Intelligent Agents Tutoring Environment

Fábio Pereira[1], Jacques Robin[2]

[1]Universidade Estadual do Sudoeste da Bahia (UESB)

[2]Centro de Informática – Universidade Federal de Pernambuco (UFPE)

{fmp2, jr}@cin.ufpe.br

Abstract. *This paper presents IATE: an intelligent agents tutoring environment. IATE explores the benefits of the integration between artificial intelligence and software engineering to build an environment that can serve for artificial intelligence and multi-agent systems teaching, research and tests. It integrates several development methods, like component based development, aspect oriented programming, model driven architecture and product line engineering through an extension of the KobrA method.*

Keywords: *artificial intelligence, tutoring environment, multi-agent systems.*

1. Introduction

Overcoming the limitations of the existent simulation environments used as framework for artificial intelligence application development, teaching and researching, we present the development of the IATE – Intelligent Agents Tutoring Environment.

The aim of this research project is the development of an extensible, fully parametric, multi-agent environment that can serve as reusable kernel for: (1) Artificial Intelligence (AI) and Multi-Agent System (MAS) tutoring environment; (2) AI and MAS controlled experiments environments; and (3) AI and MAS based team ball sport computer games. The research evaluates, adapts and extends the model-driven, aspect-oriented, component-based, product line engineering methodology KobrA [Atkinson et. al. 2001]. This is a highly multi-disciplinary research project on the intersection of software engineering, artificial intelligence, multi-agent systems, simulation, games and educational software fields.

2. A Fully Parametric Intelligent Agents Environment

A simulation environment is a framework for AI application development and researching like autonomous agents' projects, multi-agent collaboration, knowledge acquisition strategies, real-time reasoning, etc. It must be initialized with configuration parameters, which purpose is identify the relevant subclass of the environment simulated class, besides creating a random instance peculiar of that subclass. Once initialized, the environment simulator passes for its processing cycle, where it will generate the agents' perceptions starting from the current state of the environment. After having generated, those perceptions are sent to the agents. It is also function of the simulator to receive and to process the agents chosen actions, in way to update the state of the environment in order to reflect those actions.

As examples of simulation environments, we can mention the Wumpus World [Russel and Noving 2002], used in the teaching of AI, and the RoboCup [RoboCup 2005], an international initiative with the purpose of encouraging the research in AI and robotics and to integrate education projects.

Although the Wumpus World is a simple game in comparison with the modern computers patterns of games, it is an excellent environment for intelligent agents' tests. It was popularized by its use as tool for didactic explanation of several investigating problems in artificial intelligence in the text book more used to teach AI. In spite of its quality as illustrative example, it suffers of the two following limitations as teaching and execution environment for controlled experiments of empiric research in AI and MAS: (1) all of its instances possess the same characteristics: it is always treated as an inaccessible, discrete, deterministic, non-episodic, finite and mono-agent environment, and (2) there is no diversity at level of agents' classes, of their perceptions, actions and goals, as well as of the objects and locations of the environment.

The RoboCup objective is the creation of multiple robots that collaborate amongst themselves for the resolution of dynamic problems. Although the objective of RoboCup is a world cup with real robots, it offers a software platform for research accomplishment in the aspects of software [Kitano et. al. 1997]. A soccer game is a multi-agent specific real time environment, but it is very attractive in the point of view of distributed AI and multi-agent research. Although the simulated robot's soccer environment is partly complementary to the Wumpus World, for being a continuous, non-deterministic, asynchronous and multi-agent environment with great size at instance level, it shares with the Wumpus World two limitations: (1) those characteristics are fixed and (2) it not presents diversity of agents' classes, perceptions, actions, objectives, objects and locations.

Consequently, even using both environments the problems of a lot of real environments classes cannot be taught or researched empirically, in particular all the accessible or episodic environments or involving a great conceptual diversity.

Overcoming the limitations of the Wumpus World and of the RoboCup we propose the development of the IATE, a simulation environment for ends of teaching, research and tests in AI. IATE owe to be a game with parameters for all of its characteristics, allowing the creation of any possible kind of environment.

IATE is being developed accordingly a methodology based in reusable components automatically generated from an UML ontology. It should also be integrated to a web site for AI teaching purposes and to be the only one in its category.

The environment simulates a single base domain – a futuristic fusion team ball sport – in all of its possible variations in terms of accessibility, determinism, dynamicity, physics modeling granularity and agent communication, as well as the size and diversity of agent action, agent perception, objects and location sets.

3. Development Methodology

Despite agent-oriented software engineering is a very active area of research, there are essentially two approaches to agent development. One dictates that agent-oriented methodologies should be developed independently of object-oriented methodologies. The alternative approach is to take an existing OO methodology and extend it to support agent concepts. The approach of this research project is evaluates, adapts and extends the methodology KobrA to development of the environment: it extends KobrA with concepts from formal methods, agent-oriented software engineering, declarative programming and generative meta-programming. It also extends KobrA with a sub-methodology for GUI (Graphical User Interface) engineering. The KobrA method

represents a synthesis of several advanced software engineering technologies, including production line development, component based software development and the use of frameworks, among others.

The IATE project is based on some software standards and development techniques as Model Driven Architecture (MDA) [OMG 2005] and Aspect-Oriented Programming (AOP) [Colyer 2004].

4. Current Stage of Development

As a result of IATE's fist steps we already have developed the UML profile meta-model for agents' environments that can be seen in [http://www.cin.ufpe.br/~fmp2/IATE] and the abstract domain model. Profiles [Blanc 2005] provide a mechanism to manage UML extensions: standard ways for handling certain software languages, such as Java or .NET; information needed for certain business domains, such as health or insurance; or mechanisms for advanced concepts, such as frameworks, or complex real-time systems.

In IATE environment meta-model the simulator is an active class that posses some characteristics and states and controls changes in the environment. A state shows the situation of the environment entities (agents and objects) in some point of the time. Agents are components that can send actions for and receive information from the environment through interfaces. Automatic changes in the environment, behavior of entities and actions sent by agents are limited through constraints.

5. Conclusions and Further Work

In this paper, we have presented the IATE project, which is an extensible, fully parametric, multi-agent artificial intelligence tutoring environment. IATE was motivated by the limitations of the simulation environments currently existent and by the integration of the potential of AI technology in the context of software engineering. IATE purpose it to improve the development of AI teaching and research.

There are several issues remaining for future work. It include the development of a weaving engine that can automatically build MAS domain specific models from different meta-models and uses a specific model transformation language [Blanc, 2005].

References

Atkinson, C. et al. (2001) "Component-Based Product Line Engineering with UML", Component Software Series, Addison-Wesley.

Blanc, X. (2005) "MDA en action : Ingénierie logicielle guidée par les modèles", Architecte logiciel, Eyrolles, France.

Colyer, A. (2004) "AspectJ in Action - Practical Aspect-Oriented Programming", Computer Journal, Vol. 47, No. 1, p. 134-135.

Kitano, H., et al (1997) "Robocup: The Robot World Cup Initiative", In International Conference on Autonomous Agents, United States, ACM Press.

OMG (2005) "The Architecture of Choice for a Changing World: OMG Model Driven Architecture", Available at: http://www.omg.org/mda/, August.

RoboCup (2005) "Overview", RoboCup Official Web Site, Available at: http://www.robocup.org/overview/2.html, September.

Russell, S. and Norvig, P. (2002) "Artificial Intelligence: A Modern Approach" 2nd ed., Prentice Hall.