A Framework for Simulated Crowd Behavior in Virtual Environments

Giuseppe Vizzari
Complex Systems and Artificial Intelligence Research Institute
University of Milano Bicocca – via Bicocca degli Arcimboldi 8 – 21126
Milano ITALY

Giorgio Pizzi
Complex Systems and Artificial Intelligence Research Institute
University of Milano Bicocca – via Bicocca degli Arcimboldi – 21126
Milano ITALY

Flávio Soares Corrêa da Silva
Laboratory of Interactivity and Digital Entertainment Technology
University of São Paulo
Rua do Matão 1010
São Paulo BRAZIL
05504-090

Abstract

In this paper we present a framework for the definition and implementation of virtual environments inhabited by (possibly large) groups of situated autonomous agents defined according to the computational model of *Situated Cellular Agents*. We briefly describe the related work and possible application scenarios for this framework, and then we introduce the multi-agent model underlying the framework and its basic architecture. We close the paper with some potential applications of this framework, discussion and future work.

Keywords: multi-agent systems, virtual environments, simulation, 3D visualization.

1. Introduction

The design and computational implementation of virtual environments inhabited by situated autonomous entities is interesting for application areas such as:

- Human interaction through Embodied Conversational Agents (Nakanishi et all, 2003).
- Enriched interactions in virtual laboratories and classrooms (Batty and Hudson-Smith, 2005; Ishida et all, 2007; Dijkstra et all, 2003).
- Modeling, simulation and visualization of situated agents in existing, planned or reconstructed environments (Nugues et all, 2003; Papagiannakis et al, 2005; Dijkstra and Timmermans, 2002).
- Digital entertainment (such as e.g., Second Life).

These applications require expressive models for the specification of behavior of the entities situated in a virtual environment. Multi-agent systems (MAS) (Ferber, 1999) are particularly well suited to tackle the modeling issues introduced by these applications, since they aim precisely at the design and implementation of behaviors and interaction patterns among autonomous entities.

In the present paper we introduce the implementation of a framework for MAS based on *Situated Cellular Agents* (SCA) (Bandini et all, 2002). In section 2 we discuss some related work applied to a variety of application scenarios. In section 3 we briefly introduce the SCA model. In section 4 we discuss the architecture of the proposed framework, its main components and the tools to support developers who may adopt it. In section 5 we briefly discuss two potential applications of the framework comprising several hundreds of agents. Finally, in section 6 we presented some conclusions and proposed future work.

2. Related Work

The design and implementation of virtual 3D environments inhabited by autonomous entities has been the goal of several projects, commercial as well as academic. A complete and thorough description of the state of the art in this area would be rather difficult to build and result in a text whose dimensions would fall beyond the scope of this paper.

Some relevant open source projects are Mason (Mason, 2005) and Breve (Klein, 2002). Mason is a discrete-event multi-agent simulation library. Breve supports the definition of 3D simulations of MAS and artificial life.

Some relevant representatives of the category of commercial tools that can support the design and development of virtual environments are Quadstone Paramics and Massive. Paramics is a traffic micro-simulation software, which is able to generate realistic 3D visualizations. Massive is an application specifically devoted to the generation of photorealistic animation of crowd-related visual effects, which has been used in several movie productions, such as The Lord of the Rings.

A system whose goals are similar to the ones adopted in our work is Freewalk. It adopts a scenario specification language, called Q, for the specification of the environment and the behavior of agents. Freewalk, however, is not open source.

3. Situated Cellular Agents

A system of Situated Cellular Agents is denoted as a triple $\langle Space, F, A\rangle$ in which A is a finite set of agents, F is a finite set of fields, and *Space* is the environment in which the agents are situated, upon which they act autonomously and with which they interact by means of two sorts of interaction: reaction and propagation of fields.

Agents can have different types. The possibility to define different agent types introduces the opportunity to define different abilities and perceptive capabilities. Defining T as the set of types, we assume that T induces a partition on the set of agents, i.e. the agents are classified in disjoint subsets corresponding to the different types in T. An agent type τ is defined by the triple $\langle \Sigma_\tau\ Perception_\tau\ Action_\tau \rangle$ in which Σ_τ defines the set of states that agents of type τ can assume; $Perception_\tau : \Sigma_\tau \rightarrow [N \times W_{f1}] \ldots [N \times W_{f|F|}]$ associates to each agent state the vector of pairs representing respectively a receptiveness coefficient modulating the intensity of that kind of field and a sensitivity threshold represented by a specific field value. This vector of pairs is defined as

$$\left(c_\tau^1(s), t_\tau^1(s)\right), \ldots, \left(c_\tau^{|F|}(s), t_\tau^{|F|}(s)\right)$$

where for each i ($i=1 \ldots |F|$), $c^i_\tau(s)$ and $t^i_\tau(s)$ express respectively a receptiveness coefficient to be applied to the field value f_i and the agent sensibility threshold to f_i in the given agent state s; and $Actions_\tau$ denotes the set of actions that agents of type τ can perform.

The *Space* consists of a set P of sites arranged in a network. Each *site* $p \in P$ can contain at most one agent and is defined by $\langle a_p, F_p, P_p \rangle$ where $a_p \in A \cup \{\perp\}$ is the agent situated in p ($a_p = \perp$ when no agent is situated in p, in other words p is empty); $F_p \subseteq F$ is the set of fields active in p ($F_p = \varnothing$ when no field is active in p); and $P_p \subset P$ is the set of sites adjacent to p. Edges connecting sites represent a constraint to the movement of agents situated in the environment and also on the diffusion of fields, which only propagate through these connections.

A field $f_\tau \in F$ that can be emitted by agents of type τ is denoted as a tuple of the form $\langle W_\tau, Diffusion_\tau, Compare_\tau, Compose_\tau \rangle$ where $W_\tau = S \times N$ and $S \subseteq \Sigma_\tau$ denotes the set of values that the field can assume; given $w_\tau \in W_\tau$, $w_\tau = \langle s_\tau i_\tau \rangle$, where $s \in S$ represents information brought by the field (i.e. the field payload) and $i_\tau \in N$ represents its intensity; $Diffusion_\tau : P \times W_\tau \times P^{pd} \rightarrow (W_\tau)$ is the diffusion function for field type τ, i.e. $Diffusion_\tau(p_s, w_\tau p_d)$ computes the value of a field on a given destination site (p_d) taking into account in which site it was emitted (p_s) and with which initial value ($w_\tau \in W_\tau$); $Compare_\tau : W_\tau \times W_\tau \rightarrow \{True, False\}$ is the function that compares field values. It is used by the perceptive system of agents to evaluate if the value of a certain field type is such that it can be perceived; and $Compose_\tau : (W_\tau)^+ \rightarrow W_\tau$ expresses how field values of the same type have to be combined in order to obtain the unique value of a field type at a given site.

An agent $a \in A$ is defined by the triple $\langle s, p, \tau \rangle$ in which $s \in \Sigma_\tau$ denotes the *agent state* and can assume one of the values specified by its type; $p \in P$ is the site of the *Space* where the agent is situated; τ is the *agent type*, which provides the allowed states, perceptive capabilities and behavioural specification for that type of agents.

The set $Action_\tau$ is comprised by a set of actions and an action selection strategy. Actions can be selected from a set of primitives which include *reaction* (synchronous interaction among adjacent agents), *field emission* (asynchronous interaction among at–a–distance agents through the field diffusion–perception–action mechanism), *trigger* (change of agent state as a consequence of a perceived event) and *transport* (agent movement across the space). A non deterministic action selection strategy is invoked when the preconditions of more than one action are verified.

The behavior of Situated Cellular Agents is influenced by agents situated on adjacent positions and, according to their type and state, agents are able to synchronously change their states. Synchronous interaction (i.e. reaction) is a two–step process. Reaction among a set of agents takes place through the execution of a protocol introduced in order to synchronize the set of autonomous agents. When an agent wants to react with the set of its adjacent agents since their types satisfy some required condition, it starts an *agreement* process whose output is the subset of its adjacent agents that have agreed to react. An agent agreement occurs when the agent is not involved in other actions or reactions and when its state is such that this specific reaction could take place. The agreement process is followed by the synchronous reaction of the set of agents that have agreed to it.

The basic idea underlying the application of the SCA model to represent environments and situated mobile agents – such as pedestrians in an urban setting – is that their movement can be generated by means of attraction and repulsion effects (as also suggested in (Batty, 2003)). These effects are generated by means of fields that can be emitted from specific points in the environment, which can be perceived as attractive/repulsive or that can even be simply ignored by different types of agents in specific states. Additionally, the agents can also emit fields, and therefore they can also generate attraction/repulsion effects.

4. The Visualization Framework

We have adopted the open source game engine

Irrlicht for 3D visualizations of SCA models. Essentially, we have built a framework comprised by a collection of C++ classes on top of Irrlicht that implement the fundamental concepts of SCA.

The framework must manage the coordinated execution of the model defined for a specific virtual environment and update its visualization. To manage the execution of different modules and procedures, three main operative modes have been defined. The first two are characterized by the fact that agents are not provided with a thread of control of their own, and the third mode associates a thread of control to each agent. In all activation modes the environment is in charge of a regulation function that constrains the autonomy of the agents, in order to manage the consistency of the overall model and/or to manage accordingly the visualization.

5. Sample Applications

In this section we present two sample applications to show how the framework supports the definition of SCA models and corresponding effective 3D visualization. The applications were also chosen to show the potential of the framework in terms of execution of a large number of agents. Tests were carried out on a notebook configured with Windows® XP Professional operating system, an Intel® Pentium® IV 2.4 GHz processor, 320 MB RAM and an ATI® Radeon® IGP graphics card with 128 MB (shared system memory).

The first application is the simulation of the evacuation of a building. The behaviors of the agents are very simple, and only provide the movement towards specific exits. Agents reaching these exits are simply eliminated from the scenario. In this scenario the environment comprises a graph of around 1000 sites, connected by more than 3500 arcs; 150 agents are situated in the scenario and they are activated according to sequential activation strategy. The analytical results of the simulation are not relevant in this context, also because the agent models were not calibrated against real data; the simulation was executed and visualized with a number of frames per second (FPS) constantly above 60. The speed of the simulation was in fact actually limited to achieve a smooth form of visualization of the system dynamics.

The second example is the movement of agents inside a virtual museum. In this case the environment comprises around 2000 sites with around 6000 arcs connecting them; 500 agents were randomly positioned inside the building, and they were provided with a thread of control of their own. The environment and the agents were characterized by a 3D visual model. Once again, the analytical results of this simulation are not relevant, since the agent models were extremely simple and they were not calibrated against real data. The simulation was executed and visualized with a number of FPS constantly above 30.

6. Conclusions and Future Developments

The simulation of large groups of agents is a relevant issue for digital entertainment – including computer games as well as feature movies – as well as for the realistic simulation of physical environments. The algorithmic simulation, based on MAS modeling, is particularly relevant if one is interested in the dynamical simulation of social interactions.

In the present paper we have presented a framework supporting the definition and implementation of virtual environments inhabited by interacting situated agents modeled according to the SCA model. The framework supports the specification and execution of visually rich 3D virtual environment characterized by the presence of situated agents acting and interacting in it.

Some of our future work shall be devoted to improving the capabilities of the computational tools to design, build and execute virtual models of inhabited physical environments. We shall provide support, for instance, to the definition of the spatial structure of the virtual environment. We shall also work on extending the SCA model – as well as its parent model, the MMASS – in order to make it more expressive and ergonomic.

References

BANDINI, S., MANZONI, S., SIMONE, C. (2002). Heterogeneous Agents Situated in Heterogeneous Spaces. Applied Artificial Intelligence, 16(9-10):831–852, 2002.

BATTY, M. (2003). Agent-based pedestrian modeling, in *Advanced Spatial Analysis: The CASA Book of GIS*, pp. 81-106, 2003.

BATTY, M., HUDSON-SMITH, A. (2005). Urban Simulacra: From Real to Virtual Cities, Back and Beyond, *Architectural Design*, 75 (6):42-47.

DIJKSTRA, J., VAN LEEUWEN, J., TIMMERMANS, H. J. P. (2003). Evaluating Design Alternatives Using Conjoint Experiments in Virtual Reality. Environment and Planning B 30 (3), 357–367.

DIJKSTRA, J., TIMMERMANS, H. (2002). Towards a multi-agent model for visualizing simulated user behavior to support the assessment of design performance. Automation in Construction. vol. 11 (2002). Elsevier. 135-145.

FERBER, J. (1999). Multi-Agent Systems. Addison-Wesley.

ISHIDA, T., NAKAJIMA, Y., MURAKAMI, Y., NAKANISHI, H. (2007). Augmented Experiment: Participatory Design with Multiagent Simulation. IJCAI 2007, Proceedings of the

20th International Joint Conference on Artificial Intelligence (2007):1341-1346.

Klein, J. (2002). Breve: a 3D simulation environment for the simulation of decentralized systems and artificial life. In Proceedings of Artificial Life VIII, the 8th International Conference on the Simulation and Synthesis of Living Systems. The MIT Press, pp.329–334. http://www.spiderland.org/breve/breve.pdf.

MASON (2005). A Multi-Agent Simulation Environment. 2005. Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan. In Simulation 81(7) :517-527, 2005.

Nakanishi, H., Nakazawa, S., Ishida, T., Takanashi, K., Isbister, K. (2003). Can Software Agents Influence Human Relations? - Balance Theory in Agent-mediated Communities - *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2003)*, pp. 717-724, 2003.

Nugues, P., Dupuy, S., Egges, A. (2003). Information Extraction to Generate Visual Simulations of Car Accidents from Written Descriptions. ICCSA (1) 2003: 31-40.

Papagiannakis, G., Schertenleib, S., O'Kennedy, B., Arevalo-Poizat, M., Magnenat-Thalmann, N., Stoddart, A. J., Thalmann, D. (2005). Mixing virtual and real scenes in the site of ancient Pompeii. Journal of Visualization and Computer Animation 16(1): 11-24 (2005).