas de múltiples clientes. Esto hace que DeLP-Server sea muy adecuado para la programación multi-agente.

Los trabajos de investigación realizados en esta línea, se llevan a cabo siguiendo un enfoque teórico-práctico. En algunos casos, la confrontación con el mundo real se hace utilizando un grupo de tres robots móviles Khepera 2 [25], que pueden manipular y transportar objetos y poseen además distintas capacidades de sensado. Como muchos de los aspectos relacionados con el comportamiento de los robots (agentes), requieren un lenguaje de representación expresivo que refleje fácilmente los procesos de decisión realizados por los robots, se ha desarrollado en Prolog, un frame-

work [16, 20] que les permite a los robots Khepera 2 razonar y representar conocimiento en DeLP, a través de la interacción con un DeLP-Server.

### 2.3. Aspectos del razonamiento práctico

#### 2.3.1. Integrando agentes BDI y argumentación

Uno de los objetivos principales de esta línea, es el estudio y desarrollo de sistemas con agentes basados en el modelo BDI. Las arquitecturas (y modelos) BDI proponen a la trinidad **BDI (Beliefs, Desires e Intentions)** como los elementos claves del estado mental de un agente para tomar las decisiones acerca de cuándo y cómo actuar. Este tipo de enfoque ha demostrado una gran flexibilidad y efectividad en diversos problemas de gran complejidad del mundo real, lo que ha llevado a un creciente interés en la investigación de sus aspectos teóricos pero también de las plataformas que soportan el desarrollo de este tipo de agentes. En este sentido, un objetivo general de nuestro trabajo es el estudio de frameworks específicos para el desarrollo de agentes BDI como, por ejemplo, el framework de distribución gratuita Jadex.

Actualmente, nuestro trabajo principal en este área, está orientado a integrar planes escritos en Jadex con WS que proveen información relevante para distintos problemas. En particular, uno de los servicios más importantes que puede acceder un agente Jadex es el de razonamiento rebatible provisto por el DeLP-Server. De esta forma, se está logrando la integración en un mismo marco de trabajo, de capacidades de razonamiento teórico y razonamiento práctico.

#### 2.3.2. Toma de decisión multicriterio basada en argumentación

La idea de articular decisiones en base a argumentos es relevante para varios enfoques diferentes de decisión, tales como decisión bajo incertidumbre, decisión multi-criterio y decisiones basadas en reglas. Teniendo en cuenta la experiencia de la línea en el desarrollo de un framework de decisión individual basado en argumentación rebatible [19] (sistema de decisión basado en reglas), actualmente se está redefiniendo este framework para que el tomador de decisiones pueda ponderar múltiples criterios de elección al momento de tomar una decisión.

DeLP es un lenguaje expresivo que nos permitió reflejar fácilmente los procesos de decisión de los agentes que tomaban decisiones usando

el framework propuesto en [19]. Sin embargo, no puede tratar con incertidumbre explícita dado que la rebatibilidad está codificada directamente en el lenguaje objeto. Es por eso, que al momento de extender el framework de [19] para que considerara múltiples criterios, se utilizó P-DeLP [2] en lugar de DeLP. P-DeLP extiende las capacidades de razonamiento de DeLP al permitir la incorporación de incertidumbre explícita, y esta característica primordial nos facilita la integración de un método existente de agregación multi-criterio [6] al framework de decisión.

Si bien el framework de decisión multi-criterio basado en argumentación aún no está completamente formalizado, la idea conceptual consiste en utilizar el mecanismo de inferencia de P-DeLP, generando todos los argumentos posibles a favor y en contra de cada acción (comparándolas de a pares) considerando en cada argumento un único criterio de preferencia del agente. Luego, se realiza una agregación de aquellos argumentos que soporten la misma conclusión (por ejemplo, que la acción  $a_1$  es mejor que la  $a_2$ ) y el procedimiento de garantía se aplica sobre los argumentos resultantes de esa agregación.

## 3. Resultados esperados/obtenidos

Los resultados obtenidos respecto a la toma de decisiones basadas en argumentación han sido presentados en las siguientes publicaciones científicas [19, 21, 18, 17]. Estos trabajos muestran la evolución de una propuesta original para resolver un problema de aplicación de toma de decisiones basadas en argumentación [17], hasta la presentación de un modelo teórico general para la toma de decisiones ([19]). Este último, combina reglas de decisión y argumentos, y se formaliza su comportamiento de elección comparándolo con el basado en los principios de la Teoría de Decisión Clásica. Los resultados esperados en esta línea de trabajo, se orientan a lograr una propuesta más acorde a la forma en que los seres humanos deliberan y finalmente toman o entienden una elección. Uno de nuestros trabajos futuros que tiende a lograr este objetivo, es la reformulación de este framework general para que permita la agregación dinámica de múltiples criterios de preferencia.

En el área de integración de agentes BDI, WS y razonamiento rebatible, se está desarrollando actualmente un sistema que permite acceder (vía WS) a información meteorológica actualizada, que

es transformada y combinada con información sobre las preferencias de un usuario especificadas en DeLP. Luego, un proceso de razonamiento rebatible es utilizado para determinar cual es la mejor acción a tomar en cada circunstancia.

En lo referente a la comunicación de agentes basada en speech acts y argumentación, nuestra idea es extender el trabajo previo realizado en [1] para considerar protocolos de interacción más generales.

## 4. Formación de recursos humanos

Trabajos de tesis vinculados con las temáticas descriptas previamente:

- 2 tesis doctorales en ejecución (co-dirección con investigadores del LIDIA (UNS))
- 1 tesis de maestría aprobada (co-dirección con investigadores del LIDIA (UNS))
- 1 tesis de maestría en ejecución (co-dirección con investigadores del LIDIA (UNS))
- 2 tesis de Licenciatura aprobadas.
- 5 tesis de Licenciatura en ejecución.

## Referencias

- [1] G. Aguirre, M. Errecalde, and G. Simari. Actos del habla en la comunicación. una propuesta usando programación lógica rebatible. In *Proceedings del XIV Congreso Argentino de Ciencias de la Computación (CACIC 2008)*, pages 2009–2020, 2008.
- [2] Teresa Alsinet, Carlos I. Chesñevar, Lluís Godo, and Guillermo R. Simari. A logic programming framework for possibilistic argumentation: Formalization and logical properties. *Fuzzy Sets and Systems*, 159(10):1208–1228, 2008.
- [3] J. L. Austin. *How to Do Things with Words*. Harvard University Press, Cambridge, MA, Cambridge, MA, 1962.
- [4] F. Bellifemine, G. Caire, and D. Greenwood. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.
- [5] James O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer, 2nd edition, 1985.
- [6] R. Bisdorff. Preference aggregation with multiple criteria of ordinal significance. In D. Bouyssou, M. Janowitz, F. Roberts, and A. Tsouki, editors, *Annales du LAMSADE*, pages 25–44, October 2004.
- [7] Ramón F. Brena, José Luis Aguirre, Carlos Iván Chesñevar, Eduardo H. Ramírez, and Leonardo Garrido. Knowledge and information distribution leveraged by intelligent agents. *Knowl. Inf. Syst.*, 12(2):203–227, 2007.
- [8] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation-Computational Intelligence*, 2(1):14–23, 1986.
- [9] B. Bruce and D. Newman. Interacting plans. *Cognitive Science*, pages 195–233, 1978.
- [10] C. Chesñevar, A. Maguitman, and R. P. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.
- [11] C. I. Chesñevar and A. G. Maguitman. Arguenet: an argument-based recommender system for solving web search queries. In *Intelligent Systems, Proc. of 2nd International IEEE Conference*, volume 1, pages 282–287, June 2004.
- [12] Carlos I. Chesñevar and Ana G. Maguitman. An argumentative approach to assessing natural language usage based on the web corpus. In *Proc. of the 16th ECAI Conference. Valencia, Spain*, pages 581–585. IOS Press, August 2004.
- [13] Carlos Iván Chesñevar, Ana Gabriela Maguitman, and Guillermo Ricardo Simari. A first approach to argument-based recommender systems based on defeasible logic programming. In *NMR*, pages 109–117, 2004.
- [14] J. Doyle and T. Thomason. Background to qualitative decision theory. *AI Magazine*, 1999.
- [15] P. M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
- [16] E. Ferretti, M. Errecalde, A. García, and G. Simari. Khedelp: A framework to support defeasible logic programming for the khepera robots. In *ISRA*, pages 98–103, San Miguel Regla, Hidalgo, México, August 2006.
- [17] E. Ferretti, M. Errecalde, A. García, and G. Simari. An application of defeasible logic programming to decision making in a robotic environment. In *LPNMR*, volume 4483 of *LNAI*, pages 297–302, AZ, USA, May 2007. Springer.
- [18] Edgardo Ferretti, Marcelo Errecalde, Alejandro García, and Guillermo Simari. Defeasible decision making in a robotic environment. In *Anales del XIII CACIC*, pages 1335–1446, Octubre 2-5 2007.
- [19] Edgardo Ferretti, Marcelo Errecalde, Alejandro García, and Guillermo Simari. Decision rules and arguments in defeasible decision making. In *COMMA*, volume 172, pages 171–182, Toulouse, France, May 2008. IOS Press.
- [20] Edgardo Ferretti, Marcelo Luis Errecalde, Alejandro Javier García, and Guillermo Ricardo Simari. Khepera robots with argumentative reasoning. In *AMIRE Symposium*, pages 199–206, Bs.As., Argentina, October 2-5 2007.
- [21] Edgardo Ferretti, Nicolás Rotstein, Marcelo Errecalde, Alejandro García, and Guillermo Simari. Defeasible decision making in a multi-robot environment. *Research in Computing Science*, 32:150–160, September 2007. Special Issue: Advances in Artificial Intelligence and Applications.
- [22] Alejandro Javier García, Nicolás D. Rotstein, Mariano Tucát, and Guillermo Ricardo Simari. An argumentative reasoning service for deliberative agents. In *KSEM*, volume 4798 of *LNCS*, pages 128–139. Springer, 2007.
- [23] Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming: an argumentative approach. *Theory and Practice of Logic Programming*, 4(2):95–138, 2004.
- [24] Sergio A. Gómez and Carlos I. Chesñevar. A hybrid approach to pattern classification using neural networks and defeasible argumentation. In *Proc. of the Intl. FLAIRS Conference. Miami, USA*, pages 393–398, May 2004.
- [25] K-Team. Khepera 2. <http://www.k-team.com>. A miniature mobile robot designed as a research and teaching tool.
- [26] Michael Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, Chichester, England, 2002.