

A CA-Based Approach to Self-Organized Adaptive Environments: the Case of an Illumination Facility

Stefania Bandini, Andrea Bonomi, Giuseppe Vizzari
Complex Systems and Artificial Intelligence research center
Università degli Studi di Milano–Bicocca
Viale Sarca 336/14, 20126 Milano, Italy
{bandini,bonomi,vizzari}@disco.unimib.it

Vito Acconci, Nathan DeGraaf, Jono Podborseck and James Clar
Acconci Studio, 20 Jay St., Suite #215, Brooklyn, NY 11201
studio@acconci.com

Abstract

The ambient intelligence scenario depicts electronic environments that are sensitive and responsive to the presence of people. This paper will present an ambient intelligence system that is not aimed at providing some form of electronic service to its users, but whose goal is rather to enhance the experience of pedestrians moving inside the related physical environment. In particular, in our approach the environment itself is endowed with a set of sensors that perceive the presence of humans (or other physical entities such as dogs, bicycles, cars) and interact with a set of actuators that choose their actions in an attempt improve the overall experience of these users. We will adopt the example of an adaptive illumination facility to describe the problem and the proposed solution. The model guiding the interaction and action of sensors and actuators is a dissipative multilayered cellular automata, supporting a self-organization of the system as a response to the presence and movements of people inside it. The model does not require a centralized form of control, since it is made up of specific autonomous computational units locally perceiving a part of the environment and interacting with other interconnected units to coordinate their actions. The paper will introduce the model, as well as the results of simulations of its application in a concrete case study. A brief discussion of the potential use of this model and approach beyond this scenario will conclude the paper.

1. Introduction

The ambient intelligence scenario [17] depicts future human environments endowed with a large number of electronic devices, interconnected by means of wireless communication facilities, able to perceive and react to the presence of people.

The goals of these facilities can be very different, from providing electronic services to humans accessing these environment through computational devices (such as personal computers or PDAs), to simply providing some form of ambient adaptation to the users' presence (or voice, or gestures), without thus requiring him/her to employ a computational device. Ambient intelligence comprises thus those systems that are designed to autonomously adapt the environment to the people living or simply passing by in it in order to improve their everyday experience.

Besides the specific aims of the ambient intelligent system, there is an increasing interest and number of research efforts on approaches, models and mechanisms supporting forms of self-organization and management of the components (both hardware and software) of such systems. The latter are growingly viewed in terms of autonomous entities, managing internal resources and interacting with surrounding ones so as to obtain the desired overall system behaviour as a result of local actions and interactions among system components. Examples of this kind of approach can be found both in relatively traditional pervasive computing applications (see, e.g., [9]), but also in a new wave of systems developed in the vein of amorphous computing [2] such as the one on paintable computers described in [8]. In this rather extreme application a whole display architecture is composed of autonomous and interacting graphic systems,

each devoted to a single pixel, that must thus interact and coordinate their behaviours even to display a simple character.

This paper describes a biologically inspired approach to the modeling and realization of a self-organizing ambient intelligence system; in particular the adopted approach employs Cellular Automata as a model supporting self-organization among cells comprising sensors and actuators. The former can trigger the behaviours of the latter, both through the interaction of elements enclosed in the same cell and by means of the local interaction among adjacent cells. The transition rule adopted for the CA was derived by previous applications to reproduce natural phenomena such as percolation processes of pesticides in the soil, in specific percolation beds for the coffee industry and for the experimentation of elasticity properties of batches for tires [3, 6], by modeling mechanisms of reaction and diffusion. In this specific application this rule is used to manage the interactions of cells arranged through a multilayered architecture [5], better suited to represent an artificial environment comprising a set of sensors that perceive the presence of humans (or other physical entities such as dogs, bicycles, cars), and actuators that choose their actions in an attempt improve the overall experience of these users. Throughout the paper we will adopt the example of an adaptive illumination facility, that is being designed and realized by the Acconci Studio in Indianapolis.

The model that we developed is the kernel of a system supporting the design and definition of said facilities, through the simulation and envisioning of its dynamic behaviour according to specific values for the relevant parameters (e.g. parameters of the transition rule of the CA, but also the number of lights and sensors, and so on). Part of the simulator generates patterns of movement of pedestrians that represent inputs for the CA and another part of the system generates a visualization of the system dynamics, interpreting the states of the CA. These parts could be actually removed and the system could be directly interfaced to field sensors and actuators, effectively piloting them, in a centralized approach. Alternative and more adequate distributed hardware/software architectures could be employed, such as in the aforementioned approaches; nevertheless the CA and its transition rule represent a formal and executable specification of the behaviour of system components.

The following section will introduce the specific scenario in which this research effort is set, describing the requirements for the adaptive illumination system and the environment adaptation model. Section 3 introduces the modeling approach, setting it in the relevant literature, while section 4 describes the developed model in details. A description of the developed environment supporting designers will follow, then conclusions and future works will end the paper.

2. Scenario

The Acconci Studio was founded in 1988 to help realize public-space projects through experimental architecture and public art efforts. The method of Acconci Studio is on the one hand to make a new space by turning an old one inside-out and upside-down; and on the other hand to insert within a site a capsule that grows out of itself and spreads into a landscape. They treat architecture as an occasion for activity; they make spaces fluid, changeable, portable. They have recently completed a person-made island in Graz, a plaza in Memphis, a gallery in NY, a clothing store in Tokyo; they are currently working on a building façade in Milan, a park on a street median in Vienna, and a skate park in San Juan¹.

The Studio has recently been involved in a project for the renovation of a tunnel in the Virginia Avenue Garage in Indianapolis. The tunnel is currently mostly devoted to cars, with relatively limited space on the sidewalks and its illumination is strictly functional. The planned renovation for the tunnel comprises a set of interventions along the direction defined by the following narrative description of the project:

The passage through the building should be a volume of color, a solid of color. Its a world of its own, a world in itself, separate from the streets outside at either end. Walking, cycling, through the building should be like walking through a solid, it should be like being fixed in color.

The color might change during the day, according to the time of day: pink in the morning, for example, becomes purple at noon becomes blue, or blue-green, at night. This world-in-itself keeps its own time, shows its own time in its own way.

The color is there to make a heaviness, a thickness, only so that the thickness can be broken. The thickness is pierced through with something, theres a sparkle, its you that sparkles, walking or cycling though the passage, this tunnel of color. Well no, not really, its not you: but its you that sets off the sparkle a sparkle here, sparkle there, then another sparkle in-between one sparkle affects the other, pulls the other, like a magnet a point of sparkle is stretched out into a line of sparkles is stretched out into a network of sparkles.

These sparkles are above you, below you, they spread out in front of you, they light your way through the tunnel. The sparkles multiply: its you who sets them off, only you, but – when another

¹<http://www.acconci.com>



Figure 1. A visual elaboration of the desired adaptive illumination facility (the image appears courtesy of the Acconci Studio).

person comes toward you in the opposite direction, when another person passes you, when a car passes by some of these sparkles, some of these fire-flies, have found a new attractor, they go off in a different direction.

The above narrative description of the desired adaptive environment comprises two main effects of illumination, also depicted in a graphical elaboration of the desired visual effect shown in Figure 1:

- an overall effect of uniformly coloring the environment through a background, ambient light that can change through time, but slowly with respect to the movements and immediate perceptions of people passing in the tunnel;
- a local effect of illumination reacting to the presence of pedestrians, bicycles, cars and other physical entities.

The first type of effect can be achieved in a relatively simple and centralized way, requiring in fact a uniform type of illumination that has a slow dynamic. The second point requires instead a different view on the illumination facility. In particular, it must be able to perceive the presence of pedestrians and other physical entities passing in it, in other words it must be endowed with sensors. Moreover, it must be able to exhibit local changes as a reaction to the outputs of the aforementioned sensors, providing thus for a non uniform component to the overall illumination. The overall environment must be thus split into parts, proper subsystems.

However, these subsystems cannot operate in isolation, since one of the requirements is to achieve patterns of illumination that are local and small, when compared to the size

of the tunnel, but that can have a larger extent than the space occupied by a single physical entity (“sparkles are above you, below you, they spread out in front of you, they light your way through the tunnel”). The subsystems must thus be able to interact, to influence one another to achieve more complex illumination effects than just providing a spotlight on the occupied positions.

In the following part of the paper we will focus on this more dynamic and reactive part of the overall illumination facility. The need to consider a physical environment as an assembly of local subsystems arranged in a network, each able to decide on its own state according to a local stimulus and according to the influences of neighbouring subsystems led us to consider Cellular Automata as a suitable model to capture and reproduce the above described specification for the illumination facility.

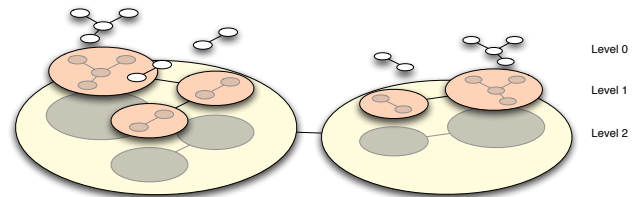


Figure 2. An example of a nested graph of level 2.

3. Related Works

Cellular Automata (CA), introduced by John von Neumann as an environment for studying self-replicating systems [18], have been primarily investigated as theoretical concept and as a method for simulation and modeling [19]. They have also been used as computational framework for specific kind of applications (e.g. image processing [16], robot path planning [7]) and they have also inspired several parallel computer architectures, such as the Connection Machine [13] and the Cellular Automata Machine [15].

Automata Networks [11] are a generalization of the classic CA, based on the introduction of the network abstraction between automata nodes. Multilayered Automata Network have been defined in [5] as a generalization of Automata Networks. The main features of the Multilayered Automata Network are the explicit introduction of a hierarchical structure based on nested graphs. Such graphs are composed of vertices and edges where each vertex can be in turn be a nested graph of lower level. An example of a nested graph is shown in Figure 2. A Multilayered Automata Network is directly obtained from the nested graph structure by introducing states and a transition function.

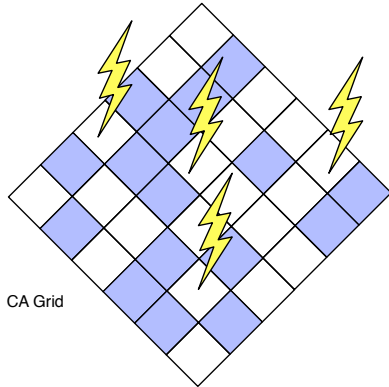


Figure 3. A schema of a Dissipative Cellular Automata in which the cells state is influenced by the environment.

Dissipative Cellular Automata (DCA), defined in [20], are also an extension of CA. DCA differ from the basic model mainly for two characteristics: while CA are synchronous and closed systems, DCA are open and asynchronous. The two approaches to the management of the elaboration of the next states of cells are visually compared in Figure 4. In particular, in DCA the cells are characterized by a thread of control of their own, autonomously managing the elaboration of the local cell state transition rule. DCA can thus be considered as an open agent system [14], in which the cells update their state independently of each other and they are directly influenced by the environment.

In order to take advantages of both the Multilayered Automata Network and the Dissipative Cellular Automata, we introduced a new class of automata called Dissipative Multilayered Automata Network (D-MAN). An informal definition of this model describes D-MAN as Multilayered Automata Network in which the cells update their state in an asynchronous way and they are open to influences by the external environment.

This extension is useful because we want to use the D-MAN as a computational environment to specify and simulate the behaviour of a distributed control system. The systems will be composed of several subsystems that are influenced by the environment in which they are situated and that are able to update their state asynchronously. Moreover each subsystem is able to communicate with its neighbours.

4. The Proposed Approach

We propose the adoption of a Dissipative Multilayered Automata Network (D-MAN) distributed control system in order to achieve the complex challenged previously presented. The proposed distributed control system is com-

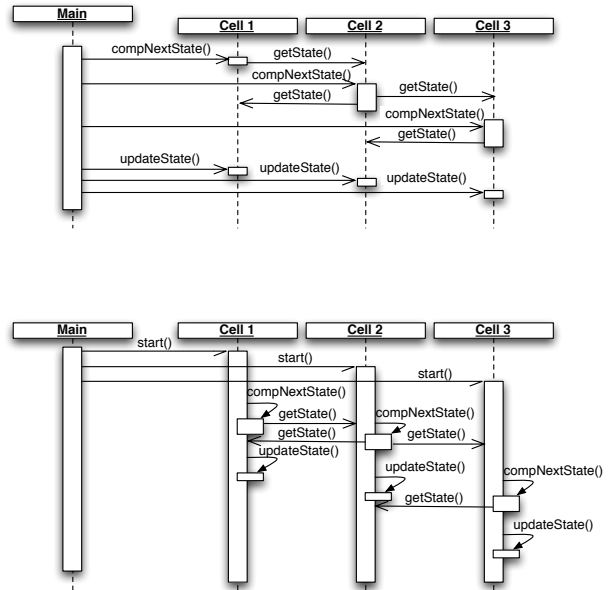


Figure 4. A UML Sequence Diagram comparing the interaction in synchronous and asynchronous CA.

posed of a set of controllers distributed throughout the system. Each controller has the responsibility of a part of the whole system and to collaborate with a subset of the other controllers (identified according to the architecture of the CA model) in order to achieve the desired overall system behavior. In the proposed architecture, every node is a cell of a D-MAN that can communicate only with its neighbours, it processes signals from sensors and it controls a predefined set of lights associated to it. The approach is fully distributed: there is no centralized control and no hierarchy between the controller, not only from a logical point of view but also a physical one. In the following sections, each of the components of the proposed approach will be described in details.

4.1. System Architecture

The designed system is an homogeneous peer system. As shown in Figure 5, every controller has the responsibility of managing the sensors and actuators belonging to a fixed area of the space. Controllers are homogeneous in terms of hardware and software capabilities. Every controller is connected to a motion sensor, which roughly covers the controlled area, some lights (about 40 LED lights) and neighbouring controllers.

As shown in Figure 6, the external layer (level 2) is the communication layer between the controllers of the system.

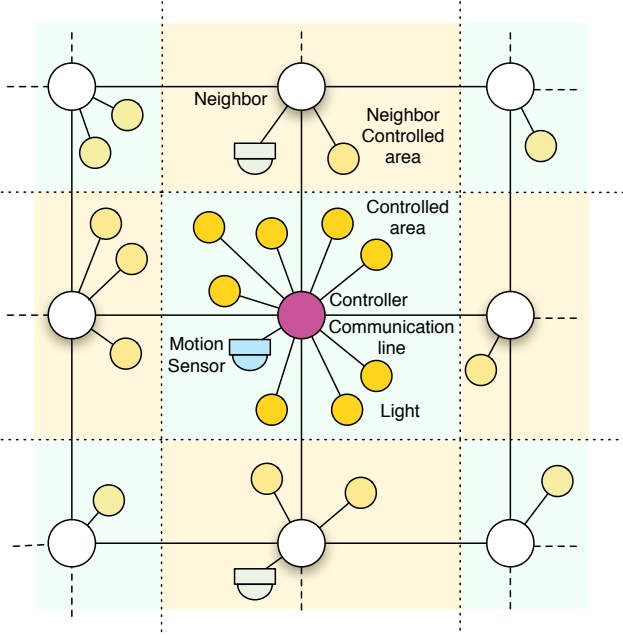


Figure 5. The proposed architecture for the distributed control system to be managed through a D-MAN approach.

Every controller is an automata network of two nodes, one node is a sensor communication layer and represents a space in which every sensor connected to the microcontroller has a correspondent cell. The other node represents the actuators' layer in which the cells pilot the actuators (lights, in our case). Since the external layer is a physical one and every cell is an independent microcontroller, it cannot be assumed that the entire network is synchronized. In some cases, a synchronous network can be constructed (for example, a single clock devices can be connected to each microcontrollers or the microcontrollers can be synchronized by a process without a master node), but the most general case is an asynchronous network.

4.2. Sensors Layer

The Sensor Layer is a Level 0 Dissipative Automata. As previously introduced, it is composed of a single cell, since only one sensor is connected to each microcontroller. It is a Dissipative Automata because the internal state of the cell is influenced by the external environment. The state of the cell is represented by a single numerical value $v_s \in \mathbb{N}_{8bit}$, where

$$\mathbb{N}_{8bit} \subset \mathbb{N}_0, \forall x : x \in \mathbb{N}_{8bit} \Rightarrow x < 2^8$$

The limit value was chosen for performance reasons because 8-bit microcontrollers are widely diffused and they

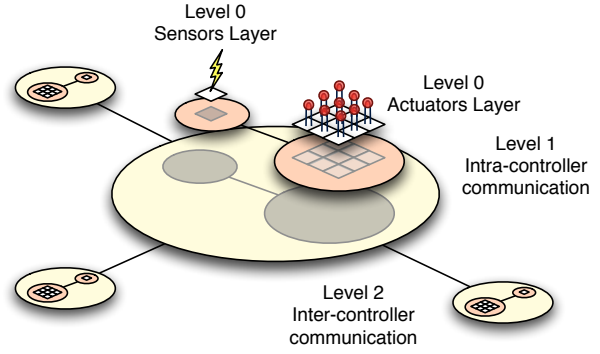


Figure 6. The proposed automata network for the D-MAN.

can be sufficiently powerful to manage this kind of situation. The value of v_s is computed as

$$v_s(t + 1) = v_s(t) \cdot m + s(t + 1) \cdot (1 - m)$$

where $m \in \mathbb{R}, 0 \leq m \leq 1$ is the *memory coefficient* that indicates the degree of correlation between the previous value of v_s and the new value, $s(t) \in \mathbb{N}_{8bit}$ is the reading of the sensor at the time $s(t)$. If the sensor is capable of distance measuring, $s(t)$ is inverse proportional to the measured distance (so, if the distance is 0, the value is 255, if the distance is ∞ the value is 0). If the sensor is a motion detector sensor (it able to signal 1 if an object is present or 0 otherwise) $s(t)$, $s(t)$ is equal to 0 if there is not detected motion, c in case of motion, where $c \in \mathbb{N}_{8bit}$ is a constant (in our tests, 128 and 192 are good values for c).

4.3. Diffusion Rule

In this section we describe the diffusion rule, that is used to propagate the sensors signals through the system. At a given time, every level 2 cell is characterized by an intensity of the signal, $v \in \mathbb{N}_{8bit}$. Informally, the value of v at time $t + 1$ depends of the value of v at time t and on the value of $v_s(t + 1)$, to capture both the aspects of interaction with neighbouring cells and the memory of the previous external stimulus caused by the presence of a physical entity in the area associated to the cell.

The intensity of the signal decreases over time, in a process we call evaporation. In particular, let us define $\epsilon_{evp}(v)$ as the function that computes the quantity of signal to decrement from the signal and is defined as

$$\epsilon_{evp}(v) = v \cdot e_1 + e_0$$

where $e_0 \in \mathbb{R}^+$ is a constant evaporation quantity and $e_1 \in$

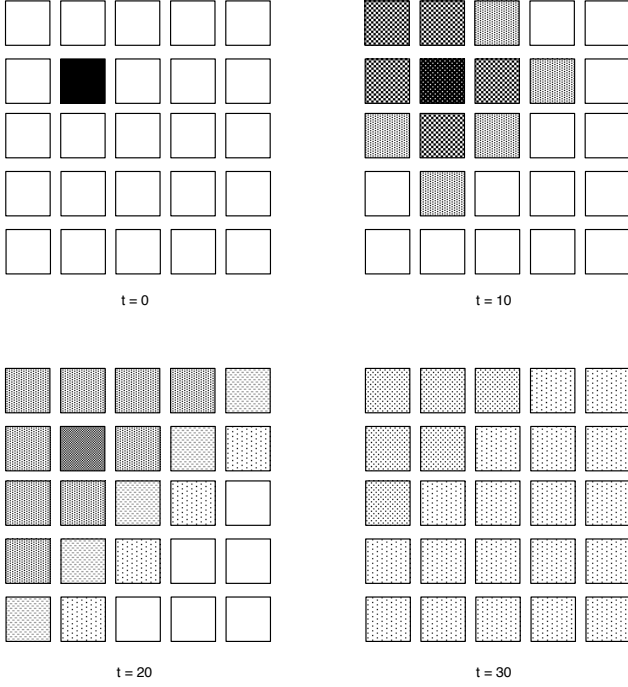


Figure 7. An example of the dynamic behaviour of a diffusion operation. The signal intensity is spread throughout the lattice, leading to a uniform value; the total signal intensity remains stable through time, since evaporation was not considered.

$\mathbb{R}, 0 \leq e_1 \leq 1$ is the evaporation rate (e.g. a value of 0.1 means a 10% evaporation rate).

The evaporation function $evp(v)$, computing the intensity of signal v from time t to $t + 1$, is thus defined as

$$evp(v) = \begin{cases} 0 & \text{if } \epsilon_{evp}(v) > v \\ v - \epsilon_{evp}(v) & \text{otherwise} \end{cases}$$

The evaporation function is used in combination with the neighbours' signal intensities to compute the new intensity of a given cell. We first present the formula for a regular neighbourhood and then we generalize to the irregular structure.

4.3.1 Regular neighbourhood

The automaton is contained in the finite two-dimensional square grid \mathbb{N}^2 . We suppose that the cell $C_{i,j}$ is located on the grid at the position i, j , where $i \in \mathbb{N}$ and $j \in \mathbb{N}$. According to the von Neumann neighbourhood [12], a cell $C_{i,j}$ (unless it is placed on the border of the lattice) has 4 neighbours (as shown in figure 8), denoted by $C_{i-1,j}$, $C_{i,j+1}$, $C_{i+1,j}$, $C_{i,j-1}$.

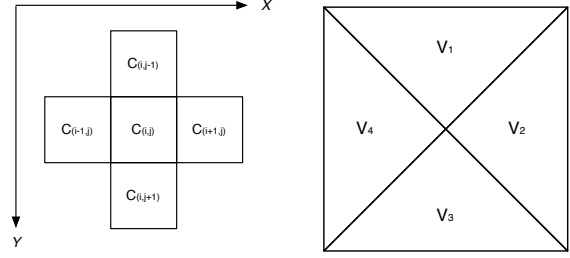


Figure 8. On the left, the von Neumann neighbourhood of the cell $C_{i,j}$, on the right, the internal structure of a cell of the regular automaton.

For simplicity, we numbered the neighbours of a cell from 1 to 4, so for the cell $C_{i,j}$, N_1 is $C_{i-1,j}$, N_2 is $C_{i,j+1}$, N_3 is $C_{i+1,j}$, and N_4 is $C_{i,j-1}$

At a given time, every cell is characterized by an intensity of the sensor signal. Each cell is divided into four parts (as shown in Figure 8), each part can have a different signal intensity, and the overall intensity of the signal of the cell is the sum of the parts intensity values. The state of each cell $C_{i,j}$ of the automaton is defined by $C_{i,j} = \langle v_1, v_2, v_3, v_4 \rangle$ where $v_1, v_2, v_3, v_4 \in \mathbb{N}_{8bit}$ represent the intensity of the signal of the 4 subparts. $V_{i,j}(t)$ represents the total intensity of the signals (i.e. the sum of the subparts signal intensity) of the cell i, j at time t . The total intensity of the neighbours are denoted by V_{N1} , V_{N2} , V_{N3} , and V_{N4} . The signal intensity of the subparts and the total intensity are computed with the following formulas:

$$v_j(t+1) = \begin{cases} \frac{evp(V(t)) \cdot q + evp(V_{N_j}(t)) \cdot (1-q)}{4} & \text{if } \exists N_j \\ \frac{evp(V(t))}{4} & \text{otherwise} \end{cases}$$

$$V(t+1) = \sum_{i=1}^4 v_i(t+1)$$

where $q \in \mathbb{R}, 0 \leq q \leq 1$ is the conservation coefficient (i.e. if q is equals to 0, the new state of a cell is not influenced by the neighbours values, if it is equals to 0.5 the new values is a mean among the previous value of the cell and the neighbours value, if it is equals to 1, the new value does not depend on the previous value of the cell but only from the neighbours). The effect of this modeling choice is that the parts of cells along the border of the lattice are only influenced through time by the contributions of other parts (that are adjacent to inner cells of the lattice) to the overall cell intensity.

4.3.2 Irregular neighbourhood

The irregular structure automata is a generalization of the regular one. The automaton is composed of cell numbered from 1 to N , so we use C_i for $0 \leq i \leq N$ to indicate the i -th cell. Every cell C_i can have an arbitrary number of neighbours $L_i, 0 \leq L_i \leq L \leq N - 1$ where L_i is the numbers of neighbours of the cell C_i and $L = \max(L_i)$ is the maximum numbers of neighbours of every cell the system. Neighbouring cells of cell i can be denoted as $N_{i,l}$.

As for the regular neighbourhood case, each cell is divided into L parts, each part can have a different signal intensity, and the overall intensity of the signal of the cell is the sum of the parts intensity values. The state of each cell C_i of the automaton is defined as $V_i = \sum_{l=0}^{L_i} v_{i,l}$ where $v_{i,l} \in \mathbb{N}_{8bit}$ represent the intensity of the signal of the L subparts. Finally, the intensity of each neighbouring cell of C_i is denoted by $V_{i,l}$.

The signal intensity of the subparts and the total intensity can thus be computed according to the following formulas:

$$v_{i,l}(t+1) = \begin{cases} \frac{exp(V_i(t)) \cdot q + exp(V_{i,l}(t)) \cdot (1-q)}{L} & \text{if } \exists N_{i,l} \\ \frac{exp(V_i(t))}{L_i} & \text{otherwise} \end{cases}$$

$$V_i(t+1) = \sum_{l=1}^{L_i} v_{i,l}(t+1)$$

In the real system, the maximum number of neighbours (L) is constrained by the number of available inputs on the microcontrollers.

4.4. Actuators Layer

The cells of the actuator layer determinate the actuators actions. In this project the actuators are LED lamps that are turned on and of according the the state of the cell. Instead of controlling a single LED from a cell, every cell is related to a group of LEDs disposed in the same (small) area.

In the case of regular neighbourhood, each controlled area in divided into 9 sub-areas and each sub-area contains a group of LEDs controlled by the same actuators layer cell. The state of each cell is influenced only by the state of the signal intensity of the upper layer cell. The correlation between the upper layer cell subparts and the actuators layer cells is shown in Figure 9.

The state of the actuators cells $A_1 \dots A_9, A_j \in \mathbb{N}_{8bit}$ is computed with the following formula:

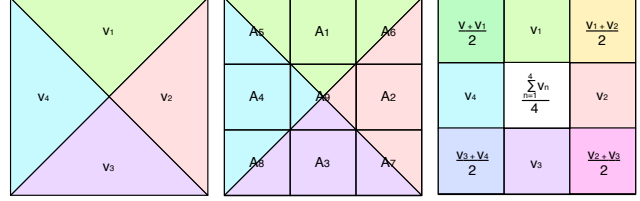


Figure 9. Correlation between the upper layer cell subparts and the actuators layer cells.

$$A_i(t+1) = \begin{cases} v_i(t+1) & 1 \leq i \leq 4 \\ \frac{v_4(t+1) + v_1(t+1)}{2} & i = 5 \\ \frac{v_1(t+1) + v_2(t+1)}{2} & i = 6 \\ \frac{v_2(t+1) + v_3(t+1)}{2} & i = 7 \\ \frac{v_3(t+1) + v_4(t+1)}{2} & i = 8 \\ \frac{1}{4} \sum_{j=1}^4 v_j(t+1) & i = 9 \end{cases}$$

There are different approaches to associate the LEDs to the cells. A first approach consists to directly connect the lights intensity to the signal levels of the correspondent cell. Another approach consists to turn on a number of LEDs proportional to the signal intensity of the controller cell.

5. The Design Environment

The design of human environments (e.g. buildings, stores, squares, roads) is a complex task, composed of several sub-tasks evolving the initial idea into a detailed project, through the production of intermediate and increasingly detailed models.

After the initial phases, in which the designer usually expresses his/her creativity with sketches on the paper or on the computer, a Computer-Aided Design (CAD) software is used to develop the project in details. CAD softwares (e.g. AutoCAD), and also 3D modelling applications (e.g. Autodesk 3DStudio Max, Blender) are used to create the digital models for the projects and to generate photo realistic renderings and animations. For a compact overview of the typical design process see [10].

Together with these software applications supporting designers in the definition of general architectural spaces, other tools supporting designers in very specific tasks can also be adopted: these tools vary from presenting the elaboration of the building shadows, to elaborating their impact

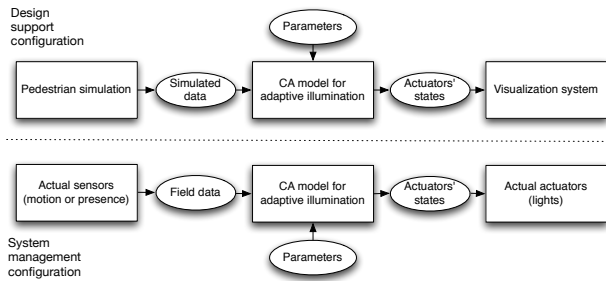


Figure 10. A schema describing the modules of the design support system prototype.

on wind conditions, up to the simulation of vehicles and pedestrian movements in the designed scenario.

The proposed design environment is one of these tools; in particular, it helps the designers in the definition and specification of an adaptive illumination facility through the simulation of its dynamic behaviour. The output of the system is not only a graphical simulation but also a static configuration of the illumination facility (number of lights and their positioning) and an unambiguous specification of their dynamic behaviour (general lights self-organization model plus specific parameters).

The design environment is composed of two main modules, also shown in Figure 10: a simulation environment (that is in turn decomposed into a pedestrian simulation module and an adaptive illumination module) and a visualization facility. In the following paragraphs these modules will be described.

5.1. The Simulation Environment

The simulation environment actually comprises two models, one managing the network of controllers (with sensors and actuators), the other simulating the environment in which the adaptive illumination facility is situated and the pedestrians situated in it. The two simulations are connected: in particular, the state of sensors of is influenced by the state of the environment simulation.

The environment simulator, that is based on the MMass [4], can be used to perform pedestrian simulation. This module actually feeds the self-organization model with simulated field data. The previously described CA model managing the self-organization of the illumination facility will react according to the current occupation of the space in the environment and according to its own parameters.

In this way, the designer can effectively envision the interaction between the people and the specified adaptive environment. The simulation environment allows the designer in configuring the network, defining the type, number, po-

sition of the sensors and actuators, and in specifying the behavior of the controllers, by means of defining the parameters of the CA model.

5.2. The Visualization Facility

The system supports both a 2D and 3D visualization of the simulated environment and the state of the two different enclosed models. The 2D visualization can be interactive, so it is possible to define an action event to be fired on a click (e.g. simulate the perception of a pedestrian when the designer clicks on a cell). This is useful because allows the designer to test the system behavior before specifying in an extensive way a pedestrian simulation scenario.

The 3D visualization is useful to understand the behavior of the system. It is not a photo realistic rendering but a real-time representation of simulated system. During the simulation, the user can navigate the 3D space, changing his/her point of view, for instance, taking the perspective of one of the pedestrians walking in the environment. It is possible to load 3D models both for the space and for pedestrian agents. The 3D visualization is based on the jMonkey engine²; the API of this open source project allows loading several 3D model formats. A screenshot of the visualization system is shown in Figure 11.

6. Future development

The paper introduced an ambient intelligence scenario aimed at improving the everyday experience of pedestrians and people passing through the related environment. A specific scenario related to the definition and development of an adaptive illumination facility was introduced, and a CA-based model supporting the specified behaviour for the illumination facility was defined. A prototype of a system supporting designers in the definition of the relevant parameters for this model and for the overall illumination facility was also introduced.

The renovation project is currently under development on the architectural and engineering side, whereas the CA-based model has shown its adequacy to the problem specification, both in order to provide a formal specification of the behaviour for the system components. The realized prototype explored the possibility of realizing an ad hoc tool that can integrate the traditional CAD systems for supporting designers in simulating and envisioning the dynamic behaviour of complex, self-organizing installations. It has been used to understand the adequacy of the modeling approach in reproducing the desired self-organized adaptive behaviour of the environment to the presence of pedestrians. We are currently improving the prototype, on one hand, to

²<http://www.jmonkeyengine.com/>

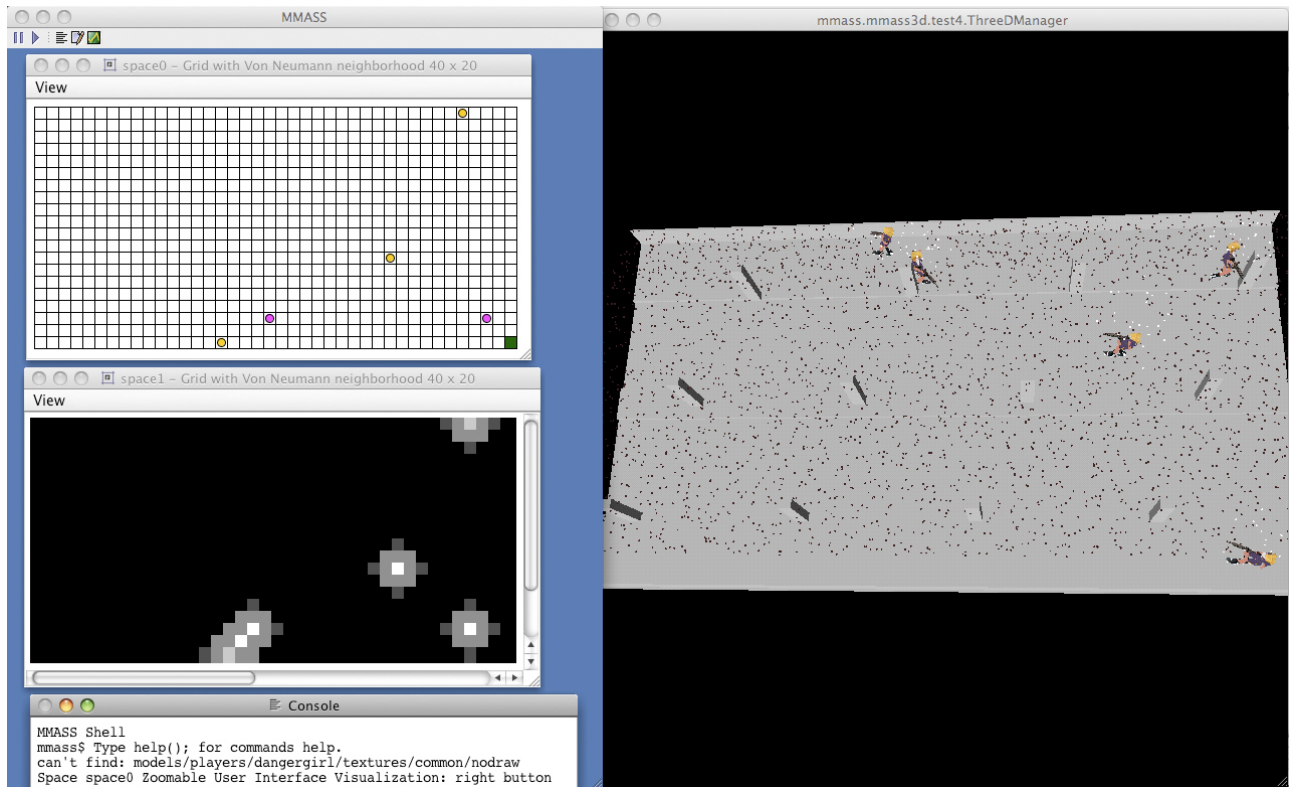


Figure 11. Screenshot of the simulation environment: on the left, the top panel shows the position of pedestrians in the environment, while the bottom one shows the intensity of cells. The right panel shows a 3D visualization of the environment, including columns, lights and pedestrians.

provide a better support for the Indianapolis project and, on the other, to realize a more general framework for supporting designers of dynamic self-organizing environments.

The modeling approach, finally, can also be adopted as a mechanism specifying and simulating the interaction of physically distributed autonomous components, for instance in monitoring and control applications.

Acknowledgements

The work presented in this paper has been partially funded by the University of Milano-Bicocca within the project “Fondo d’Ateneo per la Ricerca - anno 2007”.

References

- [1] *Proceedings of the First International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2007, Boston, MA, USA, July 9-11, 2007*. IEEE Computer Society, 2007.
- [2] H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T. F. K. Jr., R. Nagpal, E. Rauch, G. J. Sussman, and R. Weiss. Amorphous computing. *Commun. ACM*, 43(5):74–82, 2000.
- [3] S. Bandini, G. Erbacci, and G. Mauri. Implementing cellular automata based models on parallel architectures: The capp project. In V. E. Malyshekin, editor, *PaCT*, volume 1662 of *Lecture Notes in Computer Science*, pages 167–179. Springer, 1999.
- [4] S. Bandini, S. Manzoni, and G. Vizzari. Towards a platform for multilayered multi agent situated system based simulations: Focusing on field diffusion. *Applied Artificial Intelligence*, 20(4–5):327–351, 2006.
- [5] S. Bandini and G. Mauri. Multilayered cellular automata. *Theor. Comput. Sci.*, 217(1):99–113, 1999.
- [6] S. Bandini, G. Mauri, G. Pavesi, and C. Simone. Parallel simulation of reaction-diffusion phenomena in percolation processes : A model based on cellular automata. *Future Generation Comp. Syst.*, 17(6):679–688, 2001.
- [7] C. Behring, M. Bracho, M. Castro, and J. A. Moreno. An algorithm for robot path planning with cellular automata. In S. Bandini and T. Worsch, editors, *ACRI*, pages 11–19. Springer, 2000.
- [8] W. Butera. Text display and graphics control on a paintable computer. In *SASO* [1], pages 45–54.
- [9] A. E. S. Filho, E. C. Lupu, N. Dulay, S. L. Keoh, K. P. Twidle, M. Sloman, S. Heeps, S. Strowes, and J. Sventek. To-

- wards supporting interactions between self-managed cells. In *SASO* [1], pages 224–236.
- [10] J. Frazer. Computing without computers. *Architectural Design*, 75(2):34–43, 2005.
- [11] E. Goles and S. Martinez. *Neural and Automata Networks: Dynamical Behavior and Applications*. Kluwer Academic Publishers, 1990. ISBN 0-792-30632-5.
- [12] H. Gutowitz. *Cellular Automata: Theory and Experiment*. MIT Press/Bradford Books, Cambridge Mass., 1991. ISBN 0-262-57086-6.
- [13] W. D. Hillis. *The Connection Machine*. MIT Press, 1985.
- [14] N. R. Jennings. On agent-based software engineering. *Artif. Intell.*, 117(2):277–296, 2000.
- [15] N. Margolus and T. Toffoli. *Cellular Automata Machines. A new environment for modelling*. MIT Press, 1987.
- [16] P. L. Rosin. Training cellular automata for image processing. *IEEE Transactions on Image Processing*, 15(7):2076–2087, 2006.
- [17] N. Shadbolt. Ambient Intelligence. *IEEE Intelligent Systems*, 18(4):2–3, 2003.
- [18] J. von Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, 1966.
- [19] J. R. Weimar. *Simulation with Cellular Automata*. Logos Verlag Berlin, 1997. ISBN 3-89722-026-1.
- [20] F. Zambonelli, M. Mamei, and A. Roli. What can cellular automata tell us about the behavior of large multi-agent systems? In A. F. Garcia, C. J. P. de Lucena, F. Zambonelli, A. Omicini, and J. Castro, editors, *SELMAS*, volume 2603 of *Lecture Notes in Computer Science*, pages 216–231. Springer, 2002.