A Spatially Dependent Communication Model for Ubiquitous Systems

Stefania Bandini, Sara Manzoni, and Giuseppe Vizzari

Dipartimento di Informatica, Sistemistica e Comunicazione,
Università degli Studi di Milano–Bicocca,
Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
{bandini, manzoni, vizzari}@disco.unimib.it

Abstract. Models and conceptualizations are necessary to understand and design ubiquitous systems that are context–aware not just from a technological point of view. The current technological trend depicts a scenario in which space, movement and more generally the environment in which the computation takes place represent aspects that should be considered as first class concepts. The aim of this paper is to propose the Multilayered Multi-Agent Situated System (MMASS) model as a suitable support for the definition of conceptual architectures for ubiquitous systems. The model provides a strong concept of agent environment, which represents an abstraction of a physical environment and conceptual aspects as well, and the agent interaction model is strongly dependent on the spatial structure of the environment. After a brief presentation of MMASS, related concepts and mechanisms, a sample application domain illustrating how it can be adopted to model an ubiquitous system will be given.[1]

1 Introduction

The current trend of technological innovations is transforming the environment where human actors live and the way in which they perceive their interactions. Computers are "disappearing", their computational power is no more concentrated in identifiable spots, rather it is ubiquitous and can be potentially embedded in almost every object populating the environment. Interaction is also changing its nature, since it is not necessarily performed through traditional devices connected to traditional computers. Computation is spread in the environment, actors move in it carrying mobile devices of different kinds and access the "network" in different ways. In this new scenario the movement in a space and the related possibility to interact with other actors, according to the current location, represent new dimensions that must be taken into account as first class concepts. The environment influences what can be done and how tasks are performed, as the location influences communication capabilities and resources.

---

[1] The work presented in this paper has been partially funded by the Italian Ministry of University and Research within the FIRB project 'Multichannel Adaptive Information System'

Technological evolution is not combined with an equally rapid evolution of the conceptualization necessary to understand and govern the new situation [26]. The term *context–aware* has been introduced to represent new challenges and possibilities, but it is usually interpreted in technological terms, mainly, of physical localization and available resources (e.g. network connectivity). However the concept of context is a continuum of physical and logical aspects that do not only involve communication as an isolated event but also coordination and cooperation among actors moving in a logical space related to collaborative tasks. Interpreting the physical and logical space as separated worlds is a serious impediment to consider space as a basic dimension for computing systems adaptability. What we call "logical space" received a lot of attention and many approaches have been proposed to model the involved actors and their coordination as well as the involved informational entities. The emphasis is mainly on their mutual logical relationships, while the spatial one is simulated and managed in the same way as any other one, specifically without considering topology and metrics in an explicit way. On the other hand, the approaches primarily oriented to model the space give, at different degrees, a semantics to the various spatial entities and to their spatial relationships (see, e.g., [13]) but are not open to represent relationships of a different nature. Therefore a model able to handle space as a first class concept, but also to consider in a uniform way both physical and logical spaces, is still needed. These different spaces should both be considered, but not in a mixed way: a good model should reach the above goal by distinguishing the two kinds of space and at the same time by guaranteeing their interoperability (thanks to the above mentioned uniformity).

The design of ubiquitous systems cannot rely on global states or actors owning a global view of the system. On the contrary, control is fully distributed among entities owning a local state and a partial, subjective perception of their situation. Locality, perception, point of view are concepts that, once again, require a space where they can be defined. Fully distributed control and local autonomy is a typical characteristic of many agent based models [6]. Moreover the Multi-Agent Systems (MASs) approach has often been indicated as a suitable abstraction for the analysis and design of complex systems characterized by an inherent distribution of control and information sources [16]. Agent technology has also been considered an instrument to promote software interoperation (see, e.g., [9]), and the concepts and methodologies used for analysis and design of MASs (see, e.g., GAIA [25,27]) can be adopted in general for the modelling and design of complex distributed systems that, strictly speaking, are not agent-oriented from a software engineering point of view. Hence this approach can be applied to the design of a variety of systems. Agent interaction models however generally do not consider the spatial dimension of agent environment in an explicit way (topology and metrics).

The presence of different reference spaces, representing different classes of relationships among autonomous entities, requires a model able to represent a variety of spaces without imposing a hierarchy among them. In fact the behavior of a system exploiting different layers representing logical spaces and abstractions of the physical one emerges from the mutual interactions among them. The aim of this paper is to propose a model that incorporates features fulfilling the above requirements for the modelling of complex context–aware ubiquitous systems: the Multilayered Multi–Agent Situated System (MMASS) [1] model. The MMASS model provides a rich interaction model for agents, including synchronous *reaction* among adjacent entities and asynchronous interaction through the *field* emission–diffusion–perception mechanism. Both interactions are dependent on the spatial structure in which agents are placed, that can represent a physical space abstraction but also conceptual environments as well. The following section will briefly describe the MMASS model, highlighting the relationships with related works in agent interaction models and introducing the main concepts defined by the model. Section 3 will exploit MMASS in order to define a specific conceptual architecture for ubiquitous computing applications in the automotive area. Conclusions and future developments will end the paper.

2 MMASS Model

The Multilayered Multi-Agent Situated Systems (MMASS) model [1] is a formal and computational framework for the definition of systems made up of a set of autonomous entities acting and interacting in a structured environment. This section does not represent a formal description of the model (that can be found in [2]), but will briefly introduce its main concepts, specifically focusing on the environmental structure. In fact the latter deeply influences agents behaviour, as the environment is the source of their perceptions, a constraint limiting their actions (e.g. their movement), but it also provides them a medium to interact with other entities. First of all related works and their relationships with agent environment modelling will be described, then the MMASS and its main concepts will be introduced.

2.1 Agent Environment in Agent Interaction Models

Most models for agent–based systems generally provide direct–interaction mechanisms that do not consider the circumstances and context of the interaction. Agent environment is generally represented by a communication infrastructure, often implemented through a facilitator agent that is well–known by other entities. It acts as a directory, supplying agents with information related to other entities currently active in the system (often referred to as *social knowledge*), and allowing a direct information exchange among them. In some approaches in this area, the issue of agent discovery is tackled with more complex techniques, providing a set of middle agents collaborating to collect, maintain and provide social knowledge. Some of these approaches provide a thorough analysis of the structure of this organization of middle agents in order to provide specific features (e.g. robustness) [21], other propose a self-organization approach to obtain a flexible, dynamic, yet effective, way of obtaining a robust infrastructure for social knowledge [22]. Other results of the research in this area led to the

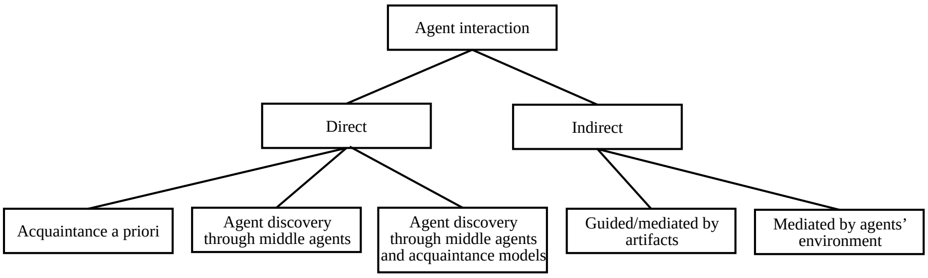

Fig. 1. A possible taxonomy of agent interaction models.

specification of acquaintance models [14] defining more precisely how this kind of agent social knowledge should be managed. However communication is generally conceived as an indiscriminate point–to–point message transfer, where messages comply to rules defined by a specific Agent Communication Language (ACL) (see, e.g., [11]). The concept of environment is thus rather weak, and in order to obtain a communication that is aware of the context in which interlocutors are placed, the involved elements (spatial or conceptual features of the environment) must be modelled and included in an agent (that can be one of the communication partners, or both of them, or even another facilitator playing the role of the environment). 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.

Other approaches provide an indirect agent interaction model, in which agents exchange information through specific artifacts and mechanisms. These artifacts represent agents' environment, at least for what concerns their means of interaction. Some models are not aimed at bringing this metaphor to the extreme, and do not mean to represent a comprehensive environmental model, but are only meant to provide a unified framework for agent interaction and coordination. In fact many of them provide extensions to the basic tuple-space-based approaches (see, e.g., Lime [19]), in order to support developer of agent based and distributed applications with a technical support for coordination in distributed and mobile environment. An interesting approach to indirect agent interaction is represented by the notion of Agent Coordination Context [17], which represents a first class abstraction to model a specific part of agents' environment focused on their social activities. In fact it captures concepts like roles, permissions and other organizational abstractions, representing also a mean for managing them at runtime, for instance in order to enforce the compliance to specific social rules.

The interaction model described in this paper differs from the previously introduced approaches as it offers interaction mechanisms that are strongly dependent on the spatial structure of the environment in which the involved entities are placed. Fig. 1 illustrates a possible taxonomy of agent interaction models, which is inspired and partly based on the one that can be found in [18]. In particular, the interaction model defined by MMASS can be placed in the category providing agent interaction mediated by agents' environment. A MAS approach that provides abstractions and concepts for environment representation and space–dependent form of communication comparable to the MMASS action–at–a–distance is Swarm [15]; other projects are based on it and propose the same kind of interaction model (e.g. Ascape [2], Repast [3], MASON [4]). Swarm is a multi–agent software platform focused on supporting the design and implementations of MASs that are based on purely reactive agents. The idea that agents should be able to understand and exploit an ACL can be unrealistic (and unnecessary) when one has to model biological systems made up of very simple entities for simulations. Moreover very simple entities exploiting their environment in order to interact among each other are able to generate fairly complex emergent behaviours. However this approach provides an explicit representation of the environment in which agents are placed, and even a mechanism for the diffusion of signals (i.e. digital pheromones) in particular versions of these structures. Recent results in the area of self-organizing systems (see, e.g., [10]) are aimed at a thorough formalization and a generalization of this kind of interaction model (often referred to as stigmergy) and its application in the engineering of MASs.

Another approach [13] provides a physically grounded model for agent interaction based on the concept of computational fields (Co-Fields). Co-Fields are signals that may be emitted either by the agents or by other elements of the environment, which supports the diffusion of those signals and thus agent interaction. In this model, agents are constantly guided by fields, that represent a mean of motion coordination, while in MMASS every perception of a field triggers a single generalized action (i.e. not strictly related to agent motion).

A different situated MASs approach [23], derived by the Influence/Reaction model [7], focuses instead on the definition of a model for simultaneous agent actions, including centralized and (local) regional synchronization mechanisms for agent coordination. In particular, actions can be independent or interfering among each other; in the latter case, they can be mutually exclusive (*concurrent* actions), requiring a contemporary execution in order to have a successful outcome (*joint* actions), or having a more complex influence among each other (both positive or negative). However, no specific mechanism for the interaction among agents occupying distant points in the environment is provided. Moreover in this approach agents' environment is related to a single layer of spatial representation.

The MMASS model provides an explicit representation of agent environment, that is made up of a set of interconnected layers whose structure is an undirected graph of sites. These layers may represent abstractions of an actual physical environment but can also be related to "logical" aspects as well (e.g.

---

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

[3] http://repast.sourceforge.net

[4] http://cs.gmu.edu/ eclab/projects/mason/

the organizational structure of a company). Between these layers specific connections (*interfaces*) can be specified. The latter are used to specify that a given field type, generated in one of these layers, may also propagate into a different one. This mechanism allows to generate interactions among different aspects and levels of the system. *Field based interaction* is the first mechanism for agent interaction, allowing a multicast form of interaction among agents occupying distant points in their environment. Adjacent agents may also perform a coordinated change of their state through a *reaction*, which is the second mechanism for agent interaction.

The model has been successfully applied to several simulation contexts in which the concepts of space and environment are key factors for the problem solving activity and cannot be neglected (e.g. crowd modelling [3], localization problems [4]). The following subsections will briefly introduce the model and the formal definitions of concepts that will be exploited to define conceptual architectures in the ubiquitous computing area.

2.2 An Overview of MMASS Model

According to the MMASS model agents are situated in sites, that is, nodes of the graphs related to a layer of the environment. Every site may host at most one agent (according to a non-interpenetration principle: “two agents cannot occupy the same site at the same time”), and every agent is situated in a single site at a given time (non–ubiquity: : “at a given time an agent occupies a single site”). Agents inherit the spatial relationships defined for the site it is occupying; in other words an agent positioned in site p is considered adjacent to agents placed in sites adjacent to p.

The adjacency relation among agents is a necessary condition for the applicability of *reaction*, the first kind of interaction mechanism defined by the MMASS model. In fact this operation involves two or more agents that are placed in adjacent sites and allows them to synchronously change their state, after they have performed an agreement. This mechanism resembles the one defined by transition rules in Cellular Automata (CA) [24], that also provide an explicit representation of a spatial structure.

CA are the model that has mainly inspired MMASS specification, and one of the main differences between the two models is the possibility to represent *action–at–a–distance*. In fact, the second interaction mechanism defined by the MMASS model provides the possibility for agents to emit *fields*, that are signals able to diffuse through the environment that can be perceived by other agents according to specific rules. This mechanism resembles pheromone approaches to agent communication (see, e.g., [10]), but fields are not just related to an intensity value and may convey more complex kind of information. Moreover for every field type a *diffusion function* can be specified in order to define how related signals decay (or are amplified) during their diffusion in the environment, from the source of emission to destination sites. Other functions specify how fields of the same kind can be *composed* (for instance in order to obtain the intensity of a given field type at a given site) or *compared*. From a semantic point of view fields themselves are neutral even if they can have related information in addition to their intensity; they are only signals, with an indication on how they diffuse in the environment, how they can be compared and composed. Different agent types may be able to perceive them or not and, in the first case, they may have completely different reaction, according to their behavioural specification. With reference to perception, an agent may perceive a field with a non–null intensity active in the site it is situated on according to two parameters characterizing its type and related to the specific field type. The first one is the *sensitivity threshold*, indicating the minimum field intensity that an agent of that type is able to perceive. The second is the *receptiveness coefficient* and it represents an amplification factor modulating (amplifying or attenuating) field value before the comparison with the sensitivity threshold. Thanks to these parameters it is possible to model dynamism in the perceptive capabilities of agents of a give type, since these parameters are related to agent state. In this way, for instance, the same agent that was unable to perceive a specific field value could become more sensitive (increase its own receptiveness coefficient) as a consequence of a change in its state. This allows to model physical aspects of perception, but also conceptual ones such as agent interests.

Reaction and field emission are two of the possible actions available for the specification of agent behaviour, related to the specification of how agents may interact. Other actions are related to the possibility to move (*transport* operation) and change the state upon the perception of a specific field (*trigger* operation). These primitives are part of a language for the specification of MMASS agents behaviour [2]. An important part of the language also provides the possibility to dynamically modify the structure of agent environment, in order to generate new sites and edges (or destroy existing ones) and create (or destroy) agents of a specific type, with a given initial state. *Agent type* is in fact a specification of agent state, perceptive capabilities and behaviour.

2.3 MMASS: Formal Definitions

A *Multilayered Multi–Agent Situated System (MMASS)* is defined as a constellation of interacting *Multi-Agent Sistuated System* (MASS) that represent different layers of the global system: $\langle MASS_1 \dots MASS_n \rangle$. A single MASS is defined by the triple $\langle Space, F, A \rangle$ where *Space* models the environment where the set A of agents is situated, acts autonomously and interacts through the propagation of the set F of fields and through reaction operations.

The structure of a layer is defined as a not oriented graph of sites. Every *site* $p \in P$ (where P is the set of sites of the layer) can contain at most one agent and is defined by the 3–tuple $\langle a_p, F_p, P_p \rangle$ where:

- $a_p \in A \cup \{\bot\}$ is the agent situated in p ($a_p = \bot$ when no agent is situated in p that is, p is empty);
- $F_p \subset F$ is the set of fields active in p ($F_p = \emptyset$ when no field is active in p);
- $P_p \subset P$ is the set of sites adjacent to p.

In order to allow the interaction between different MMASS layers (i.e. intra-MASS interaction) the model introduces the notion of *interface*. The latter specifies that a gateway among two layers is present with reference to a specific field type. An interface is defined as a 3–tuple $\langle p_i, p_j, F_\tau \rangle$ where $p_i \in P_i, p_j \in P_j$, with P_i and P_j sets of sites related to different layers (i.e. $i \neq j$). With reference to the diffusion of field of type F_τ the indicated sites are considered adjacent and placed on the same spatial layer. In other words fields of type F_τ reaching p_i will be diffused in its adjacent sites (P_p) and also in p_j.

A MMASS agent is defined by the 3–tuple $< s, p, \tau >$ where τ is the *agent type*, $s \in \Sigma_\tau$ denotes the *agent state* and can assume one of the values specified by its type (see below for Σ_τ definition), and $p \in P$ is the site of the *Space* where the agent is situated. As previously stated, agent *type* is a specification of agent state, perceptive capabilities and behaviour. In fact an agent type τ is defined by the 3–tuple $\langle \Sigma_\tau, Perception_\tau, Action_\tau \rangle$. Σ_τ defines the set of states that agents of type τ can assume. $Perception_\tau : \Sigma_\tau \rightarrow [\mathbf{N} \times W_{f_1}] \ldots [\mathbf{N} \times W_{f_{|F|}}]$ is a function associating to each agent state a vector of pairs representing the *receptiveness coefficient* and *sensitivity thresholds* for that kind of field. $Action_\tau$ represents instead the behavioural specification for agents of type τ. Agent behaviour can be specified using a language that defines the following primitives:

- $emit(s, f, p)$: the *emit* primitive allows an agent to *start the diffusion of field* f on p, that is the site it is placed on;
- $react(s, a_{p_1}, a_{p_2}, \ldots, a_{p_n}, s')$: this kind of primitive allows the specification a *coordinated change of state* among adjacent agents. In order to preserve agents' autonomy, a compatible primitive must be included in the behavioural specification of all the involved agents; moreover when this coordination process takes place, every involved agents may dynamically decide to effectively agree to perform this operation;
- $transport(p, q)$: the *transport* primitive allows to *define agent movement* from site p to site q (that must be adjacent and vacant);
- $trigger(s, s')$: this primitive specifies that an agent must *change its state* when it senses a particular condition in its local context (i.e. its own site and the adjacent ones); this operation has the same effect of a reaction, but does not require a coordination with other agents.

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

Each MMASS agent is thus provided with a set of sensors that allows its interaction with the environment and other agents. At the same time, agents can constitute the source of given fields acting within a MMASS space (e.g. noise emitted by a talking agent). Formally, a field type t is defined by

$$\langle W_t, Diffusion_t, Compare_t, Compose_t \rangle$$

where W_t denotes the set of values that fields of type t can assume; $\mathit{Diffusion}_t : P \times W_f \times P \rightarrow (W_t)^+$ is the diffusion function of the field computing the value of a field on a given space site taking into account in which site (P is the set of sites that constitutes the MMASS space) and with which value it has been generated. $Compose_t : (W_t)^+ \rightarrow W_t$ expresses how fields of the same type have to be combined (for instance, in order to obtain the unique value of field type t at a site), and $Compare_t : W_t \times W_t \rightarrow \{True, False\}$ is the function that compares values of the same field type. This function is used in order to verify whether an agent can perceive a field value by comparing it with the sensitivity threshold after it has been modulated by the receptiveness coefficient.

3 An MMASS Architecture for Ubiquitous Systems

In order to exemplify the MMASS as a model for the design of conceptual architectures in the ubiquitous computing area, a sample application scenario in the automotive context will be introduced. In fact modern cars are equipped with a large number of sensors (for instance related to the state of brakes, steering and other vehicle subsystems) and are equipped with various microcontrollers (e.g. devoted to engine control, air conditioning [20]). Information related to these devices is generally exploited to allow, enhance or maintain vehicle operation, but is otherwise wasted. The interconnection among these devices is generally developed according to some vehicular network, commonly called Controller Area Network [12]. According to this trend in automotive technology, it is thus possible to design new devices which are able to interface with existing electronic modules, in order to store relevant data, perform some kind of elaboration (e.g. check for crash conditions, perform self diagnosis), and communicate with external systems through wireless communication devices. These new technological devices could be designed in order to support new applications based on the interaction among autonomous mobile computational units spread in the environment and other fixed–position centres, that manage them in order to offer services that are aware of the context of the remote units.

Part of this concept of context is surely represented by an abstraction of the spatial structure of the environment, which may represent a map indicating conceptual communications flows among the various entities. This abstraction may be mapped to a MMASS layer, but also others aspects of the global system may be modelled through different layers interfaced to the previous one. Fig. 2 shows a possible arrangement of three MMASS layers respectively devoted to the management of the spatial aspects, of the emergency management context and to location–aware touristic information provisioning. In the following subsections more details on how these coordinated contexts may me modelled in terms of MMASS will be given.

3.1 Spatial Abstraction Layer

The typical ubiquitous computing scenario provides a number of mobile devices that are able to communicate with other entities, in order to offer some kind of

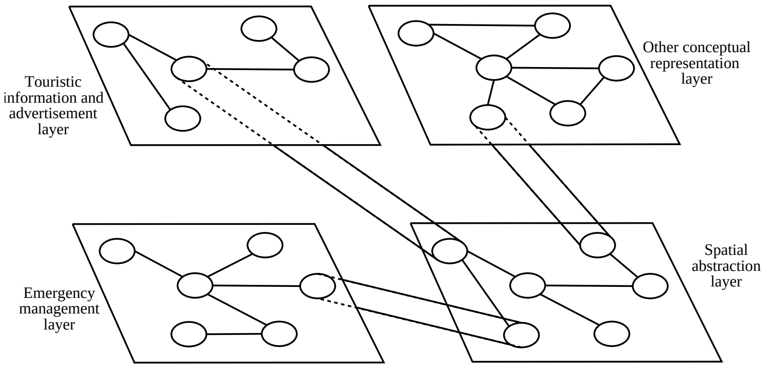

Fig. 2. The multilayered structure of the described application, highlighting interfaces among layers.

service to end–users. The nature of offered services is a key factor in determining a possible architecture for such systems, but in general those mobile devices will communicate with another kind of entity, that can be thought to have a fixed position which could even be not particularly relevant to the application. This entity could be a centralized storage and elaboration facility, or it could be part of a composite network, with different nodes that collaborate in order to supply services to the end–user.

A diagram showing this kind of architecture is shown in Fig. 3: in this case the spatial structure is an abstraction of the physical space adopted to define and manage communications in the system. In other words it is used to define which node will manage requests issued by the remote mobile entity. For example, a car fleet management system could be made up of different immobile entities, serving vehicles spread over the territory, connected through a central storage facility; the GSM standard provides a similar architecture with decentralized management of mobile terminals but a centered entity (the Home Location Register [8]) for the storage of subscribers data. In some situations such a central entity is not required, but when acquired and stored data must be analyzed (for instance with data mining techniques in order to derive profiling information) it can be appropriate to have a single data storage facility. On the other hand, if the system only has to supply an emergency assistance service to end–users represented by car drivers, there could be just decentralized centres. The area covered by the service can be partitioned into several sub–areas, and every user should be initially registered to a specific peripheral assistance centre (complex units including people, PCs, computer networks, and so on), at the moment of service subscription. The user can be handled by this unit while he/she remains in this area, and when his/her vehicle moves into another area the two peripheral centres could exchange information related to the user. Even in this case there are similarities with architectures designed for mobile wireless device. In

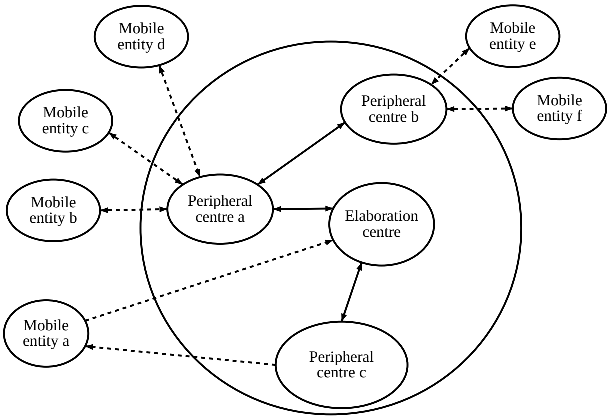

Fig. 3. A possible architecture for ubiquitous applications in the automotive domain.

fact this kind of operation involving peripheral centres can be viewed as a non–critical form of handover, and the management of this event could be derived by protocols designed in that area.

Exploiting the explicit description of the environment that the MMASS approach provides it is possible to take into account the different structures of logical connections (i.e. which peripheral centre is currently managing a specific mobile entity) to model the interactions among entities of the system. From a conceptual point of view, the previously described interactions between peripheral centres and mobile entities (i.e. authentication and login procedure, and mobile entities handover) can be modelled as *reactions*. In fact the initial interaction among mobile entities and peripheral centres can be considered a synchronous agreement process in which the former identifies itself and the latter grants access to the offered services.

Given PC the agent type that specifies features of agents related to peripheral centres, and ME the type related to mobile entities, in order to model the login procedure a reaction primitive must be included in $Action_{PC}$, the behavioural specification for peripheral centres. In particular the reaction can be specified as follows:

$$\begin{aligned}&action : react\Big(\langle S_a, S\rangle, a_m, \langle S'_a, S\rangle\Big)\\&condit : position(p), position(a_m, q), near(p, q), agreed(a_m)\\&effect : S'_a = S_a \cup \{a_m\}\end{aligned}$$

The state of an agent of type PC is a pair made up of the set of mobile entities that it is currently managing (S_a) and other internal information (S), which is not relevant for the example. The interaction takes place only if the agent and the mobile entity are adjacent in the spatial structure they are placed on and have agreed to react (i.e. the mobile entity has successfully performed

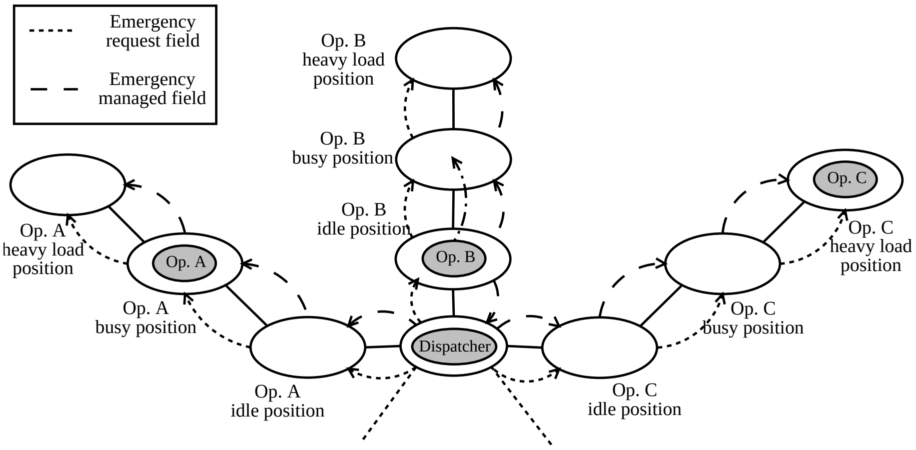

Fig. 4. A model of driver assistance context.

an authentication procedure). The effect of the reaction is the inclusion of the mobile entity a_m in the set of the ones that are served by the peripheral centre ($S'_a = S_a \cup \{a_m\}$).

Other possible interactions involving agents situated in the spatial abstraction layer are related to the diffusion of information by the peripheral centre to all the mobile entities that are currently present in its area (i.e. which are currently connected to it by an edge and included in the list of authenticated entities included in its state). In this case, the diffusion function should provide that signals reach the mobile entities unmodified, but a perceptive mechanism related to interests of the mobile user could be adopted to filter signals that are not relevant to him/her could be devised. This layer could thus also represent the basic structure for location aware diffusion of information, for instance related to road/traffic condition information or even touristic advertisement. This information could be suitably originated by different layers interfaced to this one: an example of this possibility will be described in Section 3.3.

3.2 Emergency Management Layer

The previously described layer represents just one of the aspects of the whole system, the one related to communication flows, that are dependent on spatial features, but it does not specify anything on the structure of peripheral centres and how they perform the services offered to the end–user. In order to define the behaviour of those peripheral centres a new conceptual spatial structure, interfaced to the physical spatial abstraction layer, should be defined. Fig. 4 shows a possible conceptual representation related to the operation of an organization for the management of emergency signals coming from mobile entities.

The central node, the interface to the spatial abstraction layer, hosts a *dispatcher* agent that must propagate the emergency request issued by a user or by a mobile entity in the structure defined in Fig. 3. This request is augmented with contextual information, such as data related to the user and vehicle, indications on its location, and so on. The latter can be obtained by integrating raw data transmitted by the mobile entity with a cartography and other information that might be obtained by a GIS of by a traditional information system as well. In other words, this node may enhance the information provided by the remote entity, also providing an indication on the urgency of the request by interpreting data related to the vehicle (e.g. a sudden stop may be related to a crash, and the deceleration rate may indicate the severity of this event). The dispatcher diffuses information related to the event that must be handled through a field that reaches all adjacent sites, on which idle operators are placed, but does not reach outer sites, related to busy operators. An idle operator may then *emit* a field *countering* the previous one (i.e. indicates to other operators, through the information system, the fact that he will deal with this event) and *transport* itself on the related outer site, being currently busy.

With reference to the MMASS model the previously described mechanism can be obtained through the definition of a field type related to emergencies F_e that is specified as

$$F_e = \left\langle W_e, \mathit{Diffusion}_{F_e}, \mathit{Compare}_h, \mathit{Compose}_h \right\rangle$$

where $w_e \in W_e : w_e = \left\langle id_e, type_e, int_e, d_e \right\rangle$ represents the possible values assumed by the field. Its composing parts have the following meaning: id_e represents a unique identifier of the emergency request, $type_e$ (that can be either *request* or *managed*) indicates that the field is related to the request issued by the dispatcher or represents a counter field emitted by an operator, $int_e \in \mathbb{N}$ is the intensity of the signal, and d_e is the additional data related to the emergency (which is not relevant for the example). The diffusion function specifying how this field is spread into the spatial structure is defined as follows:

$$\mathit{Diffusion}_{F_e}(p_0, f_{p_0}, p) = \begin{cases} f_e & type_e = managed \\ \left\langle id_e, type_e, int_e - dist(p_0,p), d_e \right\rangle & dist(p_0,p) < int_e \\ 0 & otherwise \end{cases}$$

The comparison function uniformly returns *true*, as all requests are perceivable by operators, and fields do not compose at all with the exception of the combination of *request* and *managed* field related to the same emergency. Formally $Compose\Big(\left\langle id_e, managed, int_e, d_e \right\rangle, \left\langle id_e, request, int_e, d_e \right\rangle\Big) = \emptyset$. With reference to field persistence in the environment, the ones marked as *request* do not vanish while *managed* ones have an instantaneous effect (i.e. they counter related *request* signals) and then are discarded.

The dispatcher performs a diffusion of a *request* field for all fields related to emergencies (which are generated in the spatial abstraction layer and forwarded to this layer thanks to a specific interface) that it perceives. The value of the

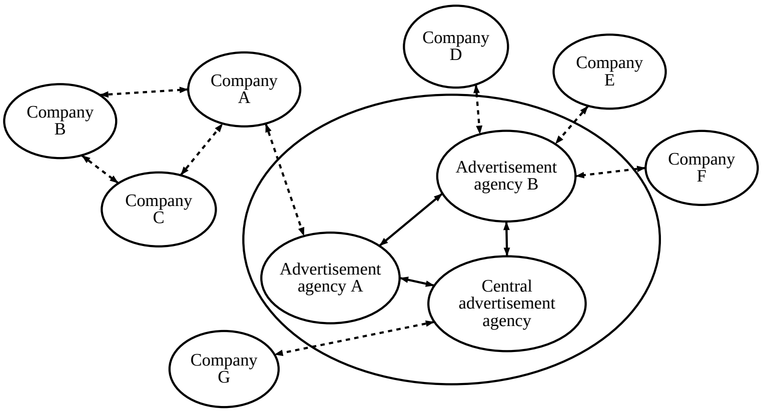

Fig. 5. A model of advertisement management context.

emitted field is $w_e^1 = \langle id_e, request, 2, d_e \rangle$, as it must be able to reach idle operators but not busy ones. An idle operator perceiving this signal and willing to manage the related request, should thus perform an emission of a counter field $w_e^2 = \langle id_e, managed, k, d_e \rangle$, that will be uniformly diffused in the environment and will cancel the request signal. After that it will *transport* itself on an outer position (the one related to the busy state). Particularly urgent requests may have a higher starting intensity, and could thus reach even outer sites. A busy operator perceiving this field may decide to delay the current lower priority task to manage the new emergency, moving to the outermost site (related to the heavy load state) in order to be shielded even from these urgent requests.

3.3 Touristic Information and Advertisement Layer

Considering another case, related to touristic information and advertisement, the layer shown in Fig. 3 defines the communication mechanism among the most suitable local information supplier and the active boxes spread over the area it is related to. However, the spatial abstraction layer does not specify anything on how companies may interact among themselves and refer to local or central advertisement agencies, in order to diffuse information related to their offers. Fig. 5 shows a possible conceptual representation related to this kind of scenario.

Thanks to the possibility to modify the spatial structure (e.g. creating or destroying sites and edges), agents related to Companies A, B and C have elected Company A as a representative that is responsible for the interaction with the Advertisement agency A. In other words it is the only one connected to the site related to the agency, with which it will interact through reaction operations. Companies D, E and F are instead interacting directly with Advertisement agency B, which will be able to diffuse information related to policies and offers through a diffusion operation. While these companies operated at a local level, Company G interacts directly with a Central advertisement agency. The latter will perform a reaction involving both Advertisement agencies A and B. The interface among this layer and the one related to the abstraction of agents' physical space, shown in Fig. 3, provides a direct connection among Peripheral centres and local Advertisement agencies, which will be able to emit specific fields that will be perceived by peripheral centres which will in turn emit signals perceivable by mobile entities positioned in their areas.

4 Conclusion

In this paper a framework for the definition of structured environments for multi–agent systems has been introduced. The MMASS model provides an explicit representation of agents' environment and interaction mechanisms that are strongly dependent on the position of involved agents and on the spatial structure of the environment.

The model, which has been previously applied to several simulation scenarios in which agent space and environment is a fundamental aspect, has been exploited to represent an ubiquitous system in the automotive area. This scenario provided mobile entities capable of storing data acquired from internal or external sensors, provided with computational and communication capabilities (i.e. active–boxes), but also to describe the interaction of entities in a specific application (i.e. emergency assistance centre). Different MMASS layers were described representing physical or conceptual abstractions specifying different aspects of the modelled system. The interaction model defined by MMASS was exploited in order to represent the communication among various entities of the system.

The design of a comprehensive software layer implementing a platform for MMASS concepts is the object of current and future developments; a first step in this direction was the analysis of distributed approaches to field diffusion [5]. Another important aspect that must be faced in order to simplify the transition from modelling to design and implementation phases is a mapping between the MMASS interaction model and possible underlying communication technologies, which are often very distant from the mechanism defined by the model.

References

1. Bandini, S., Manzoni, S., Simone, C.: Dealing with space in multi–agent systems: a model for situated mas. In: Proceedings of the first international joint conference on Autonomous agents and multiagent systems, ACM Press (2002) 1183–1190
2. Bandini, S., Manzoni, S., Simone, C.: Heterogeneous agents situated in heterogeneous spaces. Applied Artificial Intelligence **16(9-10)** (2002) 831–852
3. Bandini, S., Manzoni, S., Vizzari, G.: Situated cellular agents: a model to simulate crowding dynamics. IEICE Transactions on Information and Systems: Special Issues on Cellular Automata **E87-D(3)** (2004) 669–676
4. Bandini, S., Manzoni, S., Vizzari, G.: Multiagent approach to localization problems: the case of multilayered multi agent situated system. Web Intelligence and Agent Systems **2(3)** (2004) 155–166

5. Bandini, S., Manzoni, S., Vizzari, G.: Towards a specification and execution environment for simulations based on mmass: Managing at–a–distance interaction. In Trappl, R., ed.: Proceedings of the 17th European Meeting on Cybernetics and Systems Research, Austrian Society for Cybernetic Studies (2004) 636–641
6. Ferber, J.: Multi–Agent Systems. Addison–Wesley (1999)
7. Ferber, J., Muller, J.P.: Influences and reaction: a model of situated multiagent systems. In: Proceedings of the 2th International Conference on Multi-agent Systems, AAAI Press (1996) 72–79
8. Gabelgaard, B.: The (gsm) hlr-advantages and challenges. In: Third Annual International Conference on Universal Personal Communications, IEEE (1994) 335–339
9. Genesereth, M.R., Ketchpel, S.P.: Software agents. Communications of the ACM **37(7)** (1994) 48–ff.
10. Hadeli, K., Valckenaers, P., Zamfirescu, C., Brussel, H.V., Germain, B.S., Hoelvoet, T., Steegmans, E.: Self-organising in multi-agent coordination and control using stigmergy. In: Engineering Self-Organising Systems: Nature-Inspired Approaches to Software Engineering. Volume 2977 of Lecture Notes in Computer Science., Springer–Verlag (2004) 105–123
11. Labrou, Y., Finin, T.W., Peng, Y.: Agent communication languages: the current landscape. IEEE Intelligent Systems **14(2)** (1999) 45–52
12. Leen, G., Heffernan, D.: Expanding automotive electronic systems. IEEE Computer **35(1)** (2002) 88–93
13. Mamei, M., Zambonelli, F., Leonardi, L.: Co-fields: Towards a unifying approach to the engineering of swarm intelligent systems. In: Engineering Societies in the Agents World III: Third International Workshop (ESAW2002). Volume 2577 of Lecture Notes in Artificial Intelligence., Springer–Verlag (2002) 68–81
14. Mařík, V., Pěchouček, M., Štěphanková, O.: Social knowledge in multi-agent systems. In: Multi-Agent Systems and Applications. Volume 2086 of Lecture Notes in Artificial Intelligence., Springer–Verlag (2001) 211–245
15. Minar, N., Burkhart, R., Langton, C., Askenazi, M.: The swarm simulation system: A toolkit for building multi-agent simulations. Working Paper 96-06-042, Santa Fe Institute (1996)
16. Nwana, H.S., Ndumu, D.T.: A perspective on software agents research. The Knowledge Engineering Review **14(2)** (1999) 125–142
17. Omicini, A. In: Towards a Notion of Agent Coordination Context. CRC Press (2002) 187–200
18. Omicini, A., Zambonelli, F.: Coordination for Internet application development. Autonomous Agents and Multi-Agent Systems **2(3)** (1999) 251–269 Special Issue: Coordination Mechanisms for Web Agents
19. Picco, G.P., Murphy, A.L., Roman, G.C.: Lime: Linda meets mobility. In: Proceedings of the 21st International Conference on Software Engineering (ICSE99), ACM press (1999) 368–377
20. Prasad, V.K.: What pervasive computing brings to automotive consumer experiences, services, products & processes. Presented at *PC2001 NIST*, Gaithersberg, Maryland (2001)
21. Tichý, P.: Robustness of social knowledge in multi-agent systems. In Trappl, R., ed.: Proceedings of the 17th European Meeting on Cybernetics and Systems Research, Austrian Society for Cybernetic Studies (2004) 552–557
22. Wang, F.: Self-organising communities formed by middle agents. In: Proceedings of the 1st International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2002), ACM Press (2002) 1333–1339

23. Weyns, D., Holvoet, T.: Model for simultaneous actions in situated multi-agent systems. In: First International German Conference on Multi-Agent System Technologies, MATES. Volume 2831 of LNCS., Springer–Verlag (2003) 105–119
24. Wolfram, S.: Theory and Applications of Cellular Automata. World Press (1986)
25. Wooldridge, M.J., Jennings, N.R., Kinny, D.: The gaia methodology for agent-oriented analysis and design. Journal of Autonomous Agents and Multi-Agent Systems **3(3)** (2000) 285–312
26. Zambonelli, F., Parunak, H.V.D.: Signs of a revolution in computer science and software engineering. In: Proceedings of Engineering Societies in the Agents World III (ESAW2002). Volume 2577 of Lecture Notes in Computer Science., Springer–Verlag (2002) 13–28
27. Zambonelli, F., Wooldridge, M.J., Jennings, N.R.: Developing multiagent systems: The gaia methodology. ACM Transactions on Software Engineering and Methodology **12(3)** (2003) 317–370