Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 32 (2014) 822 – 827
Procedia Computer Science

The 1st International Workshop on Agent-based Modeling and Simulation of Cities (AgentCities14)

Multilevel Model of the 3D Virtual Environment for Crowd Simulation in Buildings

Stéphane Galland*, Nicolas Gaud, Jonathan Demange, Abderrafiaa Koukam

Multiagent Group, IRTES-SET, Université de Technologie de Belfort-Montbéliard, France - `http://www.multiagent.fr`

Abstract

Multiagent-based simulations enable us to validate different use-case scenarios in a lot of application domains. The idea is to develop a realistic virtual environment to test particular domain-specific procedures. This paper presents a multilevel model of a physic environment for the simulation of crowd in a virtual 3D building. The major contributions of this paper is the agentification of the model to support multilevel simulation of the environment. Finally, the application of the model inside an airport terminal is presented.

Selection and Peer-review under responsibility of the Program Chairs.

Keywords: Multi-agent simulation, Holon, Environment Model, Crowd Simulation, Janus Platform

1. Introduction

The simulation of complex systems is currently a major issue in many scientific disciplines such as neuroscience, biology, ecology, economy and urban management and sustainability. The simulation is an appropriate approach to study systems that cannot be directly observed or measured. The objective of the simulation is to facilitate the understanding of the dynamics of a system and try to predict its future evolution.Urban systems are typical examples of complex systems[1,2]. Structurally, an urban system can be hierarchically decomposed in streets, buildings, neighborhoods, districts, etc.[3,4] The second point of view is dedicated to the individual and collective activities, and is more related to the analysis of the traffic and the mobilities of the crowd[5,6]. The multiagent-based simulation (MABS) is well suited to the simulation of urban dynamics[7]. Indeed, it is more flexible than macroscopic approaches, which are usually based on differential equations to simulate spatial and scalable phenomena. Moreover, multiagent systems are able to simulate the behaviors of individuals and groups of individuals in place of statistic systems, usually used in macroscopic simulations. Finally, multiagent systems could be precisely situated using Geographic Information Systems (GIS) to obtain virtual models close to the reality. In this context, multiagent-based simulation of urban systems must support both the simulation of individual behaviors and the management of the environmental dynamics. The

* Corresponding author. Tel.: +33-384-583-418 ; Fax.: +33-384-583-342
E-mail address: stephane.galland@utbm.fr

1877-0509
Selection and Peer-review under responsibility of the Program Chairs.
doi:10.1016/j.procs.2014.05.497

JaSim simulation model[8] was proposed to support urban system simulation in 3D virtual environment. Two problems may occur during the execution the environment model: (i) the computational cost may be huge, and incompatible with efficient response constraints; and (ii) many times the executed algorithm is too complex and too expensive according to the topology of the crowd and obstacles; simpler and faster algorithms could be used in place with the same answer quality. Many works were proposed in literature to support different levels-of-details in the simulation, such as[9]. However, they mainly focus on the applicative agents and not on the environment itself. The problem becomes: what is the best model to support the physic environment and its dynamics? *In this paper, an organizational and agent-oriented model of the environment is defined.* Note that in the rest of this paper, the term "agent" refers to the agents, which are supporting the environment model; in opposition to the "application agents," which represent the pedestrians. Moreover, a specific type of agent is considerer: the holon[10], an agent composed of agents. *The agent-oriented model of the environment permits to adapt the overall environment's behavior dynamically during its execution. The use of agents allows to evaluate and to predict the computational costs of the algorithms, and to select the one, which is fitting the constraints in time and in quality.* The agent-oriented model is defined according to the Aspecs methodology[11], which focuses on the organizational and the agent points-of-view. This methodology and the related concepts are not the purpose of this paper and may be found in the Aspecs website[1].

This paper is structured as follows. Section 2 describes the general principles of the agentification of the environment. Section 3 details the dynamics of the model. Section 4 exposes simulation results and analysis. Finally, the paper finishes with a conclusion and a short description of future works.

2. Organizational Model of the Environment

Works on environments are numerous and mainly used for performances. They require highly specialized calculations and therefore, a significant number of resources. As shown in[12], the environment is often distributed according to places. A place is a semi-closed spatial area bounded by static objects (usually walls). Each place may have connections called portals, with its neighbor places. They are used to ease the interaction between two adjacent spaces. They also permit to use structural environment models such as Potentially Visible Set[12] for improving the computation of the perceptions of the synthetic individuals inside. Places are basically defined a priori by the designer of the simulation. They generally correspond to the structural decomposition of the environment[13,14,15]. Entities are objects inside the environment, and are located in a single place through a dedicated data-structure (usually a spatial tree or a spatial grid).

To simulate large and complex worlds, it is important to support unbalanced places in terms of entities they are containing. Indeed, the difference of population coverage by the places may cause fewer global performances to the simulator. To overcome this problem, places are decomposed in turn into a collection of dynamically built zones. In contrast to the statically defined decomposition, these zones are built during the simulation process.

Because organizational modelling is well-known as a mean to break-down the design complexity of a system, we propose to model the environment as a collection of interactions. We use the organizational metamodel crio, proposed by[11]. On Figure 1(a), the `Multilevel Simulation` organization defines the overall simulation system. Two roles are defined inside. The `Pedestrian` role is played by the agents who are participating in the simulation, *i.e.* the pedestrians. The `Environment` role is played by any agent or group of agents responsible for the overall behavior of the situated environment. Interactions between them are based on the influence-reaction model[16]; and on the computation of the pedestrian's perceptions[8]. These two models give the ability to each agent to have joint actions in the environment, and extract data from the environment. Each player of the `Environment` role must have the capacity[11] to compute the perceptions for each pedestrian. The `Environment`'s players must also have the capacity to gather influences — whises of actions — from each pedestrian.

The `Topological decomposition` organization focuses on the structure of the situated environment itself, as proposed by[15] but with an organizational model. The organization provides the capacities required by the `Environment` role in the previous organization. The `Topological decomposition` organization can contribute to the behavior of this higher-level role. The `Topological decomposition` organization is composed of interconnected places. Each

[1] `http://www.aspecs.org`

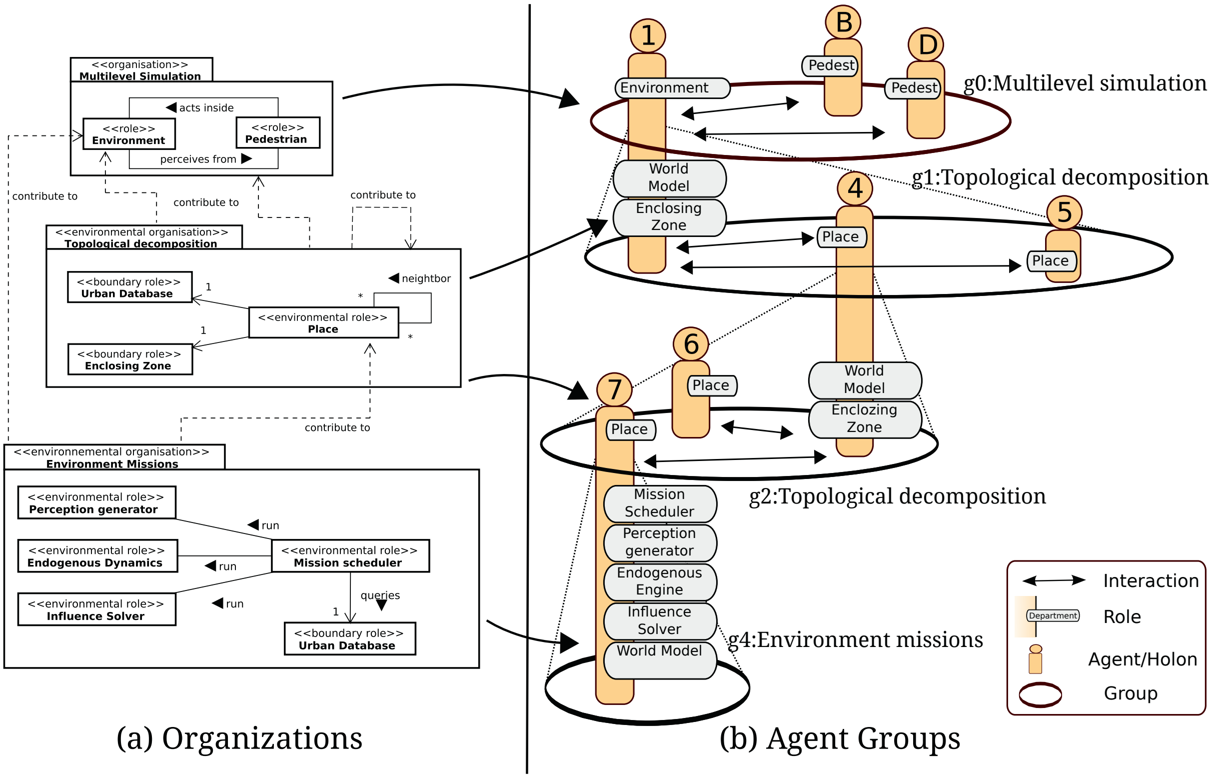

Figure 1. Example of agent society, which is managing the environment.

of them is responsible for the environment's missions in the considered space. It also manages the objects inside the zone. To realize its behavior, a `Place` role must interact with the role `Urban Database` to obtain and to change the information related to the objects inside the environment.

The role `Enclosing zone` supports the multilevel modeling of the environment (see below). The organization `Topological decomposition` represents a level within the hierarchy of composition of the environment. It is necessary that each level in this hierarchy has access to information dedicated to the multilevel dynamics. As a *boundary role*, the role `Enclosing zone` is responsible of providing to a place the state of the enclosing zone, and the indicators and the constraints given by this higher level. All these information will be described in more detail later in this paper.

The organization `Environment Mission`, inspired by[17], defines all the roles required to satisfy all the missions of the environment for a specific area. An instance of this organization is integrated as a group in all the agents, which are playing the role `Place`, which is described in the previous paragraph. This link between the two organizations is represented by the relationship `contribute to` in Figure 1(a).

The next step is to identify the agents and their behaviors in order to obtain the society agents that exhibit the expecting behavior of the organizations, and roles described above.

Figure 1(b) illustrates an instance of a society of agents, who may execute the environment behavior together. The key point is to determine for each role if a standalone agent or a group of agents is playing it. We propose to use the concept of holon: agent composed of holons[18]. In the rest of this paper, the terms agent and holons represent the same concept. The hierarchy of holon permits to supprot the hierarchical nature of the environment, which was modelled with the organizations. When one agent is managing an entire place, it is playing the role `Place` in the `Environment Model`. When a place needs to be split and managed by a group of agents, one of them must play the role `Place` in the `Topological decomposition`, and `Mission scheduler` in the `Environment Mission` organizations. The decision to decompose or not a place is the responsibility to the agent playing the `Place` role itself. It depends on several indicators from two sources: (i) the individual indicators, which are specific to an agent playing the role `Place`; and (ii) the indicators shared in the context of a group of agents, which is an instance of the organization `Topological decomposition`. Each agent playing the role `Place` can access to these indicators by interacting with the role `Enclosing zone`. The indicators are detailed in the following section.

3. Indicators of the Multilevel Simulation

At every instant of the simulation, the environment agents evaluate indicators to determine if they should change of state: (i) being a manager of a decomposed area, or (ii) the manager of an atomic area.

In case (i), the agent can be decomposed if there are enough resources to the execution of its sub-agents. Equation 1 describes the condition triggering the change of state of the agent. A super-agent must decompose when it has sufficient resources at its disposal, or the evaluation of the "almost-same" behaviors at the levels n and $n+1$ indicates that the super-agent does not approximate correctly any more the behaviors of its sub-agents. This evaluation is based on the computation of energies associated to the behaviors. If the difference between these two energies is greater than the threshold ϵ, then the upper behavior does not approximate the lower behavior.

$$\begin{aligned}\left[\left(\exists a \in D_z, \left|Eg_z - Eg_a\right| > \epsilon\right) \vee \left(\forall R, R_z \geq \sum_{p \in D_z} g_R(p) + k_z\right)\right] \wedge \\ (\max_z < i \vee \min_z > i_z) \wedge (E_z \neq \emptyset)\end{aligned} \tag{1}$$

D_z and E_z are the sets of sub-areas of and objects in the zone z, respectively. Eg_α is the energy associated to the zone α. The expression of this energy depends on the application. An example is given in Section 4. $g_R(z)$ is the estimation of the amount of ressources R that are required to execute the environment behavior for the zone z. k_z is the computation time to take the splitting/merging decision.

In case (i), the agent is decomposed into a set of sub-agents managing the sub-areas of z, the area associated with the super-agent. This determines whether to retain its sub-agents or destroy them. This last case corresponds to a change of the state of the super-agent. A super-agent can destroy its members when it does not have enough resources at its disposal to carry out the simulation and the evaluation of the consistency between the simulations at the levels n and $n+1$.

$$\left(\forall a \in D_z, \left|Eg_z - Eg_a\right| \leq \epsilon\right) \wedge \left(\forall R, R_z < \sum_{p \in D_z} g_R(p) + k_z\right) \wedge \min{}_z < i \tag{2}$$

The Equations 1 and 2 are based on the following three basis indicators: (i) The **mass of a zone** M_z indicates the importance of an area z of the environment for the simulation. This value depends on the scenario. For example, it may be proportional to the density of pedestrians in the area, or depends upon the presence of an immersed human user in this area. (ii) The **structural depth** describes the minimum or the maximum depth, $\min_z$ and $\max_z$, of the decomposition of a zone. Thus, it is possible for a role `Place` to restrict the depth of its topological decomposition. (iii) The **resource constraint** R_z describes the limits of the available resources for a place z to achieve its simulation. This constraint allows to take into account low-level information, close the operating system, such as the computation time. It is possible to impose a time constraint for approaching a real-time execution. A resource constraint can also describe the limits for any type of low-level resource (memory, network bandwidth...) The value of R_z depends on the mass of the zone: $R_z = \frac{R_u.M_z}{\sum_{n \in u} M_n}$, where u is the enclosing area of z, and n is a subarea of u (including z).

According to [19], the execution mechanism of the hierarchy follows a broad-first iteration over the agents. It means that the parent agents are executed prior to the children. According to the two cases above, the execution of the children is launched only in case (i).

Figure 2. Screenshot of the airport simulation

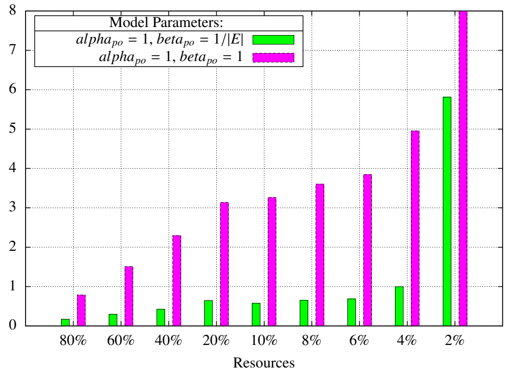

Figure 3. Evaluation of the Average Energy

4. Experiments

This section describes several experiments with the agent-based environment model on the simulation of an airport halls (illustrated by Figure 2) done on the Janus platform[2], with the 3D environment model JaSim. The airport terminal is composed of two halls, which are separated by gates. Each of these gates is a check point between the public area and the boarding area.

The behavior of the agents is decomposed on three majors activities: (i) going to check-in desk, 2/3 of the passengers need to check in their baggages, and 1/3 have only hand-baggages; (ii) passing the check points; and (iii) boarding. Figure 3 shows the evaluation of the energy according to different level of available computational resources. When this resource critera is at 100%, it means that the computer has enough resources to run the simulation at the finest level. When the resource is down at 60%, it means that only 60% percent of the micro-simulation may be run at the finest level. As explained in the previous section, the energy evaluation depends on the application. Equation 3 details a simple evaluation of this energy for the airport application. Intuitively, this energy assesses the quality of generated perceptions by the environment: more objects are not included in the perception, compared with the more precised possible perception, less is the quality of the perception. $p^{\odot}$ is the set of perceived objects that are found when it is computed at the lowest level. $p^{\ominus}$ (resp. $p^{\oplus}$) represents the objects that are lost (resp. added) at a higher level in the holarchy. $(\alpha_{po}, \beta_{po}) \in [0;1]^2$ are calibration variables (usually, $\alpha_{po} = 1, \beta_{po} = \frac{1}{|E|}$, $|E|$: total number of entities). They specify the contribution of the missed and added perceptions to the energy of the area. When value of α_{po} tends to 1, the lost of perceived objects at higher level tends to be forbidden (similar for β_{po} but for object discovery).

$$Eg_{\alpha} = \begin{cases} \dfrac{\alpha_{po}|p^{\ominus}| + \beta_{po}|p^{\oplus}|}{|p^{\odot}|} & \text{if } p^{\odot} \neq \emptyset \\ \alpha_{po}|p^{\ominus}| + \beta_{po}|p^{\oplus}| & \text{else} \end{cases} \tag{3}$$

The tests are performed with a set of 2,000 entities in the entry hall and 1,000 entities in the boarding area. Four check points are assumed to be available. The average computation time for one simulation step of the flat model[8], without holarchy, at the highest level of detail is 25.9 seconds, the equivalent holonic model (proposed in this paper) takes 41.5 seconds with a single place for the entire area, and 8.1 seconds with two places. The better performances of the agent-oriented approach are mainly due to a better balancing of the nodes of the two spatial trees, one for each place, than the balancing of the single spatial tree of the two first models.

[2] http://www.janus-project.org

5. Conclusion

Multiagent-based simulations enable us to validate different use-case scenarios in a lot of application domains. The idea is to develop a realistic virtual environment to test particular domain-specific procedures.

This paper presents an agent-oriented and multilevel model of a situated environment for the simulation of a crowd in a virtual 3D building. The major contributions in this paper are, in one hand, an accurate and efficient model of an environment, and on the other hand, the agentification of this model to support multilevel simulation of the environment. The model is successfully applied to the simulation of two airport halls. These experiments permits to evaluate the impact of the multilevel simulation on the simulation results, and the gain in terms of computational costs.

The energy formula presented within this paper may be generalized to become less application-dependent. In this paper, we propose to use energy-based indicators. Other types of indicators may be used in place to obtain accurate evaluations. Finally, the proposed model may be applied on large-scale systems to evaluate the approximation introduced by our multilevel model.

References

1. N. Farenc, R. Boulic, D. Thalmann, An informed environment dedicated to the simulation of virtual humans in urban context, in: Proceedings of EUROGRAPHICS 99, 1999, pp. 309–318.
2. G. Thomas, S. Donikian, Virtual humans animation in informed urban environments, in: CA '00: Proceedings of the Computer Animation, IEEE Computer Society, Washington, DC, USA, 2000, p. 112.
3. G. Pouyanne, Forme Urbaine et Mobilité Quotidienne, Ph.D. thesis, Université Montesquieu - Bordeaux IV (Dec. 2004).
4. F. Lamarche, S. Donikian, Crowds of Virtual Humans : a New Approach for Real Time Navigation in Complex and Structured Environments, Computer Graphics Forum (Proc. of Eurographics 2004) 23 (3).
5. L. Jian, Y. Lizhong, Z. Daoliang, Simulation of bi-direction pedestrian movement in corridor, Physica A: Statistical Mechanics and its Applications 354 (0) (2005) 619 – 628. `doi:10.1016/j.physa.2005.03.007`.
6. D. Thalman, N. Farenc, R. Boulic, Virtual human life simulation and database: Why and how, in: DANTE '99: Proceedings of the 1999 International Symposium on Database Applications in Non-Traditional Environments, IEEE Computer Society, Washington, DC, USA, 1999, p. 471.
7. A. Bretagnolle, E. Daudé, D. Pumain, From theory to modelling : urban systems as complex systems, Cybergeo, 13th European Colloquium on Quantitative and Theoretical Geography 335.
8. S. Galland, N. Gaud, J. Demange, A. Koukam, Environment model for multiagent-based simulation of 3D urban systems, in: the 7th European Workshop on Multiagent Systems (EUMAS09), Ayia Napa, Cyprus, 2009, paper 36.
9. L. Navarro, V. Corruble, J.-D. Zucker, A flexible approach to multi-level agent-based simulation with the mesoscopic representation, in: 12th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS), 2013, pp. 159–166.
10. S. A. Rodriguez, From analysis to design of Holonic Multi-Agent Systems: a Framework, methodological guidelines and applications, Ph.D. thesis, Université de Technologie de Belfort-Montbéliard (2005).
11. M. Cossentino, N. Gaud, V. Hilaire, S. Galland, A. Koukam, ASPECS: an agent-oriented software process for engineering complex systems - how to design agent societies under a holonic perspective, Autonomous Agents and Multi-Agent Systems 2 (2) (2010) 260–304. `doi:10.1007/s10458-009-9099-4`.
12. D. Cohen-Or, Y. L. Chrysanthou, C. T. Silva, F. Durand, A survey of visibility for walkthrough applications, IEEE Transactions on Visualization and Computer Graphics 9 (3) (2003) 412–431. `doi:10.1109/TVCG.2003.1207447`.
13. N. Farenc, An informed environment for inhabited city simulation, Ph.D. thesis, Lausanne (2001).
14. D. Vanbergue, A. Drogoul, Approche multi-agent pour la simulation urbaine, in: Journée Cassini 2002, 2002.
15. D. Vanbergue, Conception de simulation multi-agents: application à la simulation des migrations intra-urbaines de la ville de Bogota, Ph.D. thesis, Université Paris VI (December 2003).
16. F. Michel, The IRM4S model: the influence/reaction principle for multiagent based simulation, in: Sixth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS07), ACM, 2007. `doi:10.1145/1329125.1329289`.
17. D. Weyns, H. V. D. Parunak, F. Michel, T. Holvoet, J. Ferber, Environments for Multiagent Systems State-of-the-Art and Research Challenges, in: D. Weyns, H. V. D. Parunak, F. Michel (Eds.), Third International Workshop (E4MAS 2006), Vol. 4389 of Lecture Notes in Artificial Intelligence, Springer, Hakodate, Japan, 2006, pp. 1–47.
18. N. Gaud, S. Galland, V. Hilaire, A. Koukam, An organizational platform for holonic and multiagent systems, in: the Sixth International Workshop on Programming Multi-Agent Systems (ProMAS'08), of the Seventh International Conference on Autonomous agents and Multiagent Systems (AAMAS), Estoril, Portugal, 2008, pp. 111–126.
19. N. Gaud, S. Galland, A. Koukam, Towards a Multilevel Simulation Approach Based on Holonic Multiagent Systems, in: UKSIM '08: Proceedings of the Tenth International Conference on Computer Modeling and Simulation, IEEE Computer Society, Washington, DC, USA, 2008, pp. 180–185. `doi:10.1109/UKSIM.2008.36`.