A Hybrid Agent Architecture for Situated Agents Based Virtual Environments

Giuseppe Vizzari and Francesco Olivieri
Complex Systems and Artificial Intelligence research center
University of Milano-Bicocca, via Bicocca degli Arcimboldi 8, 20126 Milano
giuseppe.vizzari@disco.unimib.it, f.olivieri1@campus.unimib.it

ABSTRACT
This paper introduces a hybrid agent-architecture for situated agents representing pedestrians in virtual environments. The presented agent architecture is an extension of a model supporting the definition of environments encompassing situated agents, whose behavioural specification was mainly based on simple reactive rules. The new architecture supports a more compact specification of agent's behaviours by endowing agents with a qualitative form of knowledge about the environment and simple rules to autonomously construct a line

Categories and Subject Descriptors
I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems.
I.6.3 [**Simulation and Modeling**]: Applications.

General Terms
Design, Experimentation.

Keywords
Situated Multi Agent Systems, 3D visualization, crowd modeling and simulation.

1. INTRODUCTION
Virtual environments inhabited by autonomous social entities characterized by a believable, plausible or realistic behaviour have been exploited in several ways such as:

- supporting human-human forms of interaction, by introducing Embodied Conversational Agents facilitating users' interactions [9] or supplying awareness information in a visually effective form [10];

- supporting participatory design, supporting the visualization of various alternative choices to the involved stakeholders [3][5][6];

- realizing effective instruments for the modeling, simulation and visualization of the dynamics of entities situated in a representation of an existing, planned or reconstructed environment or situation [11];

- entertainment, computer games or online communities (see, e.g., Second Life[1]).

All these above applications are characterized by a strong requirement for realistic and effective visualization tools (and some of them require a thorough analysis of the system usability, due to the necessary accessibility by non-technically skilled users), they also call for expressive models supporting the specification of behaviours for the entities that inhabit these environments, as well as the interaction among them and with the environment itself. The fact that the overall performance of the system is essentially dependant on the single actions and interactions that are carried out by entities inhabiting the modeled environment leads to consider that the Multi-Agent Systems [7] approach is particularly suited to tackle the modeling issues that are posed by this scenario.

This work is set in a long-term project that provides the realization of a framework supporting the development of MAS based simulation based on the Multilayered Multi-Agent Situated System model [1] provided with an effective form of 3D visualization. The main goal of the framework is to support a smooth transition from the definition of an MMASS based model of a given situation (in terms of environment, relevant entities and their behaviours, expressed as individual actions and interactions) to the realization of simulations systems characterized by an effective 3D user interface.

The agent approach allows representing every pedestrian as an autonomous entity placed in an environment; in particular, in the MMASS approach, special structure of the environment is explicitly described together with specific rules for action and interaction. These agents can move into the space, according to a certain logic choice of destination, they can perceive their environment and react accordingly. They are thus defined reactive. The position in which an agents is situated is important, because it determines the intensity of its perceptions, which are influenced by the notion of distance into the environment. Another fundamental aspect of this model is the concept of field: a signal propagated into space and perceivable by other agents. These basic elements can be adopted to generate most forms of agent behaviour through mechanisms of attraction and repulsion generated by sources of fields. A change in an agent's state can alter the way it interprets these signals. This paper proposes a deliberative extension to the basic model granting the agent the ability to store and elaborate qualitative knowledge about its environment, in order to select those intermediate goals and related actions that will allow it to reach the point of space that contain its own goal or destination.

[1] http://secondlife.com/

2. THE MODELING APPROACH

2.1 The model in brief

A MMASS Agent is defined by the triple <*Space,F,A*> where Space models the environment where the set A of agents is situated, acts autonomously and interacts through the propagation of the set F of fields and through reaction operations (both will be described later on). More precisely *Space* consists of a set *P* of sites arranged in a network (i.e. an undirected graph of sites). The structure of the space can be represented as a neighborhood function, $N: P \rightarrow 2^P$ so that $N(p) \subseteq P$ is the set of sites adjacent to *p*; the previously introduced Space element is thus the pair <*P, N*>. Focusing instead on the single basic environmental elements, a site p∈P can contain at most one agent and is defined by the 3-tuple $<a_p, F_p, P_p>$ where: $a_p \in A_p \cup \{\varnothing\}$ is the agent situated in *p* ($a_p=\varnothing$ when no agent is situated in *p*); $F_p \in F$ is the set of fields active in *p* ($F_p = \varnothing$ when no field is active in *p*); $P_p \subset P$ is the set of sites adjacent to *p* (i.e. *N(p)*).

A MMASS agent is defined by the 3-tuple $<s,p,\tau>$ where τ is the agent type, $s \in \Sigma_\tau$ denotes the agent state and can assume one of the values specified by its type (see below for Σ_τ definition), and $p \in P$ is the site of the Space where the agent is situated. As previously stated, agent type is a specification of agent state, perceptive capabilities and behaviour. In fact an agent type τ is defined by the 3-tuple $<\Sigma_\tau, Perception_\tau, Action_\tau>$. The role of Σ_τ is to define the set of states that agents of type τ can assume.

$Perception_\tau$: $\Sigma_\tau \rightarrow [N \times W_{f1}] \ldots [N \times W_{f|F|}]$ is a function associating to each agent state a vector of pairs representing the receptiveness coefficient and sensitivity thresholds for that kind of field. $Action_\tau$ represents instead the behavioural specification for agents of type τ. Agent behaviour can be specified using a language that defines the following primitives:

- *emit(s,f,p)*: the emit primitive allows an agent to start the diffusion of field f on p, that is the site it is placed on;

- $react(s,a_{p1},a_{p2},\ldots,a_{pn},s')$: this primitive allows the specification of a coordinated change of state among adjacent agents. In order to preserve agents' autonomy, a compatible primitive must be included in the behavioural specification of all the involved agents; moreover when this coordination process takes place, every involved agents may dynamically decide to effectively agree to perform this operation;

- *transport(p,q)*: the transport primitive allows to define agent movement from site *p* to site *q* (that must be adjacent and vacant);

- *trigger(s,s')*: this primitive specifies that an agent must change its state when it senses a particular condition in its local context (i.e. its own site and the adjacent ones); this operation has the same effect of a reaction, but does not require a coordination with other agents.

For every primitive included in the behavioural specification of an agent type specific preconditions must be specified; moreover specific parameters must also be given (e.g. the specific field to be emitted in an emit primitive, or the conditions to identify the destination site in a transport) to precisely define the effect of the action, which was previously briefly described in general terms.

Each MMASS agent is thus provided with a set of sensors that allows its interaction with the environment and other agents. At the same time, agents can constitute the source of given fields acting within a SCA space (e.g. noise emitted by a talking agent). Formally, a field type *t* is defined by $<W_t, Diffusion_t, Compare_t, Compose_t>$ where W_t denotes the set of values that fields of type *t* can assume; $Diffusion_t: P \times W_f \times P \rightarrow W_t$ is the diffusion function of the field computing the value of a field on a given space site taking into account in which site (*P* is the set of sites that constitutes the space) and with which value it has been generated. It must be noted that fields diffuse along the spatial structure of the environment, and more precisely a field diffuses from a source site to the ones that can be reached through arcs as long as its intensity is not voided by the diffusion function. $Compose_t$: $(W_t)^+ \rightarrow W_t$ expresses how fields of the same type are combined (for instance, in order to obtain the unique value of field type *t* at a site), and $Compare_t: W_t \times W_t \rightarrow \{True, False\}$ is the function that compares values of the same field type. This function is used in order to verify whether an agent can perceive a field value by comparing it with the sensitivity threshold after it has been modulated by the receptiveness coefficient.

2.2 Pedestrian Modeling Approach

The basic idea of the modeling approach is that the movement of pedestrians can be generated by means of attraction and repulsion effects [4]. These effects are generated by means of fields that can be emitted by specific point of the environment, and that can be perceived as attractive/repulsive or that can even be simply ignored by different types of moving entities in specific states. Also pedestrians themselves are able to emit fields and thus, in turn, they can generate attraction/repulsion effects, and what is called an 'active walker' model. A thorough discussion of this modeling approach is out of the scope of this paper and it can be found in [2], we will now just give some indications of the main steps that must be followed to define a SCA model starting from an abstract description of a given scenario.

Definition of the spatial infrastructure of the environment – a MMASS space can represent a discrete abstraction of a physical environment, in which a site corresponds to a portion of space that can be occupied by a pedestrian. For instance, a corridor and the rooms having a door on it could be discretized in $40cm^2$ cells characterized by a Von Neumann adjacency.

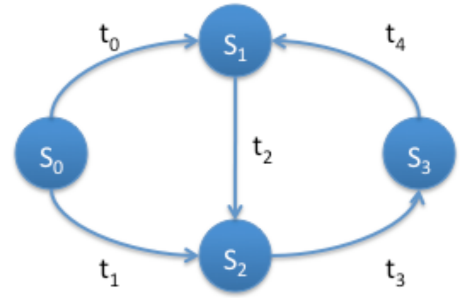

Figure 1 – State diagram of the different agent's behavioral contexts.

Definition of points of interest/reference in the environment – specific spots of the environment can represent elements of interest, reference points or constraints (e.g. gateways, doorways) influencing pedestrian movements. These elements must be associated with immobile agents (e.g. door jambs) able to emit fields indicating the presence of the point of interest/reference to pedestrians. For instance, considering a corridor the exits should be associated to suitable fields able to guide agents towards them, but also possible doorways leading to rooms should be provided with agents emitting proper fields.

Definition of the pedestrian attitude towards space – the way an individual interprets its percepts determines, for instance, whether he/she is going to be attracted by a gateway or completely ignore it, even if he/she perceives it. In the basic MMASS model this part of the behavioural specification of an agent is realized through a utility function associating agent's states to a set of weights determining how to combine the presence of fields in a site to compute its attractiveness. This function is employed by a single transport action, representing the main element of agents' behavioural specification.

In addition to this function, the agent needs to know when to change state (through trigger or react actions): in the basic MMASS model this information must be explicitly defined as part of an agent behavioural specification, through a sort of state diagram, also shown in Figure 1. States represent different attitudes towards the movement in the environment, sorts of *behavioural contexts* for the pedestrian agent.

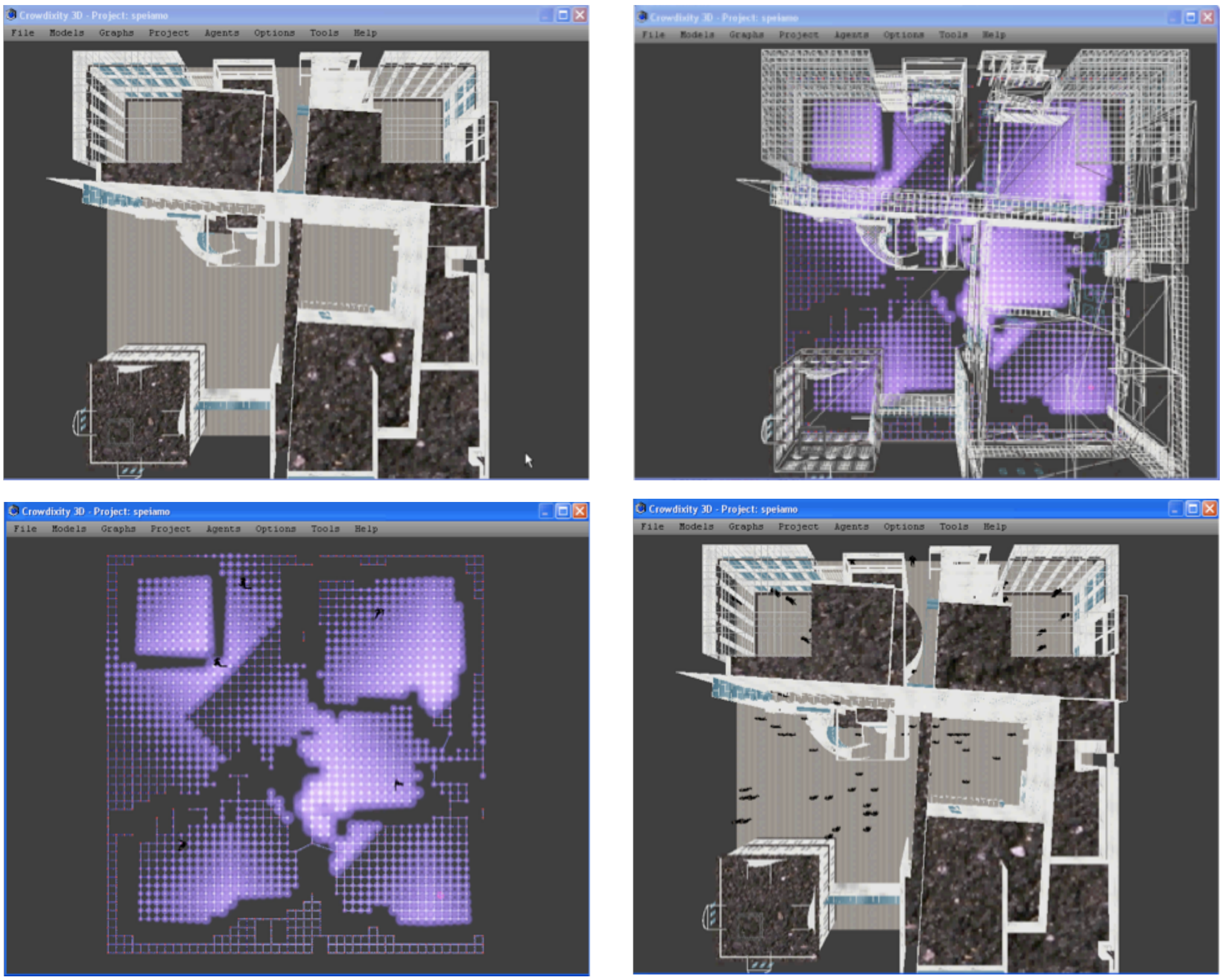

Figure 2 – A sample application of the proposed modeling approach: the museum is discretized, then rooms and gateways are associated to field sources. Agents are provided with an abstract graph representation of the environment and they can construct their plan (set of states to move from an graph of the node to their destination).

3. THE HYBRID AGENT EXTENSION

3.1 Model modifications

The proposed extension to the basic architecture of the model is to endow agents with a qualitative description of the environment they are situated in: essentially agents are provided with a relational structure in which a node is associated to every point of interest in the environment (and thus source of field). Agents are also provided with means to explore this representation in order to construct sort of 'plan' whose nodes are states, connected by transitions (i.e. trigger or react action with the necessary conditions).

A new agent action called *think* has been introduced to initialize agent's behavioural specification. Think represents the action of querying agent's knowledge and update its behavioural specification according to desires. In particular, think produces the conditions that cause a change of behavioural context in the agent; it enables the autonomous generation of a structure substituting the previously defined finite state automata, that must not be defined a priori by the modeler.

More precisely the think action is specified as follows:

action: think($Bh_\tau, K_\tau, D_\tau, f, Bh_\tau^{'}$)
condition: $a = <i_d, b_t, p, \tau>$
effect : $Bh_\tau^{'} = add(((b_1, cond_1, b_2),...,(b_n, cond_n, b_{n+1})), Bh_\tau)$,
$a = <id, b_1, p, \tau>$

Where $a = <id, b_c, p, \tau>$, is the involved agent; the passage to a partially deliberative agent architecture brought to the extension form triple which defined the old reactive agent of the MMASS model to this quadruple where: *id* is the identifier of the agent, b_c represents its current behavioural context that substitutes its state.

Even the definition of the type of the agent was modified. Now each type τ is defined by a six-tuple

$$< K_\tau, D_\tau, P_\tau, B_\tau, Perception_\tau, Action_\tau >$$

where K_τ represents agent's knowledge (i.e. a set of facts), D_τ is the set of desires of the agent (its objectives), B_τ represent the behavioural specification of the agent and it is the set of the behavioural contexts which the agent can take, $Perception_\tau$ is analogous to the original perception function with the only difference of a perception threshold for each behavioural context rather than for each state, and $Action_\tau$ which adds the think action to the previous set of available actions.

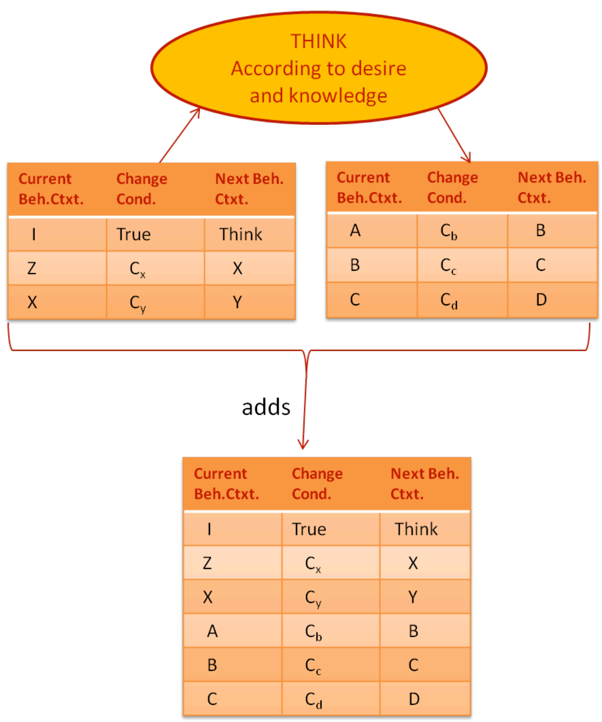

Figure 3 – Sample execution of a think action.

Given this new definition of agent, the above introduced think action will be activated only if the involved the agent is in b_t behavioural context (a conventional initial agent state that leads to the action of thinking before carrying out other actions). The action generates a list of triples *(b_1, cond, b_2)* that represents the transitions between the different behavioural contexts of the agent; this list is created by a think predicate that manipulates agent's knowledge and desires to construct this sort of plan. A sample execution of the think action is shown in Figure 3.

As previously introduced, behavioural contexts act as substitutes of agent's state in the basic MMASS model: this means that, in a given behavioural context, the agent will have a specific attitude towards the interpretation of fields. It will be thus attracted by specific points of reference/interest according to its transport action.

Beside the transport all the actions in the original MMASS agent model had a substantial modification to meet the new introduced features. For example the modeling the achievement of a (possibly intermediate) goal generally requires the adoption of a trigger or react action under specific conditions. The new format of the react action is the following:

action: react(b_1, a_2, b_2)
condition: $a_1 = <id, b_1, p, \tau>$, $a_2 = <id, b_i, q, \tau>$, $q \in P_p$,
$agreed(a_2)$, C_{b2}
effect: $a_1 = <id, b_2, p, \tau>$

Where $a_1 = <id, b_1, p, \tau>$ specifies the agent which wants interact with agent $a_2 = <id, b_i, q, \tau>$, the interaction will take place only if a_2 is situated in a site adjacent to the one in which a_1 ($q \in P_p$) is placed, and if the function agreed gives positive result (that is if the agent a_2 agrees to execute a coordinated change of state with a_1. Note that the list of conditions also includes C_{b2} that is the condition for the passage to the behavioural context b_2 (such condition must be present in the behavioural specification of the agent a_1). The trigger action was modified in an analogous way.

Consider the example in Figure 2: rooms and gateways of the square are associated to both field sources and nodes of the spatial representation. Agents can thus construct their movement plan, in terms of an ordered set of intermediate points of interest leading them from their position to their destination. Each intermediate point is mapped to an agent's behavior context, in which the agent will consider as attractive the fields generated by the related point of interest (thanks to the above introduced utility function).

3.2 Prototypal implementation

As previously introduced, this work is part of larger project aimed at developing a framework supporting an effective form of 3D visualization of the dynamics generated by the model. In particular, for the visualization we adopted Irrlicht[2], a C++ open-source 3D engine. The MAS modeling and development framework is a C++ porting and relevant refactoring of the original MMASS framework, more extensively described in [13].

The classes implementing the basic reactive agents have been integrated with SWI-Prolog to support both the representation of agents' knowledge about the environment and support the planning involved in the think action. In a head-body metaphor [12], agents' bodies (together with their environment and interaction mechanisms) are implemented in C++, while agents' heads have a Prolog implementation.

[2] http://irrlicht.sourceforge.net/

In the simulation every agent is in fact equipped with a set of Prolog facts representing its knowledge and desires; in order to manage the construction of proper triples representing transitions among behavioural contexts, these facts must also include all the foreseen behavioural contexts.

Every different agent type (e.g. employee, tourist in the previous example) will have its own "head", that is, rules to manipulate knowledge in order to implement the above described think action. Agents of the same type, in fact, share the overall behavioural mechanisms but they may have different knowledge of the environment and desires; this leads to the achievement of different results from the application of the think action of different agents. For example two employees having two different job locations will choose different paths to get to work. A sample preliminary implementation of this approach is shown in Figure 4: the left column lists the facts representing agent's desires, knowledge about the environment and the relevant entities, and all its possible behavioural contexts. The right column shows how the think action was implemented for an employee agent; in this specific scenario, it simply constructs a path leading the agent from its own initial position in the scenario (that the agent's body dynamically reads after its creation and initialization), to its own working place inside the museum.

```
% desire of employee
desire(work).

% workplace of employee
work_place(palazzo_marino).

% behavioural context of employee
bh(go_to(X)).
bh(work_to(X)).

go_to(X).
work_to(X):- work_place(X).

% place
place(palazzo_marino).
place(piazza_nord).
place(piazza_est).
place(piazza_sud).
place(scala).
place(banca).

edge(palazzo_marino,piazza_nord).
edge(piazza_nord,piazza_est).
edge(piazza_nord,piazza_sud).
edge(piazza_sud,piazza_est).
edge(piazza_est,uscita_est).
edge(piazza_sud,scala).
edge(piazza_sud, uscita_ovest).
edge(piazza_est, banca).
```

```
adjacent(X, Y) :- edge(X, Y); edge(Y, X).

% [Y] list of already visited nodes
path(X, Y, P) :- path1(X, Y, [perc(Y), go_to(Y), go_to(Y),
   react(Y), work(Y)], P).

path1(X, Y, L, [go_to(X), perc(Y) |L]) :- adjacent(X, Y).
path1(X, Y, L, P) :-
   adjacent(Z, Y),
   Z \== X,
   not(member(Z, L)),
   path1(X, Z, [ go_to(Z), go_to(Z)|L], P).

tran([X,Y,Z|W], [transition(X,Y,Z)|P]):- tran(W,P).
tran([],[]).

think(Work_place,Y,P):- work_place(Work_place),
   path(Y,Work_place,X), tran(X, P).
```

Figure 4 – Sample Prolog implementation of employee agents' knowledge-base and head.

4. Conclusions

This paper introduced a hybrid extension of MMASS agent model. The approach is based on a specialization of agent's state into more complex concepts and structures like knowledge and desires; a new action was introduced to exploit this knowledge-base to autonom ously construct a behavioural specification based on fundamental building blocks (behavioural contexts) representing tendencies, attitudes towards movement of agents.

This approach lead to the extension of an existing platform for the implementation of MMASS based models. The extended platform has been tested in small case scenarios as well as in larger environments (such as the one presented in the example). The introduced extension allowed for a much more compact definition of agents' behavioural specifications, and the resulting agents have a higher degree of autonomy. Preliminary benchmarks (with a number of agents ranging from few tens to few hundreds) shown that the introduction of a deliberative extension to the agent architecture does not cause dramatic overhead in agents' execution time.

Future work are aimed at a more general definition of the mechanisms specifying the interaction between head and body, specifically aimed at supporting the management of multiple agent's desires. To this aim, it will be necessary to consider and evaluate a significant set of existing models, either deliberative [14] and hybrid [15]. In particular, the details of the deliberation process, the balance between deliberation and action, the detection of potential problems in the execution of an agent's behavioural specification and the opportunity to reconsider it based on the perception of changed environment conditions, reflecting on a change in agent's knowledge base, are object of several existing works [16][17].

5. REFERENCES

[1] Stefania Bandini, Sara Manzoni, and Carla Simone. Heterogeneous Agents Situated in Heterogeneous Spaces. Applied Artificial Intelligence, 16(9-10):831–852, 2002

[2] Stefania Bandini, Mizar Luca Federici, Giuseppe Vizzari. Situated Cellular Agents Approach to Crowd Modeling and Simulation. Cybernetics and Systems 38(7):729-753, 2007

[3] Michael Batty, Andrew Hudson-Smith (2005). Urban Simulacra: From Real to Virtual Cities, Back and Beyond, Architectural Design, 75 (6):42-47

[4] Michael Batty. Agent-based pedestrian modeling, in Advanced Spatial Analysis: The CASA Book of GIS, pp. 81-106, 2003

[5] Dijkstra, J., J. Van Leeuwen, and H. J. P. Timmermans, (2003). Evaluating Design Alternatives Using Conjoint Experiments in Virtual Reality. Environment and Planning B 30 (3), 357–367

[6] Dijkstra, Jan and Timmermans, Harry. 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

[7] Ferber, J. (1999). Multi-Agent Systems. Addison-Wesley

[8] Toru Ishida, Yuu Nakajima, Yohei Murakami, Hideyuki Nakanishi: Augmented Experiment: Participatory Design with Multiagent Simulation. IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence (2007):1341-1346

[9] Hideyuki Nakanishi, Satoshi Nakazawa, Toru Ishida, Katsuya Takanashi, and Katherine Isbister. Can Software Agents Influence Human Relations? - Balance Theory in Agent-mediated Communities. AAMAS-2003, pp. 717-724, 2003

[10] Fabrizio Nunnari, Carla Simone: Perceiving awareness information through 3D representations. Proceedings of the working conference on Advanced visual interfaces, AVI 2004. ACM Press (2004): 443-446

[11] George Papagiannakis, Sébastien Schertenleib, Brian O'Kennedy, Marlene Arevalo-Poizat, Nadia Magnenat-Thalmann, Andrew J. Stoddart, Daniel Thalmann: Mixing virtual and real scenes in the site of ancient Pompeii. Journal of Visualization and Computer Animation 16(1): 11-24 (2005)

[12] Steiner, D.E., Haugeneder, H., Mahling, D.: Collaboration of knowledge bases via knowledge based collaboration. In: International Working Conference on Cooperating Knowledge Based Systems, Springer-Verlag (1991) 113–133

[13] Giuseppe Vizzari, Giorgio Pizzi, Flávio S. Corrêa da Silva: A framework for execution and 3D visualization of situated cellular agent based crowd simulations. SAC 2008: 18-22

[14] Anand S. Rao, Michael P. Georgeff: BDI Agents: From Theory to Practice. Proceedings of the First International Conference on Multiagent Systems, June 12-14, 1995, San Francisco, California, USA. The MIT Press 1995, pp. 312-319

[15] Zahia Guessoum: A hybrid agent model: a reactive and cognitive behavior. Third International Symposium on Autonomous Decentralized Systems, 1997(ISADS 97), pp.25-32, 1997

[16] Mehdi Dastani, Frank Dignum, John-Jules Ch. Meyer: Autonomy and Agent Deliberation. Agents and Computational Autonomy - Potential, Risks, and Solutions - Postproceedings of the 1st International Workshop on Computational Autonomy - Potential, Risks, Solutions (AUTONOMY 2003), Lecture Notes in Computer Science 2969 Springer 2004 2003:114-127

[17] William J. Clancey, Patricia Sachs, Maarten Sierhuis, Ron van Hoof: Brahms: simulating practice for work systems design. Int. J. Hum.-Comput. Stud. 49(6): 831-865 (1998)