Security in Virtual Worlds, 3D Webs, and Immersive Environments:

Models for Development, Interaction, and Management

Alan Rea
Western Michigan University, USA

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content:	Kristin Klinger
Director of Book Publications:	Julia Mosemann
Acquisitions Editor:	Lindsay Johnston
Development Editor:	Joel Gamon
Publishing Assistant:	Julia Mosemann, Natalie Pronio
Typesetter:	Natalie Pronio
Production Editor:	Jamie Snavely
Cover Design:	Lisa Tosheff

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Security in virtual worlds, 3D webs, and immersive environments : models for development, interaction and management / Alan Rea, editor.
p. cm.
Includes bibliographical references and index.
Summary: "This publication discusses the uses and potential of virtual technologies and examines secure policy formation and practices that can be applied specifically to each"--Provided by publisher.
ISBN 978-1-61520-891-3 (hardcover) -- ISBN 978-1-61520-892-0 (ebook) 1. Computer networks--Security measures. 2. Web sites--Security measures. 3. World Wide Web--Security measures. I. Rea, Alan.
TK5105.59.S442 2011
005.8--dc22
2010045520

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 7
Sociable Behaviors in Virtual Worlds

Francisco Grimaldo
Universitat de València, Spain

Miguel Lozano
Universitat de València, Spain

Fernando Barber
Universitat de València, Spain

Juan M. Orduña
Universitat de València, Spain

ABSTRACT

When simulating three-dimensional environments populated by virtual humanoids, immersion requires the simulation of consistent social behaviors to keep the attention of the users while displaying realistic scenes. However, intelligent virtual actors still lack a kind of collective or social intelligence necessary to reinforce the roles they are playing in the simulated environment (e.g. a waiter, a guide, etc). Decision making for virtual agents has been traditionally modeled under self interested assumptions, which are not suitable for social multi-agent domains. Instead, artificial society models should be introduced to provide virtual actors with socially acceptable decisions, which are needed to cover the user expectations about the roles played in the simulated scenes. This chapter reviews the sociability models oriented to simulate the ability of the agents that are part of an artificial society and, thus, interact among its members. Furthemore, it also includes a full description of a social model for multi-agent systems that allows the actors to evaluate the social impact of their actions, and then to decide how to act in accordance with the simulated society. Finally, the authors show the social outcomes obtained from the simulation of a particular 3D social scenario.

INTRODUCTION

Three-dimensional environments have significantly evolved since their beginning in the late sixties. The continuous increase of the available hardware as well as the improvement of the graphic software, in parallel with the evolution of computer networks, have brought virtual environments closer to the physical world; as science fiction already envisioned (Stephenson, 1992). 3D virtual

DOI: 10.4018/978-1-61520-891-3.ch007

worlds are at the cutting edge of the evolution of the Internet towards the new Web 2.0. Some well-known examples of such kind of applications are Massively Multiuser Virtual Environments (e.g. World of Warcraft (Blizzard Entertainment, 2009), SecondLife® (Linden Lab, 2009)...). The aim of these applications is the immersion of the users within a fictitious world. However, apart from the classical goal of immersive technologies (achieved by means of virtual reality devices such as data gloves or head-mounted displays) these 3D worlds aim at achieving the user's mental immersion by populating virtual worlds with synthetic actors, whose animated behavior resembles their equivalent in reality.

The simulation of virtual worlds is a current research topic with a great number of problems to be tackled. One of them is the challenge of populating virtual worlds with autonomous agents emulating human behaviors. Besides showing a good graphical appearance, these virtual actors must perform like-life behaviors. The behavioral animation requires the development of intelligent systems that can simulate believable behaviors for the 3D characters. This challenge involves dealing with perception, motor or animation control, goal selection, action planning and communication skills to interact with other characters or users. Therefore, this complex problem has led to the integration of different artificial intelligence techniques that reproduce intelligent skills such as autonomy, reactivity, pro-activity and sociability (Wooldridge, 1995).

Day after day, virtual worlds are incorporating new services that not only complement the originals located in the real world but also create a hybrid total experience of the physical and virtual reality, also known as interreality (Kokswijk, 2007). Consequently, the incorporation of social skills in the behavioral animation of different kinds of synthetic characters is a keystone in the development of last generation 3D virtual worlds (Williams *et al.*, 2006; Yee *et al.*, 2007). These social synthetic characters could be used to improve the user's mental immersion in Massively Multiuser Virtual Environments (Rehm & Rosina, 2008). Additionally, they could be used to model different behaviors in crowd simulations, in order to evaluate the overall impact of different policies in critical circumstances such as catastrophic events (Pelechano *et al.*, 2008).

The complexity of the behavioral animation requires splitting the problem and managing each part independently. Since virtual actors should have a reactive nature, which can be easily recognized by the users or other actors, this feature is usually considered crucial for providing credibility. According to this, the literature of virtual humans contains a high number of works focused on reactive skills (Reynolds, 1987). Secondly, proactive behaviors require the use of planning or decision making mechanisms that introduce a new intelligence layer to be integrated. Hence, there is a significative reduction in the number of works covering both behavioral aspects (Funge *et al.*, 1999). Finally, social behaviors are rarely considered, as they add a new complex problem to be integrated (Reilly, 1996). Sociability refers to the ability of agents, which are part of an artificial society, to interact among them. Some works have faced sociability by providing synthetic characters with skills such as navigation (Helbing & Molnar, 1995), emotions or affection (Lim & Aylett, 2009). Nevertheless, intelligent virtual actors still lack a kind of collective or social intelligence beyond these agent-centered skills. As virtual humans usually play a role in the simulated environment (e.g. a waiter, a guide, etc.), they generate certain expectations associated with their activities and their relationships with the rest of agents in the scene, including the user. However, virtual characters' decision-making has generally been modeled under self interested assumptions, which are not suitable for multi-agent domains. Instead, artificial society models should be introduced to provide virtual actors with socially acceptable decisions. Actors need to evaluate the social impact of their actions to decide how to act in accordance with the society. However, social decision-making entails complex cognitive processes that require an abstract knowledge of the elements of the environment. Different works have proposed the inclusion of semantic information in virtual environments (Badler *et al.*, 2000; Farenc *et al.*, 1999). Regardless of the nature of the application, the definition of a basic semantic knowledge will benefit the production, the visualization and the interaction associated to 3D virtual worlds. Moreover, the possibility of reusing ontological descriptions previously designed also constitutes an interesting feature that approaches virtual worlds to the field of the Semantic Web (Grimaldo *et al.*, 2008b).

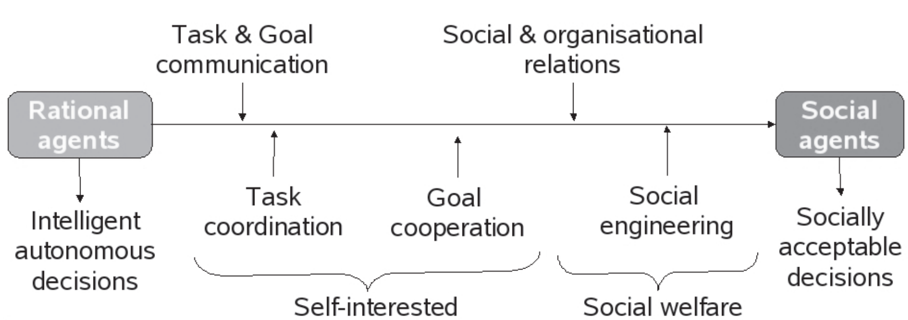
In the next section, we analyze the social decision-making techniques currently proposed and their evolution. We also review the literature around social virtual agents, as a specific use case. Then, we focus on MADeM (Multi-modal Agent Decision Making) (Grimaldo *et al.*, 2008a), a new open-source project based on the Multi-Agent Resource Allocation theory (Chevaleyre *et al.*, 2006), that provides a robust social simulation tool. MADeM is able to simulate different kinds of societies (e.g. elitist, utilitarian, etc), as well as social attitudes of their members such as, egoism, altruism, indifference or reciprocity. All these features are evaluated from the results obtained in the Virtual University Bar, a social simulation environment designed as an interesting example to evaluate the features provided by the MADeM system. The chapter ends with a discussion about the future lines of research and the general conclusions.

Background

Social and organizational models are being studied under the scope of multi-agent systems (MAS) in order to regulate the autonomy of self-interested agents. Nowadays, the performance of MAS is determined not only by the degree of deliberativeness but also by the degree of sociability. In this sense, sociability points to the ability to communicate, cooperate, collaborate, form alliances, coalitions and teams. Being assigned to an organization generally occurs in Human Societies (Prietula, 1998), where the organization can be considered as a set of behavioral constraints that agents adopt (e.g. by the role they play) (Dignum & Dignum, 2001; Hübner *et al.*, 2002).

Figure 1 shows the spectrum obtained between rational agents and social agents. The intelligent behavior provided by rational agents is not enough in environments with shared resources. On the one hand, self interested agents (i.e. agents devoted to accomplish a set of goals) easily come into conflicts in a resource bounded environment even though their goals are compatible. The conflicts are generally derived from the implicit competition produced when situating different agents in a shared environment with a finite number of resources. Coordination techniques can help in

Figure 1. From Rational agents towards social agents



this context to avoid the behavioral inconsistencies produced when displaying the animation of groups of virtual actors. Task and goal passing techniques are nomally used to provide a certain degree of coordination in this context. Furthermore, the agents must be also prepared to cooperate themselves when sharing the same goals, which is also a normal group state.

On the other hand, socially intelligent agents are autonomous problem solvers that have to achieve their goals by interacting with other similarly autonomous entities (Hogg & Jennings, 2001). Bearing this in mind, multi-agent systems are normally referred to as societies of agents, and provide an elegant and formal framework to design social behaviors for autonomous agents.

The definition of a proper MAS organization is not an easy task, since it involves dealing with three main dimensions: functioning, structure, and norms (Hübner *et al.*, 2002). On the one hand, as functionality is normally required, the MAS generally aims at achieving the best plans and cover aspects such as: the specification of global plans, the policies to allocate tasks to agents, the coordination of plans, etc. (Decker, 1998; Tambe, 1997). On the other hand, there are systems that are focused on defining the organizational structure (i.e. roles, relations among roles, groups of roles, etc.) and try to accomplish their global purpose whereas the agents follow the obligations/permissions their roles entitle them (Ferber, 1998; Fox *et al.*, 1998).

From behavioral animation area, much research has been done in virtual agents for the last few years (Badler *et al.*, 1993; Reynolds, 1987; Thalmann & Monzani, 2002). The pioneer work of Dimitri Terzopoulos showed how to design a natural ecosystems animation framework with minimal input from the animator (Tu & Terzopoulos, 1994). He simulated *Artificial fishes* in virtual underwater worlds. However, human behavior is clearly different and more complex to emulate. In (Raupp & Thalmann, 2001; Thalmann & Monzani, 2002) the goal is to design controlled agents with a high degree of autonomy. These agents are an extension of the Belief-Desire-Intention (BDI) architecture described in (Rao & Georgeff, 1991), and they include internal states such as emotions, reliability, trust and others. Emotional architectures have been also applied to virtual agents (animals and humans) to manage sociability and rationality and to produce believable groups of synthetic characters (Delgado-Mata & Aylett, 2004; Prada & Paiva, 2005).

Social reasoning has been extensively studied in multi-agent systems in order to incorporate social actions to cognitive agents (Conte & Castelfranchi, 1995). As a result of these works, agent interaction models have evolved to social networks that try to imitate the social structures found in real life (Hexmoor, 2001). Social dependence networks allow agents to cooperate or to perform social exchanges attending to their dependence relations (i.e. social dependence/power (Sichman & Demazeau, 2001)). Trust networks can define different delegation strategies by means of representing the attitude towards the others through the use of some kind of trust model (e.g. reputation (Falcone *et al*., 2004)). Finally, agents in preference networks express their preferences (normally using utility functions) so that personal attitudes can be represented by the differential utilitarian importance they place on the others' utilities. Following this preferential approach, the MADeM (Multi-modal Agent Decision Making) model (Grimaldo *et al.*, 2008a) is a market-based mechanism for social decision making, capable of simulating different kinds of social welfares (e.g. elitist, utilitarian, etc.), as well as social attitudes of their members (e.g. egoism, altruism, etc.). It considers multi-modal decisions as those that are able to merge multiple information sources received from the group. Hence, MADeM agents express their preferences for the different solutions considered for a specific decision problem using utility functions. Thus, coordinated social behaviors such as task passing or planned meetings can be evaluated to finally obtain socially acceptable behaviors. The next section fully explains the MADeM procedure as well as its implementation as an open-source library over Jason (2009), a well-known multi-agent programming framework.

THE MADEM MODEL

The MADeM model provides agents with a general mechanism to make socially acceptable decisions. In this kind of decisions, the members of an organization are required to express their preferences with regard to the different solutions for a specific decision problem. The whole model is based on the MARA (Multi-Agent Resource Allocation) theory (Chevaleyre *et al.*, 2006). Therefore, it represents each one of these solutions as a set of resource allocations. Thus, the definition domain of MADeM is composed by the following elements:

A set of agents $\vec{A} = \left\{a_1,\ldots,a_n\right\}$ where each a_1 represents a particular agent involved in the decision. A vector of weights $\vec{w} = w_1,\ldots,w_n$ is associated to each agent representing the internal attitude of the agent towards other individuals.

A set of resources $R = \{r_1,\ldots,r_m\}$ to be allocated by the agents, where each r_i represents resources in the form of *task(slot)*, where the *slot* is a parameter that needs to be assigned in order to execute the *task*. Then, it identifies each one of the solutions for a specific decision problem as an allocation *P* of elements (either agents or objects) to task-slots as follows:

$$P = \{t_1\left(s_1\right) \leftarrow e_1,\ldots,t_1\left(s_n\right) \leftarrow e_{n,},t_2\left(s_1\right) \leftarrow e_{n+1},\ldots,t_m(s_n) \leftarrow e_{n*m}\}$$

A set of utility functions $U = \{U^1,U^2,\ldots,U^q\}$. These utility functions will be used to evaluate the allocations from different points of view. Additionally, each agent will have a vector of utility weights $\vec{w}_u = w_{u1},\ldots,w_{uq}$ representing the importance given to each point of view in the multi-modal agent decision making.

- A collective utility functions *Cuf* = {*elitist*, *egalitarian*, *utilitarian*, *Nash*}, representing the social welfare of the simulated society, that is, the type of society where agents are located.

MADeM uses a market-based winner determination problem to merge the different preferences being collected according to the kind of agent or society simulated. The details of the whole decision making procedure are explained in the following subsection.

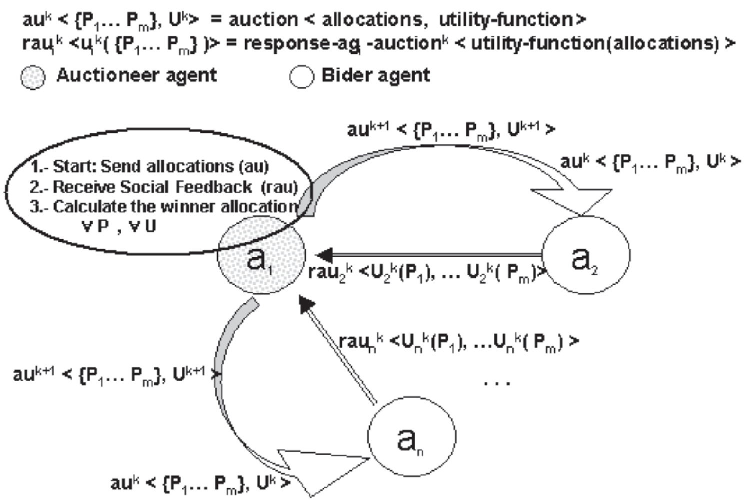
Decision Making Procedure

MADeM uses one-round sealed-bid combinatorial auctions to choose among different solutions to a decision problem. Auctioneer and bidder roles are not played by fixed agents throughout the simulation. Instead, every agent can dynamically play each role depending on his/her needs or interests. For example, an agent would be the auctioneer when he wanted to pass a task to another agent. On the other hand, agents receiving the auction would bid their utility values provided that they were interested in the task being auctioned. Thus, MADeM lies in between centralized and distributed market-based allocation.

An overview of the multi-modal decision making procedure followed by the agents is shown in Figure 2. This procedure is mainly based on the following steps:

Auctioning phase: This phase is carried out by a single agent (a_1) who wants to socially solve a decision problem (e.g. where to sit). This agent then constructs the set of allocations representing all the possible solutions for the problem ($P_1,P_2,\ldots,P_m$). These allocations have the form of task slots assignations such as $itAt(Obj_m) \leftarrow table_1$. Next, he auctions them to

Figure 2. MADeM Procedure



a particular group of agents, that we call the target agents. Each auction also includes a single type of utility function that the agent is interested in evaluating from the others ($au^k(P_1, P_2, \ldots, P_m, U^k)$). As complex decisions require taking into consideration more than one point of view, the auctioneer agent can start different auctions for the same set of allocations (au^1 through au^q).

Bidding phase: Since the auctioneer informs about both the task slot allocations and the utility functions being considered, bidders simply have to compute the requested utility functions and return the values corresponding to each auction back to the auctioneer ($rau_i^k = U_i^k(P_1), \ldots, U_i^k(P_m)$).

1. *Winner determination phase:* In this phase, the auctioneer selects a winner allocation for each launched auction. To do this, he uses a classical winner determination problem. Afterwards, he chooses one final winner allocation among these auction winners using a multi-modal decision making process. Thus, the final winner allocation will represent an acceptable decision for the society being simulated. The details of these calculations are fully described in the next section.

Winner Determination Problem

Once bidders have answered to an auction call (no answering means no preference, therefore, utility zero) the auctioneer agent has the utility values ($U_i(P_j)$) given by each bidder ($i \in A$) to every allocation being evaluated (P_j). Equation 1 groups these utility values in a set of vectors, one for each allocation.

$$\overrightarrow{U\left(P_j\right)} = U_1\left(P_j\right), \ldots, U_n\left(P_j\right) \qquad \forall j \in [1..m] \tag{1}$$

Remember that every agent had an associated vector of weights representing its attitude towards the other individuals ($\vec{w} = w_1, \ldots, w_n$). According to it, the auctioneer weighs the utility vectors in equation 1 doing a component by component multiplication with the attitude vector as shown in equation 2.

$$\overrightarrow{U_w\left(P_j\right)} = \overrightarrow{U\left(P_j\right)} * \vec{w} \qquad \forall j \in [1..m] \tag{2}$$

Attitude weights are used to model the social behavior of the auctioneer agent. For example, a whole range of behaviors between egoism and altruism can be modeled using the vector of equation 3, where p=0 represents the previous behavior, p=1 represents total altruism and p=0.5 represents an egalitarian behavior or indifference between oneself and the rest of the agents.

Egoism – Altruism:

$$\vec{w} = p,\ldots,p,1-p,p,\ldots,p \quad w_i = 1-p, w_{j\neq i} = p, i = Myself \tag{3}$$

It is also possible to model reciprocal attitudes by means of the vector $\vec{w}$. A simple example is shown in equation 4, where weights are based on the interchange of favors between agents.

$$\textit{Reciprocity: } w_i = \frac{Favors\_from(i)}{Favors\_to(i)} \tag{4}$$

In order to socially behave, the auctioneer agent attends to the social welfare value when selecting the winner allocation of each auction. Therefore, the winner determination problem chooses the allocation that maximizes the welfare of the society (equation 5).

$\forall k \in [1..q]$

Auction Winner:

$$P_{wk} \quad \leftrightarrow \quad sw\left(P_{wk}\right) = \max_{j\in[1..m]} sw\left(P_j\right) \tag{5}$$

To compute the social welfare of an allocation, the auctioneer uses Collective Utility Functions (CUFs) and the weighted utilities defined in equation 1 (as shown in equation 6). MADeM allows selecting among different CUFs when evaluating the social welfare of an allocation. At the moment, four CUFs have been integrated in MADeM, each one related to a kind of society: utilitarian, egalitarian, elitist and Nash.

$$sw\left(P\right) = cuf\left(\overrightarrow{U_w\left(P\right)}\right) \textit{ where}$$
$$\begin{cases} cuf_{utilitarian} = \sum u_w(i) \\ cuf_{egalitarian} = min\left\{u_w(i)\right\} \\ cuf_{elitist} = max\left\{u_w(i)\right\} \\ cuf_{Nash} = \prod u_w(i) \end{cases} \tag{6}$$

An agent can ask other agents about different points of view (e.g. efficiency, tiredness, etc). In order to do this, he performs several auctions with different types of utility functions (see parameter U^k in Figure 2). Once all these auctions have been resolved, the auctioneer has the winner allocation for each point of view and the social welfare obtained provided that allocation is adopted (see equation 7).

$$\left\{au^1\left(P_1,\ldots,P_m,U^1\right) \to \left(P_{w1}, sw\left(P_{w1}\right)\right),\ldots,au^q\left(P_1,\ldots,P_m,U^q\right) \to \left(P_{wq}, sw\left(P_{wq}\right)\right)\right\} \tag{7}$$

Lastly, the MADeM final winner allocation is that which maximizes the welfare of the society after having multiplied it by the corresponding utility weight $\vec{w}_u$, as shown in equation 8:

Final Winner:

$$P_w \quad \leftrightarrow \quad sw\left(P_w\right) = \max_{k\in[1..q]} w_u\left(k\right) * sw\left(P_{wk}\right) \tag{8}$$

J-MADeM: MADeM over Jason

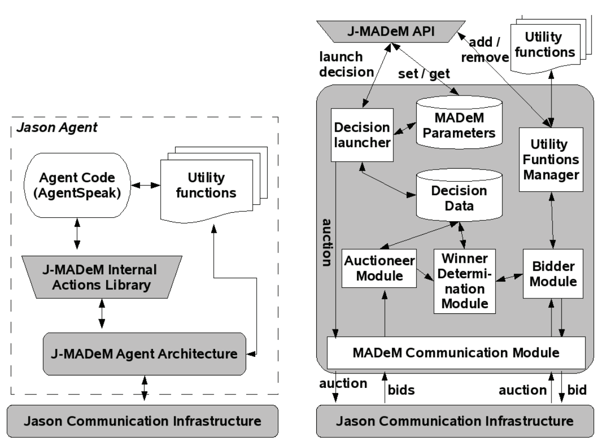
This section describes how the MADeM model can be used by an agent programming language to make socially acceptable decisions available to agents eventually part of an organization. Among several languages for agent programming, we have chosen the AgentSpeak language (Rao, 1996) and its open source interpreter Jason (Bordini *et al.*, 2007) to program this kind of social agents. This choice was made because the language is based on the well known BDI architecture and the interpreter can be easily customised to include the MADeM support. The coupling of MADeM with Jason is inspired in other extensions of Jason, in particular J-MOISE+ (Hübner *et al.*, 2007) and hence the name J-MADeM, as it joins Jason and MADeM.

The J-MADeM is built upon the *Jason Communication Infrastructure*, thus extending the communication level options available in Jason with a set of modules that provide agents with the built-in feature of performing MADeM decisions. Figure 3a illustrates how these components are integrated into Jason. The J-MADeM basically offers to the AgentSpeak programmer: (i) an agent architecture that Jason agents can use to carry out their own MADeM decisions, (ii) an interface to develop utility functions that can be used along with the MADeM model and (iii) a set of internal actions to manage the parameters of these kinds of decisions.

The *J-MADeM Agent Architecture* extends the *Jason Agent Architecture* in order to incorporate all the necessary modules that allow MADeM decisions to be automatically carried out. The main components of the *J-MADeM Agent Architecture* are shown in the figure 3b, where we can identify the following elements:

- **MADeM Parameters:** This data storage contains the MADeM context currently defined for the agent. Essentially, it stores the personal weights, the utility weights, the collective utility function and the bid timeout to be used in future MADeM decisions.
- **Decision Launcher:** This module starts the MADeM process for a particular decision. Firstly, it stores the MADeM context for this decision into the *Decision Data* storage, thus allowing other decisions to be concurrently performed with different MADeM parameters. Secondly, it auctions each of the allocations being considered as solutions to the target agents.
- **Decision Data:** This data storage holds all the information related to the MADeM decisions still in process. Therefore, it contains their MADeM context, their considered allocations and the preferences received for each of them.
- **MADeM Communication Module:** This module extends the Jason agent com-

Figure 3. (a) Overview of the J-MADeM architecture and (b) detailed view of the J-MADeM Agent Architecture.



munication module in order to deal with MADeM messages. When it receives a MADeM auction, it invokes the *Bidder Module* to get the agent's preferences over the considered allocations. On the other hand, when it receives a MADeM bid, it informs the *Auctioneer Module* about the received preferences.

- **Bidder Module:** This module manages the reception of a MADeM auction. It extracts the considered allocations and bids for them according to the agent's preferences. To express these preferences it relies on the utility values provided by the *Utility Functions Manager*.
- **Utility Functions Manager:** This component acts as an interface between the built-in MADeM mechanism and the user defined *Utility Functions*. Thus, it is in charge of locating and invoking them in order to calulate the agents' utilities for the set of considered allocations.
- **Auctioneer Module:** This module manages the reception of MADeM bids. It extracts the sender's preferences and stores them into the *Decision Data*. As soon as the preferences from all the target agents have been received, it calls the *Winner Determination Module* to solve the decision.
- **Winner Determination Module:** This module solves the MADeM winner determination problem using the information stored into the *Decision Data* for the decision being resolved (i.e. considered allocations, agents' preferences, personal weights, utility weights, social welfare…). Once resolved, it notifies the agent about the winner solution.

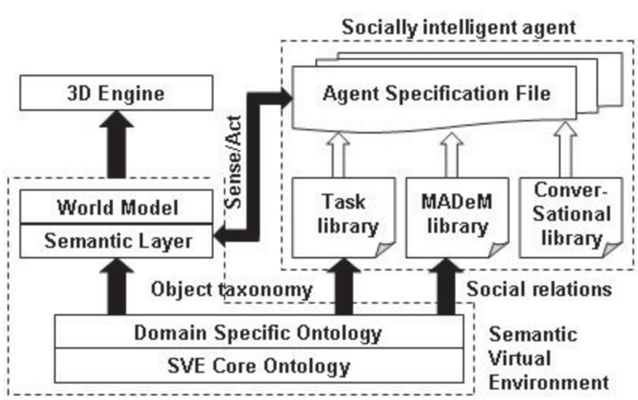
For a complete description of the utility function interface and the set of internal actions refer to the J-MADeM documentation available at the Jason website (Jason, 2009).

APPLICATION EXAMPLE

In this section, we show how we have integrated MADeM into a multi-agent framework oriented to simulate socially intelligent characters in 3D virtual environments. This framework is developed over Jason (2009), so that the J-MADeM library is used to provide BDI agents with the ability to perform MADeM decisions. Figure 4 depicts the architecture of the system, which can be basically divided into two parts:

- The *Semantic Virtual Environment* uses ontologies to define the world knowledge

Figure 4. Multi-agent simulation framework



base (i.e. the object taxonomy and the object interrelations) as well as the set of all possible relations among the agents within an artificial society. The environment is handled by the *Semantic Layer*, which acts as an interface between the agent and the world, thus sensing and executing the actions requested by the agents. The animation system – virtual characters, motion tables, etc. – is located at the *3D Engine* that can extract graphical information from the *World Model* database in order to perform visualization.

- *Socially intelligent agents* receive sensorial information from the *Semantic Layer* and compute the appropriate sequence of actions in order to achieve their goals. The agent's finite state machine is defined in the *Agent Specification File*. It calls the following libraries to enrich agent behavior: the *Task Library,* that contains the operators that sequence the actions needed to animate a task; the *J-MADeM Library,* that provides the agents with the mechanisms to make social decisions; and finally, the *Conversational Library,* that contains the set of plans that handle the animation of the interactions between characters (e.g. ask someone a favor, planned meetings, chats between friends...).

In order to test MADeM, we have created a virtual university bar where waiters take orders placed by customers. The typical objects in a bar, such as a juice machine, behave like dispensers that have an associated time of use to supply their products (e.g. 2 minutes to make an orange juice) and they can only be occupied by one agent at a time. Therefore, waiters should coordinate to avoid conflicts. Additionally, agents can be socially linked using the concepts defined in the ontology (Grimaldo *et al.*, 2008b). For example, waiters and customers create social relationships with their friends and this social network is used when deciding whether to do favors, to promote social meetings, etc.

Waiters serve orders basically in two steps: first, using the corresponding dispenser (e.g. the grill to produce a sandwich); and second, giving the product to the customer. For each task, a waiter evaluates whether to carry out the task against the chance to pass it to another waiter and perform his next task. That is, tasks are always auctioned using MADeM before their execution in order to find good social allocations. When calling MADeM, waiters take into account three points of view (i.e. utility functions): *performance*, *chatting* and *tiredness*. First, the performance utility function aims at maximizing the number of tasks being performed at the same time and represents the waiters' willingness to serve orders as fast as possible. Second, social behaviors defined for a waiter are oriented to animate chats among its friends at work. Therefore, the social utility function evaluates social interest as the chance to meet a friend in the near future, thus performing a planned meeting. Third, the tiredness utility function implements the basic principle of minimum energy, widely applied by humans at work. Finally, the type of society being simulated for waiters is elitist. That is, waiters will choose those allocations that maximize the utility functions previously defined.

On the other hand, customers place orders and consume them when served. Now, we are interested in animating interactions between customers that are consuming with their friends. Therefore, customers call MADeM to solve the problem of *where to sit*. In this case, the task slot being auctioned is the place where to sit and the candidates being evaluated are all the tables in the environment as well as the bar. Customers consider two points of view when calling MADeM: *sociability* and *laziness*. The social utility function defined for customers assigns a maximum value to a table provided that there is a friend sitting on it. To consume standing up at the bar is not considered of social interest at all, hence, its utility value is defined as zero. The laziness utility function evaluates each table according to their distance to the customer and, opposite to sociability, standing at the bar is now considered the best option. The type of society being simulated for customers is utilitarian, therefore, customers will choose those allocations that maximize the addition of the utility values previously defined. For a full description of the utility functions used by waiters and customers see (Grimaldo *et al.*, 2008a).

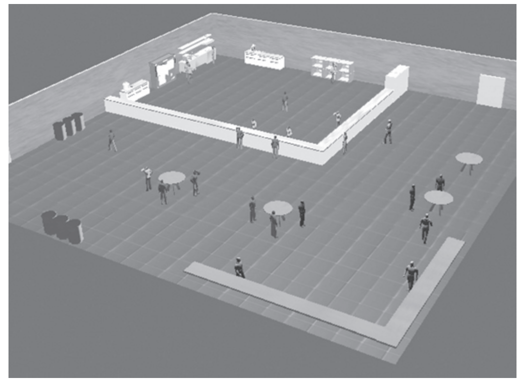
Results

In order to verify the social outcomes obtained with MADeM agents, we have simulated different types of waiters serving customers (see Figure 5 for a snapshot of the running 3D virtual environment). The results shown in this section correspond to simulations where 10 waiters attend 100 customers.

As we have previously mentioned, we have modeled an elitist society of waiters within which agents consider three points of view (i.e. performance, sociability and tiredness), each of them represented by its own utility function. In this context, utility weights can be adjusted to create different types of social waiters. For example, a *coordinated* waiter could be an agent that chooses its decisions following performance 75% of the times and following sociability or tiredness in the rest of the situations. The vector of utility weights for a *coordinated* waiter would then be $\overrightarrow{w_u} = 0.75, 0.125, 0.125$, where each component represents the importance given to each utility function being evaluated. Similarly, we have defined *social* waiters as agents with the following vector of utility weights $\overrightarrow{w_u} = 0.125, 0.75, 0.125$ and *egalitarian* waiters as agents with $\overrightarrow{w_u} = 0.125, 0.125, 0.75$.

Table 1 summarizes some performance results obtained with *coordinated*, *social* and *egalitarian*

Figure 5. 3D virtual university bar environment



waiters against *self-interested* waiters with no social mechanism included. Firstly, column *Tasks/s* results from dividing the total number of actions performed by all the waiters by the amount of time needed to serve all the customers. Secondly, column $\overline{NChats}$ is the average number of chat actions carried out by each waiter. Thirdly, column σ_{NTasks} shows the standard deviation in the number of actions performed by each waiter. According to these results, *coordinated* waiters perform better (see higher values in column *Tasks/s*) since the majority of conflicts caused by the use of the same dispenser (e.g. the coffee machine) are resolved with specialization, that is, by passing the task to another waiter already using the dispenser. On the other hand, *social* waiters take more time to serve customers but animate a greater number of chats among friends (compare the average number of chats being animated in column $\overline{NChats}$). *Egalitarian* waiters look at the tiredness utility function and try to allocate the task to the least tired waiter, therefore, the standard deviation in the number of tasks performed by each agent tends to zero (see column σ_{NTasks}). Finally, *self-interested* waiters demonstrate to perform worse than any kind of social waiter. As these agents are unable to do task passing nor

Table 1. Performance results of different types of waiters

Agent	*Tasks/s*	$\overline{NChats}$	σ_{NTasks}
Coordinated	0,91	5	6.73
Social	0.65	29.4	4.37
Egalitarian	0.62	6.6	2.74
Self-interested	0.17	-	-

chatting, columns σ_{NTasks} and $\overline{NChats}$ are not considered.

Besides the possibility to define the importance of each point of view through the vector of utility weights $\overrightarrow{w_u}$, MADeM allows for the definition of a vector of personal weights $\vec{w}$ that models the attitude of an agent towards the other individuals. Table 2 shows the task passing results obtained for the defined waiters using the models of attitude considered previously: indifference, reciprocity, altruism and egoism. In this table, column $\overline{Favors}$ refers to the average number of favors (tasks) exchanged between the agents and column σ_{Favors} refers to the standard deviation in the number of favors. Agents using *indifference* do not apply any modification over the utilities received. Therefore, we consider the results of this attitude as the base values to compare with for each type of waiter. *Reciprocity* weights utilities attending to the ratio of favors already done between the agents. This attitude produces equilibrium in the number of favors exchanged as it can be seen in column σ_{Favors}. *Altruism* has been implemented in such a way that the weight given to oneself utilities is 0.25 whereas the weights for the rest of the agents are 0.75. As expected, altruist agents do more favors, since the importance given to the other's opinions is three times the importance given to their own opinion (see high values for the average number of favors exchanged $\overline{Favors}$). On the other hand, *egoism* weights are 0.75 to oneself and 0.25 to the others, thus, agents rarely do favors (see low values in column $\overline{Favors}$).

Agent's preferences can sometimes go against personal attitudes. For example, whereas *reciproc-*

Table 2. Task passing results for different personal weights

	Coordinated		Social		Egalitarian	
Attitude	σ_{Favors}	$\overline{Favors}$	σ_{Favors}	$\overline{Favors}$	σ_{Favors}	$\overline{Favors}$
Indifference	7.57	6.9	3.52	8.7	7.58	13.6
Reciprocity	1.15	8.8	1.76	7.8	2.4	15.5
Altruism	5.94	17	6.66	12.7	4.44	17.9
Egoism	1.41	0.7	0.81	0.4	0.47	0.1

ity tries to balance the number of favors, tiredness tends to assign tasks to the least tired waiter (see the greater σ_{Favors} for egalitarian waiters). Another example is *egoism* applied to *egalitarian* waiters, in this case no task at all is passed among the agents ($\overline{Favors} = 0.1$). However, agent's preferences can also empower personal attitudes. For instance, *altruism* applied to *coordinated* waiters produces a high level of specialization. This type of agents produces big values for σ_{Favors} as the agents already using a dispenser (e.g. a juice machine) keep on getting products from the dispenser following both an altruist and a coordinated behavior that reduces collisions for the use of an exclusive resource. Despite this issue, personal weights have demonstrated to produce similar effects on the agents regardless of the kind of waiter being considered (i.e. *coordinated*, *social* or *egalitarian*).

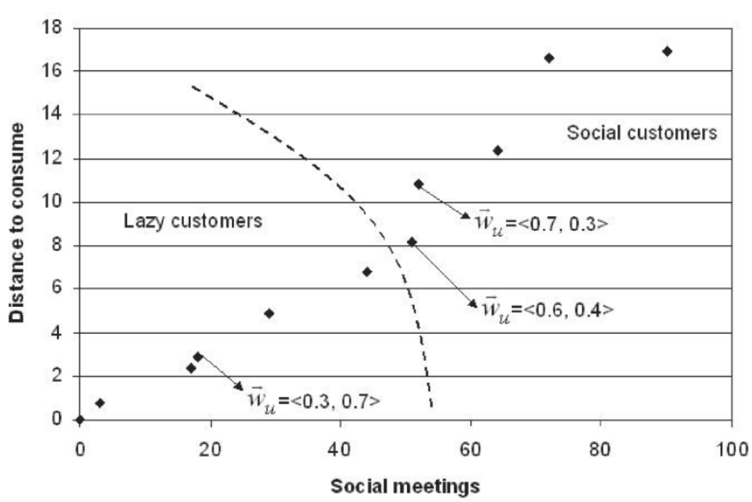
Unlike waiters, customers make decisions within a utilitarian society where they consider two points of view: *sociability* and *laziness*. Figure 6 shows the behavior obtained with different types of customers. We compare two metrics: the mean number of social meetings performed among the customers and mean distance covered to consume. Lazy customers, with low utility weights for sociability, most of the time choose to consume at the bar or to sit at a nearby table (see point $\overrightarrow{w_u} = 0.3,0.7$). Therefore, the mean distance to the consuming place is short but only a few social meetings are animated. On the other hand, social customers, with higher utility weights for sociability, perform more social meetings but they also need to move longer distances to find their friends.

Points $\overrightarrow{w_u} = 0.6,0.4$ and $\overrightarrow{w_u} = 0.7,0.3$ in figure 6 correspond to some examples of social customers.

FUTURE RESEARCH DIRECTIONS

3D virtual worlds have significantly evolved since their beginning. However, users often point to their lack of immersion due to the great number of uninhabited scenes or to the elementary interactions between avatars and autonomous humanoids. Currently, synthetic actors still need to incorporate a broad range of social techniques to enhance their behavioral animation, which will

Figure 6. Lazy vs. social customers



finally improve the mental immersion of the user within 3D virtual worlds.

Therefore, future work in this research area must cover the integration of new social techniques in the agent decision making, such as the one presented in this chapter. This goal must lead the researchers to look for the synergy among computer graphics, artificial intelligence and social sciences. This sort of socially intelligent actors will be useful to populate professional environments such as a 3D e-government office (where autonomous humans can socially follow the protocols between the administration and the citizens) as well as entertainment scenes such as the virtual bar showed in this chapter.

CONCLUSION

This chapter has presented a new social decision-making technique to provide 3D virtual agents with consistent social behaviors suitable to be animated. Firstly, we have analyzed the literature and evolution of social decision-making in multi-agent domains. Then, we focus on MADeM and its integration into Jason (J-MADeM), a new open source library oriented to create different types of simulated societies. The main feature of J-MADeM agents is that they are able to merge several points of view received from other agents. This social feedback is modeled via utility functions that express the preferences of each agent for every solution being considered. The application example presented aims at incorporating human style social reasoning for character animation. This way, we evaluate the social outcomes provided by the J-MADeM. The results obtained for the virtual university bar show how two groups of socially intelligent agents can consider different points of view in their decision making: first, a team of waiters using performance, sociability and tiredness; and second, a model of customer that evaluates sociability and laziness. Furthermore, this example allows the agents to manage several personal and collective utility functions as well as a set of weights (personal and utility-based) in order to perform elaborated social simulations.

ACKNOWLEDGMENT

This work has been jointly supported by the Spanish MEC and European Commission FEDER funds under grants Consolider Ingenio-2010 CSD2006-00046 and TIN2009-14475-C04-04.

REFERENCES

Badler, N., Bindiganavale, R., Bourne, J., Palmer, M., Shi, J., & Schuler, W. (2000). *A parametrized action representation for virtual human agents* (pp. 256–284). Cambridge, MA: MIT press.

Badler, N., Philips, C., & Webber, B. (1993). *Simulating Humans: Computer Graphics, Animation and Control*. Oxford, UK: Oxford University Press.

Blizzard Entertainment. *World of warcraft.* (2009). Retrieved February 2009 from http://www.worldofwarcraft.com.

Bordini, R. H., Hübner, J. F., & Wooldrige, M. (2007). *Programming Multi-Agent Systems in AgentSpeak using Jason*. Chichester, UK: Wiley. doi:10.1002/9780470061848

Chevaleyre, Y., Dunne, P. E., Endriss, U., Lang, J., Lemaitre, M., & Maudet, N. (2006). Issues in multiagent resource allocation. *Informatica*, *30*, 3–31.

Conte, R., & Castelfranchi, C. (1995). *Cognitive and Social Action*. London: UCL Press.

Decker, K. S. (1998). Task environment centered simulation. In *Simulating Organizations: Computational Models of Institutions and Groups* (pp. 105–128). Menlo Park, CA: AAAI Press / MIT Press.

Delgado-Mata, C., & Aylett, R. (2004). *Emotion and action selection: Regulating the collective behaviour of agents in virtual environments* (pp. 1304–1305). AAMAS.

Dignum, V., & Dignum, F. (2001). Modelling agent societies: Co-ordination frameworks and institutions. In P. Brazdil & A. Jorge, (Eds.), *Procs. of the 10th Portuguese Conference on Artficial Intelligence (EPIA'01)*, (LNAI 2258, pp. 191-204). Berlin: Springer.

Falcone, R., Pezzulo, G., Castelfranchi, C., & Calvi, G. (2004). Why a cognitive trustier performs better: Simulating trust-based contract nets. In *Proc. of AAMAS'04: Autonomous Agents and Multi-Agent Systems*, (pp. 1392–1393). New York: ACM.

Farenc, N., Boulic, R., & Thalmann, D. (1999). An informed environment dedicated to the simulation of virtual humans in urban context. In P. Brunet & R. Scopigno, (Eds.), *Proc. of EUROGRAPHICS''99, 18*(3), 309–318. London: The Eurographics Association and Blackwell Publishers.

Ferber, J., & Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agents systems. In *Proc. of the 3rd International Conference on Multi-Agent Systems (ICMAS'98)*, (pp. 128–135). Washington, DC: IEEE Press.

Fox, M. S., Barbuceanu, M., Gruninger, M., & Lon, J. (1998). An organizational ontology for enterprise modelling. In *Simulating Organizations: Computational Models of Institutions and Groups* (pp. 131–152). Menlo Park, CA: AAAI Press / MIT Press.

Funge, J., Tu, X., & Terzopoulos, D. (1999). Cognitive modeling: knowledge, reasoning and planning for intelligent characters. In *Proc. of SIGGRAPH'99*, (pp. 29–38). New York: ACM.

Grimaldo, F., Lozano, M., & Barber, F. (2008). MADeM: a multi-modal decision making for social MAS. In *Proc. of AAMAS'08: Autonomous Agents and Multi-Agent Systems*, (pp. 183–190). New York: ACM.

Grimaldo, F., Lozano, M., Barber, F., & Vigueras, G. (2008). Simulating socially intelligent agents in semantic virtual environments. *The Knowledge Engineering Review*, *23*(4), 369–388. doi:10.1017/S026988890800009X

Helbing, D., & Molnar, P. (1995). Social force model for pedestrian dynamics. *Physical Review E: Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, *51*, 4282. doi:10.1103/PhysRevE.51.4282

Hexmoor, H. (2001). From inter-agents to groups. In *Proc. of ISAI'01: International Symposium on Artificial Intelligence*.

Hogg, L. M., & Jennings, N. (2001). Socially intelligent reasoning for autonomous agents. *IEEE Transactions on Systems, Man, and Cybernetics*, *31*(5), 381–393. doi:10.1109/3468.952713

Hübner, J. F., Sichman, J., & Boissier, O. (2002). A model for the structural, functional, and deontic specification of organizations in multiagent systems. In G. Bittencourt & G. L. Ramalho, (Eds.), *Procs. of the 16th Brazilian Symposium on Artifical Intelligence (SBIA'02)*, (LNAI Vol. 2507, pp. 118-128). Berlin: Springer-Verlag.

Hübner, J. F., Sichman, J. S., & Boissier, O. (2007). Developing organised multi-agent systems using the Moise+ model: Programming issues at the system and agent levels. *International Journal of Agent-Oriented Software Engineering*, *1*(3/4), 370–395. doi:10.1504/IJAOSE.2007.016266

Jason. (2009). Retrieved 2009 from http://jason.sourceforge.net/.

Lim, M. Y., & Aylett, R. (2009). An Emergent Emotion Model for An Affective Mobile Guide with Attitude. *Applied Artificial Intelligence Journal*, *23*, 835–854. doi:10.1080/08839510903246518

Linden Lab. (2009). *SecondLife®*. Retrieved February 2009 from http://secondlife.com/.

Pelechano, N., Allbeck, J., & Badler, N. I. (2008). *Virtual Crowds: Methods, Simulation, and Control*. San Francisco: Morgan & Claypool Publishers.

Prada, R., & Paiva, A. (2005). Believable groups of synthetic characters. In *Proc. of AAMAS '05: Autonomous Agents and Multi-Agent Systems*, (pp. 37–43). New York: ACM.

Prietula, M., Carley, K., & Gasser, L. (1998). *Simulating Organizations: Computational Models of Institutions and Groups*. Menlo Park, CA: AAAI Press / MIT press.

Rao, A. S. (1996). AgentSpeak(L): BDI agents speak out in a logical computable language. In S. Verlag, (Ed.), *Proc. of MAAMAW'96*, (LNAI 1038, pp 42–55).

Rao, A. S., & Georgeff, M. P. (1991). Modeling rational agents within a BDI-architecture. In *Proc. of KR'91: The 2nd International Conference on Principles of Knowledge Representation and Reasoning*, (pp. 473–484). San Mateo, CA: Morgan Kaufmann publishers Inc.

Raupp, S., & Thalmann, D. (2001). Hierarchical model for real time simulation of virtual human crowds. *IEEE Transactions on Visualization and Computer Graphics*, *7*(2), 152–164. doi:10.1109/2945.928167

Rehm, M., & Rosina, P. (2008). SecondLife® as an Evaluation Platform for Multiagent Systems Featuring Social Interactions. In *Proc. of AAMAS'08: Autonomous Agents and Multi-Agent Systems*, (pp 1663–1664). New York: ACM.

Reilly, W. S. N. (1996). *Believable Social and Emotional Agents*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA.

Reynolds, C. (1987). Flocks, herds and schools: A distributed behavioral model. In *Proc. of SIGGRAPH'87*, (pp 25–34). New York: ACM.

Sichman, J., & Demazeau, Y. (2001). On social reasoning in multi-agent systems. *Revista Ibero-Americana de Inteligencia Artificial*, *13*, 68–84.

Stephenson, N. (1992). *Snow crash*. New York: Bantam Books.

Tambe, M. (1997). Towards flexible teamwork. *Journal of Artificial Intelligence Research*, *7*, 83–124.

Thalmann, D., & Monzani, J. (2002). Behavioural animation of virtual humans: What kind of law and rules? In *Proceedings of Computer Animation* (pp. 154–163). Washington, DC: IEEE Computer Society Press.

Tu, X., & Terzopoulos, D. (1994). Artificial fishes: physics, locomotion, perception, behavior. In *Proc. of SIGGRAPH'94*, (pp. 43–50). New York: ACM.

van Kokswijk, J. (2007). *Digital Ego: Social and Legal Aspects of Virtual Identity*.

Williams, D., Ducheneaut, N., Xiong, L., Zhang, Y., Yee, N., & Nickell, E. (2006). From Tree House to Barracks: The Social Life of Guilds in World of Warcraft. *Games and Culture*, *1*, 338–361. doi:10.1177/1555412006292616

Wooldridge, M. J., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, *10*(2), 115–152. doi:10.1017/S0269888900008122

Yee, N., Bailenson, J. N., Urbanek, M., Chang, F., & Merget, D. (2007). The Unbearable Likeness of Being Digital: The Persistence of Nonverbal Social Norms in Online Virtual Environments. *Cyberpsychology & Behavior*, *10*(1), 115–121. doi:10.1089/cpb.2006.9984

ADDITIONAL READING

Bartle, R. A. (2003). *Designing Virtual Worlds. New Riders.*

Moulin, H. (1991). *Axioms of cooperative decision making*. Cambridge University Press.

Osborne, M. J. (2004). *An Introduction to Game Theory.* Oxford University Press.