Heterogeneous Agents Situated in Heterogeneous Spaces

Stefania Bandini, Sara Manzoni, Carla Simone

Dipartimento di Informatica, Sistemistica e Comunicazione
Universita' di Milano Bicocca
Via Bicocca degli Arcimboldi, 8
20126, Milano, Italy
email{bandini,manzoni,simone}@disco.unimib.it

Abstract

This paper introduces the Multilayered Situated Multi–Agent System model (MMASS), a model for Multi–Agent Systems (MAS) situated in an environment. The main feature of the Situated MAS model is to give an explicit definition of the spatial structure of the environment in which the system of situated agents are acting and interacting. Moreover, the presented model allows to represent the heterogeneity that characterizes both agent types and interactions, and the spatial structures of the environment in which they are situated. The MMASS model allows to represent both open and closed systems and two examples of its application (one for each class of systems) will be shown.

1 Introduction

The Multi–layered Situated MAS (MMASS) model characterizes Multi–Agent Systems (MAS) situated in a multi–layered environment, defining the topological structure of agent environment as a multi–layered network of sites as composed of multi–layered *heterogeneous spaces* in whose sites *heterogenous agents* are located. Each layer of the space reproduces a physical or logical space in which the system of agents is situated. Each entity of the modelled domain may be represented by agents situated in more than one layer. The MMASS model defines a set of influences generated by agents and propagating along the edges of the multi–layered network and leaving on sites information about their presence. Agents situated in this environment are strongly influenced by their position, that is, the site of the space in which they are situated. In particular, the position in the environment defines a potentially complex combination of internal and external events and states that the agent has to take into account for its actions. A Situated MAS is made of reactive agents that are characterized by various types and thus provide heterogeneity to the MAS. Agents, according to their type, perform their actions as a consequence of the perception of stimuli coming either from other agents or from the environment. Moreover, being situated, agents are sensitive to the spatial relationships that determine their constraints, abilities and cooperation relationships.

In order to illustrate the Situated MAS model, its application to a closed system in the Multi–Agent Based Simulation (MABS) [Davidsson, 2000] domain and to an open one in the Computer Supported Cooperative Work (CSCW) [Beaudouin-Lafon, 1999] domain will be shown.

2 A model for Situated MAS

A *Multilayered Situated MAS (MMASS)* can be defined as a constellation of interacting *Situated MAS's* (MASS) through the primitive:

Construct$(MASS_1 \ldots MASS_n)$

Thus, a MMASS is defined by the set of its constituting *Situated MAS's* ($\{MASS_1 \ldots MASS_n\}$).

Each MASS of a Multilayered Situated MAS is defined by the triple

$< Space, F, A >$

where *Space* models the environment where the set A of agents are situated, act autonomously and interact via the propagation of the set F of fields. A MMASS can be also denoted by:

$$\big\langle < Space_1, F_1, A_1 > \ldots < Space_n, F_n, A_n > \big\rangle$$

where

- $\left(\bigcup_{i=1..|Spaces|} Space_i \right)$: set of *Spaces* defining the multilayered spatial structure of the MMASS;

- $\left(\bigcup_{i=1..|F|} F_i \right)$: set of *fields* acting in the MMASS;

- $\left(\bigcup_{i=1..|A|} A_i \right)$: set of *agents* situated in it.

Field emission–propagation–perception is the mechanism defined for asynchronous interaction among agents situated in the same or different MASS's: an agent emits a field that propagates throughout the *Space* and can be perceived by other agents. In order to allow MASS interaction the Situated MAS model introduces the notion of *interface*. The interface of a MASS specifies fields imported into and exported

from each MASS, and takes the following format:

Interface($MASS_i$, **export** : **E**; **import** : **I**)

where $E \subset F_i$ and $I \subset F_i$ are respectively the set of fields exported from and imported into $MASS_i$. Imported fields are used in agent actions as internal fields do. As will be better explained in section 2.3, they can be mentioned and used in the definition of agent perception function. By definition, the value of an external field in any site of the local *Space* of a MASS is the value specified at its emission. Moreover, the receiving MASS has to define if and how this value has to be internally propagated by means of local fields defined for this purpose. In fact, their distribution function (see section 2.2) is highly dependent on the structure of the local *Space* which is completely hidden to the external MASS's.

2.1 Spaces

The environment where the agents of a MASS are situated is named *Space* and is defined as made up of a set P of sites arranged in a network. Each *site* $p \in P$ can contain at most one agent and is defined by the 3–tuple

$$< a_p, F_p, P_p >$$

where: $a_p \in A \cup \{\bot\}$ 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 this way the *Space* can be considered as a undirected graph of sites. For sake of space, primitives for MMASS spatial structure manipulation will not be reported in this paper [Bandini and Simone, 2001d].

2.2 Fields

Fields acting within a MASS can be generated by agents of the MASS or have a source outside the local *Space* in the case of fields imported from another MASS or from outside the MMASS in the case of open systems. Each field of a MASS is characterized by the set of values that the field can assume during its propagation throughout the *Space*. Propagation occurs according to the diffusion function that characterizes the field and that specifies how its values propagate throughout the space according to its spatial structure. Moreover, field comparison and field composition functions are defined in order to allow field manipulation. A field $f \in F$ is defined by the 4–tuple

$$< W_f, Diffusion_f, Compare_f, Compose_f >$$

where: W_f denotes the set of values that the field can assume during its diffusion in the *Space*; $Diffusion_f : P \times W_f \times P \rightarrow (W_f)^+$ is the distribution 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 generated. Since the structure of a *Space* is generally not regular and paths of different length can connect each pair of sites, $Diffusion_f$ 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. $Compose_f : (W_f)^+ \rightarrow W_f$ expresses how these values have to be combined in order to obtain the unique value of the field at a site. $Compare_f : W_f \times W_f \rightarrow \{True, False\}$ is the function that compares the composed value of a field at a site and agent sensitivity threshold, in order to verify whether an agent can perceive the field value (see the definition of agent perception in the following subsection).

2.3 Agents

The *Space* of each MASS is populated by a set A of individuals called *agents*. An agent $a \in A$ is defined by the 3–tuple

$$< s_a, p_a, \tau_a >$$

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, and $p_a \in P$ is the site of the *Space* where the agent is situated.

Agent *type* specifies the set of *states* the agent can assume, a function to express agent sensitivity to fields, and the set of actions that the agent can perform. An agent type τ is defined by the 3–tuple

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

where: Σ_τ defines the set of states that agents of type τ can assume; $Perception_\tau : \Sigma_\tau \rightarrow [\mathbf{N} \times W_{f_1}] \dots [\mathbf{N} \times \mathbf{W_{f_{||F||}}}]$ is a function associating to each agent state a vector of pairs in which for each pair i the first element expresses a coefficient to be applied to the incoming field value, and the second one expresses the sensibility threshold to f_i in the given state (let $c_\tau^i(s)$ and $t_\tau^i(s)$ be their names). This means that an agent of type τ in state $s \in \Sigma_\tau$ can perceive a field f_i only when it is verified $Compare_{f_i}(c_\tau^i(s) \cdot w_{f_i}, t_\tau^i(s))$, that is, when the first component of the i–th pair of the perception function $(c_\tau^i(s))$ multiplied for the received field value w_{f_i} is greater than the second component of the pair $(t_\tau^i(s))$.
$Actions_\tau$ denotes the set of actions that agents of type τ can perform. $Actions_\tau$ specifies whether and how agents of type τ change their state and/or position, how they interact with other agents, and how neighboring agents can influence them. In order to explain the set of agent actions, let us consider the set $Actions_\tau$ of an agent $a =< s_a, p_a, \tau >$. In the following, conditions that must be verified and the resulting effects of each action execution will be given.

Trigger
The action $trigger(s, f, s') \in Action_\tau$ can be executed when the agent is in state s and perceives field f (i.e. $f \in F_{p_a}$ and $Compare_f(c_\tau^i(s) \cdot w_f, t_\tau^i(s)) = True$). Undertaking this action the agent state changes to s'.

Transport
The action $transport(p, f, q) \in Action_\tau$ can be executed when the agent is situated in p, perceives field f and q is a site adjacent to p with no agent situated in it (i.e. $q \in P_p$ and $q =< \bot, F_q, P_q >$). As effect of its execution, the agent changes its position to q.

Emit

Asynchronous interaction among agents takes place through a field emission–propagation–perception mechanism. When an agent state is such that it can be source for a field, it executes an $emit(s, f, p)$ action, generating and defining parameters for a field f. The field emitted at p propagates throughout the space according to its diffusion function ($Diffusion_f$). The effect of this action is thus a change at each site interested by field diffusion. In particular let us consider generic site $\bar{p}$ interested by f diffusion; the emitted field is added to $F_{\bar{p}}$ if f was not already active in $\bar{p}$, otherwise the field value already present in $\bar{p}$ and the new one are composed. Field diffusion along the space allows other agents to perceive it. $Perception_\tau$ function, characterizing each agent type, defines the second side of an asynchronous interaction among agents: that is, the possible reception of broadcast messages conveyed through a field, if the sensitivity of the agent to the field is such that it can perceive it. This means that a field can be neglected by an agent of type τ if its value at the site where the agent is situated is less than the sensitivity threshold computed by the second component of the $Perception_\tau$ function.

React

Reaction defines the *synchronous interaction* among a set of agents characterized by given states and types and pair–wise situated in adjacent sites (that is, *adjacent agents*). Synchronous interaction is a two–step process. Reaction among a set of adjacent 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. $reaction(s, a_{p_1}, a_{p_2}, \ldots, a_{p_n}, s')$ is the syntax of the reaction action of one of the involved agents. It states that when a subset of its adjacent agents (i.e. $a_{p_1}, a_{p_2}, \ldots, a_{p_n}$)) has previously agreed to react and its state is s, it changes its state to s'.

3 Situated MAS Applications

3.1 Simulation

The Situated MAS model has been applied in the MABS framework to the problem of the location of suitable sites for extra–urban shopping centers [Bandini et al., 2001a]. Intuitively, a suitable location for a shopping center should be accessible through a comfortable road system, close enough to large residential areas, far enough from other centers and large enough to host the mall and related facilities and services (e.g. parking lots, restaurants, gas stations). Simulations can be useful when suitable space is available and a good location for a new shopping center has to be
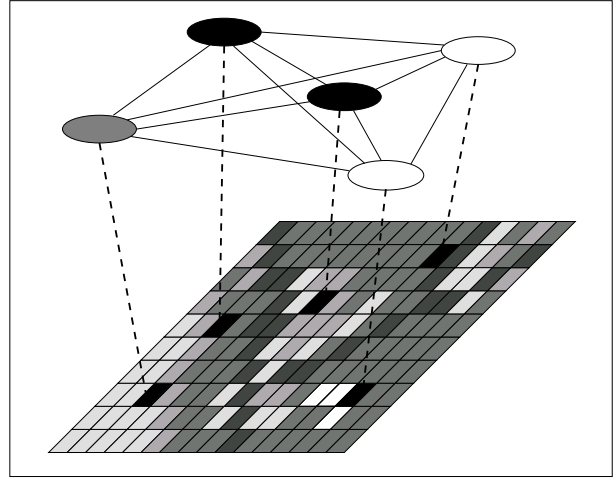

Figure 1: The Multilayered MASS composed by the *Territorial MASS* and the *Strategic MASS*. Shopping centers are represented in both MASS's: each black site of the Territorial MASS is connected by its strategic counterpart in the graph of the Strategic MASS.

chosen, or when the possibility of success in locating a new shopping center in an area already served by other retailers has to be evaluated [Couclelis, 1997]. Geographical factors, such as the proximity of other centers and residential areas, are essential for the choice of a suitable location for a shopping center. Moreover, once some centers have settled in a given area, other factors like competition should be taken into account.

In the following it will be shown how the design of a MMASS composed by two MASS's allows the simulation of the two aspects involved in this problem. Two MASS's have been defined: the *Territorial MASS* and the *Strategic MASS* (see Figure 1 for a representation of the two layers of the resulting Multilayered MASS). In the Territorial MASS, the formation or the disappearance of a shopping center is modelled by considering only geographical factors, while in the Strategic MASS already existing centers compete with one another trying to attract consumers. Each shopping center is represented in both the Territorial and the Strategic MASS. Interaction between the two MASS's is performed through field emission by agents belonging to a MASS and their perception as imported fields by agents of the other MASS.

The Territorial MASS

The Territorial MASS represents the territory where the simulation is performed. It is defined as a regular grid, that is, the territory is divided into square blocks, each one populated by an agent. Then for each site $p = <a_p, F_p, P_p>$, $a_p \neq \perp$ and $\|P_p\| = 4$.

Agent types reproduce the different types of area that can be found in an urban environment, that is: *Residential*, *Industrial*, *NotBuildable* (e.g. mountains or rivers or parks), *Road* and *Suitable* (that is areas suitable for a shopping center installation). The first four types have only one possible state, which does not change during simulations. On the contrary, agents of

f_{Sm}	$Distance1$	$Distance2$	$Distance3$
Residential	9	9	1
Industrial	1	2	0
Protected	-6	2	-2
Road	6	-1	-10
Large	-9	-7	-1
Small	-8	-4	2
Empty	3	-8	-8

Table 1: Values of field f_{Sm} according to the type of the emitting agent and the distance from the source site. The value of the field at distance greater than three sites is zero.

type *Suitable* model the formation (or the disappearance) of shopping centers by changing their state that can assume one of three possible values: *Empty*, *Small* and *Large*.

Agents of each type emit fields f_{Sm} and f_{Lg} according to their type and state in order to attract and repel agents of type *Suitable*. Positive field values of f_{Sm} (f_{Lg}) favor the formation and growth of a small (large) shopping center while negative values have a negative influence. Field values and their distribution functions change according to the type of the emitting agent. Possible values of f_{Sm}, computed according to [Engelen et al., 1995], are shown in Table 1. For instance, the presence of a large shopping center has a negative impact on the location of other centers in its surroundings, while residential areas strongly encourage the formation of shopping centers with positive field values. Every field is perceived only by *Suitable* agents that, as a consequence, change their state. For instance, if an agent in state *Empty* perceives a field value higher than a given threshold, it changes its state and becomes a *Small* or a *Large* shopping center, depending on which field has been perceived.

In order to let the Territorial MASS interact with the Strategic MASS, fields f_{Ter} and f_{Str} have been defined. *Suitable* agents of the Territorial MASS emit a f_{Ter} field when some changes in its state must be notified to its counterpart belonging to the Strategic MASS. This means that no agent of the Territorial MASS can perceive f_{Ter} fields but these fields are exported to the Strategic MASS. On the contrary, *Suitable* agents representing small or large shopping centers can perceive f_{Str} fields that are emitted in the Strategic MASS and imported into the Territorial MASS when some change occurs as effect of shopping centers competition (see the following subsection for more details). In this way, the MMASS composed by the Territorial and the Strategic MASS's is provided a mechanism for the two MASS interaction without the loss of autonomy of the two composing MASS's.

Some of the actions contained in $Action_{Suitable}$ set are:
$emit(Empty, < 3, Diffusion_{f_{Sm}}, +, <>, p)$
$trigger(Empty, f_{Sm}, Small)$
$trigger(Empty, f_{Lg}, Large)$
This means that *Suitable* agents in *Empty* state emit fields f_{Sm} with source value 3 (to positively influ-

ence the formation of small shopping centers). Moreover they change their state from *Empty* to *Small* as a consequence of the perception of a positive value of the same field while they change to *Large* when they perceive a positive value of field f_{Lg}.

The Strategic MASS
Geographical factors, such as the proximity of other centers and residential areas, are essential for the choice of a suitable location, but once some centers have settled in a given area other factors, like competition, should be taken into account. For example, one center could start low price special sales in order to attract consumers, or invest heavily on advertisement. Thus, the MMASS model allows to take into account the competition among centers already established in a given territory through the definition of the *Strategic* layer of the MMASS.

The number of sites of the Strategic MASS is equal to the number of *Suitable* agents of the Territorial MASS. In this case the graph is complete, that is, each pair of sites is connected by an edge. Moreover, in each site is situated a *Strategic* agent and it is connected to the corresponding site of the Territorial MASS in which is situated a *Suitable* agent. *Strategic* agent states are denoted by the couple $(x, f(x))$ where x is an integer value expressing the power of the agent and $f(x)$ is a function of the agent power and can assume one of the three values *Inactive*, *Defensive*, and *Aggressive*.

The state of each *Strategic* agent corresponding to an empty area on the Territorial MASS is $(0, Inactive)$. At the beginning of the simulation, when no shopping center is settled, all agents are in this state. When an *Inactive* agent perceives a f_{Ter} field, it is activated and changes its state either to *Aggressive* or *Defensive* with power proportional to the intensity of field f_{Ter} perceived.

Now, competition is modelled as a reaction between agents situated in neighboring sites. When at least one agent is *Aggressive*, it tries to "attack" the other shopping centers. The rule we adopted is simple: the most powerful one wins, and increases its power by a given value set by the user, while the loser has its power decreased by the same value.

When some change in Strategic agent state occurs, agents of this MASS emit a field (f_{Str}) with intensity corresponding to their new power value that is perceived by its counterpart on the Territorial MASS. When the field value perceived by the shopping center on the Territorial MASS drops below a given threshold, the center disappears.

3.2 Computer Supported Cooperative Work

This section illustrates how the features of the Situated MAS model can be used to construct mechanisms to support the promotion of awareness [Health et al., 1993; Rodden, 1996] within and across cooperative applications [Bandini and Simone, 2001c]. These mechanisms take facts from the applications and elaborate them in order to produce awareness information;
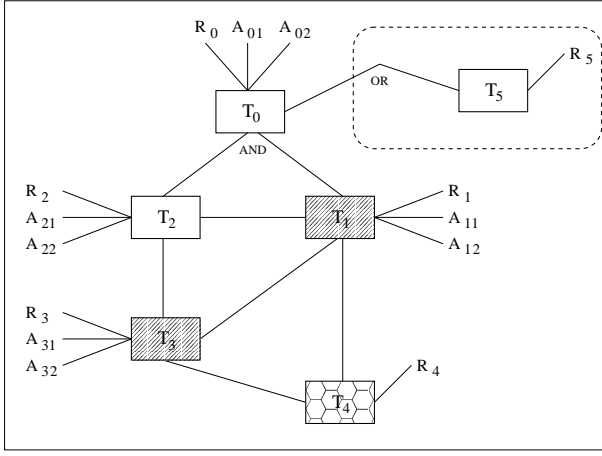
Figure 2: The awareness space associated to the work-flow after the handling of Fact(α), Fact(β) concerning T_1 and T_3, and facts managing the conclusion of T_0 and T_2

the latter is passed back to the applications and is presented at the users interface. Since in general, co-operating actors are involved in several applications at the same time, the multi–layered structure of Situated MAS is used to represent the awareness spaces associated to them to produce awareness information. Each space is populated by the proxies of entities pertaining to the associated application. Their behavior is expressed as actions of the Situated MAS model. Facts are interpreted by the model as imported fields the involved entity are always sensitive to.

In order illustrate how the above mechanisms can be constructed we consider a workflow application (WFMS) whose awareness space is defined in a natural way as a logical space populated by actors, roles and tasks. The topology of the space is defined by the relations expressing causal dependencies between tasks, assignment of responsibility for tasks to roles and allocation of actors to tasks. Moreover, tasks to be executed concurrently are also connected for sake of awareness promotion. In the same organization it is likely that another application supports the collaborative management of human resources (HRM). For sake of conciseness, we do not specify its internal structure. These two applications have associated an awareness space as shown in Figure 2. Let us consider them in turn and sketch how a whole (quite complex) mechanism can be constructed to promote awareness among the actors involved in the workflow and a specific role located in the second space. In the awareness space associated to the WFMS, each type τ has the following sets of states
$\Sigma_{Task} = \{enacted, future, concluded, in\_trouble\}$
$\Sigma_{Actor} = \{doing, done, available\}$
$\Sigma_{Role} = \{assigned\_to, in\_charge, done\}$
In the following we give some examples of the $Actions_\tau$ set and of the $Perception_\tau$ function characterizing the above types.
The enactment and execution of the WFMS generate

facts that are elaborated by the model in order to compute which awareness information has to reach each agent on the basis of its state, sensitivity and position in the (logical) awareness space. This means that the standard external field associated to each incoming fact activates a sequence of actions (script) to elaborate it. Examples of such fact–script pairs are considered here below.

$Fact(\alpha)$ = the selection of a branch of an OR node. For example in Figure 2 the left branch is selected. The rule associated to this event activates a space manipulation primitive disconnecting the right branch of the workflow. This portion of space becomes unreachable from now on for any awareness purpose.

$Fact(\beta)$ = task T_i is enacted by R_i allocating $A_{i1} \ldots A_{ik}$ to T_i. This event is interpreted by the following two actions:
1. Each involved actor A_{ij} executes:
$transport(p_j, f, q)$
where the perception of the field f corresponds to receiving the fact notification from the workflow application; q belongs to the neighborhood of the site where T_i is located.
2. Then A_{ij}, T_i and R_i execute respectively the following reactions which result in synchronous change of their states:
$react(available, A_{i1}, .., A_{i(j-1)}, A_{i(j+1)}, .., A_{ik}, R_i, T_i, doing)$
$react(future, A_{i1}, \ldots, A_{ik}, R_i, enacted)$
$react(assigned\_to(T_i), A_{i1}, \ldots, A_{ik}, T_i, in\_charge(T_i))$
The fact concerning task conclusion can be treated analogously by the execution of a reaction by A_{ij}, T_i and R_i, leading them to the state *done, concluded* and *done* respectively.

$Fact(\gamma)$ = Task T_i is going beyond its deadline, its state becomes critical. The awareness information generated by this event is as follow: the task changes its state and propagates a warning information to the other agents, with strength depending on how much they are involved, that is logically distant. This event is elaborated by T_i according to the following script:
1.$trigger(enacted, f, in\_trouble)$
where the perception of the field f is the reception of the fact from the application.
2.$emit(in\_trouble, TR, p_i)$
where TR assumes values in **N** (the set of integers); *Compose* is the sum and *Compare* is the less than relation in **N**; and

$$Diffusion_{TR}(s_0, 2, s) = \begin{cases} 2 & if \ s = s_0 \\ \frac{2}{dist(s, s_0)} & otherwise \end{cases}$$

where s_0 is the site where the field has been emitted (e.g. p_i where T_i is situated in the above case), and $dist(s, s_0)$ is the distance between s_0 and any site s. TR values are made decrease with the distance since the bigger the distance between entities the weaker

the logical relation linking them.

So for example, if T_1 is *in_trouble* then it emits the field TR with 2 as initial value. The value at the sites where R_1, A_{11} and A_{12} are situated is 2, while it is 1 at the sites where R_j and A_{j1} and A_{j2} (with $j \neq 1$) are located.

As an example of the perception function, let us consider the type *Role* and its sensitivity to the above field TR:

$$Perception_{Role}^{TR}(done) = (0,1)$$

$$Perception_{Role}^{TR}(assigned\_to) = (1,1)$$

$$Perception_{Role}^{TR}(in\_charge\_of) = (2,3)$$

The above definition of the sensitivity functions 'modulates' the perception of the field TR by the *Role* agents in the various states: in fact, they can cancel, maintain or amplify its values by playing as a coefficient for the values of TR at the sites where agents are situated.

Suppose that the information conveyed by the above TR field has to reach the awareness space associated to the HRM application. The awareness space of the HRM application contains a role, called *Personnel Manager*, which is sensitive to TR in all its possible states, with different thresholds associated to them. When it is situated in any site p and perceives TR it changes its state to *trouble_handling* and decides to assign an additional actor which is in the state *available* to the task T_1 that is *in_trouble*. Then, it emits a field *Assign_to(task)*, which is exported to the awareness space of WFMS. The following script belongs to its set of Actions:

1 $trigger(any\_state, TR, trouble\_handling)$
2 $emit(trouble\_handling, Assign\_to(T1), p)$

This implies that the type *Actor* in the WFMS awareness space has to be sensitive to *Assign_to(task)* when it is in the state *available* and its set of Actions contains the following script:

$trigger(available, Assign\_to(T1), doing)$
$transport(p, doing, q)$

where q is an empty site and belongs to the neighborhood of the site where T_1 is located.

Finally, the combination of the two awareness spaces is obtained by the **Construct**() and **Interface**() primitives as follows:

$Construct(WFMS, HRM)$
$Interface(WFMS,$
$export : TR, Import : Assign\_to(task))$
$Interface(HRM,$
$export : Assign\_to(task); import : TR)$

4 Conclusions and Future work

A model for systems of situated agents, the Multilayered Situated MAS (MMASS) model, has been presented. The MMASS model allows to represent both open and closed systems and examples of its application to both classes of systems have been shown. The paper has shown through the application of the MMASS model to two problems, how the multiple layers constituting a MMASS allows different views of a same system. This work is part of a larger project aimed to develop a support tool for designing, developing and running applications based on the Situated MAS model [Bandini et al., 2001b]. The aim of the project is to provide system designer with a language and the related infrastructure for the development of systems of agents that are characterized and influenced by their spatial position and where the spatial relationship among agents is considered in agent interaction.

References

[Bandini et al., 2001a] Bandini, S., Manzoni, S., Pavesi, G. and Simone, C. Location of Extra–Urban Shopping Centres: A Multi–Agent based Approach, Proceedings of INPUT 2001: Informatica e pianificazione urbana e territoriale, Isole Tremiti, June, 2001.

[Bandini et al., 2001b] Bandini, S., De Paoli, F., Manzoni, S., and Simone, C. *An OO Framework for Multi–Agent Systems*, Proceedings of Workshop from Object to Agent (WOA 2001), Modena, Italy, 2001.

[Bandini and Simone, 2001c] Simone, C. and Bandini, S. *Integrating awareness in coopeartive applications through the reaction–diffusion metaphor*, CSCW, The international Journal of Collaborative Computing, 2001 (to appear).

[Bandini and Simone, 2001d] Bandini, S. and Simone, C., *Integrating Forms of Interaction in a Distributed Coordination Model*, Fundamenta Informaticae, to appear.

[Beaudouin-Lafon, 1999] Beaudouin-Lafon M. (ed.), *Computer Supported Co–operative Work*, John Wiley & Sons, New York, 1999.

[Couclelis, 1997] Batty, M., Couclelis, H. and M. Eichen *Urban systems as Cellular Automata*, Special issue of Environment and Planning: Planning and Design, vol. 24 n. 2, Pion, London, 1997.

[Davidsson, 2000] Davidsson, P. (ed.), *Multi Agent Based Simulation*, WN for the 2nd International Workshop on MABS at the 4th International Conference on Multi Agent Systems, Boston, 2000.

[Engelen et al., 1995] Engelen, G., White, R., Uljii, I. and Drazan., P., *Using cellular automata for integrated modelling of socio–environmental systems*, Environmental Monitoring and Assessment, 34:203–214, 1995.

[Health et al., 1993] Health, C., Jirotka, L. Luff, P., Hindmarsh, J., *Unpacking Collaboration: Interactional Organization of Trading in a City dealing Room*, Proc. 3rd European Conference on CSCW, Milano, pp. 155–170, 1993.

[Rodden, 1996] Rodden, T., *Populating the Application: A Model of Awareness for Cooperative Applications*, Proc. the ACM Conference on CSCW, New York - ACM Press, pp. 87–96.