First workshop on Intelligent Agents in Urban Simulation and Smart Cities

in the context of the 20th

EUROPEAN CONFERENCE ON ARTIFICIAL INTELLIGENCE

ECAI 2012

Montpellier, France

August 28, 2012

Organizers:

Vincent Corruble

Fabio Carrera

Stephen Guerin

Homepage: *http://www-poleia.lip6.fr/~corruble/IAUSSCws/Home.html*

ECAI-2012 workshop on Intelligent Agents in Urban Simulation and Smart Cities

.

1 RATIONALE

As 3D modelling and graphics have made rapid progress over the last decade, new avenues have opened that make use of detailed and often realistic virtual models of real or imaginary cities. These models provide tremendous support to the study of various issues, many of them dealing with key social and political topics, such as urban planning, security or transport, or as an exceptionally rich environment, for entertainment, education or training applications (such as video-games or serious games). However, while visual realism is important to immersion, a key element for the credibility of virtual cities is the realism of its virtual inhabitants, the agents that populate its virtual world. Going from visual to behavorial credibility poses direct challenges to the agent and multi-agent community as cities provide potentially unlimited settings for urban agents to display their ability to react, act proactively, interact between themselves, or otherwise plan, learn, etc. in an intelligent, or rather human, manner.

Most of the issues typically raised within the Artificial Intelligence community can find an echo within urban simulations. These simulators can be seen as testbeds, i.e. rich environments in which to deploy AI technologies. Further, new applications coming from that area constitute new challenges to the AI community. The application domain of city simulations is attracting growing attention from various communities and could become a “killer application” for AI technologies. It is important for the community to tackle this domain head on.

In this workshop, we intend to address specific methodological and technological issues raised by the deployment of agents in rich environments such as virtual cities. We welcome contributions tackling issues related to reactive agents, cognitive architectures, the capacity to scale up to handle thousands or hundreds of thousands of agents, the ability to simulate realistic group behaviors which might be judged rational or non rational, etc., all in the context of urban agents, so that other orthogonal issues need to be kept in mind, such as the generality of the solutions proposed, the ease of use (parameterizing, interaction) by the end-user of the simulation, the ability to guarantee some level of realism for the simulation. We will also welcome contributions showcasing original applications of agent and multi-agent technologies within urban simulation, be it for design, planning, education, training, or entertainment.

2 PROGRAMME

The first IAUSSC workshop received a good number of submissions that were reviewed by the programme committee constituted of researchers active in the fields of Intelligent Agents and/or of Urban Simulation. Submissions were selected as long papers, short papers or as demo presentations, to produce a rich and diverse programme. The list of contributions is given on the next page.

3 INVITED TALKS

We are honoured to have two distinguished researchers as guest speakers:

1. **Stuart Russell**, from UC Berkeley – and visiting faculty at Université Pierre et Marie Curie, is to give a talk entitled *AI and the City*.
   Abstract: AI may supply the "smart" in the "smart city" and in the individual agents that populate city simulations. In either role, the challenges for AI are daunting, in terms of both scale and technical difficulty. Decisions and behaviors involve astronomical action spaces and long time horizons, while issues such as partial observability, relational uncertainty, and multiagent interactions are central to understanding how cities work. Some recent results in AI may be relevant, but much remains to be done.

2. **Joan Serras**, from CASA, University College London, is to give a talk entitled *Sensing and modelling the city*
   Abstract: By 2050, 70% of the world population will live in cities. This level of urbanisation implies massive challenges in congestion, segregation, demographics and resource use, to name just a few. However, the growth of 'big data', combined with significant technological advances -- particularly in the mobile industry -- offers new means of getting to grips with these challenges through modelling. Tools rooted in these sources allow us to start exploring new ways to understand and interact with cities in order to use them more efficiently. In this talk I will present the work done in this direction from several CASA projects with a focus on the field of transport.

4 WORKSHOP ORGANISATION

4.1 Worskhop chairs

•**Vincent Corruble** (contact) vincent.corruble@lip6.fr
LIP6, Université Pierre et Marie Curie (Paris 6),
France

•**Fabio Carrera**
Worcester Polytechnic Institute (WPI), USA

•**Stephen Guerin**
Santa Fe Complex, USA

4.2 Programme Committee

•Michael Batty, University College London, UK

•Andrew Crooks, George Mason University, USA

•Alexis Drogoul, IRD, Vietnam

•Amal El Fallah Seghrouchni, Université Pierre et Marie Curie, France

•Hiromitsu Hattori, Kyoto University, Japan

•Noda Itsuki, IAIST, Japan

•Geber Ramalho, UFPE, Brazil

•Stuart Russell, University of California, Berkeley, USA

•Paul Torrens, University of Maryland, USA

•Jean-Daniel Zucker, IRD, Vietnam

5 CONTENT

Using Membranes to Model a Multi-Agent System towards Underground Metro Station Crowd Behaviour Simulation

Ilias Sakellariou[1] and **Ioanna Stamatopoulou**[2] and **Petros Kefalas**[3]

Abstract. Simulation of crowd behaviours has long been an area of active research due to its high impact on urban area design. Rather recently, the problem has been tackled using agent based simulation (ABMS), a modelling approach that offers a more natural and flexible method for describing pedestrian behaviour. The current work, concerns ABMS modelling of passengers boarding an underground station using Population P Systems (PPS), a membrane computing paradigm that allows formal definition of an agent based system. It is argued that the expressive power of PPS allows modelling the simulation under study, and an implementation of the model described is provided in a well known agent simulation platform to further prove its validity.

1 INTRODUCTION

Crowd behaviour simulation has long been recognised as a valuable tool for evaluation of urban area design. Such simulations are considered more complex than those involving vehicle traffic mainly due to the fact that pedestrian navigational patterns are more freely defined and individuals are involved in more complex interactions both with the environment and other pedestrians. This fact led naturally to the introduction of agent based modelling (ABM) [4] along with other techniques discussed in a later section (section 2). The former supports a number of essential characteristics for the task, such as ability to host heterogeneous individuals in the simulation and emergence of complex behavioural patterns.

P Systems [19] is a membrane computing paradigm that is inspired by how living cells can perform computation. A P System is composed of a hierarchical structure of membranes that can transform objects through evolution rules that reside within them, and also exchange objects between them through communication rules. Membranes can dissolve or divide thus changing the configuration of the system during computation. A class of P Systems, Population P Systems (PPS) consist of cells within a bigger membrane (the environment). The cells are configured in a graph rather than a hierarchical tree structure and are connected through bonds under specific dynamic conditions. In PPS with active cells [2], cells can die or divide thus giving important features to PPS as a formal modelling language.

[1] University of Macedonia, 156 Egnatia Str, Thessaloniki 54006, Greece, email:iliass@uom.gr
[2] The University of Sheffield International Faculty, CITY College 3 Leontos Sofou Str., Thessaloniki 54626, Greece, email:istamatopoulou@city.academic.gr
[3] The University of Sheffield International Faculty, CITY College 3 Leontos Sofou Str., Thessaloniki 54626, Greece, email: kefalas@city.academic.gr

Prominent characteristics of PPS include:

- Non-deterministic communication between cells;
- Dynamic addition and removal of cells;
- Dynamic restructuring of the communications network; and finally
- Maximal and arbitrary parallelism, i.e. the ability to support selection of a *maximal*/*arbitrary* number of evolution rules with non-deterministic selection of the rule that applies at each computation step.

These features are particularly useful in MAS and this is the reason why we suggest that PPS with active cells is one of the most suitable formalisms for modelling the macro-level of biologically inspired MAS of a highly dynamic nature [13].

In this paper, we present how PPS with active cells, facilitates formal modelling of MAS targeted to crowd simulation. The aim is to demonstrate that PPS is a natural choice for this domain for developers who wish to formalise their system before proceeding to the implementation or simulation. Formalisatation is necessary when properties of the system at hand need to be proved. PPS provides an elegant way to describe simple rules for agent movement in space, multi-agent dynamic structure and organisation as well as exchange of messages. Such simple mathematical notation can be refined towards simulation and visualisation through an appropriate tool, in this case NetLogo.

The rest of the paper is structured as follows: Section 2 provides an overview of common approaches to crowd simulation; Section 3 presents the formal definition of PPS with active cells while Section 4 discusses the reasons why they are appropriate for MAS modelling; Section 5 presents the case study description together with indicative parts of the the PPS model, and Section 6 the derived NetLogo simulation; Section 7, finally, concludes the paper an discusses directions of further work.

2 CROWD SIMULATION

Pedestrian simulation has long been the topic of active research, due to its importance to urban design, as for instance evaluating a building design in case of an evacuation. There is a vast number of research results reported in the literature, that follow different approaches with respect to the method used and most importantly the granularity of the simulation, also referred to as scale.

A lot of research deals with the problem following a "macroscopic" approach, i.e. it uses for example *flow dynamics* as in [11] and [10] in order to assess designs and simulate behavioural patterns of pedestrians moving in an area of study. These models describe crowd behaviour as a set of partial differential equations and are most useful in densely populated environments.

Social force models, originally introduced in [9], consider pedestrians exposed to "force field" generated by its own intentions, other pedestrians and points of attraction. Models in this category have been thoroughly studied and in some cases combined with other approaches, as for instance agent based [14], to successfully model crowd behaviour. Social force models adopt the "microscopic" approach to crowd simulation, according to which the overall behaviour is emerging as the result of interactions of individual entities.

Cellular automata [3],[21] also fall under the same category, and describe the world as a grid of cells while behaviour is determined by a set of simple local rules that update the state of each cell, based on the state of its neighbouring cells. Gas lattice models consider pedestrians to be "particles" moving on a grid with a set of probability rules determining their next position on that grid, as for instance in [7] and [8].

Finally, *agent based modelling* (ABM) has also been employed to tackle the problem. ABM has a number of advantages [4] in the sense that it allows emergent phenomena to manifest, it is flexible and allows a natural description of the model. For instance in [6] ABM are used to model pedestrian behaviour and cellular automata to model the environment. A similar approach is used in [22] to assess design choices in cases of emergency evacuation in a metro station. Bandini et al., employ the situated cellular agent model [1] to model pedestrians in an underground train station that exhibit both cooperative and competitive behaviour, i.e passengers boarding and descending the train wagon.

The list of references provided above is by no means complete; there is a significant amount of research work and tools addressing the problem and the reader should refer to [17] and [23] for a more in depth review and assessment of models used in crowd simulation.

3 POPULATION P SYSTEMS WITH ACTIVE CELLS

3.1 Formal Definition of PPS with active cells

A Population P System with Active Cells [2] is defined as:
$PPS = (V,\ K,\ C_1,\ C_2,\ \ldots,\ C_n,\ w_E,\ \gamma,\ \alpha,\ R,\ O)$ such that:

- V is the set of all objects within the system, including those of the environment;
- K is the set of the different types of cells in the system (each cell has an associated class/type);
- $C_i, 1 \leq i \leq n$ are the n cells in the system;
- w_E is a multiset over V containing the objects initially assigned to the environment;
- $\gamma = (N, A)$ is the undirected graph representing the initial structure of the system, where: $N = \{1, 2, \ldots n\}$, and $A \subseteq N \times N$;
- α is a finite set of bond making rules;
- $R = R_e \cup R_c$ is the set of all evolution and communication rules;
- O is a partial order over the set of all rules R.

3.2 Objects in Cells and Environment

In the above definition, an object has its normal mathematical semantics, i.e. a symbol. For the purposes of this work however an object is represented as an $attribute\ :\ value$ pair and therefore $V = \{(a\ :\ v) \mid a \in Attributes,\ v \in D\}$ where $Attributes$ is a set of attributes/labels and D stands for the *domain set* of a. In PPS that deal with spatial properties, there is also a special kind of object $(\Pi_i\ :\ \pi_i) \in V$ for each cell, denoting the ith cell's position, with values $\pi_i \in \mathbb{N}_0 \times \mathbb{N}_0$ and a label Π_i that stands for *position*. In [18], a special class of PPS is defined, namely Spatial PPS or ^{sp}PPS.

3.3 Cells and Types

A cell is defined as a tuple $C = (w,\ t)$ where:

- w is the multiset of objects over V that are contained by the cell, and
- $t \in K$ is the type of the cell.

Practically, this means that cells are classified in different types containing different objects and different evolution and communication rules.

3.4 Bonds between Cells

The bond making rules in α are of the form $(t, (x\ :\ a); (y\ :\ b), p)$, where $(x : a), (y : b) \in V$ and $t, p \in K$. This rules states that in the presence of $(x\ :\ a)$ in a cell of type t and $(y\ :\ b)$ in a cell of type p, the two cells are joined, i.e. they can exchange objects. Bond making rules are responsible for constructing the undirected graph γ, which represents the communication structure of the P System.

3.5 Evolution Rules

R_e is the finite set of cell evolution rules, that is, rewrite rules which given the presence of certain object transform a cell by introducing new objects or communicating objects to other cells or transforming the type of a cell etc. Evolution rules determine the computation of the system and are of various types (in the following t and p are the types of the cell that these rules refer to):

- Transformation rules: $((x\ :\ a) \rightarrow (y\ :\ b))_t$, that is, an object $(x : a)$ becomes $(y : b)$;
- Differentiation rules: $((x\ :\ a))_t \rightarrow ((y\ :\ b))_p$, that is, in the presence of an object $(x : a)$ the cell of type t is transformed into a cell of type p containing the object $(y : b)$;
- Division rules: $((x\ :\ a))_t \rightarrow ((y\ :\ b))_t((z\ :\ c))_t$, that is, in the presence of an object $(x\ :\ a)$ the cell is divided into two cells of the same type. The two cells contain the objects $(y : b)$ and $(z : c)$ respectively;
- Death rules: $((x\ :\ a))_t \rightarrow \dagger$, that is, in the presence of an object $(x : a)$ the cell disappears from the system.

Evolution rules are shown diagrammatically in Fig. 1. The outer membrane in each diagram represents the environment whereas the inner membranes are the cells.

3.6 Communication Rules

Communication rules are responsible for importing/exporting objects from/to the environment as well as importing/exporting objects from/to other cells. Communication rules R_c are of the form (in the following t is the type of the cell that these rules refer to):

- $((x\ :\ a); (y\ :\ b), enter)_t$, that is, in the presence of an object $(x : a)$ the object $(y : b)$ is imported from the environment;
- $((x : a); (y : b), enter_{copy})_t$ that is, in the presence of an object $(x : a)$ a copy of the object $(y : b)$ is imported from the environment;

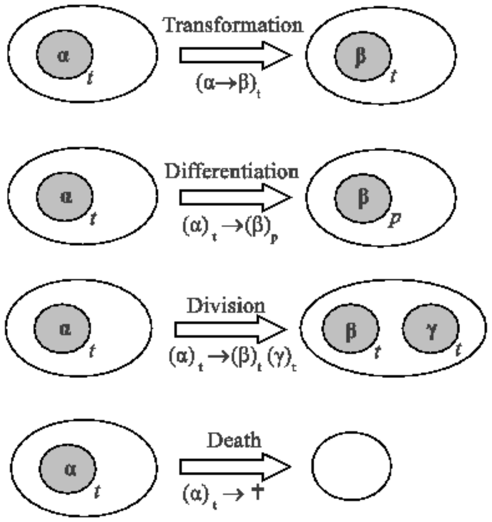

Figure 1. Evolution Rules in Population P systems with active cells.

- $((x : a); (y : b), in)_t$, that is, in the presence of an object $(x : a)$ the object $(y : b)$ is imported non-deterministically from a neighboring cell;
- $((x : a), exit)_t$, that is, when present, the object $(x : a)$ is exported to the environment.

Communication rules are also shown diagrammatically in Fig. 2.

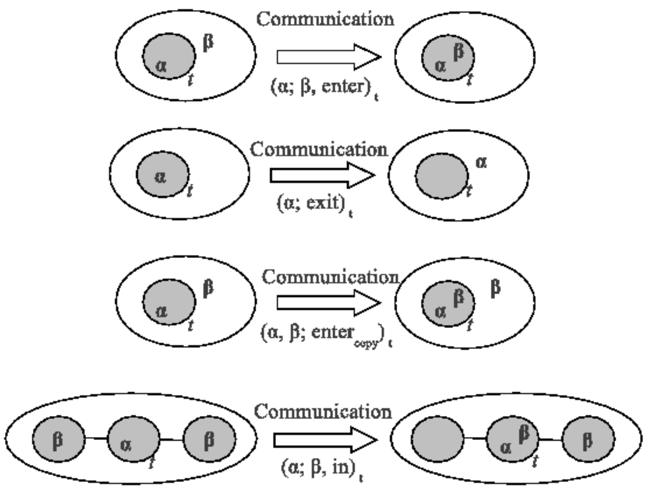

Figure 2. Communication Rules in Population P systems with active cells.

3.7 Computation

The computations step in a cell of a PPS consists of firstly applying in a non-deterministic order all the evolution rules that are triggered, and afterwards applying in a non-deterministic order the triggered communication rules. Since, however, this does not always serve the purposes of ABM, the partial order *O* augments the model, imposing a priority to the rules so that they applied in an order that fits the model at hand, e.g. $...r_i \succeq r_j....$ The computation in each cell is performed independently of other cells, as implied by the maximal and arbitrary parallelism characteristic of PPS.

4 POPULATION P SYSTEMS FOR MODELLING OF MAS

Formal methods such as PPS are particularly suited to modelling MAS [12]. Individual agents:

- perceive their environment by receiving stimuli as input;
- receive messages from other agents;
- revise their beliefs based on both the percepts as well as the information encoded in the received messages;
- react based on a specific set individual behavioural rules;
- engage in a deliberation process to decide on the next action;
- compile and send messages to other agents;
- act, with the effects of their action appearing in the environment.

Not all the above are present in every agent. For example, reactive agents do not deliberate, while "smarter" proactive agents do. Also, communication between simple biological agents is rather primitive and mostly done through the environment, in contrast to more elaborated direct communication that may follow a strict protocol. Therefore, in order to create a model of an agent, one would require non-trivial data structures, means of encoding rules and behaviour, representation of the internal state of the agent, etc. It is evident that PPS can provide such attributes in an agent (cell) through the set of objects and the set of evolution rules.

In a MAS, each agent operates in parallel with other agents, and a mode of interaction determines the way in which agents exchange messages. Agents have roles and are organised in a way that the communication flow is facilitated. The features of PPS are well suited for all the above; maximal/arbitrary parallelism, types of cells and bond making rules.

At MAS level, formal modelling would require ways to deal with exchange of messages between agents, a method for expressing the asynchronous computation of individuals, the addition and removal of agent instances "on the fly" and means for structuring and restructuring the organisation "on the fly" (structure mutation). Again, it is evident that PPS can provide such attributes for modelling MAS through communication rules, division and death rules as well as bond making rules.

Finally, in the current context of crowd behaviour, a MAS also requires special handling of spatial characteristics of agents, something that PPS can neatly provide. To summarise, PPS, through their characteristics, seems to offer an interesting and elegant way to formally model the features of spatial MAS such as those met in crowd behaviour.

5 CASE STUDY: METRO STATION CROWD BEHAVIOUR

The case study employed to demonstrate the application of the PPS approach concerns the simulation of underground metro station, as the latter is discussed in [1]. The scenario concerns a common everyday situation where passengers enter an underground station and try to board a wagon. The experimental setting includes an entrance area, where passengers initially entering the station are located, the platform, the wagon, the door area, an intermediate target for the passengers to reach before completing their boarding, and the seats (Fig. 3). Briefly, each passenger:

- Upon entering the station, selects one wagon door and moves towards it.
- The passenger then waits in the platform area for the doors to open.
- When the doors open the passenger moves to the door area in order to board the wagon.
- The passenger completes the boarding and is now located in the wagon.
- If the passengers perceives an empty seat they move towards it to get seated.

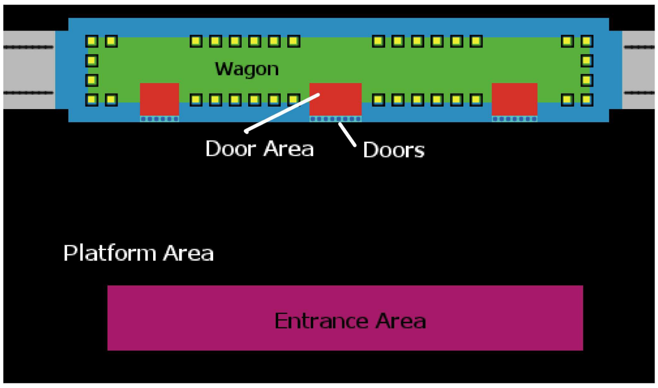

Figure 3. Areas in the underground simulation scenario.

For the experimental setting described above, we consider space to be discrete, divided to a grid, where each grid position can accommodate a single passenger. The latter move from a position to the next instantly. This is in accordance with almost all discrete models of pedestrian simulation reported in section 2. All the assumptions stated here are going to be reflected in the PPS model described in the following.

For the purposes of modelling the given scenario three distinct cell types are required. Cells of the first type $p \in K$ will be used to model the individual passengers. Each passenger cell should include objects that correspond to the area it is positioned, and beliefs, such as:

$(area : platformWaiting)(\Pi : (x, y))(goal : door_1)$

indicating that the particular passenger is currently waiting at the platform, in a particular position (x, y) with the goal of going towards the first door when the doors open.

Given a passenger in the above state, for them to perceive the doors have opened we need a communication rule that will retrieve the information (object $(doors : open)$) from the environment:

$((area : platformWaiting); (doors : open), enter_{copy})_p$

For the agent to move towards its intended direction, i.e. $door_1$ as specified by the $goal$ object, the neighbouring environment needs to be perceived for empty positions:

$((\Pi : (x, y))(goal : door_1);$
$(door_1 : (nx, ny))(empty : (nx, ny)), enter)_p$

such that (nx, ny) is a the neighbouring position that is closer or belonging to $door_1$.

After an empty position has been perceived, the passenger moves to this position (transformation rule), updating its environment to inform that its previous position is now empty (communication rule):

$((\Pi : (x, y))(empty : (nx, ny)) \rightarrow (empty : (x, y))(\Pi : (nx, ny)))_p$
$((empty : (x, y)), exit)_p$

Perceiving that it has arrived to the door area (communication rule), the agent updates its area (transformation rule):

$((\Pi : (x, y))(goal : door_1); (door_1 : (x, y)), enter_{copy})_p$
$((\Pi : (x, y))(area : platformWaiting)(goal : door_1)(door_1 : (x, y))$
$\rightarrow (\Pi : (x, y))(area : door_1)(goal : wagon))_p$

Passengers are removed from the system when they are in the entrance area exiting the station:

$((area : entranceExiting)(\Pi : (x, y))(entrance : (x, y)) \rightarrow \dagger)_p$

A second type of cell is required to model the arrival of new passengers at the entrance of the platform. The idea is that we have a generator cell (type $g \in K$) positioned at the entrance, that only contains one $divide$ object and two division rules. One rule will actually generate a new passenger at the entrance area whereas the other will be a "fake" division rule that does not generate a passenger. In this manner one of the two rules will be non-deterministically applied in each computation step possibly generating a new passenger. The two rules are:

$((divide) \rightarrow (divide)(\varepsilon))_g$, and
$((divide) \rightarrow (divide)$
$((state : entrance)(goal : dock)(\Pi : (x, y)))_p)_g$

where (x, y) is any position such that in the environment there are the objects $(entrance : (x, y))$ and $(empty : (x, y))$.

Finally, a third type of cells is required, to be responsible for the opening and closing of doors (type $d \in K$). One cell of the given type is necessary so that at specific intervals it expels to the environment the $(doors : open)$ and $(doors : close)$ objects with the use of two communication rules. To model the time intervals a transformation rule is used, which produces a copy of a $tick$ object at every step. Assuming, for example that doors have to open in 30 time units the rules would be:

$(tick \rightarrow tick^2)_d$, increasing the time (superscript denotes the multiplicity of the object in the multiset)

$(tick^{30} \rightarrow tick(doors : open))_d$, generating the $(doors : open)$ object and reseting the clock , and

$((doors : open); (doors : open), exit)_d$, for expelling to the environment so that the passengers are informed. A similar process takes place for closing the doors.

6 FROM FORMAL MODELLING TO SIMULATION

In the recent years a large number of agent simulation platforms have been proposed in the literature [15], differing in the programming language used, development environment, documentation, etc. Probably, Repast [16] and NetLogo [20] are the most widely adopted, and better supported platforms with a large community of active users. Out of the two, NetLogo was selected as the platform of choice for implementing the PPS model of the underground station, mainly due to its unparalleled simplicity in developing and maintaining simulation experiments.

NetLogo is an agent simulation platform, aiming at the simulation of complex social and biological phenomena, involving a large number of agents. The platform offers a complete simulation development environment, supporting the creation of a large number of agents (turtles) "living" in a 2D or 3D grid (patches), whose behaviour is controlled by a domain specific language. The platform

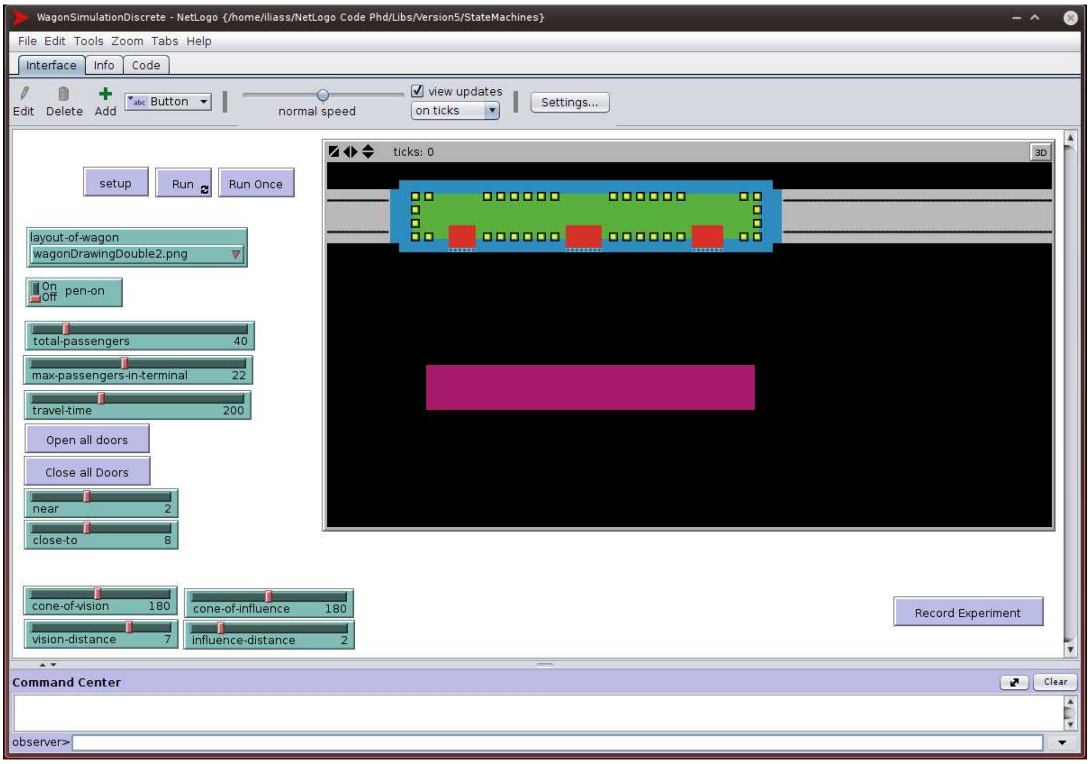

Figure 4. The underground station simulation scenario in NetLogo. Note that a number of controls allows modification of experiment parameters.

also supports easy GUI creation, thus greatly facilitating the development of a simulation experiment. It should be noted that it has been used as a tool for ABMS in a large number of cases, and in pedestrian simulation as well [5], [24]. Figure 4 depicts the NetLogo simulation environment of the case study of the paper.

Supporting direct execution of PPS defined agents in NetLogo presents a number of problems. Rules, as described in section 5, require support for one-way pattern matching (rewrite rules), a feature that is difficult to support in the functional-like domain specific language of NetLogo. Additionally, the declarative nature of the agent rule base presents a few difficulties when encoded in this programming paradigm. Therefore, we have chosen to implement the agent "PPS engine" in an NetLogo[4] extension, NetProlog, that allows execution of Prolog code from the simulation platform.

This greatly facilitated the development of our agent model, since after the initialisation of the experiment, all relevant objects regarding the environment and the agents are asserted as Prolog facts, in the NetProlog engine. During simulation execution each agent runs the engine and updates its position and state according to the results of the execution.

Figure 5 presents a snapshot of the simulation showing agents boarding the metro wagon. Although in the figure all passengers are shown white, in the actual simulation, they are color coded depending on their state (seated, boarding etc.). In Fig. 6 all passengers have successfully boarded the metro wagon and some have managed to find a place to seat.

Although the NetProlog extension offers an easy and elegant way to implement the PPS model, we are considering the option of auto-

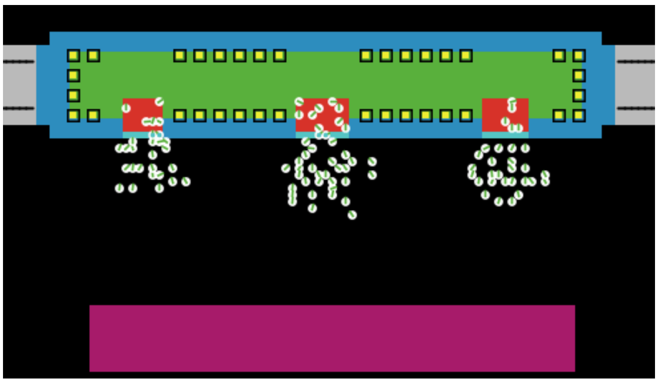

Figure 5. Passengers (shown in white) Boarding the Wagon.

[4] NetProlog is available at http://www.cs.us.es/~fsancho/NetProLogo/

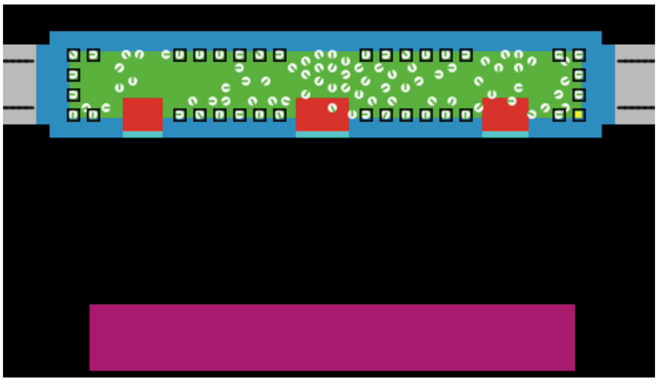

Figure 6. All Passengers on Board. Note that some passengers are seated, occupying the corresponding space in the wagon.

matically compiling rules to the NetLogo programming language, mainly for efficiency reasons, since it will reduce the overhead of updating the Prolog interpreter in each cycle.

7 CONCLUSIONS AND FURTHER WORK

With this work we aim at demonstrating how PPS with active cells is a suitable formalism for modelling MAS for the purposes of simulating crowd behaviour. We have used metro station boarding as an example to illustrate that PPS provide the necessary constructs both for modelling individual agents, with the use of objects that represent the agent's beliefs and evolution rules that update these beliefs, as well as for dealing with the structure reconfigurations that take place in such dynamic systems, with the use of bond-making, division and death rules. Additionally, although not explicitly demonstrated in the presented case study, PPS also allow the communication between agents through objects' exchange.

The initial NetLogo simulation described in this paper, acts as a proof of concept in using the PPS formalism to such a complex scenario. We are considering further developing tools and techniques for automatic translation of PPS specifications to NetLogo and possibly other platforms such as Repast, considering further issues such as simulation scalability, possibly through parallel/distributed execution. Additionally, we are considering the development of more complex pedestrian behaviours, as for example in panic situations in evacuation scenarios, where object exchange mentioned above could provide a flexible modelling approach.

REFERENCES

[1] S. Bandini, M.L. Federici, and G. Vizzari, 'Situated cellular agents approach to crowd modeling and simulation', *Cybernetics and Systems*, **38**(7), 729–753, (2007).

[2] F. Bernardini and M. Gheorghe, 'Population P Systems', *Journal of Universal Computer Science*, **10**(5), 509–539, (2004).

[3] V.J. Blue and J.L. Adler, 'Cellular automata microsimulation for modeling bi-directional pedestrian walkways', *Transportation Research Part B: Methodological*, **35**(3), 293–312, (2001).

[4] E. Bonabeau, 'Agent-based modeling: Methods and techniques for simulating human systems', *Proceedings of the National Academy of Sciences of the United States of America*, **99**(Suppl 3), 7280–7287, (2002).

[5] F. Camillen, S. Capri, C. Garofalo, M. Ignaccolo, G. Inturri, A. Pluchino, A. Rapisarda, and S. Tudisco, 'Multi agent simulation of pedestrian behavior in closed spatial environments', in *Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference*, pp. 375–380, (sept. 2009).

[6] J. Dijkstra and H. Timmermans, 'Towards a multi-agent model for visualizing simulated user behavior to support the assessment of design performance', *Automation in Construction*, **11**(2), 135–145, (2002).

[7] M. Fukamachi and T. Nagatani, 'Sidle effect on pedestrian counter flow', *Physica A: Statistical Mechanics and its Applications*, **377**(1), 269–278, (2007).

[8] R.Y. Guo and H. J. Huang, 'A mobile lattice gas model for simulating pedestrian evacuation', *Physica A: Statistical Mechanics and its Applications*, **387**(2-3), 580–586, (Jan 2008).

[9] D. Helbing and P. Molnar, 'Social force model for pedestrian dynamics', *Physical Review E*, **51**, 4282, (1995).

[10] L. Huang, S. C. Wong, M. Zhang, C.W. Shu, and W.H.K. Lam, 'Revisiting Hughes' dynamic continuum model for pedestrian flow and the development of an efficient solution algorithm', *Transportation Research Part B: Methodological*, **43**(1), 127–141, (2009).

[11] R. L. Hughes, 'A continuum theory for the flow of pedestrians', *Transportation Research Part B: Methodological*, **36**(6), 507–535, (2002).

[12] P. Kefalas and I. Stamatopoulou, 'Modelling multi-agent systems: Experiences with membrane computing and future challenges', in *Proceedings of the 1st International Workshop on Application of Membrane Computing, Concurrency and Agent-based Modelling in Population Biology (AMCA-POP 2010)*, pp. 71–83, (2010).

[13] P. Kefalas and I. Stamatopoulou, 'Towards modelling of reactive, goal-oriented and hybrid intelligent agents using P systems', in *Proceedings of the 11th International Conference on Membrane Computing*, pp. 265–272, Berlin, Heidelberg, (2010). Springer-Verlag.

[14] Q. Lin, Q. Ji, and S. Gong, 'A crowd evacuation system in emergency situation based on dynamics model', in *Interactive Technologies and Sociotechnical Systems*, eds., Hongbin Zha, Zhigeng Pan, Hal Thwaites, Alonzo Addison, and Maurizio Forte, volume 4270 of *Lecture Notes in Computer Science*, 269–280, Springer Berlin / Heidelberg, (2006).

[15] C. Nikolai and G. Madey, 'Tools of the trade: A survey of various agent based modeling platforms', *Journal of Artificial Societies and Social Simulation*, **12**(2), 2, (2009).

[16] M.J. North, T.R. Howe, N.T. Collier, and J.R. Vos, 'A declarative model assembly infrastructure for verification and validation', in *Advancing Social Simulation: The First World Congress*. Springer, Heidelberg, FRG, (2007).

[17] E. Papadimitriou, G. Yannis, and J. Golias, 'A critical assessment of pedestrian behaviour models', *Transportation Research Part F: Traffic Psychology and Behaviour*, **12**(3), 242–255, (2009).

[18] I. Petreska and P. Kefalas, 'Population p systems with moving active cells', in *Proceedings of the 12th International Conference on Membrane Computing, Fontainebleau, France*, pp. 421–432, (2011).

[19] G. Păun, 'Computing with membranes', *Journal of Computer and System Sciences*, **61**(1), 108–143, (2000). Also circulated as a TUCS report since 1998.

[20] U. Wilensky, *NetLogo*, Center for Connected Learning and Computer-Based Modeling, Northwestern Univ., Evanston, IL., 1999. http://ccl.northwestern.edu/netlogo/.

[21] H. Yue, H. Hao, X. Chen, and C. Shao, 'Simulation of pedestrian flow on square lattice based on cellular automata model', *Physica A: Statistical Mechanics and its Applications*, **384**(2), 567–588, (2007).

[22] N. Zarboutis and N. Marmaras, 'Searching efficient plans for emergency rescue through simulation: the case of a metro fire', *Cognition, Technology and Work*, **6**, 117–126, (2004).

[23] X. Zheng, T. Zhong, and M. Liu, 'Modeling crowd evacuation of a building based on seven methodological approaches', *Building and Environment*, **44**(3), 437–445, (2009).

[24] W. Zhu and H. Timmermans, 'Modeling pedestrian shopping behavior using principles of bounded rationality: model comparison and validation', *Journal of Geographical Systems*, **13**, 101–126, (2011).

Contextualized Information Assessment in Smart Cities

Beibei Hu and **Abdelghani Chibani** and **Yacine Amirat**[1]

Abstract. In this paper, we present a rule-based architecture for the provision of information that is highly relevant for the user's activities. The architecture provides an ontology-based context model that can characterize the contextual situations of users who act in specific roles in urban environments. The architecture supports assessing the relevance of information relying on a rule engine, which allows to configure the behavior of systems in different situations. The evaluation based on different emergency scenarios has shown that our approach is effective for selecting relevant information given the user's context.

1 Introduction

The users involved in dynamic urban environments need to have up-to-date information to support their daily activities. Consider the fact that unexpected events, such as a traffic incident, could arise at any time in urban environments, people affected by such events act in different roles (e.g., citizens or first responders), and they need to be aware of activity-relevant information at real-time, in order to perform effective decision-making. For instance, police officers targeted at a traffic incident need to know the number of involved objects, while citizens need to be informed about the road condition.

To adapt the behavior of systems according to the current activities of users, a rich body of research focuses on realizing ubiquitous information services, such as a tourist guide and traffic routing. Context appears as a fundamental key to enable systems to determine the relevant information from large amounts of available information. Besides the physical context like location and temperature that can be measured by hardware sensors, the logical context, such as activities or tasks users are performing, is also important for characterizing the user's situations. *Context-awareness* refers to the capability of an application or a service being aware of its physical environment or situation and responding proactively and intelligently based on such awareness [9].

Urban ecosystems are required to explore distributed information sources, from environmental measures to people social network. The popularity of urban-specific linked data, such as LinkedGeoData[2] and OpenStreetMap[3], has attracted much research attention to improve these data relying on Semantic Web technologies [4]. However, our literature review has guided us to recognize that few solutions are presented to determine the relevance of information given the contextual situations of users involved in urban ubiquitous environments. Our research thus focuses on addressing the issue of contextualized information assessment in smart cities.

We have been involved in the European PEOPLE (Pilot smart urban Ecosystems leveraging Open innovation for Promoting and enabLing future E-services) project[4], which aims at creating pilots of Smart Open Innovation Urban Ecosystems in four European cities, namely Vitry sur Seine (in France), Bilbao (in Spain), Thermi (in Greece) and Bremen (in Germany). Our Vitry-sur-Seine pilot aims to create a network of smart bus stops and services related to urban social connection, mobility information management and security. For improving the public safety in critical situations, we consider emergency events (i.e., incidents) as our use cases.

Our contributions in this paper are three-folds: (1) We designed an architecture for the provision of information tailored to the specific needs of users in urban environments. (2) We presented an ontology-based model which is extensible to characterize the contextual situations of users, who are involved in certain events and acting in specific roles. (3) We developed a rule-based mechanism for computing the relevance of information given context by the specification of declarative rules.

The rest of the paper is organized as follows: Section 2 overviews the state-of-art of research in context-aware urban services. Section 3 illustrates the motivating emergency scenarios. Section 4 presents the overall system architecture and the functionality of each component. Section 5 presents an ontology-based context model. Section 6 elaborates the implementation of a rule engine. Section 7 shows our evaluation results based on scenarios. Section 8 concludes the paper and outlines the future research.

2 Related Work

In the field of ubiquitous computing, there is a lot of research on the use of context-awareness as a technique for developing applications, which are flexible, adaptable, and capable of acting autonomously on behalf of users [11]. Modelling context is considered as a fundamental issue. Ontology-based context modelling has advantages in terms of sharing a common understanding of the structure of context information among users, devices as well as services, and also reusing the domain knowledge. The existing ontologies for modelling the context that are most influencing for our work are: COSAR-ONT, which represents activities, artifacts, persons, communication routes and symbolic locations [12]; and BeAware!, which incorporates spatio-temporal relation types and represents situation types [2].

There exists many research efforts on the integration of Semantic Web technologies for the realization of an adaptable and scalable context-aware infrastructure. For example, the framework SCOUT (Semantic COntext-aware Ubiquitous scouT) [14] for personalized information delivery, which allows developers to seamlessly combine and query information from different web presences; and the MobiSem context framework [15], aiming for intelligently assisting mobile users by selectively replicating RDF data from remote data sources to a mobile device. In general, the use of ontologies as a

[1] Lissi Laboratory, University of Paris-Est Creteil (UPEC), Creteil, France, email: beibei.hu,chibani,amirat@u-pec.fr

[2] `http://linkedgeodata.org/`

[3] `http://openstreemap.org/`

[4] `http://www.people-project.eu`

key component for building a context aware computing architecture is widely acknowledged [15, 10]. In addition, many rule-based architectures are also explored, building on their flexibility and their simplicity in specifying the behavior of systems.

A variety of innovative solutions have been proposed to provide urban information services [13]. For instance, the PaTac platform designed for providing urban ubiquitous services to citizens and tourists [3]; an u-Service platform for urban computing environment [6]; and a semantically-based unit-task description model for supporting spontaneous interactions among users [8]. We focus on the provision of critical decision support for citizens and also first responders in urban situations.

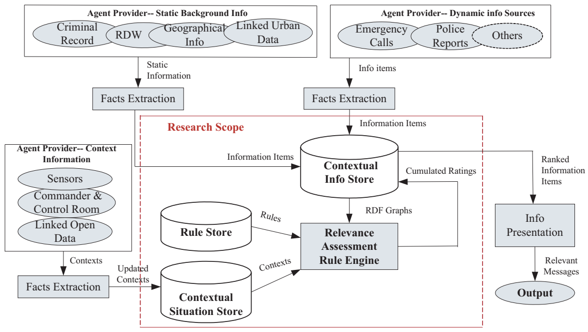

Figure 1. A Rule-based Architecture

3 Motivating Scenarios and Requirements

Assume in a smart city, where bus stations and streets are equipped with sensors to track the physical context of objects in real-time. The command center operators use a system to coordinate the traffic and manage the emergency events. Let us consider the following two scenarios.

Scenario I – Collision Incident. Around 18:00, two buses collide at the corner of the Rue Rene and the Rue Sacco. While there is no injury, the buses are disabled and cannot be driven away, resulting in a blocked lane. After 5 minutes, a robbery incident occurs nearby on the Rue Rene.

When police officers are dispatched to handle this collision incident, they need to learn the number of involved objects and also navigation information to arrive at the spot in time. After arrival, they need to know the road activity conditions to evacuate the traffic. Before starting to administer a breathalyzer test for involved drivers, they should know the personal background information of the drivers, such as illegal weapon holding records. Besides, citizens located at the bus station should be informed about a delay in the arrival of the bus.

Scenario II – Citizen Rescue. At the Paul Armangot bus stop around 17:00, an old man has suddenly fallen down and remains unconscious. One emergency ambulance is sent to rescue this citizen. Since a crowd of people who are employees of company ABC is having a strike march on the Paul Armangot street, a police unit is requested to assist the rescue task.

The ambulance crews, who are approaching to rescue a heart attack patient, need navigation information; when they are engaged in rescuing activity, the medical profiles of the patient (e.g., his disease history) should be informed. Police officers, who start to provide a free passage for the ambulance, need information includes the number of involved citizens and the traffic situations. Besides, the strikers should receive an alert message that requests them not to block the Paul Armangot street. In addition, the relatives of the patient need to know which hospital the patient will be sent. From the above scenarios that describe emergency events in a smart city, we can derive some fundamental requirements as follows:

- In contrast to the standard office work environments, the situations in which users perform their activities in urban spaces are characterized by various types of context. Beyond the spatio-temporal context, daily activities could be affected by unexpected events, such as bus collisions and traffic jam.
- In the same environment, the same piece of information may have different levels of significance for people acting in specific roles (i.e. citizens, police officers or ambulance crews). We can identify several general information needs as: (1) the navigation information is important to guide first responders to arrive at the incident spot in the fastest way; while such information is also relevant to citizens who are nearby the spot. (2) The personal safety information, such as violence behaviors, and criminal records of involved objects as well as their social relations, is highly relevant to first responders who are targeted at the objects. However, such type of information is irrelevant to citizens. (3) The medical profiles of victims are important for ambulance to take rescue actions. (4) Information about nearby incidents is relevant to first responders when it could influence their targeted incidents. (5) The bus schedule is only relevant to citizens who are waiting at that bus station.
- The most important information should be selected from different sources by taking advantage of user-related tasks and activities. Moreover, those information should be presented to the users in a proactively way. For example, a traffic jam alert should be triggered when the ambulance is approaching to the spot.

From the above analysis, it can be seen that the provision of context-aware urban information services involves the following challenges: (1) How can we design an architecture which supports contextualized relevance assessment by exploiting contextual information from distributed information sources? (2) How can we model the contextual situations of multiple user groups acting in specific roles? (3) How can we define a strategy for deciding the relevance of urban linked data given the user's contextual situations?

4 A Rule-based Architecture for Contextualized Information Delivery

According to the above requirements, we designed a rule-based architecture for contextualized information delivery (CIDA) as shown in Figure 1. Relying on the middleware presented in [5], context information can be provided from distributed agents. The main function of each component is explained below:

Information Item Store. It stores different types of information items needed to be assessed their relevance from different sources, including dynamic sensor data, high-level context derived from sensor data, background information from standard information systems (e.g., medical records) and linked open data (e.g., social relation). All information is formalized according to the RDF[5] (Resource Description Framework) data model, which facilitates the integration of multiple data sources and allows us to create expressive vocabularies with RDF schema.

Contextual Situation Store. It represents the contextual situations of users relying on an ontology-based model. We consider the

[5] http://www.w3.org/RDF/

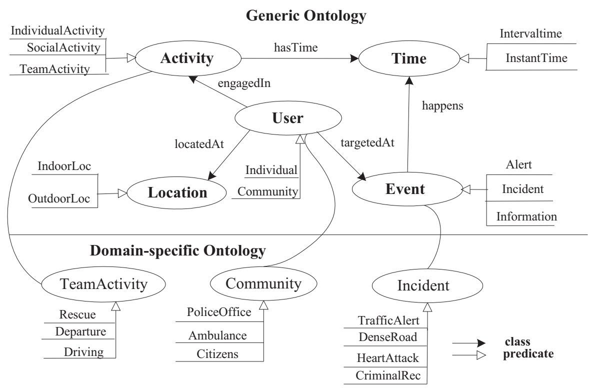

Figure 2. Context Model

context dimensions for the purpose of assessing the relevance of information, including role, activity, targeted event, involved objects, and spatio-temporal context.

Rule Store. It contains a set of rules which specifies how the relevance of context items for certain users is updated when a list of conditions is fulfilled. Inspired by the SWRL[6] (Semantic Web Rule Language), we defined a rule language (CRRL: context relevance rating language) for the purpose of generating a cumulative rating for a given user/info pair.

Relevance Assessment Rule Engine (RARE). RARE fulfills the core functionality of assessing the relevance of information by executing CRRL rules, while taking into account conditions, such as information type and a spatio-temporal relation between an event and a user's activity. The relevance ratings of context items are collected and updated according to the accumulated values generated by RARE.

5 An Ontology-based Context Model

Compared to the current context models, our ontology-based context model has the following advantages: (1) beyond the spatio-temporal context, it incorporates the activities that an individual or a group of users are engaged in and the events user are targeted at and also the social relations; and (2) it can explicitly describe the relationships between the context dimensions, such as the causal relation between an activity and an event. Our context model shown in Figure 2, consisting of a generic ontology and a domain-specific ontology, defines the context concepts of user, activity, event, location and time.

The **User** class is used to describe both the concept Individual and the Community. Each user has a current location, identity, contact information and role. The properties of the Individual involve the personal profile information (including name, gender, age, position and date of birth) and social relations. The concept Community has properties including organization hierarchy, responsibility and service. The subclasses of the Community are First Responders, Working Community and Student Community. For instance, a police unit is described by: PoliceUnit $\equiv$ Community $\sqcap$ $\exists$ hasRole.PoliceOfficers.

The **Activity** class is employed to describe the activities users are engaged in. An activity can be set by definition or can be derived from available contextual information. The concept activity is defined as: Activity $\equiv$ IndividualActivity $\sqcup$ SocialActivity $\sqcup$ TeamActivity. Specifically, the individual activity is an activity having exactly one actor; the social activity is an activity with at least two actors who are friends; and the team activity is an activity within a community who share the same responsibility.

The **Event** class is for describing events that happen in a real-world situation and might have influence on the user's activities and goals. This class has the three subclasses: Information (events that do not affect activities); Alert (events that may affect activities); and Incidents (events that endanger activities).

The **Time** class is employed to record when an event has occured, when an activity has started (InstantTime), how long an activity has performed (PerformDuration), and how long an event has been taken (IntervalTime). Relations between time intervals are represented based on Allen's Temporal Algebra [1].

The **Location** class is employed to represent where an event has occurred (isSpotOf), where a user is located (locatedAt) and where an activity is performed (performedIn). It also has the predicates adjacentLoc, disjointedLoc and nearby to describe particular spatial relations.

The relationships between context dimensions are represented by the predicates in the context model. For example, the *isTargetedBy* predicate, which has as domain Event and as range User, represents which event is targeted by which user; the *engagedIn* predicate, which has as domain User and as range Activity, reveals which user is engaged in which activity.

6 Relevance Assessment Rule Engine

We implemented a rule engine for relevancy assessment of information items.

6.1 CRRL Engineering

We defined CRRL according to the requirements from our information need studies [7]. In general, CRRL rules were formalized by two steps: (1) the conditions specified by the if statement were expressed by triple patterns and combined by logical operators; and (2) the degree of relevance was mapped to a numerical rating level in line with users' judgements, i.e., highly relevant corresponds to a numerical rating level of $+5.0$, moderately relevant corresponds to $+2.0$ and irrelevant corresponds to -1.0.

General rules deriving from basic requirements. According to the information needs for navigation information explained in Section 3, let us consider some general rules adapted to our two scenarios: **If** an information item is of type navigation information, **and** it describes an event which is targeted by a user, **and** the location described in this message is the user's destination, **and** the user has a role of first responders, **and** the user starts a departure activity at a certain time, **and** this item is reported no later than 30 minutes before the activity has started, **then** this item is highly relevant. The CRRL rule is:

R1.1: **infoType** ($?info$, NaviInfo) & **describedEvent** ($?info$, $?event$) & **targetedBy** ($?event$, $?user$) & (**hasRole**($?user$, Ambulance)| **hasRole**($?user$, Police)) & **engagedIn** ($?user$, DepAct) & **reportTime** ($info$, $?repTime$) & **startTime** (DepAct, $?actTime$) & [$?repTime \geq ?actTime$ - 30m]
$\Longrightarrow$ ***updateRating***$(+5.0)$

If an information item is of type navigation information, **and** it describes an event happened nearby the location of a user, **and** the user has a role of citizen, **then** this item is moderately relevant. The CRRL rule is:

[6] http://www.w3.org/Submission/SWRL

R1.2: **infoType** ($?info$, NaviInfo) & **describedEvent** ($?info$, $?event$) & **locatedAt** ($?event$, $?eveLoc$) & **nearby** ($?eveLoc$, $?userLoc$) & **isLocationOf** ($?userLoc$, $?user$) & **hasRole**($?user$, Citizen)
$\Longrightarrow$ ***updateRating***$(+2.0)$

6.2 CRRL Syntax

The CRRL grammar is an LL(1) grammar, and it is represented by the 41 corresponding rules. We present parts of the grammar in the following:

RG1	$< Rule >$	$:=$	$< Body >$ "$\Longrightarrow$" $< Head >$
RG2	$< Body >$	$:=$	$< RuleExp >$ | "(" $< RuleExp >$ ")"
RG3	$< Head >$	$:=$	"$updateRating$" "(" $< Decimal >$ ")"
RG4	$< RuleExp >$	$:=$	$< OrExp >$
RG5	$< OrExp >$	$:=$	$< AndExp >$ ("|" $< AndExp >$) $*$
RG6	$< AndExp >$	$:=$	$< TriExp >$ ("&" $< TriExp >$) $*$
RG7	$< TriExp >$	$:=$	$< Triple >$ | "(" $< OrExp >$ ")" | $< TimeFilterExp >$ | $< NegationExp >$
RG8	$< Triple >$	$:=$	$< Predicate >$ "(" $< Subject >$ "," $< Object >$ ")"

The rule body contains a list of conditions that are combined together with the intersection operator (logical conjunction) "&", the union operator (logical disjunction) "|", the negated operator "!", and arithmetic as well as comparison operators for temporal expressions. The basic term of conditions can be a triple or a time filter expression or a negation expression. Each triple is represented in the form of *predicate(subject, object).*

6.3 CRRL Semantics

The rule body specifies conditions that need to be fulfilled in a certain context. Two special variables are appointed in the rule body: ?info and ?user. When all conditions in the rule body are satisfied, the relevance rating for the combinations of given items (?info) for given users (?user) is incremented with the value indicated in the rule head. It is important to note that the relevance of a given item for a specified user can be modified by different rules concurrently according to different assessment criteria.

6.4 Implementation and Validation

The rule engine consists of three main components: (1) the rule parser, (2) the query generator, and (3) the query evaluator. The rule parser, which relies on JavaCC[7] to generate the parser of the rules, provides an abstract syntax tree (AST) to the query generator component. The query generator transforms these ASTs into SPARQL[8] queries and passes them along to the query evaluator which evaluates the queries using the Sesame query engine.

When evaluated, each query returns info/user pairs bindings. A query consists of three parts: the PREFIX declarations, the SELECT clause and the WHERE clause. The above Rule2.1 and Rule2.2 are translated to SPARQL queries as:

RARE is triggered to execute CRRL rules once context is updated. For each rule, RARE translates the rule body into a SPARQL query which is evaluated by the Sesame query engine, and computes

```
PREFIX  rdf: <http:// www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  cd:  <http://ruleEngine.org/ contextData />

SELECT  ?info  ?user
WHERE
{ ?info    rdf: type     cd: NaviInfo .
  ?info    cd: describedEvent     ?event .
  ?event   cd: targetedBy    ?user .
  { ?user   cd: hasRole    cd: Ambulance } UNION
  { ?user   cd: hasRole    cd: Poilce } .
  ?user   cd: engagedIn      cd: DepAct .
  ?info    cd: reportTime   ?repTime .
  cd: DepAct   cd: startTime    ?actTime .
   FILTER(?repTime >= ?actTime - "1800000" ^^xsd:long)
}

SELECT  ?info  ?user
WHERE
{ ?info    rdf: type     cd: NaviInfo .
  ?info    cd: describedEvent     ?event .
  ?event    cd: locatedAt    ?eveLoc .
  ?eveLoc   cd: nearby     ?userLoc .
  ?userLoc   cd: isLocationOf    ?user .
  ?user     cd: hasRole  cd: Citizen}
}
```

the relevance rating indicated in the rule head. All the translated rules interact together to increase or decrease the relevance, adding up the effect to produce a final ranking in a given context. To update the relevance, all the triples of type *?rating cd:value ?value* are firstly removed and then SPARQL construct queries are executed to create triple patterns as: *cd:newRating cd:newValue newValue^^xsd:decimal*.

7 EVALUATION

We evaluated the feasibility of our rule-based approach based on two emergency scenarios with respect to different configurations.

7.1 Datasets Definition

Corresponding to the collision and rescue scenarios, we constructed two datasets denoted as D_c and D_r, consisting of 48 information items as well as contextual information that concerns five context dimensions (i.e., role, time, location, event and activity). Note that we intended to add some items as "noise" information (e.g., items related to a robbery event), to test whether the system could distinguish relevant items from irrelevant ones.

Contextual Situation Representation. Let us consider a specific situation in scenario II: "around 10am, at the Paul Armangot station, an emergency event involving a patient named 'John' is targeted by an ambulance. A police unit located nearby is engaged in providing a clean passenger for the ambulance." In such a snapshot, the corresponding contextual situation can be represented in our context model as Figure 3 shows. It can be explained as follows:

1. An emergency event (urgentLevel) happens at Paul Armangot station (locatedAt) at 17:00 (startTime). This event involves a patient (involvedObject), whose son (socialRelations) lives in a residence behind Villejuif Arago bus stop.
2. A group of strikers (hasRole), who are the employees of company ABC (socialRelations), are holding a strike march nearby (co-located event).
3. An ambulance (hasRole) is approaching from the south of Paul Armangot street to avoid the strikers and thus can rescue the patient in time (targetedEvent & hasGoal).

[7] `http://javacc.java.net`
[8] `http://www.w3.org/Submission/SPARQL-Update/`

```
PREFIX cd: <http://ruleEngine.org/contextData/ >

cd :AmbulanceUnite1 ;
        cd :hasRole     cd :Ambulance  ;
        cd :targetedAt    cd :HeartAttackPatientEvent1  ;
        cd :hasGoal       cd :RescuePatient  .

cd :PolieUnite1
        cd :hasRole     cd :PoliceOffice ;
        cd :targetedAt   [
                cd :invovledObject  [
                        cd: hasCommunity  "CompanyABC"^^xsd:string   ;
                ];
        cd :engagedIn   [
              cd :taskType  cd :TeamTask  ;
              cd :taskDescription   cd :TrafficControl ;
              cd :influencedBy      cd :StrikeMarchEvent1;
              cd :startTime      "2012-03-23T10:10:00Z"^^xsd :dateTime
        ].

cd :HeartAttackPatientEvent1
        cd :isTargetedBy   cd :AmbulanceUnite1 ;
        cd :nearbyEvent    cd :StrikeMarchEvent1 ;
        cd :happensAt       "2012-03-23T10:10:00Z"^^xsd:dateTime  ;
        cd :locatedAt        cd :HeartAttackPatientEvent1Loc  ;
        cd :involvedObject  [
                cd :identity  "John"^^xsd:string  ;
                cd :hasSon (
                        cd:identify   "Peter"^^xsd:string
                        cd:livedIn   "Villejuif Arago"^^xsd:string
                        );
                                ].

cd :HeartAttackPatientEvent1Loc
        cd :isSpotOf     cd :HeartAttackPatientEvent  ;
        cd :isNearby     cd :StrikeMarchEventLoc ;
        cd :locName      [
               cd :country  "France"^^xsd:string  ;
               cd :city        "Villejuif"^^xsd:string  ;
               cd :street     "Paul Armangot"^^xsd:string
               ].
```

Figure 3. Contextual Situation Representation

4. A police unit (hasRole) is engaged in the task of traffic control (TeamActivity), which is influenced by the strike march event.

7.2 Rule Sets Definition

Method for extending rule set. For D_c, we defined a set of initial rules that is denoted as Rul_c consisting of 30 rules. To adapt to a new case of rescuing a patient, an extended rule set denoted as Rul_{c+r} was developed for the dataset D_r by adding 15 additional rules. Thus, the current rule set is extended in such a way that additional rules are always asserted on top of the current rules without changing the current ones.

Additional rules specific for a new case. To deal with new types of information, such as medical history in scenario II, additional rules can be easily developed by reusing and modifying of triple patterns. Two additional rules in Rul_{c+r} are illustrated in the following:

Rule 2.1: **infoType**($?info$, MedicalHistory) & **describedObject** ($?info$, $?person$) & **isInvolvedIn** ($?person$, $?event$) & **isTargetedBy**($?event$, $?user$) & **engagedIn** ($?user$, RescueActivity) & **hasRole**($?user$, Ambulance)
$\Longrightarrow$ ***updateRating*** $(+5.0)$

Rule 2.2: **infoType**($?info$, MedicalHistory) & **hasRole**($?user$, ?role) & ! **hasRole**($?user$, Ambulance)
$\Longrightarrow$ ***updateRating*** (-1.0)

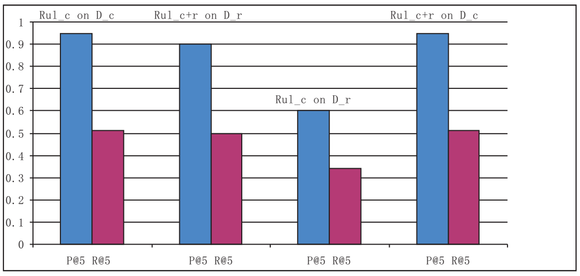

Figure 4. The Results for Four Different Configurations

7.3 Evaluation Method

We evaluated the behavior of the system with respect to different configurations. First, the effectiveness of the system is evaluated by applying rule sets on corresponding datasets, i.e., Rul_c on D_c; and Rul_{c+r} on D_r. Second, to investigate whether the rules developed for a specific scenario can provide a base performance for a new case, the performance of Rul_c was evaluated on dataset D_r. Lastly, the rule set Rul_{c+r} was applied to the dataset D_c to assess the influence of additional rules.

7.4 Evaluation Measures

The results were measured in terms of precision and recall. Precision denotes the ratio of correctly retrieved information items over all retrieved items ($top - 5$). Recall quantifies the ratio of correctly retrieved information items over all relevant items in a gold standard (GS). We evaluated the top 5 results, since we assume that it is a reasonable number of information items shown on mobile devices. Precision and recall are defined as the following formulas ($k = 5$):

$$P@k = \frac{|Top - k \cap GS|}{k}, \quad (1)$$

$$R@k = \frac{|Top - k \cap GS|}{|GS|}, \quad (2)$$

7.5 RESULTS

Figure 4 presents the results with regard to different configurations. We can observe the following results:

- With regard to the configuration of applying initial rules Rul_c on the corresponding dataset D_c, the precision is 0.95 while the recall is 0.51. It shows that the system is effective for selecting relevant information by taking advantage of relationships between context dimensions.
- The precision is 0.90 while the recall is 0.50, achieved by the rule set Rul_{c+r} on dataset D_r. It proves that the system can be adapted to a new scenario easily by only adding additional rules.
- With respect to the configuration of applying Rul_c on a new use case D_r, the precision is 0.60 while the recall is 0.34. Thus, the current rules engineered for a specific scenario are not over-fitted and they provide a reasonable performance on a new case.
- The results achieved by the rule set Rul_{c+r} on dataset D_c remain the same compared to that generated by Rul_c on D_c. Therefore, the additional rules don't degrade the performance of the system on a backward use case.

8 Conclusion and Future Work

In this paper we presented a rule-based architecture (CIDA) supporting for contextualized information delivery. CIDA is able to represent the characteristics of urban environments relying on an ontology-based context model, and it supports the development of rule-based systems for assessing the relevance of information given the user's context. The evaluation results with regard to different configurations show that CIDA is adaptable for dynamic urban environments and it is feasible for selecting relevant information for specific users.

In the next step, we will implement a component for the causal and temporal reasoning of events relying on the Event Calculus, in order to derive high-level context knowledge. We will use the reasoning power of ontologies to check and solve inconsistent context knowledge. To update the relevance ratings that guide better information services, we will explore machine technologies to learn from user's feedback. Through the evaluation studies based on large datasets, we will further investigate the feasibility our research approaches for the ubiquitous urban domain.

Acknowledgments

This work has been supported by the European FP7 ICT PSP PEOPLE project.

References

[1] James F. Allen, 'Maintaining knowledge about temporal intervals', *Commun. ACM*, **26**(11), 832–843, (1983).

[2] N. Baumgartner, W. Gottesheim, S. Mitsch, W. Retschitzegger, and W. Schwinger, 'BeAware! –Situation awareness, the ontology-driven way', in *Data & Knowledge Engineering*, volume 69, 1181–1193, Elsevier Science, (2010).

[3] Luigi Ceccaroni, Victor Codina, Manel Palau, and Marc Pous, 'Patac: Urban, ubiquitous, personalized services for citizens and tourists', in *Proceedings of the 2009 Third International Conference on Digital Society*, ICDS'09, pp. 7–12. IEEE Computer Society, (2009).

[4] I. Celino, Simone Contessa, Marta Corubolo, Daniele Dell'Aglio, Emanuele Della Valle, Stefano Fumeom, and Thorsten Krüger, 'Urbanmatch linking and improving smart cities data', in *Linked Data on the Web Workshop (LDOW 2012)*, pp. 1–10, (2012).

[5] Y. Chibani, A.and Amirat, 'Qos driven context awareness using semantic sensors infrastructure', in *Quality of Service Mechanisms in Next Generation Heterogeneous Networks: Utopia or Reality?*, ed., A. Mellouk, 407–430, Knowledge/John Wiley and Sons Inc., (2008).

[6] Yongyun Cho and Hyun Yoe, 'An U-service model based on a smart phone for urban computing environments', in *Security-Enriched Urban Computing and Smart Grid*, eds., Tai-hoon Kim, Adrian Stoica, and Ruay-Shiung Chang, Communications in Computer and Information Science, 600–605, Springer Berlin Heidelberg, (2010).

[7] Bei Bei Hu, *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work*, Ph.D. dissertation, Delft University of Technology, Delft, The Netherlands, 2011.

[8] A. Jimenez-Molina, Kang Byung-Seok, Kim Jun-Sung, and Ko In-Young, 'A task-oriented approach to support spontaneous interactions among users in urban computing environments', in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 8th IEEE International Conference on*, pp. 176–182, (2010).

[9] Michael Knappmeyer, Nigel Baker, Saad Liaquat, and Ralf Tönjes, 'A context provisioning framework to support pervasive and ubiquitous applications', in *Proceedings of the 4th European conference on Smart sensing and context*, EuroSSC'09, pp. 93–106. Springer-Verlag, (2009).

[10] Davy Preuveneers, Jan Van den Bergh, Dennis Wagelaar, Andy Georges, Peter Rigole, Tim Clerckx, Yolande Berbers, Karin Coninx, Viviane Jonckers, and Koen De Bosschere, 'Towards an extensible context ontology for ambient intelligence', in *Ambient Intelligence*, eds., Panos Markopoulos, Berry Eggen, Emile Aarts, and James Crowley, volume 3295, 148–159, Springer Berlin / Heidelberg, (2004).

[11] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen, 'Contextphone a prototyping platform for context-aware mobile applications', *IEEE Pervasive Computing*, **4**(2), 51–59, (2005).

[12] Daniele Riboni and Claudio Bettini, 'Context-aware activity recognition through a combination of ontological and statistical reasoning', in *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing*, UIC'09, pp. 39–53. Springer-Verlag, (2009).

[13] Emanuele Della Valle, Irene Celino, and Daniele Dell'Aglio, 'The experience of realizing a semantic web urban computing application', *Transactions in GIS*, **14**(2), 163–181, (2010).

[14] William Van Woensel, Sven Casteleyn, and Olga De Troyer, 'A framework for decentralized, context-aware mobile applications using semantic web technology', in *On the Move to Meaningful Internet Systems: OTM 2009 Workshops*, volume 5872, 88–97, (2009).

[15] Stefan Zander and Bernhard Schandl, 'Semantic web-enhanced context-aware computing in mobile systems: Principles and application', *Mobile Computing Techniques in Emerging Markets: Systems, Applications and Services.IGI Global*, 47–96, (2012).

Towards Procedurally Generated Perceptually Plausible Inhabited Virtual Cities: A Psychophysical Investigation

Stuart O'Connor[1] and **Szymon Fialek**[1] and **Etienne B. Roesch** [2] and **Christopher Peters**[1]

Abstract. Creating convincing inhabited virtual environments not only requires suitable modelling, but it also relies on sound evaluation methodologies. In this paper, we provide an overview of our preliminary work towards creating and evaluating inhabited, procedurally-generated, virtual cities. Our methodology involves the use of parameterised procedural generation models to provide a large variety of stimuli for supporting psychophysical evaluations of the human perception of agent behaviour in varying urban configurations. Of significance here is the context in which behaviour is perceived, its relationship with the mental models of observers and links to their impressions of realism. The research is envisaged to feed back into improving the urban modelling process while uncovering information about human mental models of urban environments. It is applicable to domains where the perceived realism of virtual crowds in urban environments is of importance, for example, in computer entertainment applications, serious games and cultural heritage recreations, with more general links envisaged to urban design and the psychology of human perception.

1 INTRODUCTION

Procedural generation refers to the automatic generation of content, rather than the use of traditional manual methods. While there has been a great focus on the procedural generation of cities [1], and recent studies have started to consider the credibility of the crowd scenes taking place in manually defined urban environments [2], few studies have investigated the technical challenges of creating inhabited procedurally generated environments that are as credible as possible, and the concomitant perceptual implications. This paper provides an overview of our efforts to establish a method to evaluate the credibility of procedural inhabited urban environments. The methods reported in this paper form the basis towards addressing two novel research challenges:

1. Evaluating procedurally generated cities: The creation of credible procedural virtual cities requires suitable generation routines and the identification of key generation parameters and their ranges from a potentially large candidate set. While many generation techniques are available, literature investigating methods for evaluating procedurally generated urban environments is scant.
2. Perception of crowd behaviour in an urban context: A significant issue here is the effect of the context in which the crowd behaviour takes place on its perception by a viewer. For example, specific types of pedestrian behaviour may appear realistic at a cross-roads or constrained path, but may not do so in a more open urban spaces. Such behaviours range from locomotion and navigation, to grouping and queuing. While many studies have focussed on modelling behaviour and more recently on evaluating it perceptually, an open challenge involves accounting for behaviour in the wide variety of contexts that occur in everyday situations. This not only relates to agent behaviours, but also to the automatic generation of annotations in the urban environment to support them.

We highlight the complementarity of both of these domains of endeavour: the context (e.g. environment) in which behaviour takes place is important to how it is perceived [3][2] and ultimately, for defining modelling priorities when creating plausible AI models. Procedural city generation techniques provide large data sets of specific, reproducible, parameterised stimuli suitable for psychophysical evaluation of crowds, capable of accounting for a multitude of varying urban contexts and configurations. Reciprocally, the modelling and evaluation of urban environments benefits from the addition of virtual inhabitants, who not only add a sense of life and realism to virtual urban environments, but are of great use as a tool for assessing the functionality of architectural and urban planning designs in a cost effective manner. Ultimately, this work seeks to support the development of credible, procedurally-generated inhabited virtual cities that may run in real-time (i.e. at interactive rates) on standard contemporary hardware configurations.

1.1 Overview of paper

This paper is organised as follows: In Section 2, we discuss related work on procedural city generation, behaviour annotation, crowd simulation and perceptual evaluation of crowds. Our initial modelling attempts for road network generation and crowd simulation are described in Section 3. The aim of the current modelling stage is not to create realistic looking cities, but to define algorithms allowing a large corpus of parameterised stimuli to be created for use in perceptual evaluation studies based on user experiments. These perceptual studies are envisaged to feed back into the modelling process, with the aim of allowing generation parameters to be automatically chosen that will maximise observer's impressions of plausibility. The general evaluation methodology is presented in Section 4, in addition

[1] Dept. of Computing, Coventry Univ., United Kingdom, email: Christopher.Peters@coventry.ac.uk
[2] School of Psychology, Univ. of Reading, United Kingdom.

to two sample experiment designs. Section 5 concludes with a summary of our contribution and details of our future implementation and evaluation plans.

2 BACKGROUND

Our research relates directly to two established domains: procedural city generation (see Section 2.1) and crowd simulation (see Section 2.2).

2.1 Procedural generation

Procedural generation refers to the automatic generation of content, rather than more traditional manual methods. Important procedural techniques have been presented for the generation of models, from natural objects [4] and phenomena, to complex man-made structures [1]. Procedural generation consists of abstracting the characteristics of structures into a set of procedures that are executed by the computer when needed. Procedural generation is based on sets of input parameters that can be adjusted to create variations. This approach seeks to capture the essence of an object or structure, without restricting it to one particular instance and is a powerful way of creating large amounts of varied content automatically.

Different procedural techniques have been proposed for city generation, including grid based layouts, L-systems, agent-based simulations and split grammars (see [5] for an overview). One of the most notable examples of a city generator is Esri CityEngine [6], which uses L-system grammars and is capable of modelling a complete city using a comparatively small set of statistical and geographical input data. It has been used by Maïm et al. [7] to recreate a virtual replica of ancient Pompeii[3] populated with crowds of virtual agents.

2.2 Agents and crowds

Research in modelling crowd behaviour in real-time can be divided into at least three general themes of direct relevance to this paper: the modelling of real-time crowd algorithms for guiding behaviours, the definition of annotation information to support behaviour in the environment, and the perceptual evaluation of generated behaviours at different levels.

2.2.1 Crowd modelling

When animating large groups of characters, a variety of different approaches have been taken, most notably the use of social force models [10] and steering approaches [11], augmented with path finding and reactive navigation [12][13], behavioural and cognitive capabilities incorporating synthetic perception [14], learning [15], and sociological effects [16]. Modelling approaches based on real input data are also available [17][18].

2.2.2 Behaviour annotation and authoring

Approaches to generating smart behaviour usually involve some form of behaviour scripting for artificial entities [19]. In addition, rather than pursuing the simulation of complex autonomous agents, considerable savings in processing and complexity are possible by adding semantic behaviour-supporting details into the virtual objects [20] or environment descriptions [21], which decision-making systems can then utilise. When dealing with large number of agents, managing them can become problematic. Authoring approaches, such as [22], aid in this by allowing users to paint crowds and their characteristics through a user-friendly interface.

2.2.3 Perception of crowds

An important issue in level of details is the perception of the fidelity of resultant characters' appearances and behaviours. Research has been conducted regarding the perception of the motion of individuals [8], groups [9] and crowds [21][2]. If properly designed, the results of perceptual studies can be linked to synthesis in order to inform the creation of more credible animations [24]. Perception can also be leveraged to drive level of detail schemes, for example, by disabling collision detection [25] for characters who are not clearly visible, or by accounting for user focus and events to enable a dynamic level of simulation detail [26]. In [27], a perceptual simulation is generated that processes a small portion of the world at a time with inexpensive approximations that are perceptually indistinguishable from those of the original simulation.

3 MODELLING AND SIMULATION

The primary purpose of our modelling efforts is to develop a set of procedural, parameterised models of urban generation and behaviour simulation, allowing a large set of stimuli to be reproduced and altered for experimental evaluation (see Section 4). Since the aim is to probe human impressions of urban environments and inhabitants, unlike standard modelling and behaviour approaches, these models must be capable of producing a wide variety of stimuli and not solely configurations that appear to the modeller to be realistic. The perceptual studies are envisaged to feed back into informing the modelling process, allowing generation parameters to be automatically selected to maximise viewer impressions of plausibility.

3.1 Overview

There are three distinct modelling areas in the current phase of work: road network generation (see Section 3.2) defines algorithms for creating a broad variety of parameterised road networks, currently focussing on grid-like networks and city structures. Behavioural annotation (see Section 3.3) concerns the definition of information in the virtual environment that provides a basis for supporting crowd AI. Crowd simulation (see Section 3.4) focusses on real-time algorithms for updating the movement of agents following paths in the city, utilising underlying annotations to support collision avoidance behaviours with pedestrians and the environment.

3.2 Road network generation

For the first phase of development on the road network generation algorithm, features were chosen empirically, based on a visual inspection of grid-like cities from Google Maps. The current emphasis is on road network structure rather than graphical fidelity, for example, modelling, lighting, texturing and shading. The network is generated in six stages (see Figure 1). First, a basic grid is generated that includes vertices as well as tertiary horizontal and vertical roads joining each vertex to its neighbours. At this stage the number of cells are entered as parameters to the generator. In the second stage, a number of streets (collection of connected roads) are removed. Primary, secondary and tertiary roads/streets are selected and diagonal

[3] See http://bit.ly/procedural-pompeii

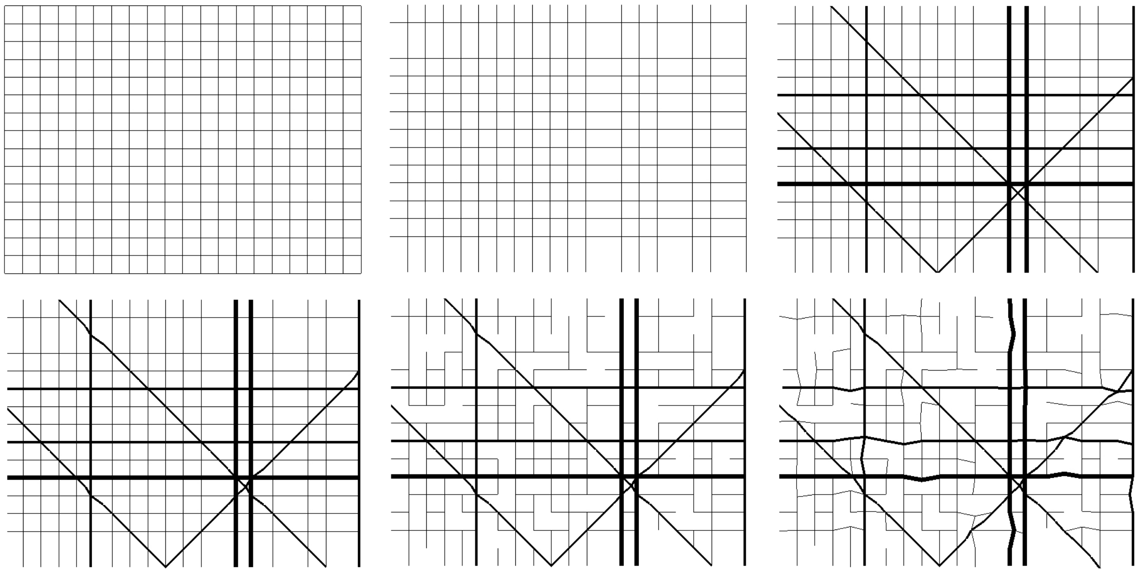

Figure 1. Procedural generation of the grid-based road network. From left to right, top to bottom: (1) basis grid, (2) remove streets, (3) select primary, secondary and tertiary roads and add diagonals, (4) move streets, (5) remove tertiary roads, (6) displace vertices.

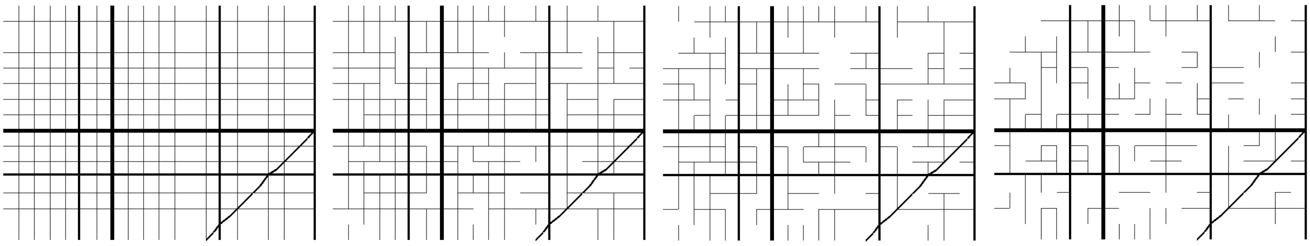

Figure 2. Early sample results of varying the proportion of tertiary roads removed. From the left: none removed, 10% removed, 20% removed, 30% removed. These networks will be used as stimuli in evaluation studies probing the human perception of computer generated inhabited urban areas (see Section 4).

streets added. A number of streets are moved/offset from their starting locations. A proportion of tertiary roads are removed and finally, in the last stage, vertex positions are altered. The number of vertices that change positions along x, y or both axis, as well at the number vertices comprising primary, secondary and tertiary roads and maximum displacement can be parameterised. An example of outputs varying the parameter for the proportion of roads removed is shown in Figure 2. The purpose of the generation algorithm is not to produce necessarily the most realistic structures, but a range of varied parameterised stimuli for experimentation.

3.3 Behavioural annotation

Behavioural annotation is achieved by labelling features within the environment so that agents can utilise them for adapting their behaviour accordingly. Labelling can consist of simple values designating the presence of a feature, or contain more complex control information [20]. The most fundamental annotation in the system is the addition of path-finding nodes. The run-time agent simulation conducts global navigation using an A* graph, and navigates locally using steering. For the urban crowd simulation the agents utilise this detection mechanism to find labelled features in nearby proximity. Example annotation features in the current system include:

1. Pedestrian Crossings: Agents' velocities are reduced and form bi-flowing lanes in order to cross.
2. Roads: Agents' velocities increase as they will be attempting to quickly cross the road.
3. Stationary Positions: Agents move to the stationary position if it is unoccupied and occupy it becoming stationary for an amount of time.

It is hoped that the addition of these behavioural features will provide a clearer relationship between the agents and their immediate environment for perceptual evaluation.

3.4 Crowd simulation

At run-time, simulating large crowds of agents requires constant updating on a frame by frame basis. Each agent has its own data structure encompassing path movement variables. When the simulation is initialised, each agent is initialised and set with key data regarding its coordinates, node position, vector, velocity and other essentials. Agents path-find using an A* graph of the city network. Currently each agent is updated as follows:

1. If the path has finished, search for a connecting node to the agent's current node.

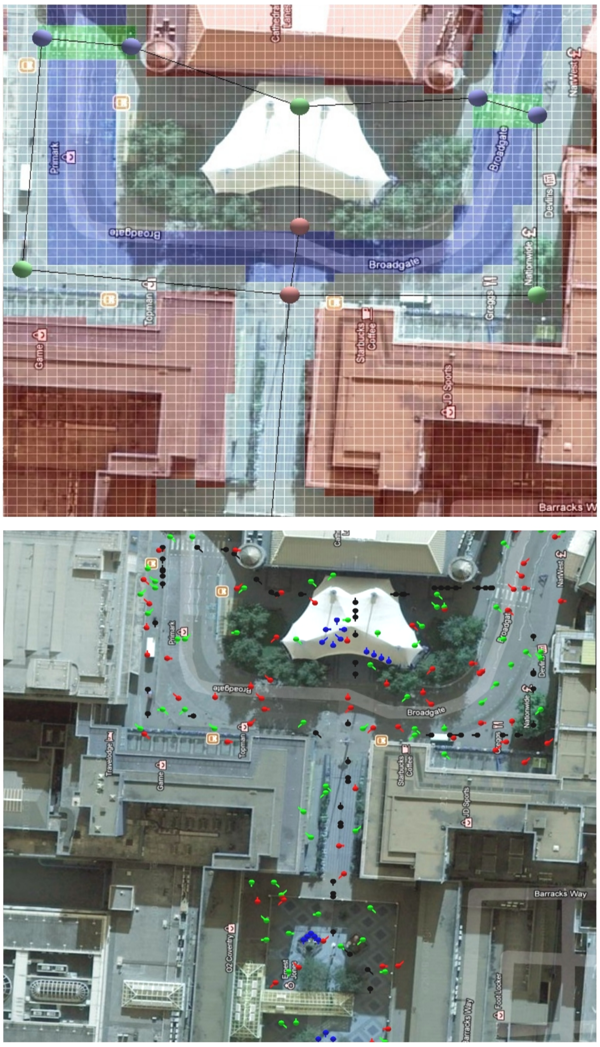

Figure 3. Crowd annotation and simulation prototype for a manually defined urban area. The left image displays behavioural annotation details for this environment, based on an image from *Google Maps* [28] (see Section 3.3). Nodes form a walking network in the city, coded according to usage. Here, blue and red nodes represent two different crossroad types respectively. A grid is also overlaid on the environment and aids collision avoidance between agents and the environment. The right image displays the final scene (also based on a Google Maps image) where agents are colour coded according to behaviour: black agents stay on paths, red agents wander, blue agents form groups and green agents are stationary.

2. If the path has not finished, conduct the movement routine and check to see if the next node has been reached.
3. If the node has been reached, then the path is considered to be finished and the agent's current node is set to the new connecting node.
4. If the node has not been reached, then calculate the vector difference between the current node and the destination node and update the agent's position according to its velocity.

3.5 Implementation

All implementation efforts are being conducted using C++ and the cross-platform OpenGL graphics library for visualisation. All algorithms documented in this section run in real-time. While this is not a necessary requirement for the development of experimental stimuli, these algorithms (subsequent to substantial improvement and optimisation) will eventually form the core of a real-time inhabited city environment: therefore the real-time constraint is still valid. It is especially pertinent for the behavioural simulation, as it is envisaged that city generation components will run as an offline pre-process in any eventual substantial implementation effort.

4 USER EXPERIMENTS

At the core of the method are user experiments designed to explore the parameter spaces manipulated by our generation algorithms. In particular, with a view to evaluating the plausibility of the visual material produced, we aim to identify the combinations of parameters that produce the most credible material over populations of participants. The results of these user experiments are intended to be used to fine tune our algorithms, leading to the automatic generation of environments and behaviours that will maximise viewer impressions of plausibility.

4.1 General methodology

Our general experimental methodology consists of an analysis-synthesis-perception approach [2], consisting of the following components:

1. Analysis: Use real-world instances and scenarios to inform algorithm construction and identify features of interest.
2. Synthesis: Synthesise virtual replicas of real-world instances and modify selected features to produce a corpus of synthesised original and modified experimental stimuli.
3. Perception: Conduct psychophysical studies, testing if participants can differentiate between the original and modified versions of the synthetic stimuli and identifying perceptual thresholds.

The output stimuli we produce are in the form of static images (for the road network experiments) and/or video clips. Consequently, we plan to employ online survey systems, such as Mechanical Turk [29] or SurveyMonkey [30] to recruit large numbers of participants, for the large scale evaluation of the material produced. Participants will be presented with the output of our algorithms, and asked to rate credibility on a number of dimensions. An important consideration here is set on employing appropriate techniques for managing quality control of the surveyed data [31] and evaluating the parameter space of our algorithms. With a view to fine tuning our experiments, we thus intend to conduct two preliminary studies based on the psychophysical method.

A key objective of the psychophysical method is to related phenomenal experience to physical stimuli [32][33]. "The art of psychophysics is to formulate a question that is precise and simple enough to obtain a convincing answer" [34] such that it is possible to study the perceptual effects of particular physical dimensions. Investigating the limits of visual perception, an experimenter may, for instance, decide to parametrically vary the display duration of images, with the aim to measure the threshold below which the images are no longer perceived [35]. The experimenter is provided with correlates of perception that are as opposite and clear-cut as possible, facilitating greatly the interpretation of the data [36]. Of further importance is the fact that the emphasis lies in the perceptual abilities of that particular participant.

In what follows, we describe the experimental design and objectives of two preliminary studies, the first relating to the perception of road networks (Section 4.2) and the second relating to the perception of basic crowd movements along paths in an urban context (Section 4.3). In both of these experiments, we explore a great number of continuous parameters, by asking participants to evaluate the visual material that our algorithms produce online: We employ the psychophysical method to explore the vast parameter space which our algorithms manipulate; a new visual stimulus is created upon each new trial, which the participant is asked to evaluate. This evaluation is fed back into the algorithm until we reach the threshold that determines the niche in the parameters space that produced visual material that looks the most credible.

4.2 Road network experiment

Participants will be presented with static images showing procedurally generated road networks as well as representations of real road networks (based on examples from Google Maps) and asked to rate the credibility of these networks. Images will be presented for a short amount of time and followed by a blank screen to separate the trials and allow for response. This short exposure time is chosen to avoid conscious, rational analysis of the presented network and prompt for an initial reaction to the visual material. For each test, participants are presented with two-alternative forced-choice, “According to you, was this road map representative of a real city, yes or no?”. Four parameters will be used in the experiment:

1. Proportion of vertex displacement
2. Amount of vertex displacement
3. Proportion of streets and roads deleted
4. Proportion of primary roads to secondary roads.

Sample images for the outputs of different values of the proportion of roads removed parameter are shown in Figure 2. Before the start of the experiment, participants will be asked to provide a demographic data, including their age, gender, occupation, type and level of education. Upon completion, participants will be asked to fill in questionnaires containing the following questions:

1. Was there anything that helped you to perform the task, any particular element of the road network that determined you decision?
2. How often do you use paper maps or online services like Google Maps or similar?
3. Rank, on scale form one to five, how difficult you found the task (one - very easy; five - very difficult).
4. Do you have any other comments?

The QUEST Bayesian adaptive staircase procedure [38][36] will be used to estimate thresholds of believability. Upon each new trial, the participant is presented with a new set of parameters to evaluate, which is determined by their reaction to the previous trial. The result of this evaluation is then used to adjust the parameter set, and new material is produced for the next trial in a manner that attempts to close in on the perceptual limits of plausibility. The QUEST algorithm takes into account perceptual interindividual differences to navigate a parameter space and determine perceptual thresholds. The parameter value is increased or decreased, according to the evaluation results from the previous trial. The threshold is estimated as the value for which a certain number (usually the best quantile) of positive responses is obtained[4].

[4] See http://etienneroes.ch/quest for information about the QUEST algorithm.

4.3 Crowd behaviour experiment

To measure the perceived realism of the simulation and its various features, a series of tests have been devised utilising a series of perceptual evaluations. The data from the tests involving participants will be examined in order to discover which features of the simulation are perceived to improve scene realism so that they can be ranked and cross examined with the data gained from the comparisons of real-world locations. Each participant will be shown a series of short video clips from the crowd annotation and simulation prototype (see Figures 3 and 4). The video clips will contain a different version of the same scene with different conditions. Each video clip will be followed by a blank screen to separate the trials and allow for response.

Initial experiments will consider variations in agent movement velocity. At least two velocity conditions are envisaged: a uniform velocity condition and a varying velocity condition. In the uniform velocity case, agents all move at the same speeds while in the varying velocity case, different agents will move at different speeds. This will test if agents that move uniformly are perceived to be more realistic than agents moving in a more chaotic manner, and especially, how the differing context of the environment and situation may change ratings. For example, in high density situations in constrained areas, agents may be forced to move at uniform speeds, while in more open urban areas, movement may be expected to vary more.

Data will be collected during the test using a questionnaire asking participants which of the two scenes shown for each clip is the more realistic. After the trials are complete, a short interview will be conducted for each participant in which the data from the completed test will be discussed.

5 CONCLUSIONS AND FUTURE WORK

We have presented details of our work-in-progress towards a psychophysical investigation of inhabited virtual cities. While the modelling methods and evaluation details are preliminary, the general methodology has been developed and will inform our modelling and evaluation efforts. It consists of the analysis of real-world instances and scenarios, synthesis of virtual replicas and modified outputs as experimental stimuli, and perceptual evaluation testing participants impressions of realism of behaviour in a variety of urban contexts. It is based on the insight that cities and their inhabitants cannot be evaluated in isolation: perception of behaviour is dependent on context, and inhabitants are inextricable parts of their urban environments. Reciprocally, virtual inhabitants not only add a sense of life and realism to virtual urban environments, but are of great use as a tool for assessing the functionality of architectural and urban planning designs in a cost effective manner.

The research is envisaged to feed back to improving urban modelling processes while uncovering human mental models of urban environments through user experiments. More generally, the perception of automatically generated behaviour and environment has consequences for reality. Crowd members do not act based on reality, but based on the perception and interpretation of limited data concerning events and environments, employing some form of mental model to do so. Investigating mental models through perceptual studies and user experiments may add new perspectives enabling AI systems to better predict or understand real behaviour or to better design real urban environments.

We are in the process of creating more sophisticated traditional generation procedures and visualisations consisting of radial and

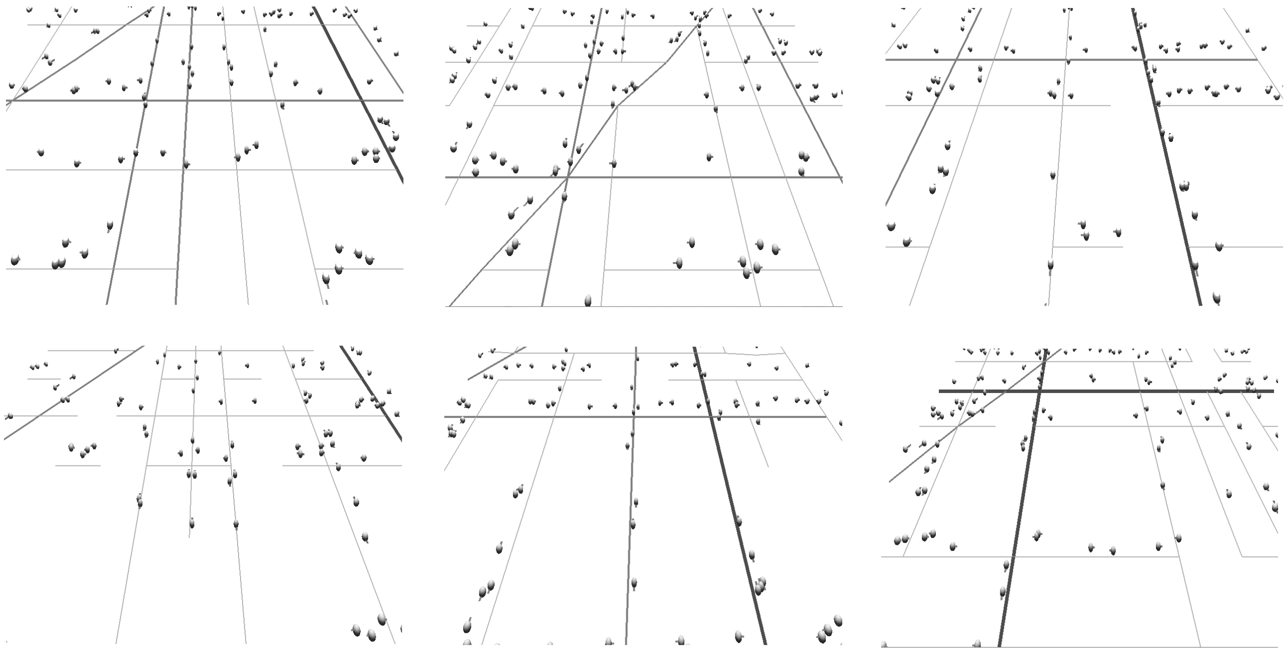

Figure 4. An early example of outputs of the procedural road network generation algorithm used to define paths for agents. The current emphasis is on defining a set of suitable parameterisations for road structure and agent behaviour, with a view to experimentation. Future work will focus on the graphical and behavioural fidelity of the environment once suitable base parameters have been established through pilot studies, as part of the iterative methodology.

evolved road networks. For the city generation phase, while the emphasis thus far has been on structural modelling rather than graphical fidelity (for example, models, lighting, texturing and shading), we are in the process of developing more sophisticated 3D graphical representations of the city. These will need to be carefully managed however, as experimentally, the culture, region, age or a host of other contextual factors can be expected to set participant expectation in relation to city plausibility. In this respect, the use of simple stimuli for experimentation is thus an important part of the methodology for attempting to identify candidate features that may be generic across contexts.

In terms of agent behaviour, a basic infrastructure has been completed and more annotation details will be added to nodes and grid zones to support the more complex behaviours that we are currently pursuing. In particular, we wish to improve local navigation and introduce proper collision avoidance behaviour between pedestrians, as this is likely to facilitate experimentation relating to the impact of simulation of behaviour on viewer credibility.

We hope that after intensive experimentation, the generation and simulation algorithms (and identified parameter sets) will eventually form the core of an automatic method for generating virtual real-time inhabited cities that are perceptually plausible.

ACKNOWLEDGEMENTS

The authors would like to thank the referees for their constructive and informative comments which helped to improve the quality of the paper.

REFERENCES

[1] Parish, Y.I.H. and Müller, P. Procedural modeling of cities. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01). ACM, New York, NY, USA, pp. 301-308, 2001

[2] Ennis, C., Peters, C. and O'Sullivan, C. Perceptual effects of scene context and viewpoint for virtual pedestrian crowds. ACM Trans. Appl. Percept. 8(2):10, 2011

[3] Peters, C., Ennis, C., McDonnell, R. and O'Sullivan, C. Crowds in context: Evaluating the perceptual plausibility of pedestrian orientations. In Proceedings of Eurographics Short Papers. Springer-Verlag, Berlin, 2008

[4] Runions, A., Fuhrer, M., Lane, B., Federl, P., Rolland-Lagan, A.-G., and Prusinkiewicz, P. Modeling and visualization of leaf venation patterns. ACM Trans. Graph. 24, 3 (July 2005), pp.702-711, 2005

[5] Kelly, G. and McCabe, H. A Survey of Procedural Techniques for City Generation, ITB Journal, Issue 14, December 2006

[6] Esri CityEngine, http://www.esri.com/software/cityengine/

[7] Maïm, J., Haegler, S., Yersin, B., Müller, P., Thalmann, D. and VanGool, L. Populating ancient pompeii with crowds of virtual romans. In Proceedings of the 8th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST'07). 26–30, 2007

[8] McDonnell, R., Newell, F. and O' Sullivan, C. Smooth movers: Perceptually guided human motion simulation. In: SCA '07: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on computer animation, pp 259–269, 2007

[9] Ennis, C., McDonnell, R. and O'Sullivan, C. Seeing is believing: Body motion dominates in multisensory conversations. ACM Trans Graph 29(4):1–9, 2010

[10] Helbing, D. and Molnár, P. Social force model for pedestrian dynamics, Phys. Rev. E, 51(5): 4282–4286, American Physical Society, May 1995

[11] Reynolds, C. W. Flocks, herds, and schools: A distributed behavioral model. SIGGRAPH '87 Proceedings, M. C. Stone, Ed., vol. 21, 25–34, 1987

[12] Paris, S., Pettré, J. and Donikian, S. Pedestrian Reactive Navigation for Crowd Simulation: a Predictive Approach. Computer Graphics Forum, 26(3), Eurographics 2007 Conference Proceedings, September 2007, p. 665-674, 2007

[13] Lamarche, F. and Donikian, S. Crowd of Virtual Humans: a New Approach for Real Time Navigation in Complex and Structured Environments. Computer Graphics Forum, 23(3): 509-518, Eurographics 2004 Conference Proceedings, 2004

[14] Ondřej, J., Pettré, J., Olivier, A.-H. and Donikian, S. A Synthetic-Vision-Based Steering Approach for Crowd Simulation, ACM SIGGRAPH

2010 Papers, 2010
[15] Shao, W. and Terzopoulos, D. Autonomous pedestrians. In Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation (SCA ‘05). ACM, New York, NY, USA, 19-28, 2005
[16] Durupinar, F., Allbeck, J., Pelechano, N. and Badler, N. Creating crowd variation with the ocean personality model. In AAMAS ‘08: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems (Richland, SC, 2008), International Foundation for Autonomous Agents and Multiagent Systems, pp. 1217–1220, 2008
[17] Lerner, A., Chrysanthoum, Y., Andlischinski, D. Crowds by example. Eurographics ‘07: Computer Graphics Forum 26(3), 2007
[18] Pettré, J., Ondȓej, J., Olivier, A-H., Crétual, A. and Donikian, S. Experiment-based modeling, simulation and validation of interactions between virtual walkers, Proceedings of the 2009 ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA‘09), 2009
[19] Anderson, E.F. Scripted smarts in an intelligent virtual environment: behaviour definition using a simple entity annotation language. In Proceedings of the 2008 Conference on Future Play: Research, Play, Share (Future Play ‘08). ACM, New York, NY, USA, pp.185-188, 2008
[20] Peters, C., Dobbyn, S., MacNamee, B. and O’Sullivan, C. Smart objects for attentive agents, Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2003
[21] Peters, C. and Ennis, C. Modelling groups of plausible virtual pedestrians, IEEE Computer Graphics and Applications, Special Issue on Virtual Populace, vol. 29, no. 4, pp. 54–63, July/Aug, 2009
[22] Ulicny, B., de Heras Ciechomski, P. and Thalmann, D. Crowdbrush: Interactive authoring of real-time crowd scenes. In SCA ‘04 Proceedings of the 2004 ACM SIGGGRAPH/Eurographics Symposium, pp. 243-252, 2004
[23] Yersin, B., Maïm, J., Pettré, J. and Thalmann, D. Crowd Patches: Populating Large-Scale Virtual Environments for Real-Time Applications, I3D‘09: Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games, 2009
[24] McDonnell, R., Larkin, M., Hernandez, B., Rudomin, I., O’Sullivan, C. Eye-catching crowds: Saliency based selective variation. ACM Trans Graph 28(3):110, 2009
[25] Paris, S., Gerdelan, A. and O’Sullivan, C. CA-LOD: Collision Avoidance Level of Detail for Scalable, Controllable Crowds, Lecture Notes in Computer Science, Volume 5884/2009, pp. 13-28, 2009
[26] Navarro, L., Flacher, F. and Corruble, V. Dynamic Level of Detail for Large Scale Agent-Based Urban Simulations. In The Tenth International Conference on Autonomous Agents and Multiagent Systems, pp. 701–708, International Foundation for Autonomous Agents and Multiagent Systems, 2011
[27] Sunshine-Hill, B. and Badler, N. Perceptually realistic behavior through alibi generation. Proc. Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE), 2010.
[28] Google Maps, http://maps.google.com
[29] Mechanical Turk, http://www.mturk.com
[30] SurveyMonkey, http://www.surveymonkey.com
[31] Welinder, P. and Perona, P. Online crowdsourcing: Rating annotators and obtaining cost-effective labels. Workshop on Advancing Computer Vision with Humans in the Loop, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR, 2010
[32] Link, S.W. Rediscovering the Past: Gustav Fechner and Signal Detection Theory. Psychol Sci 5, pp. 335-340, 1994.
[33] Fechner, G.T. Elemente der Psychophysik. Breitkopf & Härtel: Leipzig, 1860.
[34] Ehrenstein, W.H. and Ehrenstein, A. Psychophysical Methods. Modern techniques in neuroscience research pp. 1211–1241, 1999.
[35] Roesch, E.B., Sander, D., Mumenthaler, C., Kerzel, D. and Scherer, K.R. Psychophysics of emotion: The QUEST for Emotional Attention. Journal of Vision, 10(3):4, 1-9, 2010.
[36] Watson, A.B. and Fitzhugh, A. The method of constant stimuli is inefficient. Perception & Psychophysics, 47, pp. 87-91, 1990.
[37] Farell, B. and Pelli, D.G. Psychophysical methods, or how to measure threshold, and why. In R. H. S. Carpenter & J. G. Robson (Eds.), Vision research: A practical guide to laboratory methods (pp.129-136). New York: Oxford University Press, 1999.
[38] Watson, A.B. and Pelli, D.G. Quest: A Bayesian adaptive psychometric method. Perception & Psychophysics, 33, pp. 113–120, 1983.

Safer Smarter City

Fabien Flacher, Laurent Navarro and Christophe Meyer[1]

Abstract. The purpose of this extended abstract is to present SE-Star, a Thales proprietary multi-agent simulator, together with the content of a scenario demonstrating its capabilities at defining urban safety procedures and new services in Smart Cities simulations.

1 SE-STAR

Large-scale 3-dimensional environments, populated with dozens to hundreds of thousands humans showing complex behavior, are now widely used in many fields such as Defense, Security, Smart Cities, Smart Grid, Transportation, Training, Entertainment and researches in Social Sciences. They usually aim to play scenarios that require credible human activity for planning, training and optimization purposes. However, such simulations are difficult to realize due to the complexity of the behavioral models and the huge amount of data involved.

We present here **SE-Star**, a Thales proprietary multi-agent simulator, which is able to animate a large number of agents within a large-scale 3D environment. Each agent is fully autonomous and unique, with specific physical and psychological traits. It is driven by a motivational tree, possesses internal states and motivations, and can perceive its environment through several advanced artificial sensors – such as a retina (Figure 1). This architecture allows it to fulfill its goals by interacting with intelligent objects. However, communication and advanced coordination abilities are not implemented yet within the simulator.

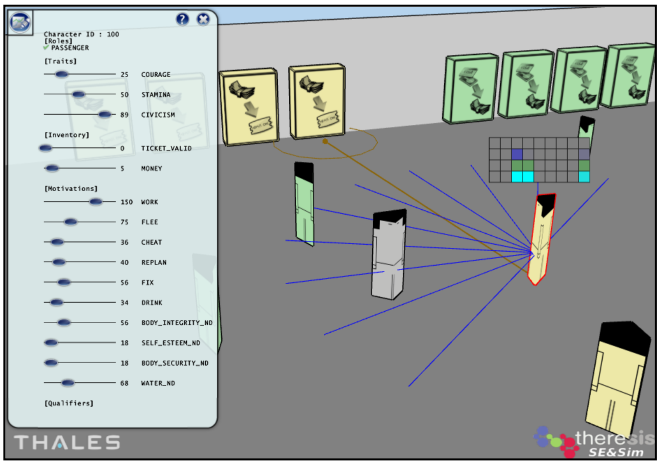

Figure 2. View of an agent about to buy a ticket, with its internal state (on the left) and its retina (above it).

SE-Star is a powerful and modular application which can interface with supervision systems for different purposes. First, it can simulate the business applications connected to the supervisor, allowing it to test its orchestrated responses in various contexts, and validate operational procedures. Then, SE-Star can stimulate the supervisor by reproducing the data originally sent by real equipment, such as video streaming or any kind of sensor (Figure 2). This allows the training of operators on scenarios that are difficult or impossible to recreate in real life. Finally, it can bring advanced visualization capabilities to supervisors through real data aggregation.

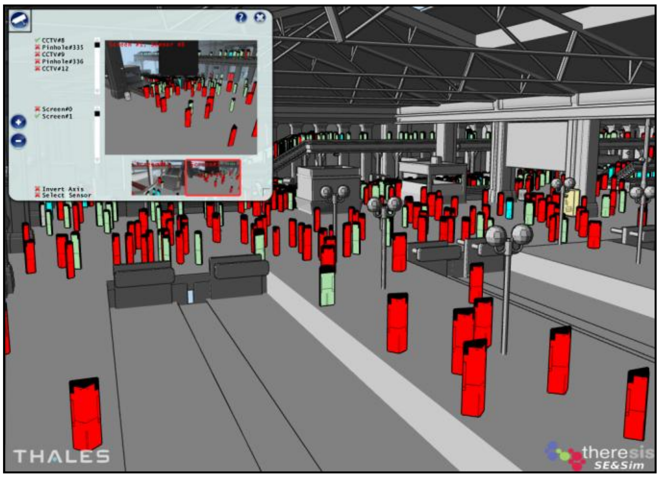

Figure 1. Simulation of the Gare du Nord train station in Paris, with a video stream of a security camera (on the top left).

SE-Star offers facilities for the implementation for agents' and objects' behavioral models, through modifiable repositories. Moreover, its fast prototyping tools allow the modification of the environment, by adding and removing objects and agents during the simulation. Thus, the simulator can quickly adapt a specific business or research field, and help non-expert users to design, evaluate and optimize business models, procedures and scenarios.

2 OVERVIEW OF THE SCENARIO

During this demonstration, we propose to simulate an area of 2.25 square kilometers of a fictional city (Figure 3) populated with 100.000 pedestrians and 1.000 vehicles. The area itself is segmented into homes, shops and offices and contains also a highway and a railway line.

Each agent is driven by a dozen different motivations, such as going to work, satisfying its basic biological needs, buying several

[1] Thales, 20-22 rue Grange Dame Rose, 78140 Vélizy, France. Email: {firstname.lastname}@thalesgroup.com

goods at the available shops, wandering in the city or returning home. These high-level actions are automatically broken down into basic tasks that the agent can perform by interacting with various street furniture or classic urban service providers.

For example, an agent going to work may need to take the train at one of the two available train stations. To do so, he may want to buy a ticket, thus to interact with a ticket machine located near the station and maybe with an ATM if he does not earn enough money. However, a less civic agent can also jump over the ticket barriers, showing that the plan of each agent also depends on its physical and psychological traits.

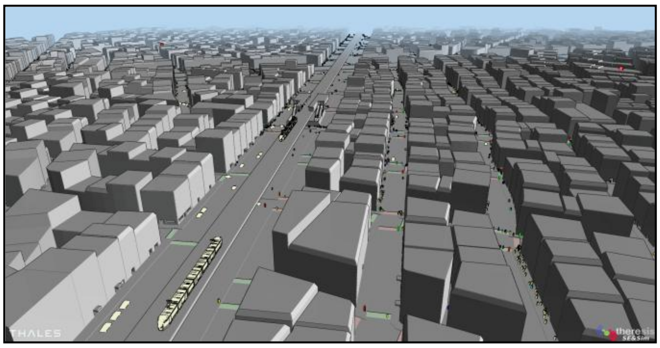

Figure 3. View of the city neighborhood with the highway and the railway line.

In this context, we can demonstrate various scenarios related to the industrial and academic fields cited above. We propose here a Security scenario, based on the evacuation of a critical infrastructure and the automatic activation of urban safety services. The action takes place in one of the two train stations, during the morning rush hour when the whole neighborhood is highly crowded with pedestrians and vehicles. The main steps of the scenario are as follow:

1. A fire occurs near the train station, and spread gradually.
2. The surrounding agents start to stress and flee.
3. Smoke is detected by sensors close, which trigger an alarm in the station.
4. The passengers are massively leaving the station.
5. Ticket barriers are opened, facilitating the evacuation of passengers.
6. The city’s fire stations are alerted and send fire trucks.
7. The train traffic is stopped and the train station is closed.
8. Agents are affected by fire and smoke, possibly leading them to death.
9. First responders arrive on site and start to extinguish the fire
10. Fire is extinguished, the trucks return to the fire stations.
11. The alarm is stopped and the train station is opened again.
12. Pedestrians start to use the station again.

Such scenario demonstrates that Smart City simulators can help answering specific Security use cases, such as the definition of urban safety procedures and new services in critical situations with their testing and optimization. Those new services can be created to the convergence of the different integrated systems. It shows that a large-scale environment populated with a large number of autonomous agents is critical for the credibility and the understanding of those scenarios, in order to use Smart City simulators as powerful analysis, planning and decision-making tools.

The City Knowledge Platform: an Agent-oriented Framework for Intelligent City Maintenance, Management and Modeling

Fabio Carrera[1] and **Stephen Guerin**[2]

Abstract. In this demo, we introduce a new approach to Urban Asset Management and Simulations that relies on a City Knowledge [1] platform composed of intelligent agents, each "representing" an individual urban element, much in the same manner as human agents manage actors and athletes. Our intelligent urban agents act on behalf of urban assets and communicate with other agents to maintain an up-to-date picture of the status of the asset, with the ability to issue alerts when critical situations are detected, based on publish-and-subscribe feeds coming from legacy systems, as well as from mobile apps or web reports.
Our City Knowledge Platform employs agents to represent mobile as well as immobile assets, making it possible both to maintain and manage urban assets, as well as to model activities such as vehicular or pedestrian traffic, even in the presence of sparse data, leveraging and re-utilizing data from legacy administrative systems, from existing sensor networks, as well as from human sensors deploying custom-designed smartphone apps.

1 INTRODUCTION

Urban Data Management and Modeling have long relied on tried and true technologies like relational databases and geographic information systems, coupled, at times, with some higher-level modeling. In this demo, we introduce a new approach to Urban Asset Management and Simulations that relies on a City Knowledge Platform composed of intelligent agents, each "representing" an individual urban element, much in the same manner as human agents manage actors and athletes [2].
Our intelligent urban agents act on behalf of urban assets and communicate with other agents to maintain an up-to-date picture of the status of their own asset, with the ability to issue alerts when critical situations are detected, based on publish-and-subscribe feeds coming from legacy systems, as well as from mobile apps or web reports. Each asset's data is visualized as an individual wiki page, where the information is immediately published and can thus be reviewed (and partially modified) by citizens and municipal officers. Any change to the attribute set of an urban asset is immediately reflected on its wiki page. As a sort of "guardian angel", each agent will monitor what happens to the asset under its direct tutelage, by scanning official documents, citizen reports, work orders and the like, while also keeping an eye on what is happening to other nearby assets of the same type, by subscribing to other assets' wiki feeds and also by unleashing background "bots" that are constantly rummaging through API-enabled municipal systems.

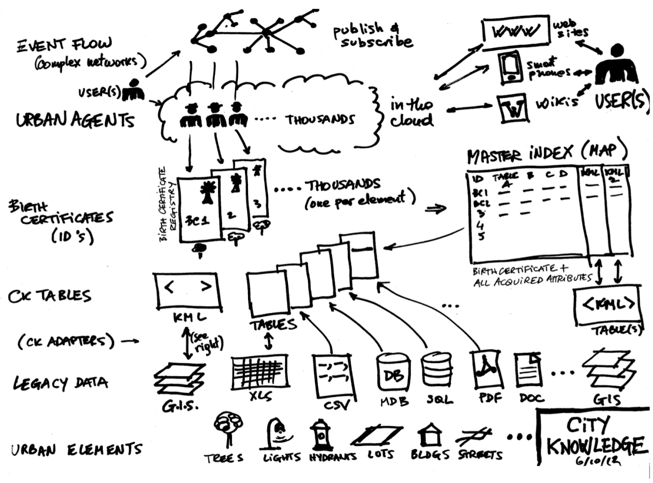

Figure 1. A conceptual diagram of the City Knowledge system

For example, each streetlight in a city will have its own wiki page, with a set of structured attributes (e.g. info boxes) that will be constantly updated thanks to its own agent. Citizen-based mobile apps (like our own *StreetBump* [3] and *VeniceNoise* [4]) will interact with these agents to provide up-to-date reports, such as the fact that a light is out, which in turn will trigger the agent to send alerts to the public works department to request that the light bulb be replaced. Scouring the city's work order system, the agent representing the defective streetlight will notice that there is a plan to replace the bulb of a nearby light pole, and will therefore request that its own light fixture be replaced concurrently, thus saving the city the cost of an extra trip to that street. The agent will also calculate the lifespan of the burnt bulb, based on its thorough knowledge of the maintenance history of the light pole it is in charge of. By communicating with all other agents in charge of streetlights in the city, it will discover that some are getting twice as many hours out of their bulbs, thanks to new technologies, such as fluorescent or LED lights, and will thus request that its pole be equipped with these latest fixtures, instead of the old inefficient ones. And so on...

[1] Interdisciplinary and Global Studies Division, Worcester Polytechnic Institute, Worcester, Massachusetts, USA. email: carrera@wpi.edu
[2] City Knowledge LLC and Redfish Group LLC, Santa Fe, New Mexico, USA. email: stephen@redfish.com

2 DESCRIPTION OF DEMONSTRATION

Our demo introduces our City Knowledge platform (Figure 1), which allows a variety of legacy datasets about the urban environment to be uploaded (via the *City Knowledge Console*), archived in a real-time database (*Acequia*) and managed through individually assigned agents (*AgentsCloud*). Multiple applications, both mobile and web-based, can simultaneously access the data in the CK system, as demonstrated by the project we just completed, under the auspices of UNESCO, for the crowdfunding of the preservation of material culture in Venice, Italy [5].

The demonstration shows how the CK Console was used to acquire essential datasets for the UNESCO project, including several thousand records about all of the public art objects visible from the streets and canals of Venice, Italy. The CK Console is capable of accepting CSV files containing existing data, GIS maps, and accessory media, such as images and sounds.

During the upload sessions, the City Knowledge system automatically gives each urban asset a 'Birth Certificate', with a unique ID and creates a number of web-addressable data branches on the real-time database tree (currently *FireBase*) where each attribute of each record is stored (see Figure 1).

The demonstration then illustrates how the UNESCO mobile app, that we developed on top of our CK platform, allows pieces of Venetian public art that need attention to directly request assistance from app users, via the real-time interaction between the urban agents managed via our *AgentsCloud*. When a user is face-to-face with an object, the object can also make a direct plea for a micro-donation (< €10) straight to the user phone, depending on the funding it knows it needs for its own upkeep (thanks to its "agent").

The calculation of restoration costs is conducted by yet another application, at PreserVenice.org), which also automatically re-calculates preservation priorities, as the system gets updated by various users, via the smarphone app, or via internal searches conducted by its agent, or via user input on the wiki page or the preservenice web app itself. The results of the preservenice calculations are in turn mutually shared with the mobile app and the wiki, thanks to our CK platform.

As clear demonstration of the real-time nature of the agent-mediated system, we show how the PreserVenice site instantly displays each micro-donation made from the phone app as a short "tweet" on the main web page, proving, once again, the real-time nature of the connection between our CK Platform and any app built on top of it.

3 TECHNICAL DETAILS

The City Knowledge platform and CK Console were developed in *Javascript* and are currently implemented using *Firebase* as the real-time repository of data trees. We expect that our own real-time database and publish/subscribe system (called *Acequia)* will replace *Firebase* in future deployments. *AgentsCloud*, which manages the deployment of Agents and the corresponding message-passing and querying capabilities was programmed using *Node.js*.

The mobile app for Venetian Public Art was developed using *Trigger.io* to allow for multi-platform deployment of the underlying HTML5 code, which was also re-used in the companion web application at PreserVenice.org. We expect that our own proprietary technology (called *SensorDomo*) will replace Trigger.io in future deployments, to allow access to the phone sensors from the HTML5 layer.

4 EVALUATION AND FUTURE WORK

After over 20 years of urban research projects and agent-based modeling of cities, we are transitioning to an *agent-oriented programming* approach for urban agents, where we are deploying active software agents into the municipal sensor/actuator fabric [6]. This demonstration represents the official debut of the alpha version of the first functioning system that fully adopts the City Knowledge principles and concepts [1].

Work remains to be done at the lowest level of our system, primarily within *AgentsCloud*, where most of the true advantages of the CK system will be realized (Figure 2).

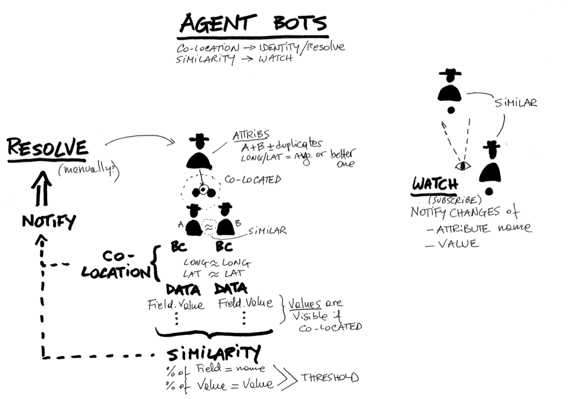

Figure 2. Examples of internal agent behaviors

We will officially release the existing alpha version of the UNESCO system at the time of this workshop and we will track its use for evaluation and debugging purposes.

We hope that this demonstration will stimulate the imagination of the audience with respect to all of the numerous uses of our CK platform in several contexts and domains, both within the Smart Cities movement, and without.

The development of the City Knowledge platform, AgentsCloud, Acequia, SensorDomo and other core technologies will continue in the second phase of the UNESCO project, and in conjunction with other application-specific projects in Boston (StreetBump, parking and street lights), in Park City, UT (regional planning) and in Venice (modeling of pedestrian mobility).

REFERENCES

[1] F. Carrera, *City Knowledge. An Emergent Information Infrastructure for Sustainable Urban Maintenance, Management and Planning*, Doctoral Dissertation, Massachusetts Institute of Technology, (2003).

[2] S. Guerin and F. Carrera, *Autonomous Urban Agents*: *a Santa Fe Approach to City Knowledge*, UGIS Summer Assembly, Santa Fe, June 22-23, 2009.

[3] See *Streetbump.org*

[4] See *Venicenoise.org*

[5] F. Carrera, *Crowdfunding the Preservation of the Public Art of Venice*, American Institute for Conservation, Annual Conference, Albuquerque, May 2012.

[6] F. Carrera and S. Guerin, 'Urban Agents and Citizen Sensors' in *Cities 2.0*, TED book, upcoming.

Reification of emergent urban areas in a land-use simulation model in Reunion Island

Daniel David [1] and **Yassine Gangat** [2] and **Denis Payet** [2] and **Rémy Courdier** [2]

Abstract. Emergent phenomena are often relevant for users and developers of simulation models. But the potential reification of these phenomena raises many questions, conceptually (should they be reified?) and technically (how to do it?). In this paper, we show that such a reification can be considered as an effective way to refine simulation models in which direct modifications, that are made laborious by the multiplicity of the entities and behaviors, often leads to the destabilization of the entire system. We propose a reification technique of the emergent phenomena that do emerge in an agent-based simulation. We illustrate this proposition through the reification of new urban areas, an emergent phenomenon observed in a model that we created to simulate land-use evolutions in Reunion Island.

1 INTRODUCTION

Emergence is a fascinating concept for scientists from different backgrounds. In the context of modeling and simulation, it is often known as a concept encouraging the choice of MultiAgent Systems (MAS) in comparison to other existing techniques. Thus, lots of works have allowed definitions and classifications of emergent phenomena observed in a system, while some of them have tackled the question of their potential reification.

We consider that today the problem is not really to succeed in reifying a known phenomenon, but it is to answer to why its reification should be done or not, and what are the steps that can lead such a reification. Therefore we can legitimately think that there are some cases where it will be useful, while there are other cases where its usefulness remains more uncertain. Consequently, the initial question is what are the cases where the reification of potential emergent phenomena takes an interest, and we believe that this knowledge relies on the context in which each study is done.

In this paper, we briefly present the DS Model, a model that simulates land-use evolutions in Reunion Island. This model is the result of the work of many researchers since 2007. We then present the new urban areas, an emergent phenomenon that is regularly observed in our simulation results. We propose a general architectural framework that will lead us to the reification of such a phenomenon in Agent-Based Simulations (ABS), and we illustrate this proposal with new urban areas in DS.

[1] CIRAD, UPR 78 Recyclage & Risque,
Station de la Bretagne, BP20, 97408 Saint-Denis cedex9, La Réunion
email: daniel.david@cirad.fr
[2] Université de La Réunion, EA 2525 LIM IREMIA,
2 rue Joseph Wetzel, 97490 Sainte-Clotilde, La Réunion
email: {yassine.gangat, denis.payet, remy.courdier}@univ-reunion.fr

2 REUNION ISLAND AND THE DS MODEL

Reunion Island is a French territory of 2500 km^2 in the Western Indian Ocean. It has a strong growth in a limited area with an actual population of 800,000 inhabitants that will probably be more than 1 million in 2030 [1]. This demographic change opens the door to many issues, including housing: even when assuming high densification hypothesis, the demand for urban land will increase of several thousand hectares.

The evolution of this territory must then be done according to a clear urbanization policy and planning documents regulating the evolution of urbanization of the island should take into account these projections within the bounds of possibility, as it is evident that a rule-less urbanization is not a viable long-term scenario. In addition, since 2010, 100,000 hectares of natural areas of the island are included in the UNESCO World Heritage due to their beautiful landscapes and their amazing biodiversity potential. And 40,000 hectares, including historical sugarcane areas, are used by agricultural activities that need to be at least preserved.

So, as noted in [1], in terms of land-use planning, Reunion Island must take up the challenge of hosting a growing population while developing its agricultural land and protecting its natural areas and outstanding landscape. In such a context, and in order to fill the blank in terms of tools dedicated to land-use foresight [6], the implementation of the DS Model (named by the contraction of the names Domino and Smat, the two projects that led to its realization) has been initiated in 2006 [3, 6]. This model is fruit of a collaboration between many partners [2, 7], researchers (CIRAD, Reunion Island University, IRD...) and decision-makers (Reunion Island Regional Council). This model can simulate at the same time the evolution of the population and the land-use (urban, agricultural or natural) changes on the island territory.

Implemented on the simulation platform GEAMAS-NG, its successive developments have made it a model in which there are a large number of entities whose behaviors and interactions are rich and varied: thousands (up to 250,000) of *Parcels* agents (land units of approximately one hectare) live together with agents representing the different institutional layers of the island (1 *Region* agent, 4 *Micro-regions* agents, 24 *Cities* agents...). Results from simulations are used to illustrate (*e.g.* in the form of cartographic outputs such as in Fig. 1) locally large scenarios of land use.

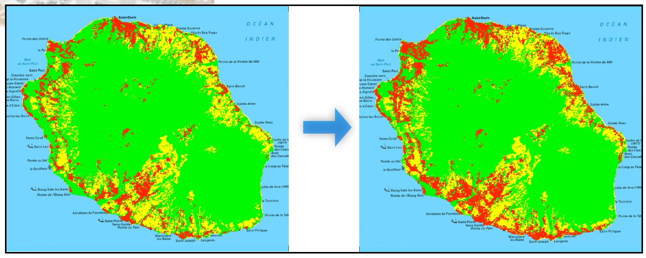

Figure 1. A simulation with the DS Model (Initial state in 2003 on the left, Final state in 2030 on the right). Urban areas are in red, agricultural areas in yellow and natural areas in green.

3 EMERGENCE OF NEW URBAN AREAS

When the DS Model is used to perform some simulations, whatever the scenarios, the results show us the very important part of urbanization in the island. Indeed, the population of Reunion Island is such increasing that even with strong assumptions of housing densification (configurable in DS), the need for housing and various constructions related to this increase (business parks, commercial areas…) inevitably increase.

When studying more closely the simulation results, we realize that some specific urban areas appear over time: areas that are urbanized rapidly, during just a few simulation cycles and that correspond to territories very concentrated, once considered to be natural or agricultural areas.

This kind of phenomena is locally well known as many areas of this type can be spotted on the island's territory. They generally correspond to areas for which regulations have been modified in the planning documents: PLU (Local Urbanization Plans, at the communal scale), SCoT (Territorial Coherence Schemes, at the micro-regions scale) and SAR (Regional Planning Scheme, at the regional scale). This is especially what happens when large tracts of agricultural land are degraded, making them buildable for rapid urbanization, whether for homes designed to fill the need for new housing or facilities, warehouses, halls, which will appear for companies setting up in new areas of activity.

In this paper, we will consider the virtual development of an area of Saint-Pierre. This area, still virgin a few years ago, is now occupied by many commercial buildings, and, according to our simulations, seems designated to experience greater urbanization. A sample output of a simulation (from 2003 to 2030) over an area containing this plot is given in Fig. 2.

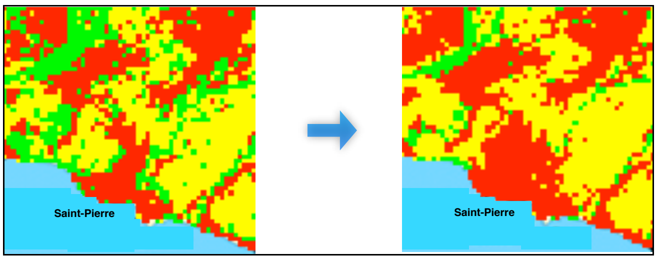

Figure 2. A simulation on the Saint-Pierre area (Initial state in 2003 on the left, Final state in 2030 on the right). Urban areas are in red, agricultural areas in yellow and natural areas in green.

In this figure, there is a significant development of urbanization. But compared to other samples that we are able to product, it is difficult to visually distinguish and to define the emergent phenomenon corresponding to a new urban area (or even just to say whether we are witnessing such a phenomenon). If we wish to observe such phenomena, it requires the use of detection techniques other than the simple eyes of experts who focus on the results.

But when we faced this type of phenomena, we often have (legitimately?) the desire to go beyond their “simple” detection and approach the broader issue of their reification.

4 THE REIFICATION OF EMERGENT PHENOMENA

Reification of emergent phenomena is a subject often mentioned in works related directly or indirectly to the emergence, particularly in the MAS community, but which is rarely defined. In our case, we consider that the reification of an emergent phenomenon in an ABS is a process that takes place in two phases (possibly dissociated): a detection phase and a phase of materialization [4, 5]. This process raises many conceptual and technical questions.

4.1 To reify or not to reify?

We can legitimately wonder about the validity of the simple desire of reification of an emergent phenomenon. Of course, experts and users of a model have nothing to lose (but everything to gain?) when they hope to detect any emergent phenomena, because they will improve their knowledge on the model studied and thereby on the real system modeled. Moreover, as emergent phenomena are often considered part of the expected results of a simulation, they should therefore be highlighted.

We will not discuss here on the various software techniques that can detect emergent phenomena, they are numerous (research techniques pattern [12], techniques based on building of interaction graphs [8], techniques based on *emergence laws* and *emergence revelators* [4, 5]) if we consider (as in our case) that an emergent phenomenon is only contingent of the eye that looks at it and the level of expertise associated with it. For example, a given emergent phenomenon would be obvious for a geographer but would not even exist in the eyes of an economist (or vice versa, obviously). It will be the same in virtual systems in which emergent phenomena could be detected easily with the degree of knowledge of the system itself (or entities or mechanisms responsible of that detection).

But regarding the materialization phase, which would fill out the reification of an emergent phenomenon, the question of its being arises for good reasons. Indeed, on a philosophical level, if one seeks to give shape to an emergent phenomenon within a system in which it would have emerged, doesn't it lose its emergent nature? Moreover, reifying an emergent phenomenon in a system also means that we tend to change the original model with the risk of destabilizing it and lose its essence, this very one which leads to the emergence of the considered phenomenon.

The choice to complete the reification of emergent phenomena potentially detected is a choice we should not trifle with. If this choice is made, we must ideally do this reification without destabilizing the initial model produced and implemented, as long as this implementation does not require a thorough look to make possible the desired reification (which is often the case in large scale projects). In order to do this, we propose the use of special *emergence structures* that allow materialization of the emergent phenomenon in an ABS.

4.2 Emergence structures

In general, emergent phenomena detected in the real world often manifest behaviors that make the very existence of these phenomena affect the real world entities: some of these entities may participate directly in the emergence of phenomena, while others are influenced by these phenomena, and still others have their perception modified by the presence of these phenomena. This is obviously the same in the virtual world that we are handling in an ABS, since our goal is to reproduce phenomena that occur in systems or processes of the real world. Thus it is important to offer solutions in order to represent these phenomena in the architecture of an ABS platform. That's why we propose to use two types of ***emergence structures***: *emergence agents* and *interposition elements*, which are shown in Fig. 3.

An ***emergence agent*** is an intelligent agent that runs within an ABS platform. It evolves in the same environment(s) as all other agents in the system and interacts with them through mechanisms of influence and perception that underlie the host platform. If necessary, several agents can be created to reify the same emergent phenomenon.

An ***interposition element*** is a structure allowing change of one or more agents from one or more environments in which they operate. Such a structure modifies (as appropriate by altering them, improving them, restricting them, etc.) mechanisms of perception or influence used by the agents of the ABS.

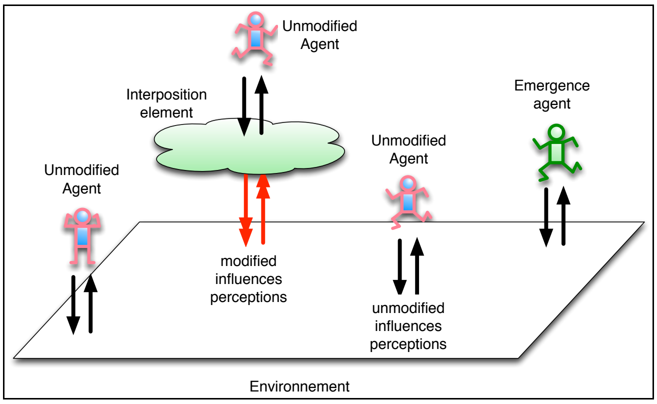

Figure 3. *Emergence structures* (an *emergence agent* and an *interposition element*) with modified perceptions/influences

Both types of structures, seen as complementary or independent, allow us to take into account different types of phenomena that occur in a studied ABS. Thus, in an example of intrinsic emergence (as defined in the classification of Boschetti [2]), like the apparition of a school of fish. The school of fish will be represented directly in the system by the *emergence agent*. And the different fishes that constitute the school of fish will continue to move in their environment but will have their perceptions and influences changed by *interposition elements*.

In the same way of thinking, we can include an example of low emergence (as it is defined in [11]): twigs (objects in the environment), that have been stockpiled by termites agents, making emergence of a pile of stick. Here the woodpile does not have its own behavior, there is therefore no need to create an *emergence agent* to represent it. However, if certain entities of the system must perceive the woodpile as such, this will be possible through *interposition elements* that will change perceptions and influences of these entities.

One of the real benefits of this technique is that it does not require modifications of the code of the agents involved in our emergent phenomenon. Changes of these agents' behaviors are only a side effect due to the presence of *emergence structures* related to them. Therefore, the agents that are not concerned by the emergence of a particular phenomenon (which generally constitute the vast majority of agents in the ABS) would never be destabilized.

5 THE REIFICATION OF URBAN AREAS IN THE DS MODEL

The phenomenon of new urban areas previously described corresponds, in terms of urbanization, to a real emergent phenomenon that requires further study. In particular, it would be interesting to allow the detection of such phenomena in the DS Model. Thus they would be noticed before the (meticulous and fastidious) analysis of experts from the results and maps generated by the simulations. And it would also be interesting to consider the emergence of these new urban areas in the system to test various hypotheses of urbanization associated with them and the consequences they induce.

Obviously, although the *Parcels* agents that compose it have all an urban state, every new urban area is not considered in the same way. We can easily imagine that their potential behaviors may differ depending whether the recent urbanization is for example housing areas or business parks. We can therefore use the *emergence structures* that we have defined in order to detect new emergent urban areas in the DS Model and to materialize them, so we would be able to experiment different scenarios and assumptions.

In the following part, we will therefore describe the detection and materialization phases whose realization leads to the reification of the considered phenomenon.

5.1 Detection phase in the DS Model

The first stage of the reification process of new urban areas in the DS Model is to be able to detect the formation of these zones during simulations. This is done through platform mechanisms we have implemented in GEAMAS-NG [9, 10] to which we (as users of the system) must give elements to describe the emergent phenomenon as a new urban area. Thus, Fig. 4 illustrates a new urban area, appeared in the simulation shown in Fig. 2, which is composed of thirty *Parcels* agents. All of them were detected using an indicator to detect the emergence of *Parcels* that are at least 5 in number, in the same proximity, and had their urbanization performed in the same time period of 5 years. Geographer experts and specialists in urbanization have indicated these numerical values to us during the experimentation process.

Figure 4. The emergent phenomenon "new urban area" that has been detected (in pink) with the actual aerial view of its area

In this figure, we can notice the set of cells that are grouped together in the detected new urban area. This area is also well known locally, because it is a recent ZAC (a local activities development zone) of Saint-Pierre: the Canabady ZAC). We can also easily see the town of Saint-Pierre, already highly urbanized, and the little urbanized area in which our emergent phenomenon will occur in our simulations (the circled area), but in which there are already the first buildings of the Canabady ZAC on the right part. In this aerial view, we can also note many farmland areas, cultivated, that, according to the choices made in the different simulation scenarios, could be devoted to urbanization in the coming years.

This detection of new urban areas that are likely to appear in the simulations with the DS Model, allows us to provide support to experts dealing with the analysis of the simulation results. In that sense, this experiment is therefore an important proposition of progress by the possibilities offered by the stable and utilized versions of the model. But to go beyond the "simple" assisted detection, we will now show how the new urban areas that have emerged can be materialized in the ABS.

5.2 Materialization phase in the DS Model

The first thing to be done, in order to materialize new urban areas that emerge in a simulation, is to examine how this phenomenon will be integrated into the ABS, through *emergence structures*. We can also ask ourselves, according the value we want to give to the phenomenon, if it must be materialized using only *interposition elements*, if we should use an *emergence agent* only, or if we should move towards a joint use of both types of structures.

In our example of new urban areas, we began by analyzing how the system entities are involved in this phenomenon. Naturally, there are *objects* of the environment and *agents* (*Parcels*, *Cities*, *Micro-regions*…) which are within its geographical area and are directly concerned by the emergence of a new urban area, while *agents* and *objects* that are quite far from it and are not directly involved in its emergence. If we want the materialization of new urban areas in the ABS to be useful, it is obvious that we must at least establish *interposition elements* with *Parcels* agents who are concerned with the emergent phenomenon. This will allow us to test different assumptions affecting the evolution of these agents that were internal to the DS Model, without editing them directly.

But we should bear in mind that the smallest entities are not the only ones affected by the emergence of a new urban area. Indeed, the DS Model is composed of different levels of agents and each new urban area emerges within a particular *City*, in particular *Micro-region*, and inside a global *Region* itself. So we must take this into account for the corresponding agents and implement elements of interposition for the *Region* agent, for each *Micro-region* agent and for each *City* agent concerned by the emergent phenomenon.

Finally, we can materialize our new urban areas by establishing for each new urban area, an *emergence agent* that will assign a specific behavior to the phenomenon and that can interact with the environment of the DS Model *via* its own influences and perceptions. Again, this will allow us to test various hypotheses of evolution.

In our experiment, we chose to reify each new urban area that would emerge in a simulation using:

- An emergence agent.
- An interposition element used for Micro-regions agents (in our example only for the Southern Micro-region agent, but it is possible that a new emergent urban area emerges on the territories of several Micro-regions agents).
- An interposition element used for Cities agents (in our example only the Saint-Pierre City agent, but it is possible that a new emergent urban area emerges on the territories of several Cities agents).

These *interposition elements* are enough in our experiments, as we consider reasonable to assume that all agents of the same kind that are involved in an emergent phenomenon will be affected the same way. However, it seems obvious that agents of different types (and scales) will be affected differently. In our example, these choices can be explained because the *Southern Micro-region* agent and the *Saint-Pierre City* agent are the ones in the middle of the urbanization process for the new urban area detected at Saint-Pierre.

Regarding the *emergence agent* representing the phenomenon, it will help us, through the behavior that it will be given, to test various hypotheses in order to refine the general behavior of the DS Model in relation to the emergence of this particular phenomenon. This intelligent agent is the *Urbanization Manager* agent of the area.

5.3 Results

The main interest that emerges from the reification process of new urban areas is to help users of the DS Model. They could test different hypotheses that can refine the behavior of the model in order to reflect the specific requirements of the phenomena that have been put forward.

Indeed, the DS Model allows, in its original version, to take into account the behavior at the scale of the *Parcel*, the *Region* and the *Micro-regions*. It is clear that the hundreds of thousands of small *Parcels* agents, that all have specific characteristics related to their location, have therefore a sufficient precision to assume that the treatments they perform are adequate on these specificities. But for agents of larger scale, such as *Cities* agents, which are in the middle of the hierarchy of *Parcels*/*Cities*/*Micro-Regions*/*Region* agents, the initial expected behavior in the DS Model are sometimes too general to consider specificities such as new urban areas.

With the *emergence structures* we established, we could indirectly alter the evolution of the entities of the DS Model. In our example, the conducted experiments allow us to test hypotheses in order to refine the behavior of the *Saint-Pierre City* agent in the area of our new urban area. Concretely, if we want to make this area a commercial area rather than a housing area, the *interposition* *elements* used with the *Saint-Pierre City* agent and the *Southern Micro-region* agent allow us to "hide" from them the *Parcels* agents affected by the new urban area. For example, they can no longer consider them when they are looking forward to allocate the new population calculated on the territory. And, considering relevant informations like land-use policies and development wills, the *Urbanization Manager* agent of the new urban area can decide to influence entities of the ABS that are in the territory of the new urban area. It allows for example to increase the attractiveness potential of the territory of the reified new urban area and that would then simulate a faster urbanization.

Fig. 5 illustrates the results of two simulations where the exact same rapid urbanization has been observed and where we have detected the new urban area present in the city of Saint-Pierre. But we materialized them in two different ways. The two images have a gradation of red to represent the population density observed at the end of both simulations:

- In the first case (on the left), the new urban area is materialized by considering that it would be mostly dedicated for residential units.
- In the second case (on the right), the new urban area is materialized by considering that it would be mostly dedicated to commercial and business buildings.

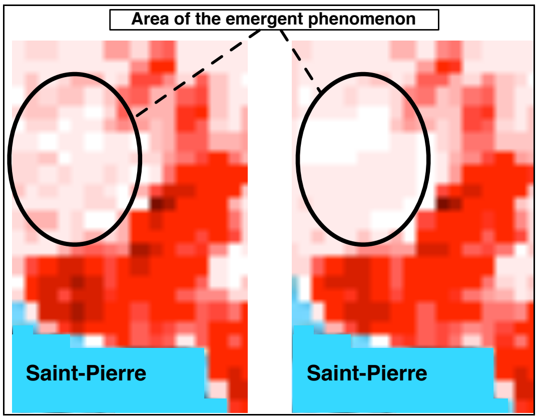

Figure 5. The population density observed in two simulations results (on the left the new urban area is materialized by considering that it is for housing, on the right it is materialized by considering that it is for business premises; the population density of the area is higher on the left)

We can note the difference in shades of red in the encircled area, indicating logically that people began to be distributed over the area when its urbanization in the first case but not in the second one, where for which, disturbed by elements of interposition, the *Southern Micro-region* agent and the *Saint-Pierre City* agent have not assigned new population to *Parcels* agents contained in the area. In all this experiment, the behavior of the system has change while the behaviors (and so the code) of the *Parcels*, *Micro-Regions*, *Cities*, and *Region* agents have never been modified directly.

6 CONCLUSION

In this paper, we studied the problem of reification of phenomena that emerge in an ABS. To that end, we presented the DS Model, a model that allows us to simulate land-use evolutions in Reunion Island. We focused on the study of new urban areas, a particular phenomenon that emerges in many simulations. As an experiment, we have detailed how it was possible to make the process of reification of these new urban areas by relying on the use of *emergence structures* that we have defined and mechanisms that we have implemented in the GEAMAS-NG platform.

These proposals are not aimed to deliver *the* solution to take into account any emergent phenomenon in an ABS, because to achieve the reification of a phenomenon we must, as we have seen through the experimentation of new urban areas and particularly during their phase of materialization, go through (sometimes fastidious) stages of analysis, modeling and programming of the *emergence structures* constituted by *interposition elements* and *emergence agents*.

But the sequence of this experiment shows that our approach allows integrating the consideration of emergent phenomena in simulation models in which it was not anticipated. And we can extend the functionalities of real-case models (whose complex structure often makes difficult any changes of behaviors of certain entities without causing a global imbalance of the complete model itself) like the DS Model (which was used by local decision-makers in Reunion Island) to refine their general behavior in order to reflect new specificities. This puts us therefore in the middle of the processes of injection and production of knowledge *in* and *through* a simulation.

[1] Agorah, *Une politique foncière, une des clefs pour aménager le territoire*, Agence d'urbanisme de La Réunion, 2006.

[2] Boschetti, F., Gray, R., A Turing test for emergence, *Advances in applied self-organizing systems*, pp. 349-369, (2007)

[3] Botta, A., Daré, W., Antona, M., Leclerc, G., Integration of multi-scale stakes in governance by applying companion modelling to land-use foresight, *MODSIM'09 International Congress on Modelling and Simulation*, pp. 2377-2383 (2009)

[4] David, D., Courdier, R., See Emergence as a Metaknowledge, a Way to Reify Emergent Phenomena in Multiagent Simulations?, *International Conference on Agents and Artificial Intelligence* (ICAART'09), pp. 564-569 (2009)

[5] David, D., Courdier, R., Emergence as Metaknowledge: Refining Simulation Models through Emergence Reification, *European Simulation and Modelling Conference* (ESM'08), pp. 25-27 (2008)

[6] David, D., *Prospective territoriale par simulation orientée agent*, PhD Thesis, Université de La Réunion, 2010.

[7] Lagabrielle, E., Botta, A., Daré, W., David, D., Aubert, S., Fabricius, C., Modelling with stakeholders to integrate biodiversity into land-use planning - Lessons learned in Réunion Island, *Environmental Modelling and Software*, vol. 25, n. 11, pp. 1413-1427 (2010)

[8] Moncion, T., Amar, P., Hutzler, G., Automatic characterization of emergent phenomena in complex systems, *Journal of Biological Physics and Chemistry*, vol. 10 (2010)

[9] Payet, D., Courdier, R., Sébastien, N., Ralambondrainy, T., Environment as support for simplification, reuse and integration of processes in spatial MAS, *Information Reuse and Integration Conference* (IRI'06), pp. 127-131 (2006)

[10] Ralambondrainy, T., Courdier, C. Synthesizing agent interactions through the concept of conversation, *Summer Computer Simulation Conference* (2007)

[11] Stephan, A., Emergence, *Encyclopedia of cognitive science*, Vol. 1, pp. 1108-1115 (2009)

[12] Tranouez, P., Bertelle, C., Olivier, D., Changing levels of description in a fluid flow simulation, *Emergent Properties in Natural and Artificial Dynamical Systems*, Understanding Complex Systems series, pp. 1108-1115 (2006)