Fundamenta Informaticae XX (2005) 1–21
IOS Press

Supporting Action–at–a–distance in Situated Cellular Agents

Stefania Bandini
Dipartimento di Informatica, Sistemistica e Comunicazione
Università degli Studi di Milano–Bicocca
via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
bandini@disco.unimib.it

Giancarlo Mauri
Dipartimento di Informatica, Sistemistica e Comunicazione
Università degli Studi di Milano–Bicocca
via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
mauri@disco.unimib.it

Giuseppe Vizzari [C]
Dipartimento di Informatica, Sistemistica e Comunicazione
Università degli Studi di Milano–Bicocca
via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
giuseppe.vizzari@disco.unimib.it

Abstract. The aim of this paper is to describe algorithms and structures required to support action–at–a–distance in Situated Cellular Agents (SCA). This model provides the possibility to define heterogeneous entities placed in regular or irregular spatial infrastructures. Different mechanisms supporting field diffusion within these structures will be described and analyzed, with reference to their performance. A sample application of SCA model (a variation of Conway's Game of Life) will also be illustrated, while a more complex usage of action–at–a–distance for pedestrian simulation will be sketched.

Keywords: Multi agent systems, action–at–a–distance, diffusion

1. Introduction

Action–at–a–distance plays a very important role in the simulation of complex systems, for instance in the area of urban planning (see, e.g., [5, 16]) or in the analysis of crowd's emerging behaviour [14], and it was also exploited in multi–agent approaches to localization problems [4]. Discrete models, based on Cellular Automata (CA [18]), are commonly used in these areas since they are very simple and elegant and are generally well understood by experts in these application domains. Nonetheless the management

[C]Corresponding author

of action–at–a–distance represents a weakness of this model. In fact the interaction between cells is limited by the definition of cell neighborhood, and the influence determined by distant entities can be obtained as a consequence of sequential applications of local transition rules. The main way to obtain this kind of effect instantly is to vary the concept of neighborhood beyond the simple Moore and Von Neumann definitions (see, e.g., [17]). This and other kinds of modifications (for instance with reference to lattice structure and cell state) are often so deep that the resulting models can hardly be considered CAs and in some cases they are just cellular models of the related application domains.

Multi Agent Systems (MAS [8]) is a conceptual and computational approach that provides the implementation of a complex global system's behaviour through local actions and interactions of its composing parts. While this model is generally endowed with complex and rich means of communication between agents, most MAS models lack an explicit representation of the spatial structure of agents' environment. Situated Cellular Agents (SCA [3]) are a particular class of Multilayered Multi Agent Situated Systems (MMASS [1]), a model that has been designed for applications requiring an explicit representation of the spatial structure of the environment [2]. Said structure can be regular or irregular and agents' behaviour is strongly influenced by their position, as it is determined as a consequence of synchronous interaction with other adjacent entities (i.e. *reaction*) or according to the perception of signals asynchronously emitted by at–a–distance agents (i.e. *field diffusion*). Such a remote interaction represents a mean of modelling the concept of locality, while reaction can represent a direct cooperation between neighbours. The main difference between a MMASS agent and a Situated Cellular Agent is the fact that the latter may only diffuse information included in their internal state. Moreover a MMASS can model different and possibly superposed spatial structures, representing layers of physical or conceptual relations, whereas SCA model provides just one spatial layer.

The SCA model provides a dedicated support for action–at–a–distance through field diffusion and the perception–deliberation–action model for agent behaviour. A formal comparison between CA and SCA is out the scope of this paper, nonetheless a sample application has been developed in order to show how the latter could be exploited to model a typical cellular automaton: Conway's Game of Life. The SCA model allows to represent heterogeneous agents, according both to field sensitivity and emission, so different experiments were performed to show how a system based on the SCA model can reproduce the typical behaviour of a CA, and have a qualitative indication of the effects caused by the introduction of elements of heterogeneity in the system. The main aim of this paper is to describe and analyze various mechanisms supporting field diffusion, with reference to computational costs in terms of memory occupation of the required structures, and time complexity of the related algorithms. The following section will define the elements of the SCA model, with an indication of related works both in the MAS and in the CA areas, while Section 3 will describe and analyze different possible mechanisms supporting field diffusion. After that a sample application of SCA (a variation of Conway's Game of Life) will be illustrated, while a more complex usage of action–at–a–distance for pedestrian simulation will be sketched. Conclusions and future developments will end the paper.

2. SCA model description

2.1. The basic structure

A system of Situated Cellular Agents is denoted by:

$$< Space, F, A >$$

where A is a finite set of agents, F is a finite set of fields (that will be described later on, in Subsection 2.3), and *Space* is a single layered environment where agents are situated, act autonomously and interact by means of reaction or through the propagation of fields. More precisely the *Space* consists of a set P of sites arranged in a network (i.e. an undirected graph of sites). We will represent the structure of the space by a neighborhood function, $N : P \longrightarrow 2^P$ so that $N(p) \subseteq P$ is the set of sites adjacent to $p \in P$. Each *site* $p \in P$ can contain at most one agent and a subset $F_p \subseteq F$ is the set of fields that are active in p ($F_p = \emptyset$ when no field is active in p).

2.2. Situated Cellular Agents

With reference to agents, the model can represent heterogeneity in terms of the possibility to define different abilities and perceptive capabilities for various entities placed in the environment. Agents that share these characteristics are said to have the same type. Denoting by T the set of types, it is appropriate to partition the set of agents in disjoint subsets corresponding to different types. The set of agents can thus be defined as

$$A = \bigcup_{\tau \in T} A_\tau$$

where $A_{\tau_i} \cap A_{\tau_j} = \emptyset$ for $i \neq j$. An agent type τ is defined by the triple

$$< \Sigma_\tau, Perception_\tau, Action_\tau >$$

where:

- Σ_τ defines the set of states that agents of type τ can assume;
- $Perception_\tau : \Sigma_\tau \rightarrow [\mathbb{N} \times W_{f_1}] \ldots [\mathbb{N} \times W_{f_{|F|}}]$ is a function associating to each agent state the vector of pairs

$$\left(c_\tau^1(s), t_\tau^1(s)\right), \left(c_\tau^2(s), t_\tau^2(s)\right), \ldots, \left(c_\tau^{|F|}(s), t_\tau^{|F|}(s)\right)$$

  where for each i ($i = 1 \ldots |F|$), $c_\tau^i(s)$ and $t_\tau^i(s)$ express respectively a receptiveness coefficient to be applied to the field value f_i and the agent sensibility threshold to f_i in the given agent state s. The field perception mechanism will be clearer after details of fields will be given, in Subsection 2.3.

- $Actions_\tau$ denotes the set of actions that agents of type τ can perform, and will be described in following subsection.

So, an agent $a \in A$ is defined by $< \tau_a, s_a, p_a >$, where: τ_a is the *agent type*, $s_a \in \Sigma_{\tau_a}$ denotes the *agent state* and can assume one of the values specified by its type; $p_a \in P$ is the site of the *Space* where the agent is situated. Note that, since a site $p \in P$ can contain no more than one agent, for $a, b \in A, a \neq b$ it must be that $p_a \neq p_b$; equivalently the function $f : A \rightarrow P$ associating an agent with its site must be injective.

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

$action : reaction(s, a_{p_1}, a_{p_2}, \ldots, a_{p_n}, s')$
$condit : state(a) = s, p_a = p, agreed(a_{p_1}, a_{p_2}, \ldots, a_{p_n})$
$\mathit{effect} : state(s')$

where $state(a) = s$, $p_a = p$ and $agreed(a_{p_1}, a_{p_2}, \ldots, a_{p_n})$ are verified when the state of agent a is s, its position is p and agents situated in sites $\{p_1, p_2, \ldots, p_n\} \subseteq P_p$ have previously agreed to undertake a synchronous reaction. This process of agreement implies that the possibility of an agent to perform a reaction depends on the state of other involved entities. The effect of a reaction is the synchronous change in state of the involved agents; in particular, agent a changes its state into s'.

Other possible actions are related to the asynchronous interaction model, related to field emission and to the perception–deliberation–action mechanism. Agent emission can be defined as follows:

$action : emit(s, f, p)$
$condit : state(a) = s, p_a = p$
$\mathit{effect} : added(f, p)$

where $state(a) = s$ and $p_a = p$ are verified when the agent state is s and its position is p. The effect of the emit action is a change in the active fields related to sites involved in the diffusion, according to $\mathit{Diffusion}_f$, the diffusion function that will be defined in Section 2.3.

The effect of an agent perception of a certain field f_i is defined through a $trigger$ action, that is specified as follows:

$action : trigger(s, f_i, s')$
$condit : state(a) = s, p_a = p, perceive(f_i)$
$\mathit{effect} : state(s')$

where $perceive(f_i)$ is verified when the agent is able to perceive the field f_i (the mechanism of field perception will be more thoroughly described in the following subsection). The effect of the trigger action is a change in agent's state according to the third parameter. The last possible action for an agent causes a change in its position and can be specified as follows:

$action : transport(p, f_i, q)$
$condit : p_a = p, empty(q), near(p, q), perceive(f_i)$
$\mathit{effect} : p_a = q, empty(p)$

where $empty(q)$ and $near(p, q)$ are verified when $q \in P_p$ and $q =< \perp, F_q, P_q >$ (q is adjacent to p and it does not contain agents). The effect of a transport action is thus to change the position of the related agent. This aspect clearly requires to consider the case of concurrent attempts to move to the same site. Currently this conflict is handled on a first–in–first–served basis, considering the site as a shared resource whose access is mutually exclusive, with a non–deterministic choice in case of contemporaneous requests (i.e. in current implementations the $transport$ operation may fail). A more through analysis of the issues related to conflicts deriving from agents behaviour, including movement, and various strategies devoted to their management is currently being performed and will be described in a future work.

2.3. Fields

Situated Cellular Agents can emit fields that will spread across their environment, according to propagation functions defined in the field specifications. In other words, following the physical metaphor, fields might weaken from the starting intensity during the propagation from the source to other sites. Other agents may perceive these fields, if the actual value in their site (i.e. the starting one modified according to the propagation function) exceeds their sensitivity threshold for that field type. Agent choose their next action according to a perception–deliberation–action mechanism and thus fields represent a mean of remote (i.e. among non–adjacent agents) and asynchronous interaction between agents.

A field $f_\tau \in F$ that can be emitted by agents of type τ is denoted by

$$< W_\tau, Diffusion_\tau, Compare_\tau, Compose_\tau >$$

where:

- $W_\tau = S \times \mathbb{N}$, with $S \subseteq \Sigma_\tau$; given $w_\tau \in W_\tau$, $w_\tau =< s_\tau, i_\tau >$, where $s \in S$ represents information brought by the field, which is a subset of the related agent state, and $i_\tau \in \mathbb{N}$ represents its intensity.
- $Diffusion_\tau : P \times W_\tau \times P \rightarrow (W_\tau)^+$ is the diffusion function of the field computing the value of a field on a given site taking into account in which site and with which value it has been emitted. Since the structure of a $Space$ is generally not regular and paths of different length can connect each pair of sites, $Diffusion_\tau$ returns a number of values depending on the number of paths connecting the source site with each other site. Hence, each site can receive different values of the same field along different paths. For most applications of the model, only the shortest one (according to some concept and metric for distance among sites) will be considered.
- $Compare_\tau : W_\tau \times W_\tau \rightarrow \{True, False\}$ is the function that compares field values, and it is used to verify whether an agent can perceive a certain field or it is not sensitive enough. The previously introduced $perceive$ function is $true$ when $f_i \in F_p$ (i.e. the field is active in the site where the possibly perceiving agent is placed) and $Compare_\tau(< s_\tau, c^i_\tau \cdot i_{f_i} >, t^i_\tau) = true$ (in other words, field intensity modulated by an receptiveness coefficient exceeds the sensitivity threshold of the possibly perceiving agent for that field). The $Compare$ may rely solely on field intensity, but could even use the other part of field information in order to build some additional filter. For instance, in order to model the fact that an agent is only interested in a certain kind of signal, this function could be defined as $False$ for every signal $w_\tau =< s_\tau, i_\tau >$ where $s_\tau \neq s_k$, where s_k is the kind of signal the agent is interested in.

- $Compose_\tau : (W_\tau)^+ \to W_\tau$ expresses how field values have to be combined in order to obtain the unique value of a certain field type at a site.

Fields may have an *instantaneous* effect on the involved agents and disappear after the diffusion operation has been performed. On the other hand they might *persist* in the environment until another field cancels them through a suitable composition. There is also the possibility to define a sort of *evaporation function* in order to obtain a gradual decrease of field intensity. The definition of these aspects is strictly related to the application domain: for instance an evaporation function for synchronous systems will be discrete while asynchronous systems will require a continuous function. A through analysis of these issues is out of the scope of this paper and will be performed in future works.

2.4. Related works

Despite the number of problems that require an explicit representation of spatial relationships, most MAS models disregard these aspects, providing a direct interaction model for agent communication. Agents are aware of each other, or they can obtain acquaintance information from a well–known facilitator (an agent that plays the role of *white–pages*), and can directly exchange messages, according to specific rules defined by an *Agent Communication Language* [11]. In this models, concepts related to the spatial context in which interaction takes place must be introduced in interacting entities (that should regulate their own interaction) or in third–part modules (e.g. agents responsible for the management of interactions). In this way conceptual elements (i.e. interaction and spatial context management) are mixed–up with other aspects related to domain specific issues (e.g. agents behaviour) and often delegated to ad–hoc implementations. The interaction model described in this paper differs from the common point–to–point message–passing communication model adopted by most multi–agent models, and offers the possibility to have a multicast communication that is strongly dependant on the spatial context of the involved entities. A similar approach [12], providing a physically grounded model for agent interaction, was applied to urban traffic management. The concept of Co-Field (Computational Field) allows to model the environment through these signals that are spread by agents and by an infrastructure that supports the diffusion of those signals and thus agent interaction. In this model, agents are constantly guided by fields, while in SCAs every perception of a field triggers a single action.

Another MAS approach that provides abstractions and concepts for environment representation and space–dependant form of communication comparable to action–at–a–distance is Swarm [13]; other projects are based on it and propose the same kind of interaction model (e.g. Ascape [1] and Repast [2]). Swarm is a multi–agent software platform that provides explicit representation of the environment in which agents are placed, and even a mechanism for the diffusion of signals in particular versions of these structures. This diffusion mechanism is not well documented and, even though allows a certain degree of configurability (e.g. through the definition of constants regulating signals diffusion and evaporation), it does not allow the definition of specific diffusion functions. Moreover the possibility to distribute signals over the spatial structure and the mechanism regulating this process are considered particular spatial features. Therefore diffusion is strictly related to the definition and implementation of the spatial structure and, as a consequence, is only available for particular regular lattices (square and hexagonal grids).

[1] http://www.brook.edu/dybdocroot/es/ dynamics/models/ascape/README.html.
[2] http://repast.sourceforge.net.

In the CA area there are many different cases of systems adopting the strategy of neighborhood extension in order to obtain action–at–a–distance, but even languages and platforms considered this possibility. Both Cellang [7] and CDL++ [9] allow the definition of an extended neighborhoods (the former providing an extensive definition of the neighbor cells and the latter through the definition of a *distance*). Moreover they provide abstractions for the definition of objects and their motion over the cellular space. CDL++ allows the definition of Moving Objects (MOBs) (considering and managing conflicts arising from the possibility of different MOBs to move towards the same cell), while Cellang uses the term agent to denote the same concept. These abstractions can be useful for the definition of very simple entities wandering over a cellular space according to precise, homogeneous and uniform rules, but do not allow the representation of complex situations in which the non–uniform behaviour of single, possibly heterogenous individuals have a role in the dynamics of the whole system. In this framework, there are even other approaches (see, e.g., [6]) that provide agents characterized by roles, goals and beliefs that are placed over a cellular space, that can be perceived by agents and influences their behaviour.

3. Diffusion mechanism

Section 2.3 formally introduced the concept of field and diffusion as a suitable strategy for broadcast and multicast communication, as fields can propagate throughout the space reaching several (possibly all) agents. This interaction form, combined with the reaction mechanism, which allows synchronous interaction among adjacent agents, can effectively represent a wide range of concrete and conceptual domains. Both of those interaction means are space mediated, so the price of this expressiveness is the development of suitable infrastructures that, more than just reflecting a spatial representation, have to support agent interactions. In the following, issues and proposed solutions related to the design of mechanisms supporting diffusion will be discussed.

3.1. Centralized approach

A centralized approach to the field diffusion management provides a unique module whose task is to receive information related to emitted fields from various agents and suitably update global data structures related to the spatial representation. Such a monolithic implementation of these mechanisms clearly presents a bottleneck in this module, as it should handle all remote interactions between agents. In particular, a single global data structure may represent an issue, as write accesses (i.e. field emission) must be mutually exclusive. The granularity of this structure must thus reach a deeper level of detail, providing the representation of the single site as an entity of its own, to allow a reasonable chance of concurrent handling of different fields diffusion. Even doing so, splitting the spatial structure into smaller parts corresponding to sites, a centralized approach providing a single entity to manage the diffusion of all fields emitted in the space might represent an issue. As many agents may emit fields concurrently, a suitable queuing mechanism should be designed and implemented, and a sort of synchronicity should be introduced.

These considerations are clearly significant for asynchronous systems, as in a synchronicity assumption the field diffusion can be performed just after the emission, before any other agent action can be performed. The centralized approach in general seems not appropriate for the implementation of sys-

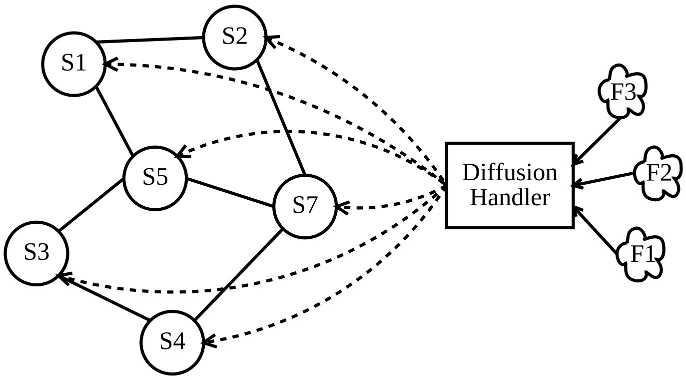


Figure 1. A centralized diffusion handler must provide a mechanism for concurrent diffusion management.

tems based on the SCA model, and should only be considered for the implementation of synchronous SCA-based systems.

3.2. Distributed approach

A suitable approach to field diffusion management should thus provide asynchronous and possibly distributed mechanisms. These considerations lead to an analysis of how existing algorithms, designed in the area of distributed systems and networking [15] (more precisely routing protocols), could be used or adapted to this situation.

The first and simplest algorithm we considered was flooding: to emit a field an agent must interact with the underlying site, which sends it to the adjacent ones that will in turn do the same (avoiding to send back the field to the source site). Field intensity must be modified according to the diffusion function: this operation can be performed by the source or destination site, but the former choice avoids unnecessary field transmissions (i.e. totally decayed fields). No additional structure is required and this algorithm is very robust (that is not crucial in this situation), but this feature is obtained through a massive replication of sent messages. This is more than a performance issue, as a specific field could reach a site through different paths (for instance, see Figure 2). In several situations fields should reach destination sites only once, generally through the shortest path, and thus flooding would not represent a good choice. In fact a field could reach a site through a path different from the shortest one, and its intensity would thus be inconsistent. This problem can be solved by associating a unique identifier to every emission, in order to allow the recognition of previously diffused fields. In this way, sites can avoid diffusing fields that have already passed through them, unless their intensity is higher (or lower, if the diffusion function grows with the distance) than the current one. This modification to the basic flooding algorithm assures that inconsistent situations are temporary and will be corrected, and also limits the number of field transmissions between sites. However the duration and cost of a diffusion operation cannot be determined *a priori* as it only relies on a possibly irregular spatial structure.

While flooding does not require or exploit additional structures to perform diffusion, other approaches provide the creation of secondary structures to guide diffusion, for instance in order to avoid field replication. Said structures could be computed once for every site (when the application is started) and then stored, if the number of agents–sites ratio is high and the emission of fields is frequent. Other-

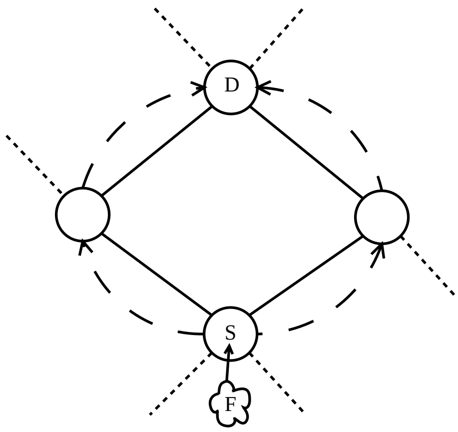


Figure 2. A flooding approach to diffusion determines field replication over the spatial structure. The destination site receives two copies of field F, possibly with different intensity values.

wise a more economic approach should provide the generation of those structures by request and a partial storage of them in a sort of cache, to keep limited memory occupation. The following Section will focus on different solutions adopting a distributed approach exploiting additional structures guiding diffusion.

3.3. Proposed algorithm and structures

The algorithm described in this section provides the generation of infrastructures to guide field diffusion and a specification of how sites should perform it, according to the diffusion function related to the specific field type. It was designed under the assumption of an irregular space (i.e. a non–directed, non–weighted graph), with a high agents–sites ratio and very frequent field emissions. Fields propagate instantly throughout the space, according to the modulation specified by the field type diffusion function; in general fields could diffuse throughout all sites in the space. Under these assumptions we considered the possibility of storing a spatial structure representation for each site, and namely a minimum spanning tree (MST) connecting it to all other sites, as the use of said structures is frequent and the overhead for their construction for every diffusion operation would be relevant.

There are several algorithms for MST building, but previously explained design choices led to approaches that could be easily adapted to work in a distributed and concurrent environment. The breadth first search (BSF) algorithm starts exploring the graph from a node that will be the root of the MST, and incrementally expands knowledge on the structure by visiting at phase k nodes distant k hops from the root (for instance, see Figure 3). This process can be performed by nodes themselves (sites, in this case), that could offer a basic service of local graph inspection that could even be useful in case of dynamism in its structure. The root site could inspect its neighborhood and require adjacent sites to do the same, iterating this process with newly known sites until there is no more addition to the visited graph. An important side effect of this approach is that this MST preserves the distance between sites and the root: in other words the path from a site to the root has a number of hops equal to its distance from the root. Fields propagate through arcs of the MST and thus the computation of the diffusion function is facilitated. The complexity of the MST construction using this approach is the order of $O(n + e)$ where n is the number of sites and e is the number of edges in the graph. Such an operation should be performed

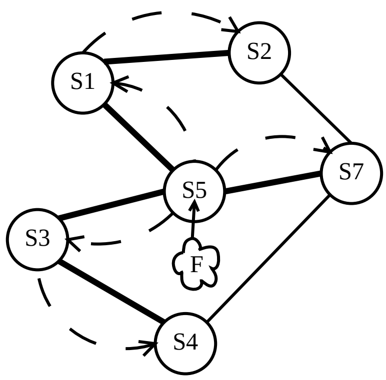
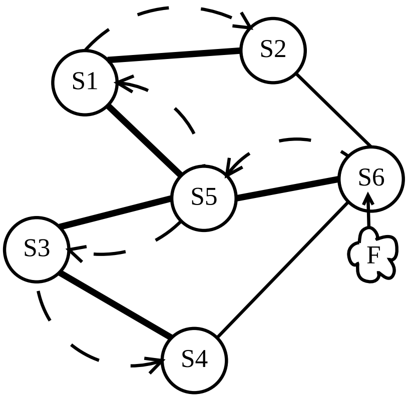


Figure 3. The MST centered in the source site (S5) guides the diffusion of field F. At the k-th transmission, the diffused field reaches sites distant k hops from the emission site.

by every site, but with a suitable design of the underlying protocol they could proceed in parallel. Field diffusion requires at most $O(\log_b n)$, where b is the branching factor of the MST centered in the source site and the field propagation between adjacent sites is performed in constant time. The issue with this approach is the memory occupation of all those structures, that is $O(n^2)$ (in fact it is made up of n MSTs, each of those provides $n - 1$ arcs); moreover if the agents–sites ratio is not high or field emission is not very frequent to keep stored the MST for every site could be pointless, as many of those structures could remain unused.

There are different correctives to this approach, and two possibilities were considered: the first provides the storage of just one MST, requiring a central data structure with the distances between different sites, the other provides the construction of MSTs by request and the caching of this structure in a limited buffer. The first approach uses a single MST whose edges are used to propagate fields that will thus reach every site just once (see Figure 4). For the diffusion of a field, the source site becomes the root of the tree and exploits it to send the related information to other sites. However this structure doesnt preserve the actual distance between nodes so, to effectively compute the actual value of a field in the destination site according to the diffusion function, another structure is needed to store distances among nodes. This structures size is $O(n^2)$, so according to the asymptotic memory occupation this approach is comparable to the previous one, and the same holds for the time complexity related to the building of these structures. To perform a diffusion operation, instead, a field must visit n-1 sites in the worst case. This approach seems thus comparable to the previous one according to the space occupation, and even worse according to the complexity of the diffusion operation, but this comparison is related to asymptotical costs and related constants could be relevant (at least for a certain values of n) so these results should be supported (or possibly confuted) by tests and benchmarks.

The second modification does not bring substantial changes in the infrastructures and algorithms, but limits the memory occupation by defining a MST cache facility that will allow the storage of a constant number of these structures. By doing so the space complexity is reduced to $O(n)$, but the diffusion operation is more complex, because it depends on the cache hit–rate. If the MST centered on the source site has already been built and is still in the cache the cost of diffusion is at most $O(\log_b n)$, otherwise the MST must be constructed and the cost grows to $O(n + e)$. This approach seems thus the most effective, especially in a scenario where the agents–sites ratio is not high and agents do not have a high mobility.



	S1	S2	S3	S4	S5	S6
S1		1	2	3	1	2
S2	1		3	2	2	1
S3	2	3		1	1	2
S4	3	2	1		2	1
S5	1	2	1	2		1
S6	2	1	2	1	1	

Figure 4. A single MST allows the diffusion of a field over the space without replications, but a separate structure is needed to compute field intensity in destination sites. Gray squares are unnecessary, but even not considering them the structure size is $O(n^2)$.

A suitable strategy for cache management (possibly derived by virtual memory handling policies), and especially for MST replacement in the related buffer, should be defined and implemented.

3.4. Discussion

To design an algorithm for field diffusion over a generally irregular spatial structure, in the context of the SCA model, several aspects must be taken into account. The model is meant to be general and thus makes no assumption on the synchronicity of the system, and on the form and nature of the diffusion function, directing the field propagation in the environment. Results of our analysis are summarized in Table 1. We claim that there is not a generally optimal algorithm, but every situation presents specific features that must be taken into account in the choice (or design) of a strategy for field diffusion. For instance, the flooding approach seems generally not convincing, especially given the massive field replication that must be controlled, but if the situation provides fields able to reach sites through all the possible paths, it could represent a good choice. Building MSTs covering all sites in the space could be an unnecessary burden if the area of propagation of a field is a very small portion of the whole environment. These are just two examples of how the specific the situation and reality to be modelled and its requirements have a deep impact on the choice related to the strategies to handle field diffusion, and thus on structures and algorithms. Moreover this analysis was done on a non–directed, non–weighted graph. The work that was started with this analysis aims to the design of a general framework, including computational counterparts of abstract concepts defined in the SCA model, with some basic implementations of strategies derived by a general theoretical analysis and direct experiences on specific applications of the model.

4. Sample application

Conway's Game of Life is a typical example of a cellular automaton, but it is very interesting as it represents one of the simplest examples of emergent behaviour. Elaborate patterns and behaviours emerge from its simple rules, based on the concept of neighborhood in a regular grid. Cells can be either *dead*

	Structures building complexity	**Diffusion complexity**	**Memory occupation**	**Unique path for diffusion**	**Distance preservation**	**Comments**
Flooding (corrected)	0	generally not known	0	no, correctives needed (i.e. emission Id)	Problematic (transient inconsistency)	Unknown diffusion complexity can be a serious issue
MST for every site	$O(n+e)$	$O(\log_b n)$	$O(n^2)$	yes	yes (built via BSF)	
MST + distance table	$O(n+e)$	$O(n)$ (worst case)	$O(n^2)$	yes	yes	The distance table could represent a bottleneck in a distributed environment
MST cache	0 (dynamically built)	$O(\log_b n) + P(\bar{c}) \times O(n+e)$ ($\bar{c}$ means cache failure)	$O(n)$	yes	yes	Requires a suitable cache management strategy

Table 1. Presented approaches and features are shown in this table.

or *alive*, and transition rules are essentially based on two concepts:

- *loneliness*: a cell that has not enough live neighbors will die or remain dead;
- *overcrowding*: a cell that has too many live neighbors will die or remain dead.

Balance between loneliness and overcrowding determines the conditions that allow a cell to remain alive or switch its state from dead to alive. Modeling this situation with a SCA requires the definition of a mechanism to represent transition rules in terms of reaction between agents or through field based interaction. The concept of crowding modelled in Life is strictly bound to neighborhood, while a more general idea of locality could be modelled through the use of a *vitality* field. Every live agent is a source of this kind of field that is diffused in the space, degrading during its propagation in the space from the source site. Every agent is characterized by its state (dead or alive), by an indication of its *strength* (i.e. the intensity of the vitality field it emits), and by two values representing respectively the loneliness and overcrowding thresholds. Agent behaviour could be specified as follows:

- if it is alive it emits a vitality field whose starting intensity is equal to its strength;
- if the intensity of the vitality field in its place is below the loneliness or above the overcrowding thresholds its next state will be dead;
- otherwise (i.e. local vitality intensity is between the thresholds) its next state will be alive.

In the following subsection this situation will be formally described in terms of the SCA model, and a brief description of performed experiments will follow.

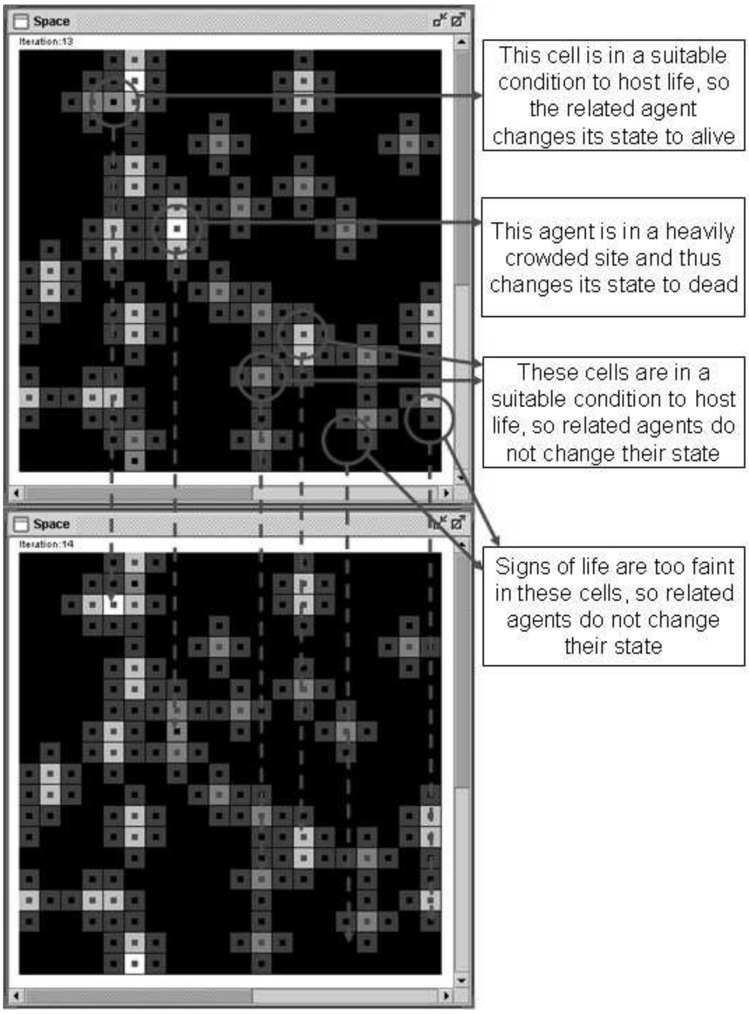


Figure 5. Some screenshots taken from the evolution of an homogeneous SCA system.

4.1. Modelling

The spatial structure of Life is a regular grid of uniform cells. This means that suitably defined homogeneous agents must be placed in every cell of the environment, that will be a regular grid of sites. Agent state, $\Sigma_{LifeSCA}$ is defined by

$$< A, S, L, C >$$

where $A = \{dead, alive\}$ indicates if the agent is dead or alive, $S = \mathbb{R}^+$ represents agent strength, $L = \mathbb{R}^+$ is the loneliness threshold, $C = \mathbb{R}^+$ is the crowding threshold; $c > l$ is a non–triviality condition expressed to avoid inconsistent situations (an agent cannot feel alone and pressed from overcrowding at the same time).

The vitality field F_l assumes values in $\mathbb{R}^+$, its *Compose* and *Compare* functions are respectively the sum and the less than relation between field intensity, and $\forall p \in P$

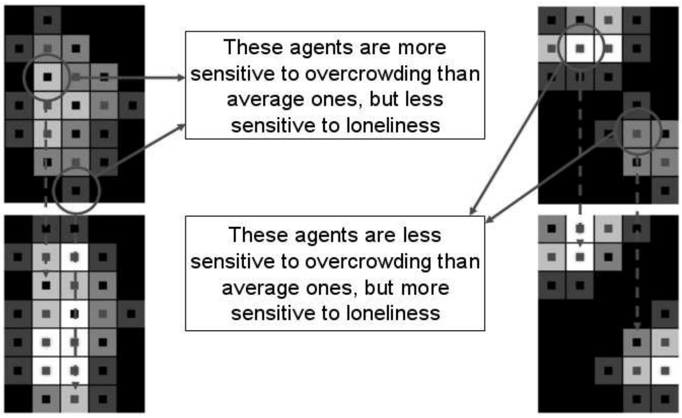


Figure 6. A fragment from a simulation showing the effects of heterogeneity in perceptive capabilities.

$$Diffusion_{F_l}(p_0, f, p) = \begin{cases} f - dist(p_0, p) & f > dist(p_0, p) \\ 0 & otherwise \end{cases}$$

Perception function for agents of this type maps any state s into $\langle 1, < strength, 0 > \rangle$, in other words agents do not alter field intensity, as their receptiveness coefficient is 1, and perceive any not null vitality field. Agent behaviour can be specified as follows:

- $action : emit\Big(< a, s, l, c >, \langle s, Diffusion_{F_l}, <, + \rangle, p\Big)$
  $condit : (a = alive), position(p)$
  $effect : added\Big(\langle s, Diffusion_{F_l}, <, + \rangle, p\Big)$

- $action : trigger\Big(< a, s, l, c >, \langle f, Diffusion_{F_l}, <, + \rangle, < d, s, l, c > \Big)$
  $condit : (a = alive), position(p), (f < l), (d = dead)$
  $effect : state\big(< d, s, l, c > \big)$

- $action : trigger\Big(< a, s, l, c >, \langle f, Diffusion_{F_l}, <, + \rangle, < d, s, l, c > \Big)$
  $condit : (a = alive), position(p), (f > c), (d = dead)$
  $effect : state\big(< d, s, l, c > \big)$

- $action : trigger\Big(< d, s, l, c >, \langle f, Diffusion_{F_l}, <, + \rangle, < a, s, l, c > \Big)$
  $condit : (a = alive), position(p), (f > l), (f < c), (d = dead)$
  $effect : state\big(< a, s, l, c > \big)$

The first element specifies that alive agents must emit a vitality field whose intensity is equal to agent's strength; the second and third conditions represent conditions bringing to agent death, respec-

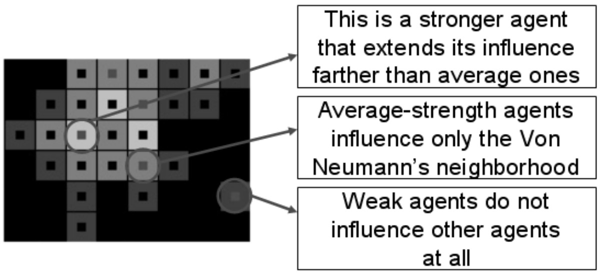


Figure 7. A fragment from a simulation showing action–at–a–distance as effect of a high strength agent.

tively from loneliness and over overcrowding; the last point specifies when an agent switches from dead to alive state.

Having included agent strength, loneliness and overcrowding thresholds in its state it is possible to refer to them in conditions and actions, but even to modify them, obtaining thus more complex, dynamic and possibly heterogeneous situations.

4.2. Experiments description

Basing on the previously defined model various experiments were performed, starting from a simple homogeneous scenario, then introducing heterogeneity with reference to agents' sensitiveness to loneliness and overcrowding, and with reference to their strength. The goal of this activity was to show how a system based on the SCA model can reproduce the typical behaviour of a CA, and have a qualitative indication of the effects caused by the introduction of elements of heterogeneity in the system.

All the experiments were based on a simple environment, a regular grid 20×20, with a SCA agent placed on every cell; about 10% of the agents are alive in the first iteration. The first experiment provides homogeneous agents with a strength $s = 2$, a loneliness threshold $l = 2$ and an overcrowding threshold $c = 3$. This example is similar to Conway's Life, using diffusion to implement the equivalent of a Von Neumann neighborhood. Some screenshots of one of these experiments are shown in Figure 5. The cell color represents the value of the vitality field in the related site; brighter color corresponds to a higher value for this field. Inside every cell there is also a smaller square representing the state of the related SCA agent: red means that the agent is alive, while it will be black if the state is dead. The Figure shows that some patterns and oscillators emerge quite soon.

The same starting situation evolves in a very different way introducing heterogeneity in agent sensitivity. In a second kind of experiment 5% of the agents are more sensitive to loneliness and more resistant to overcrowding (loneliness threshold $l = 3$ and an overcrowding threshold $c = 4$), and another 5% of the agents are instead less sensitive to loneliness and more susceptible to overcrowding (loneliness threshold $l = 1$ and an overcrowding threshold $c = 2$). The effect of this modification is that there are particular positions that can withstand harder conditions, in terms of either a long or short distance from lively areas, preserving an alive state or allowing the transition from the dead state. This means that some areas cannot be densely populated, while other are more sensitive to even faint signs of life. Figure 6 is a graphical representation of the effects related related to different sensitivity thresholds.

Another element of heterogeneity is represented by differences in agents' strength. Following the previous schema, in a third kind of experiment 5% of the agents are weaker (strength $s = 1$), and 5%

are stronger (strength $s = 3$) than the average agent. The latter are thus essentially able to implement a neighborhood of radius $r = 2$. This means that some cells can transfer vitality fields at a greater distance, but can even cause overcrowding in adjacent cells (and possibly even in their own one), while other ones are nearly undetectable by close agents because of the weakness of the emitted vitality field. Figure 7 shows the influence of agents with different strength values, and thus the possibility of action–at–a–distance.

Some experiments seemed to point out that the former play an important role in the first iterations, spreading life signals across the board, but generally do not survive as their own emission kills themselves. The latter can instead survive without causing troubles to adjacent agents, as their emission is really faint, but as a consequence are not able to generate life.

A fourth kind of experiment provided the combination of the previously highlighted modifications, obtaining thus an heterogeneous system both in terms of agents sensitivity and strength. Some screenshots of this kind of experiment are shown in Figure 8.

A quantitative analysis of the effects of this heterogeneity is out of the scope of this paper, but starting from the same distribution of living agents, the average number of alive agents per turn was greater in the heterogeneous system. This variable seemed less stable than in a complete homogeneity situation, but generally the system entered a cyclic oscillation. Figure 8 also shows that a number of "weak" agents was able to survive after more than two hundred iterations, while no "strong" one is still in an alive state. A few isolated alive agents point out that even agents that are less sensitive to loneliness are more likely to survive, given the starting conditions. Specific patterns and oscillators appear even in this situation: a complex oscillator of period 4 is located in the lower–right quadrant of the grid.

4.3. Action–at–a–distance in the crowding context

While the Life example has shown how concepts and elements defined in the SCA model can be exploited in a typical CA context, in the following we are going to give some indications and examples of how they could be used for the representation of pedestrians and related behaviours. Pedestrians and individuals can be modelled as agents, and the environment where they are placed can be suitably represented as a graph of sites, that is a discrete abstraction of a physical environment. The latter is divided into sites occupying approximately $40 \times 40\ cm^2$ (i.e. the typical space occupied by a pedestrian in a dense crowd [14]). This choice has an impact on the realism of the simulation, as it determines the maximum pedestrian density, since one site may host at most one agent.

One of the main elements that characterizes the SCA model, especially compared to CA, is the possibility to define goals and behaviours of specific entities. A possible way to specify the tendency of an agent to move towards specific areas or influence each other is the definition of attractive fields and specific actions to define a suitable reaction to their perception. A simple attractive field F_a assumes values in $\mathbb{R}^+$, its *Compose* and *Compare* functions are respectively the sum and the less than relation between field intensity, and $\forall p \in P$

$$\mathit{Diffusion}_{F_l}(p_0, f, p) = \begin{cases} f - dist(p_0, p) & f > dist(p_0, p) \\ 0 & otherwise \end{cases}$$

A simple example of perception function for agents receptive to this kind of field maps any state s into $\langle c, f_t \rangle$,where c may be part of agent's state (and thus a possibly dynamic value) indicating its

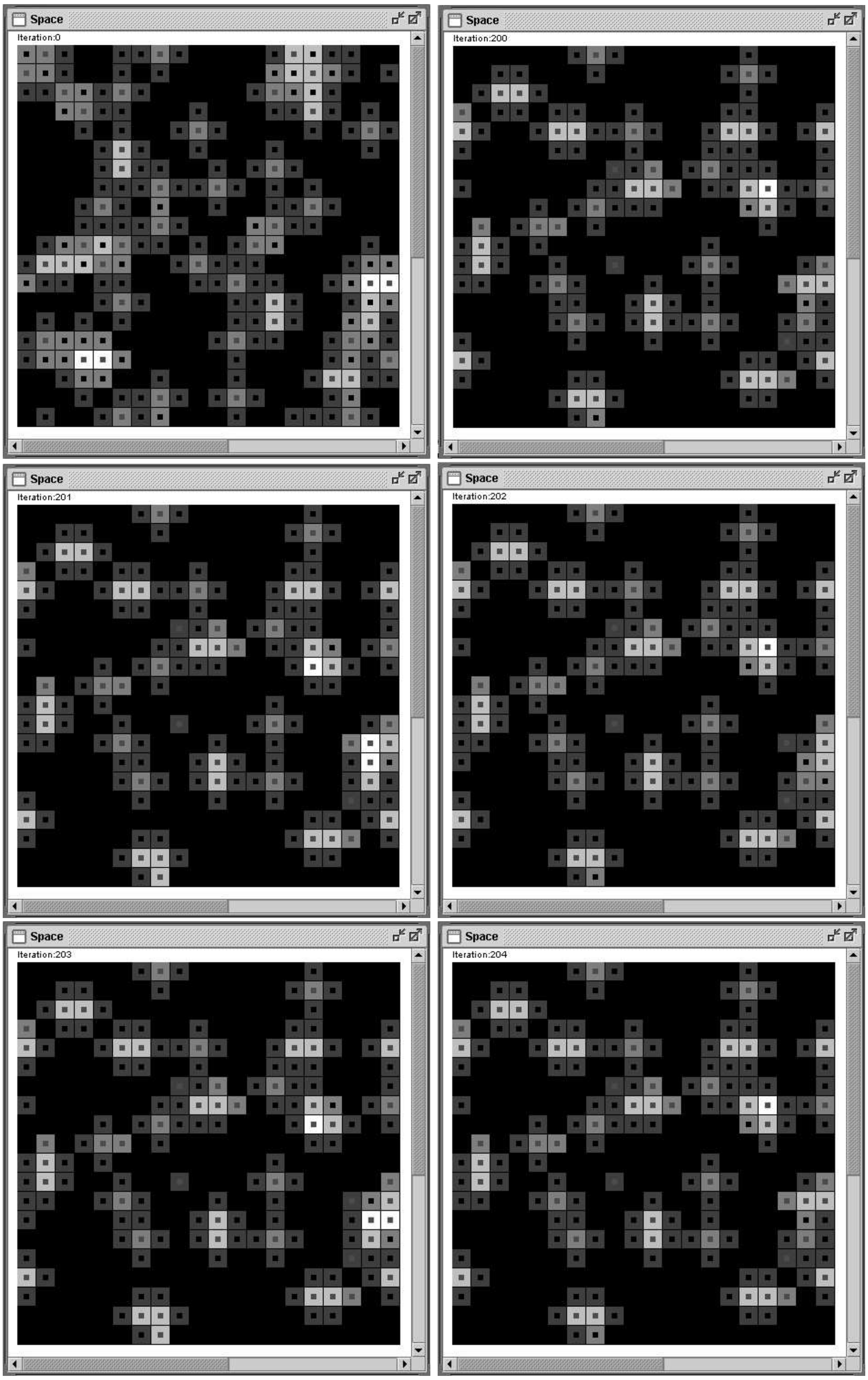


Figure 8. Some screenshots taken from the evolution of a heterogeneous SCA system (according to both sensitiveness and emission).

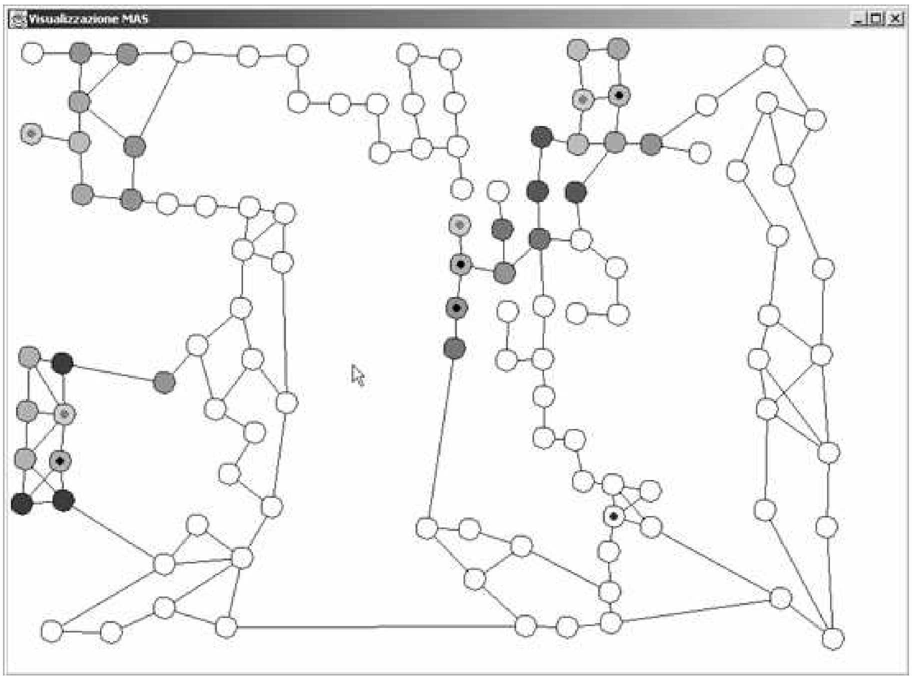


Figure 9. A sample implementation of a an attractors–followers system. Dark agents are attracted by light ones, that spread a signal of their presence in an irregular space.

receptiveness to this field type, while f_t is a field whose intensity represents a threshold for filtering low intensity fields. Given an agent situated in the site p, the set of adjacent sites where the attractive field is active and no other agent is placed is defined by $P_p^f = \{p' \in P_p | a_{p'} = \perp \wedge f \in F_{p'}\}$. In this framework, the behaviour specification for agents that should seek the site where this field's intensity is (locally) maximum should include at least a transport action very similar to the following one:

$action : transport(p, f_p, q)$
$condit : position(p), q \in P_p^{f_p}, f_q > f_r \; \forall r \in P_p^{f_p}$
$effect : position(q), empty(p)$

This kind of mechanism is not suited to model situations in which an agent should try to keep at a certain distance from the source of certain field, for instance to generate a flow of pedestrians that follow each other without getting too close to each others. They could be defined as sources of specific attractive fields, but only sensitive to the signal emitted by the preceding one. In this case the transport action could be specified as follows:

$action : transport(p, f_p, q)$
$condit : position(p), q \in P_p^{f_p}, f_q > f_l, f_h > f_q$
$effect : position(q), empty(p)$

where f_l and f_h are thresholds indicating the minimum and maximum field desirable field intensity for that kind of agent. The values related to these thresholds must be carefully determined, considering that a low f_h could prevent movement in crowded environments. Moreover this approach requires the definition of many different field types, related to the various entities present in the system. In order to tackle this second issue, the value related to an attractive field could be extended in order to include an identifier related to the emitting entity. More precisely a more flexible attractive field could be defined as follows:

$$F_h = < W_h, \mathit{Diffusion}_h, \mathit{Compare}_h, \mathit{Compose}_h >$$

with $W_h = \mathbb{N} \times \mathbb{R}$. For every $w_h \in W_h$, $w_h =< f_{Id}, f_{In} >$ where the first component indicates the identifier of the field source, while the latter represents its intensity. The agent state should be expanded to include the identifier of the preceding entity, and could thus be defined as a two–tuple $< \Sigma, Sens >$, where the latter indicates the identifier of the entities whose fields will be considered, while the former contains other agent internal information, that is not relevant to this subject. In this framework the transport action could be modified as follows:

$action : transport(p, < f_{Id}, f_{In} >, q)$
$condit : position(p), state(< \Sigma_p, f_{Id} >), q \in P_p^{f_p}, f_q > f_l, f_h > f_q$
$\mathit{effect} : position(q), empty(p)$

The possibility to alter the state of an agent can offer the chance to suitably combine the search for local maximum values for a certain fields (i.e. field sources) in order to force agents to follow a specific path. Static field sources can be placed as way–points, and agents should be provided with two kinds of actions and precisely one to follow the field gradient and another one to switch to the next way–point. In this way agent movement is less influenced by crowding, as long as there are vacant sites to move to which are more attractive, having a higher value of the attractive field. For instance these actions could be defined as follows:

- $action : transport(p, < f_{Id}, f_{In} >, q)$
  $condit : position(p), state(< \Sigma_p, f_{Id} >), q \in P_p^{f_p}, f_q > f_r \; \forall r \in P_p^{f_p}$
  $\mathit{effect} : position(q), empty(p)$

- $action : trigger(< \Sigma_p, f_{Id} >, < f_{Id}, f_{In} >, < \Sigma_p, f_{Id'} >)$
  $condit : position(p), f_{In} > f_{next}$
  $\mathit{effect} : f_{Id'} = f_{Id} + 1$

The same software framework developed and used for the Life application was also adopted to implement simple attractors–followers systems, with some agents that randomly roam throughout the environment and emit a field indicating their presence and other ones that are sensitive to those signals and are attracted by places where they are more intense. The user interface is similar: agents are represented as circles in sites (light gray circles are attractors, black ones are followers), while fields related to attractors are light gray and become darker when degraded by the distance. White sites are empty, with no agent and no active field. A screenshot of this system is shown in Figure 9. This example was developed to test the proposed approaches supporting field diffusion in case of an irregular spatial structure.

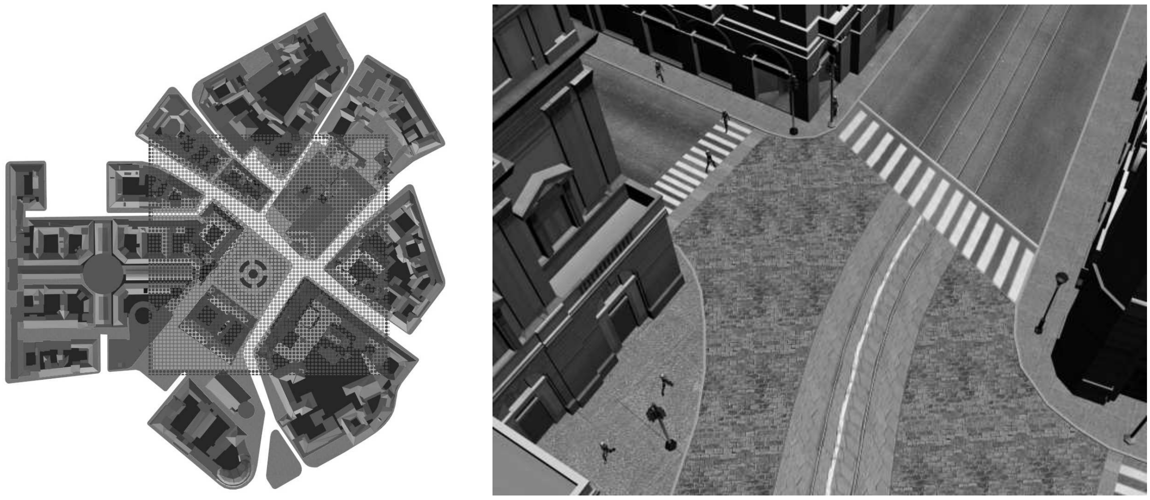


Figure 10. The left part of the figure sketches the abstraction of a physical space, while the right one presents a screenshot of the 3D system, with pedestrian following paths generated by attractive fields.

However SCAs models very similar to those described in this subsection, combining effects of attraction and repulsion generated by environmental elements and by agents themselves, were also adopted for the representation of crowd behaviour in 3D system for the simulation of believable agents in realistic environments. A screenshot of the system, showing a pedestrian following paths generated by attractive fields, is shown in Figure 10[3].

5. Conclusions and future developments

In this paper issues related to the implementation of diffusion mechanisms as means for obtaining action–at–a–distance in the SCA model were introduced. Elements of this model were briefly described, then different approaches to the implementation of the diffusion operation were introduced and analyzed with reference to memory occupation of the related structures and time complexity of the involved algorithms. No generally optimal solution was devised, as the model is very general and can be used to represent deeply different situations whose specific features might require or indicate specific mechanisms to optimize field diffusion.

A sample application of the model, very similar to Conway's Game of Life, was also illustrated, to show how the SCA model can be used to describe and implement a typical CA. The related experiments have given some qualitative indications of the effects caused by the introduction of elements of heterogeneity in the system. A formal comparison between SCA and CA was out of the scope of the paper, as well as a quantitative analysis of the performed experiments; in particular the latter will be the subject of a future work. Other future developments provide the applications of this model to the areas of urban modelling and crowd behaviour simulation. The sketch of a possible way to describe pedestrian behaviour with SCA model concepts, and in particular field diffusion, was also illustrated.

[3]The 3D model of Scala Square appears courtesy of Geosim systems.

References

[1] Bandini, S., Manzoni, S., Simone, C.: Dealing with Space in Multi-Agent Systems: a model for Situated MAS, *Proceedings of the First International Joint Conference on Autonomous Agents and Multi–Agent Sytems (AAMAS 2002)*, pp. 1183–1190, ACM press, 2002.

[2] Bandini, S., Manzoni, S., Simone, C.: Enhancing Cellular Spaces by Multilayered Multi Agent Situated Systems, *Proceeding of the 5th International Conference on Cellular Automata for Research and Industry (ACRI 2002)* (S. Bandini, B. Chopard, M. Tomassini, Eds.), LNCS 2493, pp. 156–167, Springer–Verlag, 2002.

[3] Bandini, S., Manzoni, S., Simone, C.: Situated Cellular Agents in Non–Uniform Spaces, *Seventh International Conference on Parallel Computing Technologies (PaCT–2003)*, LNCS 2763, pp. 10–19, Springer–Verlag, 2002.

[4] S. Bandini, S. Manzoni, and G. Vizzari. MultiAgent Approach to Localization Problems: the Case of Multilayered Multi Agent Situated System. *Web Intelligence and Agent Systems An International Journal*, IOS Press, Vol.2, No.3, pages 155–166, 2004.

[5] Batty, M., Couclelis, H., Eichen, M.: Urban systems as cellular automata, Editorial, *Environment and Planning B: Planning and Design*, vol. 24, no. 2, pp. 159–164, 1997.

[6] Dijkstra, J., Timmermans, H.J.P., Jessurun, A.J.: A Multi–Agent Cellular Automata System for Visualising Simulated Pedestrian Activity, *Proceedings of the 4th International Conference on Cellular Automata for Research and Industry (ACRI 2000)*, pp. 29–36, Springer–Verlag, 2001.

[7] Eckart, J. D.: A cellular automata simulation system: version 2.0, *SIGPLAN Notices*, vol. 27, no. 8, pp. 99–106, ACM, 1992.

[8] Ferber, J.: *Multi-Agents Systems*, AddisonWesley, 1999.

[9] Hochberger, C. and Hoffmann, R., Waldshmidt, S.: The Cells Start Walking: Moving Object in CDL++, *Proceedings of the 3rd Conference on Cellular Automata for Research and Industry (ACRI 1998)*, pp. 271–282, Springer–Verlag, 1998.

[10] Jiang, B.: Agent–based approach to modelling urban and environmental systems within GIS, *Proceedings of the 9th International Symposium on Spatial Data Handling*, Beijing, 2000.

[11] Labrou, Y., Finin, T., Peng, Y.: Agent communication languages: the current landscape, *IEEE Intelligent Systems*, No. 14, Vol. 2, pp. 45–52, 1999.

[12] Mamei, M., Leonardi, L., Zambonelli, F.: A Physically Grounded Approach to Coordinate Movements in a Team. In *Proc. 1st International Workshop Mobile Teamwork*, IEEE CS Press, pp. 373–378, 2002.

[13] Minar, N., Burkhart, R., Langton, C., Askenazi, M.: The Swarm Simulation System: A Toolkit For Building Multi-Agent Simulations, *Santa Fe Institute Working Paper 96-06-042*, http://www.swarm.org/archive/overview.ps, 1996.

[14] Schadschneider, A.: Cellular Automaton Approach to Pedestrian Dynamics - Theory, *Pedestrian and Evacuation Dynamics* (M. Schreckenberg, S. D. Sharma, Eds.), pp. 75–86, Springer–Verlag, 2002.

[15] Tanenbaum, A.: *Computer Networks*, Prentice Hall, third edition, 1996.

[16] Torrens, P. T., O'Sullivan, D.:Cellular automata and urban simulation: where do we go from here?, Editorial, *Environment and Planning B: Planning and Design*, vol. 28, no. 2, pp. 163–168, 2001.

[17] White, R., Engelen, G.: Cellular Automata as the Basis of Integrated Dynamic Regional Modelling, *Environment and Planning B: Planning and Design*, vol. 24, no. 2, pp. 235–246, 1997.

[18] Wolfram, S.: *Theory and Applications of Cellular Automata*, World Press, 1986.