Coordination and Sociability for Intelligent Virtual Agents

Francisco Grimaldo, Miguel Lozano, and Fernando Barber

Computer Science Department, University of Valencia,
Dr. Moliner 50, (Burjassot) Valencia, Spain
{francisco.grimaldo,miguel.lozano,fernando.barber}@uv.es

Abstract. This paper presents a multi-agent framework designed to simulate synthetic humans that properly balance task oriented and social behaviors. The work presented in this paper focuses on the social library integrated in BDI agents to provide socially acceptable decisions. We propose the use of ontologies to define the social relations within an artificial society and the use of a market based mechanism to reach sociability by means of task exchanges. The social model balances rationality, to control the global coordination of the group, and sociability, to simulate relations (e.g. friendliness) and reciprocity among agents. The multi-agent framework has been tested successfully in dynamic environments while simulating a virtual bar, where groups of waiters and customers can interact and finally display complex social behaviors (e.g. task passing, reciprocity, planned meetings).

1 Introduction

Multi-agent systems are sometimes referred to as societies of agents and provide an elegant and formal framework to animate synthetic humans. When designing such agents, the main concern has normally been with the decision-making mechanism, as it is the responsible for the actions that will be finally animated. Virtual actors normally operate in dynamic resource bounded contexts; thus, multi-agent simulations require group coordination, as self-interested agents easily come into conflicts due to the competition for the use of shared resources (i.e. objects in a virtual environment). These obstructions produce low quality animations where characters do not act realistically. Moreover, virtual humans represent roles in the scenario (e.g. a virtual guide, a waiter, a customer, etc.) and the social network formed by the relations among the members of the society should also be considered when animating their behaviors.

This paper presents a multi-agent simulation framework to produce good quality animations where the behavior of socially intelligent agents better imitates that of real humans. We aim at incorporating human style social reasoning in virtual characters. Therefore, we have developed a market based social model [1] which coordinates the activities of groups of virtual characters and incorporates social actions in the agent decision-making. Our approach is inspired in reciprocal task exchanges between agents [2] and uses ontologies to define the

J.S. Sichman et al. (Eds.): COIN 2007 Workshops, LNAI 4870, pp. 58–70, 2008.

social relations within an artificial society. According with the main parameter of the model, that is sociability, the agents can balance their task-oriented behaviors (e.g. a virtual waiter should serve customers) and their social skills (e.g. negotiate with other waiters to gain access to a resource, assume external actions/favors, or simple chats).

The structure of the paper is as follows: in section 2 we describe briefly some previous literature on the field. In section 3 we present the multi-agent simulation framework and the main components of the social model. Section 4 describes an illustrative example modeled to test our framework. Lastly, section 5 summarizes the first results extracted and analyzes them.

2 Related Work

Many interactive games and virtual communities put human users together with synthetic characters. In this context, some research has been done on the believability issues of virtual actors, usually centred on the interactions either between a human user and a single character [3] or among the synthetic characters themselves [4]. These interactive scenarios often present tasks to the participants that must be solved collaboratively [5]. Therefore, behavioral animation has broadly been tackled from the field of coordinated multi-agent systems (e.g. Generalized Partial Global Planning (GPGP) [6], the TAEMS framework [7] or the RETSINA system [8]). Moreover, task coordination has been applied to HSP-based (Heuristic Search Planning) virtual humans in [9] and [10] to adapt better to the dynamism of shared environments.

Social reasoning has also been extensively studied in multi-agent systems in order to incorporate social actions to cognitive agents [11]. As a result of these works, agent interaction models have evolved to social networks that try to imitate the social structures found in reality [12]. Social dependence networks in [13] allow agents to cooperate or to perform social exchanges attending to their dependence relations (i.e. social dependence and social power). Trust networks in [14] are used to define better delegation strategies by means of a contract net protocol and fuzzy cognitive representations of the other agents as well as of the dynamic environment. In preference networks, such as the one presented in this paper, agents express their preferences using utility functions and their attitude towards another agent is represented by the differential utilitarian importance they place on that agent's utility.

Semantic information can be of great value to the agents inhabiting a virtual world. As demonstrated in [15], the use of semantics associated to objects can enhance the interaction of virtual humans in complex environments. Environment-based approaches are also emerging to provide semantic interoperability among intelligent agents through the use of coordination artifacts [16]. Furthermore, ontologies are very useful to model the social relations between the agents involved in graphical and interactive simulations [17]. In MOISE+ [18], ontological concepts join roles with plans in a coherent organizational specification. Another example can be found in [19] where a functional ontology for reputation is proposed.

Although the results obtained by the previous approaches show realistic simulations for many task-oriented behaviors, synthetic characters should also display pure social behaviors (e.g. interchanging information with their partners or grouping and chatting with their friends). MAS-SOC [20] aims at creating a platform for multi-agent based social simulations with BDI agents, which is also our purpose. In this context, work is ongoing in order to incorporate social-reasoning mechanisms based on exchange values [21]. The multi-agent framework presented here is oriented to simulate socially intelligent agents able to balance their rationality and sociability, a key point to finally display high quality behavioral animations.

3 Multi-agent Simulation Framework

The multi-agent simulation framework presented in figure 1 has been developed over Jason [22], which allows the definition of BDI agents using an extended version of AgentSpeak(L). The animation system (virtual characters, motion tables, etc) is located at the 3D engine, which can run separately. The environment is handled by the *Semantic Layer*, which acts as an interface between the agent and the world. It is in charge of perceiving the state of the world and executing the actions requested by the agents, while ensuring the consistency of the *World Model*. Ontologies define the world knowledge base using two levels of representation: the *SVE Core Ontology* is a unique base ontology suitable for all virtual environments and it is extended by different *Domain Specific Ontologies* in order to model application-specific knowledge.[1]

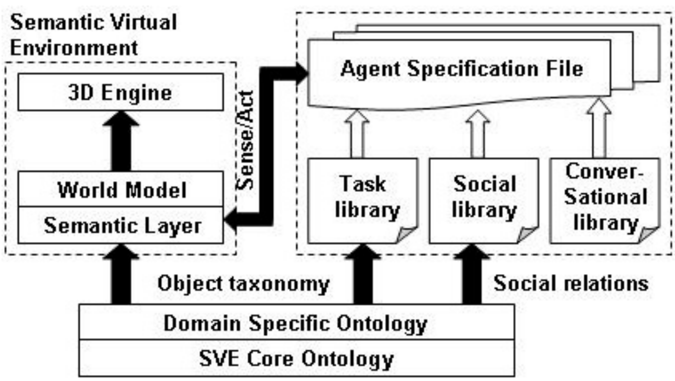

Fig. 1. Multi-agent simulation framework

The agent decision-making is defined in the *Agent Specification File*. This file contains the initial beliefs as well as the set of plans that make up the agent's finite state machine. The *Task library* contains the set of plans that sequence

[1] See [15] for details on ontologies and their use to enhance agent-object interaction.

the actions needed to animate a task. For instance, a virtual waiter serving a coffee will go to the coffee machine to get the coffee and will give it to the customer afterwards. Here, modularity is guaranteed since the *Task library* can be changed depending on the environment and the roles being simulated. As stated above, only rational behaviors are not enough to simulate agent societies. Therefore, we have extended the ontologies to define the possible social relations among the agents of a society and we have included a *Social library* to manage different types of situations. This library is based on an auction model and uses social welfare concepts to avoid conflicts and allow the agents to behave in a coordinated way. The *Social library* also incorporates a reciprocity mechanism to promote egalitarian social interactions. Finally, the *Conversational library* 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...).

3.1 Social Ontology

The set of possible social relations among the agents within an artificial society can be ontologically represented in the form of interrelations between classes of agents. Figure 2 shows the extensions made to the object ontology previously presented in [15] in order to hold agent relations. We distinguish two basic levels of social relations: the level of individuals (i.e. *agentSocialRelations*) and the institutional level (i.e. *groupSocialRelations*). When one agent is related with another single agent, an *agentSocialRelation* will link them. Different application domains can need specific relations; thus, Domain Specific Ontologies are used to inherit particular relations from the core ontology. For instance, the property *workFriend* is used by the waiters in the virtual bar presented in section 4 to model the characteristic of being a friend of a workmate. Other examples of individual relations are family relations such as to be parent of or to be married with another agent. In this case, there is not only semantic but also structural difference, since *parent* is a unidirectional relation whereas *marriedWith* is bidirectional.

On the other hand, *groupSocialRelations* can be used to represent an agent belonging to a group. The social network created by this type of relation can be explored to get the rest of the agents of the same group, thus modeling a one-to-many relation. The *Group* class is an abstraction of any kind of aggregation. Therefore, we can model from physical groups such as the players of a football team to more sophisticated mental aggregations such as individuals of a certain social class or people of the same religious ideology. Although not considered in this paper, many-to-many relations between groups could also be created using this ontological approach. The dynamics of how these relations are created, modified and terminated falls out of the scope of this paper. Thus, at the moment relations are set off-line and do not change during the simulation.

3.2 Social Library

The simulation of worlds inhabited by interactive virtual actors normally involves facing a set of problems related to the use of shared limited resources and the

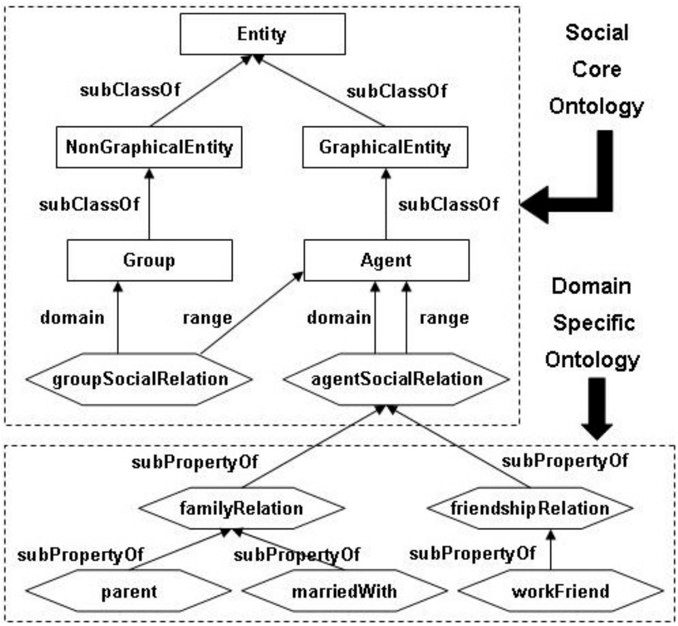

Fig. 2. Social Ontology

need to animate pure social behaviors. Both types of problems are managed by the *Social library* by using a Multi-agent Resource Allocation approach [1]. This library allows any agent to auction tasks in order to reallocate them so that the global social welfare can be increased. Tasks are exchanged between agents using a first-price sealed-bid (FPSB) auction model where the agents express their preferences using *performance and social utility functions.*

The performance utility function $U^i_{perf}(\langle i \leftarrow t \rangle)$ of a bidder agent i reflects the efficiency achieved when the task t is allocated to the agent i ($\langle i \leftarrow t \rangle$). There can be many reasons for an agent to be more efficient: it may perform the task faster than others because of his know-how or it may be using a resource that allows several tasks to be performed simultaneously (e.g. a coffee machine in a virtual bar can be used by a waiter to make more than one coffee at the same time). The utility function has to favor the performance of the agents, but high performances can also be unrealistic for the animation of artificial human societies. For example, if all agents work as much as they can, they will display unethical or robotic behaviors. Furthermore, agents should also show pure social behaviors to animate the normal relations between the members of a society.

Whereas the performance utility function modelled the interest of an agent to exchange a task from an efficiency point of view, we introduce two additional social utilities to represent the social interest in exchanging a task. The aim of social utilities is to promote task allocations that lead the agents to perform social interactions with other agents (e.g. planned meetings with their friends). Therefore, these functions take into account the social relations established between the agents and defined in the ontology to compute the value that expresses their social preferences. Negotiation of long sequences of actions is not very interesting for interactive characters, as plans are likely to be thwarted due to the dynamism of the environment and to other unpredictable events. Thus, we define the following social utility functions:

– Internal social utility ($U^i_{int}(\langle i \leftarrow t, j \leftarrow t_{next}\rangle)$): is the utility that a bidder agent i assigns to a situation where i commits to do the auctioned task t so that the auctioneer agent j can execute his next task t_{next}.
– External social utility ($U^i_{ext}(\langle j \leftarrow t\rangle)$): is the utility that a bidder agent i assigns to a situation where the auctioneer agent j executes the auctioned task t while i continues with his current action.

The winner determination problem has two possible candidates coming from performance and sociability. In equation 1 the welfare of a society is related to performance, hence, the winner of an auction will be the agent that bid the maximum performance utility. On the other hand, equation 2 defines the social winner based on the maximum social utility received to pass the task to a bidder (see $U^*_{int}(t)$ in equation 3) and the maximum social utility given by all bidders to the situation where the task is not exchanged but performed by the auctioneer j (see $U^*_{ext}(t)$ in equation 4). To balance task exchange, social utilities are weighted with a reciprocity matrix (see equations 3 and 4). We define the reciprocity factor w_{ij} for two agents i and j, as the ratio between the number of favors (i.e.tasks) that j has made to i (see equation 5).

$$winner_{perf}(t) = \left\{ k\epsilon Agents | U^i_{perf}(t) = \max_{i\epsilon Agents} \{U^i_{perf}(\langle i \leftarrow t\rangle)\} \right. \tag{1}$$

$$winner_{soc}(t) = \begin{cases} j \; U^*_{ext}(t) >= U^*_{int}(t) \\ i \; U^*_{ext}(t) < U^*_{int}(t) \wedge U^i_{int}(t) = U^*_{int}(t) \end{cases} \tag{2}$$

$$U^*_{int}(t) = \max_{i\epsilon Agents} \{U^i_{int}(\langle i \leftarrow t, j \leftarrow t_{next}\rangle) * w_{ij}\} \tag{3}$$

$$U^*_{ext}(t) = \max_{i\epsilon Agents} \{U^i_{ext}(\langle j \leftarrow t\rangle) * w_{ji}\} \tag{4}$$

$$w_{ij} = \frac{Favours_{ji}}{Favours_{ij}} \tag{5}$$

At this point, agents can decide whether to adopt this kind of social allocations or to be only rational as explained previously. They choose between them in accordance with their *Sociability* factor, which is the probability to select the social winner instead of the rational winner. *Sociability* can be adjusted in the range [0,1] to model intermediate behaviors between efficiency and total reciprocity. This can provide great flexibility when animating characters, since *Sociability* can be dynamically changed thus producing different behaviors depending on the world state.

4 Application Example

In order to test the presented social multi-agent framework, we have created a virtual university bar where waiters take orders placed by customers (see figure 3a). The typical locations in a bar (e.g. a juice machine) behave like resources 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. Agents can be socially linked using the concepts defined in the *Social Ontology*. According to them, all waiters are related through a *groupSocialRelation* to *Waiters*, a group representing their role (see figure 3b). Moreover, they can be individually related with other waiters through *workFriend*. This relation semantically means that the agents are friends at work and, in this application, it has been modeled as bidirectional but not transitive. For example, in figure 3b, Albert is friend of Dough and John but these later ones are not friends of each other. Moreover, we have also specified three possible groups of customers: teachers, undergraduates and graduates. The social network specified by them is used to promote social meetings among customers in the university bar.

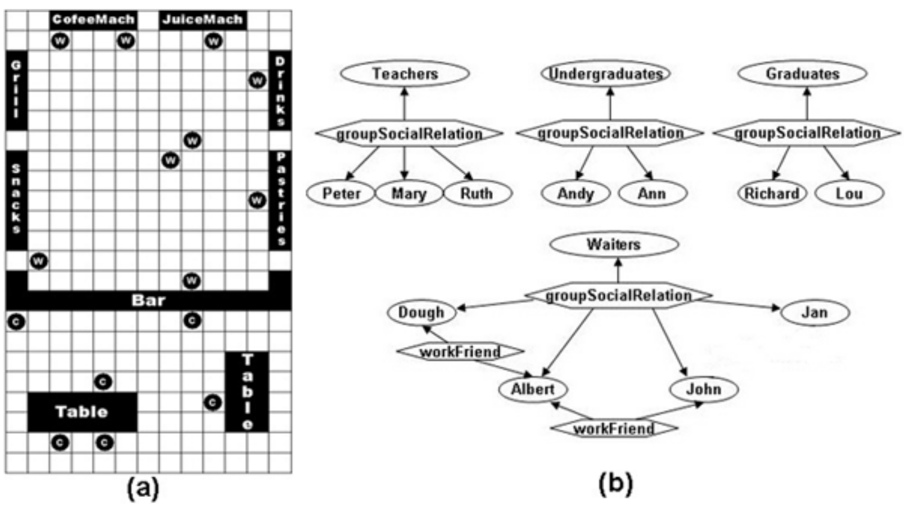

Fig. 3. (a) Virtual university bar environment, (b) Social relations between agents

The waiters are governed by the finite state machine[2] shown in figure 4a, where orders are served basically in two steps: first, using the corresponding resource (e.g. the grill to produce a sandwich) and second, giving the product to the customer. Tasks are always auctioned before their execution in order to find good social allocations. Equations 6 and 7 define the utility values returned by the performance utility function for these tasks. This function aims at maximizing

[2] Specified by means of plans in Jason's extended version of AgentSpeak(L).

the number of tasks being performed at the same time and represents the waiters' willingness to serve orders as fast as possible. Social behaviors defined for a waiter are oriented to animate chats among his friends at work. Therefore, waiters implement the internal and external social utility functions detailed in equations 8 and 9, where *Near* computes the distance between the agents while they are executing a pair of tasks. These functions evaluate social interest as the chance to meet a *workFriend* in the near future, thus performing a planned meeting.

$$U^i_{perf}(\langle i \leftarrow \text{'Use'}\rangle) = \begin{cases} 1 \text{ if } [(i = Auctioneer) \wedge IsFree(Resource)] \vee \\ \quad [IsUsing(i, Resource) \wedge not(IsComplete(Resource))] \\ 0 \text{ Otherwise} \end{cases} \tag{6}$$

$$U^i_{perf}(\langle i \leftarrow \text{'Give'}\rangle) = \begin{cases} 1 \text{ if } [(i = Auctioneer) \wedge nextAction = NULL] \vee \\ \quad [currentTask = \text{'Give'} \wedge not(handsBusy < 2)] \\ 0 \text{ Otherwise} \end{cases} \tag{7}$$

$$U^i_{int}(\langle i \leftarrow t, j \leftarrow t_{next}\rangle) = \begin{cases} 1 \text{ if } IsWorkFriend(i, j) \wedge Near(t, t_{next}) \wedge \\ \quad ExecTime(t_{next}) > RemainTime(currentTask) \\ 0 \text{ Otherwise} \end{cases} \tag{8}$$

$$U^i_{ext}(\langle j \leftarrow t\rangle) = \begin{cases} 1 \text{ if } IsWorkFriend(i, j) \wedge Near(currentTask, t) \\ 0 \text{ Otherwise} \end{cases} \tag{9}$$

On the other hand, customers place orders and consume them when served. At the moment, we are not interested in improving customer performance but in animating interactions between the members of a social group (i.e. teachers,

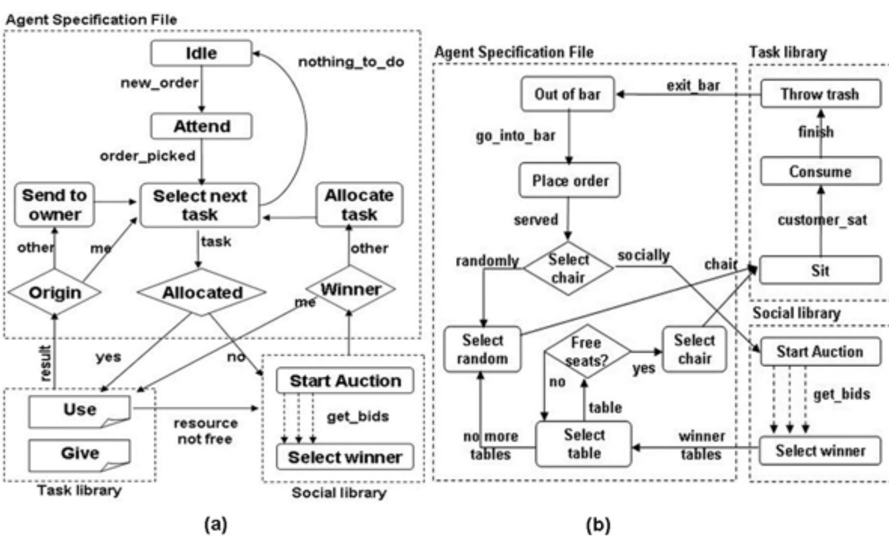

Fig. 4. (a)Waiter specification, (b) Customer specification

undergraduates and graduates). The finite state machine in figure 4b governs the actuation of customers that use auctions to solve the problem of *where to sit*. Depending on his or her sociability factor, a customer can randomly choose a chair or start an auction to decide where to sit and consume. This auction is received by all customers in the bar, which use the external social utility function defined in equation 10 to promote social meetings. This function uses the *groupSocialRelations* to determine if two individuals belong to the same group. We define the performance and the internal social utility functions as 0 since task passing is not possible in this case (i.e. no-one can sit instead of another customer). Finally, when a social meeting emerges, both waiters and customers use the plans in the *Conversational Library* to sequence the speech-acts needed to animate commitments, greetings or simple conversations.

$$U^i_{ext}(\langle j \leftarrow \text{'Sit'}\rangle) = \begin{cases} 1 \text{ if } IsSameGroup(i,j) \wedge IsConsuming(i, auctionedTable) \\ 0 \text{ Otherwise} \end{cases} \tag{10}$$

5 Results

To illustrate the effects of the social techniques previously defined we have simulated the virtual university bar example with up to 10 waiters serving 100 customers, both with different sociability factors. We estimate the social welfare of our society using two metrics explained along this section: $Throughput$ and *Animation*. $Throughput$ is an indicator in the range $[0, 1]$ that estimates how close a simulation is to the ideal situation in which the workload can be distributed among the agents and no collisions arise. Thus, equation 11 defines $Throughput$ as the ratio between this ideal simulation time (T^*_{sim}) and the real simulation time (T_{sim}), where N_{tasks} and N_{agents} are the number of tasks and agents respectively and $\overline{T_{task}}$ is the mean time to execute a task.

$$Throughput = \frac{T^*_{sim}}{T_{sim}} = \frac{N_{tasks} * \overline{T_{task}} / N_{agents}}{T_{sim}} \tag{11}$$

Figure 5a shows the $Throughput$ obtained by different types of waiters versus self-interested agents (i.e. agents with no social mechanisms included). In this first social configuration, all waiters are friends and customers are automatically assigned a group (teacher, undergraduate or graduate) when they come into the scenario. Self-interested agents collide as they compete for the use of the shared resources and these collisions produce high waiting times as the number of agents grows. We can enhance this low performance with elitist agents ($Sociability = 0$) which coordinately exchange tasks with others that can carry them out in parallel, thus reducing the waiting times for resources. Nevertheless, they produce unrealistic outcomes since they are continuously working if they have the chance, leaving aside their social relationships (in our example, chats between friends). The *Sociability* factor can be used to balance rationality and sociability. Therefore, the $Throughput$ for the sort of animations we are pursuing should be placed somewhere in between elitist and fully reciprocal social agents (*Sociability* = 1). On the other hand, figure 5b demonstrates that the higher the *Sociability* factor is, the larger the number of social meetings that will be performed by the customers when they sit at a table.

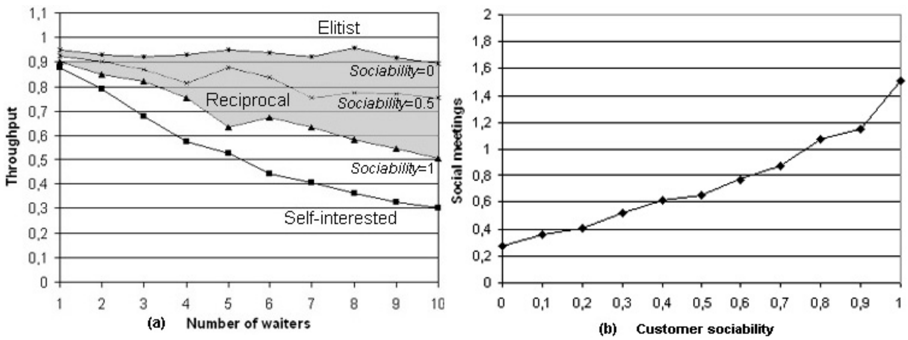

Fig. 5. (a) Waiter *Throughput*, (b) Customer social meetings

Throughput is an estimator for the behavioral performance but, despite being a basic requirement when simulating groups of virtual characters, it is not the only criterion to evaluate when we try to create high quality simulations. Therefore, we have defined another estimator that takes into account the amount of time that the designer of the simulation wants to be spent in social interactions. According to this, we define the following simulation estimator:

$$Animation = \frac{T^*_{sim} + T_{social}}{T_{sim}}, \tag{12}$$

where T_{social} represents the time devoted to chatting and to animating social agreements among friends. In our virtual bar we have chosen T_{social} as the 35% of T^*_{sim}. Figure 6 shows the animation values for 10 reciprocal social waiters with 4 degrees of friendship: all friends, 75% of the agents are friends, half of the agents are friends and only 25% of the agents are friends. As we have already mentioned, low values of *Sociability* produce low quality simulations since the values obtained for the animation function are greater than the reference value ($Animation = 1$). On the other hand, high values of *Sociability* also lead to low quality simulations, especially when the degree of friendship is high. In these cases, the number of social conversations being animated is too high to be realistic and animation is far from the reference value. The animation function can be used to extract the adequate range of values for the *Sociability* factor, depending on the situation being simulated. For example, in our virtual bar we consider as good quality animations those which fall inside ±10% of the reference value (see shaded zone in figure 6). Hence, when all the waiters are friends, good animations emerge when $Sociability \in [0.1, 0.3]$.

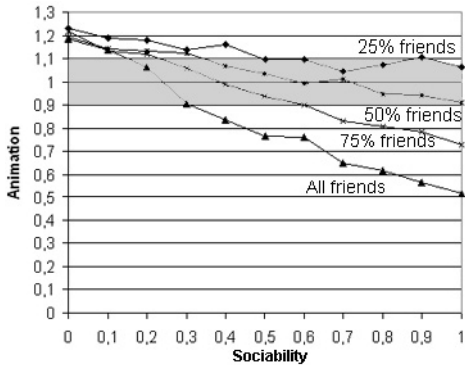

Fig. 6. Animation results obtained for waiters

Table 1. Time distribution for 10 waiters in the bar (time values are in seconds)

	Sociability = 0				*Sociability* = 1			
Agent	T_{wait}	T_{use}	T_{give}	*Balance*	T_{wait}	T_{use}	T_{give}	*Balance*
1	0	32	19	-6	16	69	34	-2
2	3	4	26	-3	18	58	24	-2
3	14	52	1	28	41	45	16	0
4	3	16	28	-3	48	60	27	3
5	0	7	30	-16	34	58	12	-1
6	3	37	17	-1	48	64	14	-2
7	0	67	4	21	18	48	24	1
8	0	45	17	1	33	45	24	4
9	7	5	23	-11	46	36	21	0
10	1	6	41	-10	27	56	20	-1

Finally, table 1 compares the amount of time devoted to executing each type of task in executions with 10 elitist waiters ($Sociability = 0$) and 10 fully reciprocal social waiters ($Sociability = 1$). The irregular values in the columns T_{use} and T_{give} on the left side of the table demonstrate how some agents have specialized in certain tasks. For instance, agents 2, 5, 9 and 10 spend most of their time giving products to the customers while agents 3 and 7 are mainly devoted to using the resources of the bar (e.g. the coffee machine, etc). Although specialization is a desirable outcome in many multi-agent systems, egalitarian human societies need also to balance the workload assigned to each agent. On the right side of the table, fully reciprocal social waiters achieve equilibrium between the time they are giving products and the time they are using the resources of the environment (see columns T_{use} and T_{give}). Furthermore, the reciprocity factor balances the number of favors exchanged among the agents (compare *Balance* columns). A collateral effect of this equilibrium is the increase in the waiting times, since social agents will sometimes prefer to meet his friends in a resource than to reallocate the task (compare columns T_{wait}).

6 Conclusions and Future Work

The animation of groups of intelligent characters is a current research topic with a great number of behavioral problems to be tackled. We aim at incorporating human style social reasoning in character animation. Therefore, this paper presents a technique to properly balance social with task-oriented plans in order to produce realistic social animations. We propose the use of ontologies to define the social relations within an artificial society and the use of a market based mechanism to reach sociability by means of task exchanges. The multi-agent animation framework presented allows for the definition of different types of social agents: from elitist agents (that only use their interactions to increase the global performance of the group) to fully reciprocal agents. These latter agents extend the theory of social welfare with a reciprocity model that allows the agents to control the emergence of social interactions among the members of a society.

Work is ongoing to provide the agents with mechanisms to self-regulate their *Sociability* factor depending on their social relations and on their previous intervention. Thus, agents will be able to dynamically adjust to the situation in order to stay within the boundaries of good quality animations at all times.

Acknowledgements

This work has been jointly supported by the Spanish MEC and European Commission FEDER funds under grants Consolider Ingenio-2010 CSD2006-00046 and TIN2006-15516-C04-04.

References

1. Hogg, L.M., Jennings, N.: Socially intelligent reasoning for autonomous agents. IEEE Transactions on System Man and Cybernetics 31(5), 381–393 (2001)
2. Piaget, J.: Sociological studies. Routledge (1995)
3. Bickmore, T., Cassell, J.: Relational agents: A model and implementation of building user trust. In: Proc. of CHI 2001: Conference on Human Factors in Computing Systems, ACM, New York (2001)
4. Tomlinson, B., Blumberg, B.: Social synthetic characters. In: Proc. of Computer Graphics. vol. 26 (2002)
5. Prada, R., Paiva, A.: Believable groups of synthetic characters. In: Proc. of AAMAS 2005: Autonomous Agents and Multi-Agent Systems, pp. 37–43. ACM, New York (2005)
6. Decker, K.S., Lesser, V.R.: Designing a family of coordination algorithms. In: Readings in Agents (1997)
7. Decker, K.: Environment Centered Analysis and design of Coordination Mechanisms. PhD thesis, University of Massachusetts, Amherst (May 1995)
8. Giampapa, J.A., Sycara, K.: Team-oriented agent coordination in the RETSINA multi-agent system. Tech. report CMU-RI-TR-02-34, Robotics Institute-Carnegie Mellon University (2002)
9. Ciger, J.: Collaboration with agents in VR environments. PhD thesis, École Polytechnique Fédérale de Lausanne (2005)

10. Grimaldo, F., Lozano, M., Barber, F., Orduña, J.: Integrating social skills in task-oriented 3D IVA. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, Springer, Heidelberg (2005)
11. Conte, R., Castelfranchi, C.: Cognitive and Social Action. UCL Press, London (1995)
12. Hexmoor, H.: From inter-agents to groups. In: Proc. of ISAI 2001: International Symposium on Artificial Intelligence (2001)
13. Sichman, J., Demazeau, Y.: On social reasoning in multi-agent systems. Revista Ibero-Americana de Inteligencia Artificial 13, 68–84 (2001)
14. Falcone, R., Pezzulo, G., Castelfranchi, C., Calvi, G.: Why a cognitive trustier performs better: Simulating trust-based contract nets. In: Proc. of AAMAS 2004: Autonomous Agents and Multi-Agent Systems, pp. 1392–1393. ACM Press, New York (2004)
15. Grimaldo, F., Barber, F., Lozano, M.: An ontology-based approach for IVE+VA. In: Proc. of IVEVA 2006: Intelligent Virtual Environments and Virtual Agents (2006)
16. Viroli, M., Ricci, A., Omicini, A.: Operating instructions for intelligent agent coordination. The Knowledge Engineering Review 21, 49–69 (2006)
17. Kao, E.C.C., H-M., C.P., Chien, Y.H., Soo, V.W.: Using ontology to establish social context and support social reasoning. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, Springer, Heidelberg (2005)
18. Giménez-Lugo, G., Sichman, J., Hübner, J.: Addressing the social components of knowledge to foster communitary exchanges. International Journal on Web Based Communities 1(2), 176–194 (2005)
19. Casare, S., Sichman, J.: Towards a functional ontology of reputation. In: Proc. of AAMAS 2005: Autonomous Agents and Multi-Agent Systems, ACM, New York (2005)
20. Bordini, R.H., da Rocha, A.C., Hübner, J.F., Moreira, A.F., Okuyama, F.Y., Vieira, R.: MAS-SOC: A social simulation platform based on agent-oriented programming. Journal of Artificial Societies and Social Simulation 8(3) (2005)
21. Ribeiro, M., da Rocha, A.C., Bordini, R.H.: A system of exchange values to support social interactions in artificial societies. In: Proc. of AAMAS 2003: Autonomous Agents and Multi-agent Systems, ACM Press, New York (2003)
22. Bordini, R.H., Hübner, J.F.: Jason (March 2007), Available at `http://jason.sourceforge.net/`