

SISTEMAS MULTI-AGENTE

Presentación del Curso

Dr. Alejandro Guerra-Hernández
<https://www.uv.mx/personal/aguerra>
aguerra@uv.mx

Maestría en Inteligencia Artificial

IIIA - Instituto de Investigaciones en Inteligencia Artificial
Universidad Veracruzana
Campus Sur, Calle Paseo Lote II, Sección 2a, No 112
Nuevo Xalapa, Xalapa, Ver., México 91097

21 de agosto de 2023

Este documento presenta la experiencia educativa de Sistemas Multi-Agente (SMA) del IIIA en la Universidad Veracruzana (UV). Aunque se trata de un curso general sobre los SMA, el contenido del mismo está influenciado por las Líneas de Generación y Aplicación del Conocimiento del IIIA: Asumimos una clara preferencia por el enfoque *Belief-Desire-Intention* (BDI) para la agencia racional, tal como se modela en AgentSpeak(L) y se implementa en Jason; al igual que por los ambientes basados en artefactos CArtAgO. Las aplicaciones sobre aprendizaje, minería de datos y simulación social son también resultado de tal influencia. El material está organizado para ofrecer un curso de 60 horas, durante un semestre, con dos sesiones de dos horas por semana. El resto del documento presenta los objetivos del curso, la manera de evaluar el desempeño, los recursos y las herramientas que utilizaremos; el calendario de las sesiones; y la bibliografía básica contemplada.

1. Objetivos

1. El estudiante identificará los conceptos fundamentales de la agencia racional y los sistemas multi-agentes: intencionalidad e interacción.
2. El estudiante adquirirá las habilidades entender la formalización de estos conceptos usando lógicas BDI y semánticas operacionales.
3. El estudiante adquirirá las habilidades para resolver problemas complejos con el lenguaje de Programación Jason y tecnologías asociadas, *e.g.*, ambientes y organizaciones.
4. El estudiante podrá aplicar las habilidades previas en la resolución de problemas típicos del área, *e.g.*, simulación, minería de datos distribuida, cooperación, etc.

2. Evaluación

La nota final, será calculada de la siguiente forma:

- Las tareas que cubren el 70 % de la nota final. Las tareas incluyen escritura de ensayos, exposiciones y, por supuesto, ejercicios de programación.
- El proyecto integrador cubre el 30 % de la nota final.

Para obtener una **nota aprobatoria**, el alumno deberá haber obtenido notas aprobatorias en todas estas actividades, es decir, no aprobar cualquier elemento parcial de la evaluación, implica no aprobar el curso.

3. Tareas

Las tareas pueden requerir investigación bibliográfica, la resolución de ejercicios teóricos, la experimentación en computadora y/o exposiciones frente a grupo. La entrega y evaluación de éstas se realizará conforme a los siguientes lineamientos:

Entrega. Las tareas se entregan al inicio de clase del día designado para ello. El mérito de las mismas decrece 25 % por cada 24 horas de retraso. El calendario del curso marca las fechas tentativas para cada tarea, su entrega suele ser dos semanas más tarde.

Formato. Las tareas se deben procesar con L^AT_EX, siguiendo la plantilla de este mismo documento. En todas las partes que involucran código de computadora o algoritmos, estos deberán documentarse apropiadamente con comentarios y ejemplos de corridas. En algunos casos los ejercicios propuestos indican explícitamente las corridas que deben probarse. Por ejemplo, Si se les pide que implementen en Prolog un predicado `allPerms` que regrese todas las permutaciones de una lista y lo prueben con las lista `[1,2,3]`, la solución debería reportarse como sigue:

```
1  %% allPerms computa todas las permutaciones de los elementos
2  %% de la lista L en la lista de listas L2.
3  %% ?- allPerms([1,2,3],Resp).
4  %% Resp = [[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1],
5  %%          [3, 1, 2], [3, 2, 1]].
6
7  allPerms(L,L2) :-
8    findall(Perm, permutation(L,Perm), L2).
```

Evaluación. Cada ejercicio de las tareas tiene asignado el puntaje a obtener. Para que un ejercicio o pregunta de la tarea sea evaluado, deberá estar resuelto completamente. Es más redituable invertir el tiempo en contestar una pregunta de manera correcta y completa, que responder a dos de manera incompleta.

Plagio. Cualquier forma de plagio causa la expulsión definitiva del curso, y por consiguiente, de la maestría. Esto incluye: Reportar trabajo de otros como propio y no citar pertinentemente las referencias usadas y el código de otros. La responsabilidad en el uso de IAs generativas recae en el alumno, *i.e.*, la IA no plagia, plagia el alumno; la IA no se equivoca, se equivoca el alumno que no la corrige.

3.1. Proyecto integrador

Se trata de un proyecto práctico que requiere la aplicación las diversas técnicas introducidas en el curso, posiblemente complementado con el contenido de otros cursos de este semestre. Se comienza a definir a la mitad del mismo para tratar de considerar los intereses de investigación de los estudiantes. Las fechas de las revisiones parcial y final del proyecto integrador aparecen más adelante en este documento, en el calendario de actividades.

4. Recursos del curso

Los recursos del curso incluye algunas páginas web de soporte, los lenguajes de programación a utilizar y diversos ambientes de desarrollo. En lo que sigue, las ligas llevan a estos recursos.

4.1. Soporte

- La página web del curso es la referencia de soporte por excelencia.
- Usaremos como complemento el libro *An Introduction to MAS* de Michael Wooldridge.
- The missing semester es un curso sobre diversas herramientas computacionales que todo el mundo espera que sepas usar ¡Y éste no es el caso! Muy útil para todas las experiencias educativas del programa.
- Overleaf debería ser la forma más sencilla de comenzar a trabajar con \LaTeX . Su principal limitante es que al ser un editor en línea, requiere una conexión a internet para su uso. Si esto no es posible, instalen \LaTeX en el sistema operativo en que estén trabajando.

4.2. Lenguajes de programación básicos

- Jason.
- Netlogo.

La bibliografía básica está a disposición de ustedes en la biblioteca del departamento o en mi oficina. Algunos de los artículos están disponibles en versión electrónica gracias a las suscripciones de nuestra biblioteca central.

5. Calendario

Este año las sesiones se llevarán a cabo los martes de 12:00 a 14:00 hrs., en el Aula Híbrida; y los viernes de 10:00 a 12:00 hrs., en el Centro de Cómputo, ambos en el tercer piso del edificio C del IIIA. Las sesiones se organizarán como sigue:

| Fecha | Tema | Tarea |
|------------|------------------------------------|-------|
| 22/08/2023 | Agencia | |
| 25/08/2023 | | |
| 29/08/2023 | Intencionalidad | |
| 01/09/2023 | | |
| 05/09/2023 | Interacción | |
| 08/09/2023 | | T1 |
| 12/09/2023 | Lógicas BDI | |
| 15/09/2023 | | |
| 19/09/2023 | | |
| 22/09/2023 | AgentSpeak(L) | |
| 26/09/2023 | | T2 |
| 29/09/2023 | Jason | |
| 03/10/2023 | | |
| 06/10/2023 | | |
| 10/10/2023 | Medios ambientes | |
| 13/10/2023 | | P1 |
| 17/10/2023 | | |
| 20/10/2023 | | T3 |
| 24/10/2023 | Comunicación | |
| 27/10/2023 | | |
| 31/10/2023 | Acuerdos | |
| 03/11/2023 | | |
| 07/11/2023 | Cooperación distribuida | T4 |
| 10/11/2023 | | |
| 14/11/2023 | Minería de Datos basada en Agentes | |
| 17/11/2023 | | |
| 21/11/2023 | | |
| 24/11/2023 | Simulación | |
| 28/11/2023 | | |
| 01/12/2023 | | |
| 05/12/2023 | | |
| 08/12/2023 | | |
| 09/01/2024 | | P2 |

Referencias básicas

Wooldridge (2009) ofrece una introducción general a los SMA que complementaremos con el libro de Weiß (2013). Huns and Singh (1998) recopilan

una serie de artículos fundacionales que definieron la transición entre la IA Distribuida y los SMA. Shoham and Leyton-Brown (2008) ofrece una introducción a los SMA más centrada en Teoría de Juegos. Mahmoud (2020) abunda en esa dirección. Evidentemente, Russell and Norvig (2022) dan cuenta más claramente de la interacción entre los SMA y la IA en general. Bordini et al. (2007) es la referencia básica de Jason. Boissier et al. (2020) proveen una revisión más reciente del lenguaje, en el contexto de la programación de ambientes e instituciones. Henderson-Sellers and Giorgini (2005) y Shehory and Sturm (2014) ofrece una revisión de diversas metodologías de ingeniería de software aplicadas a la programación orientada a agentes.

Referencias

- Boissier, O., Bordini, R. H., Hübner, J. F., and Ricci, A. (2020). *Multi-Agent Oriented Programming: Programming Multi-Agent Systems using JaCaMo*. Intelligent Robotics and Autonomous Agents. MIT Press, Cambridge, MA, USA.
- Bordini, R. H., Hübner, J. F., and Wooldridge, M. (2007). *Programming Multi-Agent Systems in Agent-Speak using Jason*. John Wiley & Sons Ltd.
- Henderson-Sellers, B. and Giorgini, P. (2005). *Agent-Oriented Methodologies*. IDEA Group Publishing, Hershey, PA, USA.
- Huns, M. and Singh, M., editors (1998). *Readings in Agents*. Morgan Kaufman Publisher, San Mateo, CA, USA.
- Mahmoud, M. S. (2020). *Multiagent Systems: Introduction and Coordination Control*. CRC Press, Boca Raton, FL, USA.
- Russell, S. and Norvig, P. (2022). *Artificial Intelligence: A Modern Approach*. Pearson, New York, NY, USA, fourth, global edition.
- Shehory, O. and Sturm, A. (2014). *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*. Springer-Verlag, Berlin, Germany.
- Shoham, Y. and Leyton-Brown, K. (2008). *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press.
- Weiß, G. (2013). *Multiagent Systems*. Intelligent Robotics and Autonomous Agents. The MIT Press, Cambridge, MA, USA, second edition.

Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. John Wiley & Sons, LTD, West Sussex, England, 2nd edition.