ELMS: An Environment Description Language for Multi-agent Simulation

Fabio Y. Okuyama\textsuperscript{1}, Rafael H. Bordini\textsuperscript{2,1}, and Antônio Carlos da Rocha Costa\textsuperscript{3,1}

\textsuperscript{1} Programa de Pós-Graduação em Computação, Universidade Federal do,
Rio Grande do Sul (UFRGS), Porto Alegre RS, Brazil
okuyama@inf.ufrgs.br

\textsuperscript{2,\ast} Department of Computer Science, University of Liverpool,
Liverpool L69 3BX, U.K
R.Bordini@csc.liv.ac.uk

\textsuperscript{3} Escola de Informática, Universidade Católica de Pelotas (UCPel),
Pelotas RS, Brazil
rocha@atlas.ucpel.tche.br

Abstract. This paper presents ELMS, a language used for the specification of
multi-agent environments. This language is part of the MAS-SOC approach to
the design and implementation of multi-agent based simulations. The approach is
based on specific agent technologies for cognitive agent programming and high-
level agent communication, as well as ELMS. We here concentrate on introducing
ELMS, which allows the description of environments in which agents are to be
situated during simulations. The ELMS language also allows the definition of the
agents’ perceptible properties and the kinds of (physical) interactions, through
action and perception, an agent can have with the objects of the environment or
the perceptible representations of the other agents in the environment.

1 Introduction

The goal of our overall project is to develop an approach and platform for the development
of multi-agent based social simulations, incorporating agent technologies for specifying
and running cognitive agents. When a multi-agent system is fully computational (i.e.,
not situated in the real world, the Internet, etc.), the specification of the (simulated) envi-
ronment where agents are situated is an important task in the engineering of the system,
which is not, however, normally addressed in the literature: environments are usually
simply considered as “given”, or sometimes environments are themselves modelled as
agents. Nevertheless, the characteristics of environments are quite different from those
of cognitive agents. Therefore, in our practical work, we identified the need for the use
of a language specifically designed for the specification of multi-agent environments.

Based on that experience, we have developed a prototype of an interpreter for an
environment definition language, presented in detail in \cite{1} and mentioned in \cite{2}. The

\textsuperscript{\ast} Current affiliation: Department of Computer Science, University of Durham, Durham DH1 3LE,
U.K. E-mail: \texttt{R.Bordini@durham.ac.uk}

© Springer-Verlag Berlin Heidelberg 2005
language has been designed to support the description of environments for our multi-
agent based social simulations (although it may turn out to be useful more generally). Besides the basic environment properties and objects, the language provides the means
for the specification of the “physical” representation of a simulated agent, which we refer
to as the “body of an agent”\footnote{Note that in referring to agent’s bodies we do not mean to say that our approach is only applicable
to \textit{embodied agents}. By “body” we simply mean whatever physical properties of an agent that
may be \textit{perceptible} by other agents in the environment. This is quite general: if an \textit{environment}
metaphor is present at all in the multi-agent system being developed, in all likelihood some
characteristics of the agents will be perceivable by other agents.}, as well as the various kinds of physical interactions, through action and perception, among agents and objects or other agents in the environment.

This paper is structured as follows. Sect. 2 covers the main ideas of the MAS-SOC
approach to the development of multi-agent based social simulation. We discuss the
classes of environments that can be modelled with ELMS in Sect. 3. Sect. 4 presents
ELMS itself, the language we introduce in this paper and that is designed specifically for
the modelling of multi-agent environments. Then we describe how ELMS environments
are run in Sect. 5. Besides small examples given in Sect. 4 we also give a complete
example in Appendix A.

2 The MAS-SOC Project

The main goal of the MAS-SOC project (\textbf{Multi-Agent Simulations for the \textbf{SOC}ial Sci-
ences}) is to provide a framework for the creation of agent-based social simulations that,
ideally, should not require much experience in programming from users \footnote{This will be available in \textit{Jason} soon, as reported in \cite{5}.}. In particular,
it should allow for the design and implementation of cognitive agents and their social
actions. A graphical user interface facilitates the specification of environments, agents
(their beliefs and plans), and the simulation as a whole. It also helps the management of
libraries of simulation components. From the information input by the user, the system
generates source codes for the interpreter used for agent reasoning (from the representa-
tions of agents’ mental attitudes), and for the ELMS interpreter, whereby environment
objects and agents’ bodies are simulated.

Agents’ practical reasoning is specified in AgentSpeak(L) \footnote{This will be available in \textit{Jason} soon, as reported in \cite{5}.}, using the \textit{Jason} inter-
preter \footnote{This will be available in \textit{Jason} soon, as reported in \cite{5}.} (see also \cite{5}). We do not discuss here the AgentSpeak(L) programming
language, but one can refer to the papers mentioned above, as well as \cite{6,7}, for a com-
plete account of that language. We here concentrate on presenting the ELMS language
and its interpreter.

The interaction between the interpreters (for agents and the environment) and the
graphical interface for creating and controlling the simulations is made possible by the
SACI toolkit, developed as part of the work reported in \cite{8}. This tool also supports the
interactions of agents with the environment (perception and action) as well as speech-act
based agent communication, including interactions such as plan exchange. SACI also
provides the infra-structure that makes it possible for us to run distributed simulations,
thus facilitating large-scale simulations with cognitive agents.