A spatial cognitive map and a human-like memory model dedicated to pedestrian navigation in virtual urban environments.

Romain Thomas[1] and Stéphane Donikian[2]

[1] formerly PhD student at IRISA/INRIA,
Campus de Beaulieu, 35042 Rennes Cedex, France
abberom@gmail.com
[2] IRISA/CNRS,
Campus de Beaulieu, 35042 Rennes Cedex, France
donikian@irisa.fr

Abstract. Many articles dealing with agent navigation in an urban environment involve the use of various heuristics. Among them, one is prevalent: the search of the shortest path between two points. This strategy impairs the realism of the resulting behaviour. Indeed, psychological studies state that such a navigation behaviour is conditioned by the knowledge the subject has of its environment. Furthermore, the path a city dweller can follow may be influenced by many factors like his daily habits, or the path simplicity in term of minimum of direction changes. It appeared interesting to us to investigate how to mimic human navigation behavior with an autonomous agent. The solution we propose relies on an architecture based on a generic model of informed environment, a spatial cognitive map model merged with a human-like memory model, representing the agent's temporal knowledge of the environment, it gained along its experiences of navigation.

1 Introduction

One of the most important skills for a virtual human is its ability to navigate inside an environment, as it is part of a large number of behaviours. Agent navigation in a virtual environment is an important problem of interest, and a lot of researches has been dedicated to it in various research fields, such as cognitive science [2, 30], robotic [13, 16, 15], and behavioural animation [17, 23]. Reproducing the navigating activity requires more information than the geometric representation of the environment. It is necessary to provide additional data such as mereotopological and semantic information. Gibson states in his theory of affordance [10] that *animals perceive the environment in terms of what they can do with and in it.* The *what ... with* aspect has been addressed by M. Kallmann [14] with the notion of smart objects. The *in it* aspect has been addressed by N. Farenc [9] and G. Thomas [25] with their respective models of informed urban environments. This kind of approach is based on an omniscient point of view, where the agent can obtain any information from the informed environment to plan its path, which limits the realism of the simulation. Another approach based on artificial vision is using information retrieval from an image captured by a camera located on the head of the virtual human. This process, using the well known Z-buffer, has been introduced by O. Renault et al. [24] to compute the perception of a virtual human. More recently, N. Courty [6] has extended this kind of approach by blurring the peripheral vision area in the perceived image and by introducing a saliency map calculation. Salient points are extracted from the perceived image and used in the visual attention guidance of a virtual human navigating without any goal in an urban environment. C. Peters et al. [22] use also an artificial vision model, but in their case it is coupled to a human memory model (based on the Memory Model of Atkinson and Shiffrin [3]) that allow to manage scanning and object retrieval techniques inside an apartment.

Furthermore, in simulation, heuristics employed to guide the navigation focus most of the time on the shortest path computation. However, Duckham et al. [8] propose a path planning algorithm based on the most simple journey instead of the shorter one, while Hochmair et al. [12] investigate least-angle and initial segment strategies for route selection in an unknown environment. The navigation behaviour of an agent in classical simulations differs a lot from what is observed in studies involving human subjects [30], in the sense that agents simulation exhibits standardized behaviours which lack the apparent "fuzziness" and plurality of human behaviours. A human being navigating in an environment is confronted with the problem of his own spatial localization. Indeed, most of the cognitive works on that subject state that the knowledge a pedestrian can have of his environment differs a lot from what is really the environment: his perception and memory are most of the time incomplete and distorted [29]. Then it seemed interesting to us to study what should be added to the classical navigation architecture in behavioural animation, to simulate a more realistic navigation behaviour. After a careful study of formerly performed research in the field of cognitive psychology, the introduction of a spatial cognitive map and a human-like memory appeared necessary. The map is necessary to restrict the omniscience of the agent, due to the fact that planning will be computed with incomplete knowledge of the environment. As the memory is dedicated to take into account a temporal factor in the simulation, the major consequence is that, paths taken to go from one location to another will not be the same at different moment during the simulation as the state of the agent's memory is continuously evolving.

In the rest of this paper, we will present the different components of our architecture, as "add-on" to the classical perception-decision-action architecture. Section 2 will describe our architecture. Section 3 to 5 will describe respectively the model of environment we use, the model of spatial cognitive map we have created and will sketch our memory model. Then, finally, Section 6 will explain the agent's navigation process using these various elements.

2 The overall architecture

Let us introduce the overall architecture (cf figure 1) of our system. An environment is modeled by an Informed Hierarchical Topological Graph, containing two layers: the Simple Space Layer and the Global Area Layer. Simple spaces are small-surfaced spaces where the agent navigates. They come from the convex partitioning of the environment. Practically a simple space can be either a building or a navigable space. The Spatial Cognitive Map contains subjective information the agent has acquired on its environment. It is composed of a Filter Informed Hierarchical Topological Graph, a Memory Controller, a Working Memory and a Landgraph. The Filter Informed Hierarchical Topological Graph contains three layers which are topological graphs: the Filter Simple Space Layer, the Filter Global Area Layer and the Local Space Layer. A navigation module, managing the navigation algorithms, uses the data collected from the working memory to elaborate a navigation plan. All these components are presented in the next sections.

3 The environment

The virtual environment M where the simulation takes places is modeled with an Informed Hierarchical Topological Graph named IHT-Graph. It has two layers: the Simple Space Layer (named as SSL) and the Global Area Layer (named as GAL), corresponding to the two types of objects contained in the environment.

$$M = \{E, F, \gtrdot, \xrightarrow{r}\}$$

where

- $E = E_s \cup Z$ with
  E_s the set of simple spaces of the world M
  Z the set of global areas of the world M

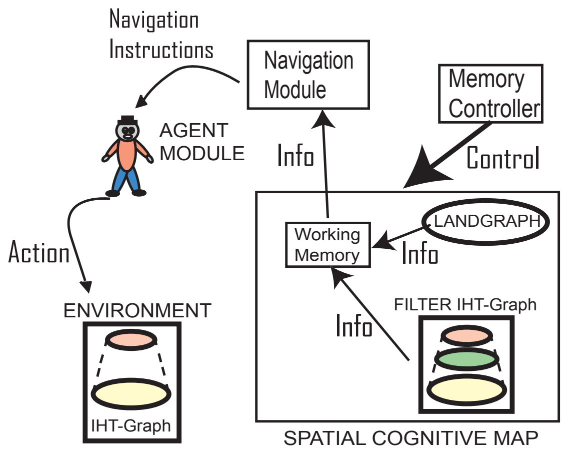

Fig. 1. The navigation system architecture

- $F = \phi_F \cup \phi_{NF}$ the set of boundaries associated to E. ϕ_F and ϕ_{NF} are respectively the sets of passable and impassable boundaries of E
- $\gtrdot$ the *part of* relation in M, linking simple spaces to global areas they belong to.
- $\xrightarrow{\tau}$ the connexity relation, that can be applied among both simple spaces and global areas.

Let us define $Ref(E)$, the set of identifiers, such as $Ref(E) = Ref(E_s) \cup Ref(Z)$, with:

- $Ref(E_s) \subset \mathbb{N}$ is the subset of $\mathbb{N}$ that contains the set of identifiers of elements of E_s.
- $Ref(Z) \subseteq \Big(Ref(E_s) \times \mathcal{P}(Ref(E_s))\Big)$ is a set of couples, whose first component is the identifier of the kernel and the second component is the set of identifiers of the ring of the global area (with $\mathcal{P}$ the set of parts).

It contains a unique identifier for each space of E.

Let us define also the association function p such as:

$$\begin{aligned} p \ : \ E &\longrightarrow Ref(E) \\ e &\longrightarrow i \end{aligned}$$

For a space $e \in E$, it returns its identifier number $i \in Ref(E)$.

Likewise, we define the inverse function p^{-1}:

$$\begin{aligned} p^{-1} \ : \ Ref(E) &\longrightarrow E \\ i &\longrightarrow e \end{aligned}$$

For an identifier $i \in Ref(E)$, it returns the associated space $e \in \mathbb{E}$.

A simple space is an object which is pictured by a convex polygon. Those polygons can be either navigable portions of the environment or building of the urban layout. The set of simple spaces E_s is obtained by partitioning the layout with a constrained Delaunay triangulation.

E_s is defined as follows:

$$E_s = \{e = (i,\ a,\ o,\ s)/i \in Ref(E_s),\ a \in B \cup N\ ,\ o \subseteq O_p\ ,\ s \in \Re\} \tag{1}$$

with

B the set of convex polygons corresponding to buildings of M
N the set of navigable convex polygons of M
O_p the set of punctual objects part of M
s is the saliency associated to the space e

An example of the layout partitioning is shown in fig.2. Furthermore a simple space contains three types of information:

- *Geometrical*: within each simple space, the associated polygon is stored as the set of its borders.
- *Topological*: the SSL is a topological graph and each simple space has at least one neighbour. Therefore, the connections between a simple space and its neighbours are stored in the simple space.
- *Semantic*: the type value of simple space is stored. It can be a building or a navigable zone.

Moreover, each simple space is gifted a *saliency parameter s*. This parameter represents the visual and thematic importance a space can have for the majority of the city dwellers. A train station will be more salient than a dead end located in a suburb. At this stage of our work, this parameter is set empirically for each space.

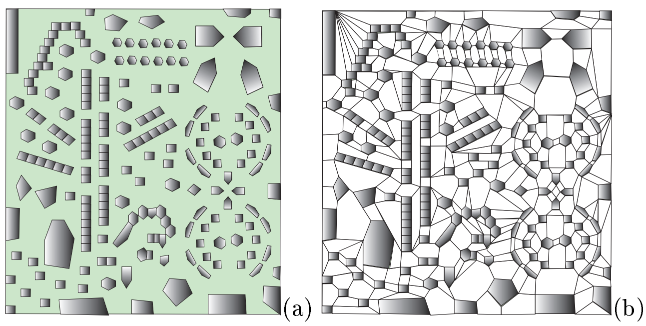

Fig. 2. (a) An urban layout, representing only buildings (b) the simple spaces of the layout.

Let us define also four functions for $e = (i,\ a,\ o,\ s) \in E_s$:

- $Ct(e) = a$. It gives the boundary of e, and is represented as an ordered list of points.
- $Ob(e) = o$. It gives the set of objects associated to e
- $\phi(e) = f$. It gives the set $f \in (\phi_F \cup \phi_{NF})$ of boundary segments of e.
- $S(e) = s$. It gives the saliency parameter of the space e.

A global area(**GA**) is an object which is a conceptual blend of a Lynch quarter [19], and a Penn local area [21]. The implementation of quarters for real and virtual cities in our model was crucial. Indeed, the route planning and navigation in an environment is a multi-level planning in term of abstraction of spaces. For instance, Lynch [19] states that a pedestrian who is not familiar with a town will be prone to position himself and navigate, using the major subdivisions of space like quarters and general orientation. Similarly the more the pedestrian's knowledge of the environment grows, the more he will navigate using very specific landmarks related to his past experiences. In our model, global areas simulate the notion of big area exhibiting a single identity.

The set Z of global areas of M is defined as follows:

$$Z = \{z_g = (i,\ n,\ c)/i \in Ref(Z),\ n \in E_s\ ,\ c \in Z_C\}$$

Practically a GA gathers simple spaces of the environment which can be related to another simple space of interest. This particular simple space is called the **kernel** of the global area (n). It symbolizes the identity of the GA. Then, the set of spaces linked to this kernel is called the **ring** of the global area (c). So each global area has a ring and a kernel.

Let us define four functions operating on a global area.

- $N(z_g) = n$. It returns the kernel of z_g
- $C(z_g) = c$. It returns the ring of z_g
- $\mathcal{R}(z_g) = r$. It returns the gathering potential of the kernel of the global area z_g
- $Ct(z_g) = a$: It returns the contour of the ring of z_g, as a list of boundaries belonging to $\phi_F \cup \phi_{NF}$.

Definition 1. The part of relation:

$$\forall z_g \in Z\ , \forall b \in E_s\ ,\ z_g > b \Leftrightarrow b \in N(z_g) \cup C(z_g)$$

A simple space b belongs to the global area z_g if and only if it is part of the ring or the kernel of z_g.

Definition 2. The connexity relation:

For simple spaces: $\xrightarrow{T}: \forall a, b \in E_s\ : a \xrightarrow{T} b \Leftrightarrow \exists f \in F, (f \in \phi(a)) \wedge (f \in \phi(b))$
For global areas: $\xrightarrow{T}: \forall a, b \in Z\ : a \xrightarrow{T} b \Leftrightarrow C(a) \cap C(b) \neq \emptyset$

Let us define the topological distance between two simple spaces $a, b \in E_s$. a and b are at a topological distance n, as defined by the following relation $a \xrightarrow[n]{\mathcal{T}} b$:

$$\xrightarrow[n]{\mathcal{T}} : \quad - \; a \xrightarrow[1]{\mathcal{T}} b \Leftrightarrow a \xrightarrow{\mathcal{T}} b$$
$$- \; a \xrightarrow[n]{\mathcal{T}} b \Leftrightarrow \exists c \in E \,,\; (a \xrightarrow{\mathcal{T}} c) \wedge (c \xrightarrow[n-1]{\mathcal{T}} b)$$

The first step in computing a GA is to determine its kernel. We have been inspired by the work of Cutini [7] and his K index, whose role is to evaluate the gathering potential of city dwellers an open space can have. We adapted the index K_{ga}, and for each simple space of the urban layout, we compute the value of this index defined by:

$$K_{ga} = \frac{\mathcal{T}(s)}{\mathcal{I}(s)}$$

where $\mathcal{I}(s)$ is the spatial integration, a very common measure in the Space Syntax community.

$$\mathcal{I}(s) = \frac{\sum_{k=1}^{card(N)} i_k}{card(N)}$$

with $i_k = min(\{i \in \mathbb{N} / e \xrightarrow[i]{\mathcal{T}} e_k\})$ and $N \subset E_s$ the set of navigable simple spaces.

It represents the mean value of the topological minimum distance between the space s and all the other simple spaces of the environment. Practically, if the topological distance between spaces $s1$ and $s2$ equals 3, it means that, there is at least 2 simples spaces lying on the path from $s1$ to $s2$.

For $e_1, e_2 \in E_s$. The visibility relation between e_1 and e_2 is represented by the following relation $\mathcal{V}(e_1, e_2)$. This relation is true if and only if:

$$\exists p \in e_1,\; \exists f \in \phi(e_2) \forall f_B \in \phi_{NF},\; \left(\overline{B(e_2)p} \cap f_B = \emptyset\right) \wedge \left(\overline{M(f)p} \cap f_B = \emptyset\right)$$

and

$\mathcal{V}(e_1, e_2) = 1$ if e_1 is visible from e_2
$\mathcal{V}(e_1, e_2) = 0$ otherwise

$\mathcal{T}(s)$ is the neighbourhood size. It is defined as the ratio of the number of spaces from which the simple space s is visible to the total number of simple spaces of the environment.

$$\mathcal{T}(s) = \frac{\sum_{k=1}^{card(N)} \mathcal{V}(s,\; e_k)}{card(N)}$$

Once the index K_{ga} is computed for each simple space, the algorithm selects spaces having the highest K_{ga} (the number of spaces selected depends on the user will). Those spaces will be kernels of all the global areas of the environment. Once the kernels are selected, their corresponding rings are computed. We derive a **radiation measure** for each kernel, depending on the kernel K_{ga} and its neighbours ones and on the kernel surface. Then, we compute the radius of the ring for each kernel. The more the kernel radiation grows the more the ring radius will be important.

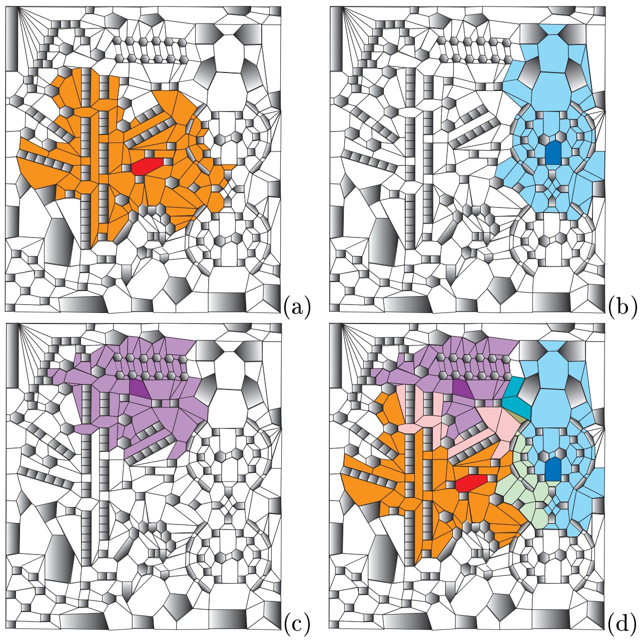

Fig. 3. (a)(b)(c) Three different Global areas computed in our sample environment (d) overlapping of Global areas (pink and green regions).

Fig.3 shows three global areas computed in our sample environment. In the figure, the kernel of each global area is darker than the spaces taken from the ring. The rings of global areas can overlap, as shown in Fig3.d: a space can belong to the rings of many global areas. Let's precise that a global area is connected to another in the global area layer if and only if their ring overlap. The SSL and GAL form a hierarchical graph. Fig.4 schematically illustrates an IHT-Graph modelling two non-connected GAs and their respective rings in the SSL.

4 The spatial cognitive map

In the previous section, we described the model of environment where the simulation takes place. The Spatial Cognitive Map (SCM) will have almost the same

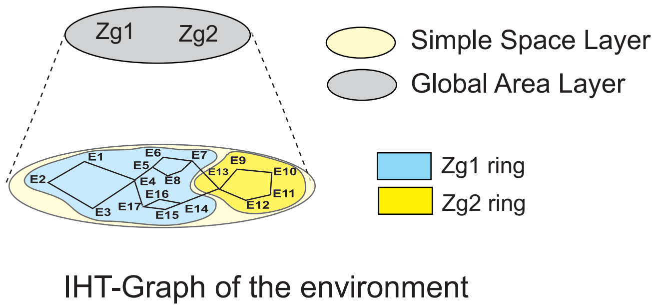

Fig. 4. An example of Hierarchical Topological Graph.

shape like the environment, given that it will model the knowledge on the environment the agent has already acquired during the simulation. The main idea is that the SCM acts as a filter between the environment and the agent's decisional module. In our architecture, the perception is simulated by a query to the database representing the environment: if the agent wants to obtain information on a space, the decisional module will query the IHT-Graph for this information. A limitation of this type of model is that it gives the agent an omniscience which impairs the realism of simulations. Limiting this omniscience to the set of spaces the agents previously discovered, allows them to exhibit navigation behaviours more related to the ones human subjects can show: the agent navigating from an origin point to a destination one, will be able to build its route using spaces it has already discovered or is currently discovering. More details on the navigation process will be given in section 6.

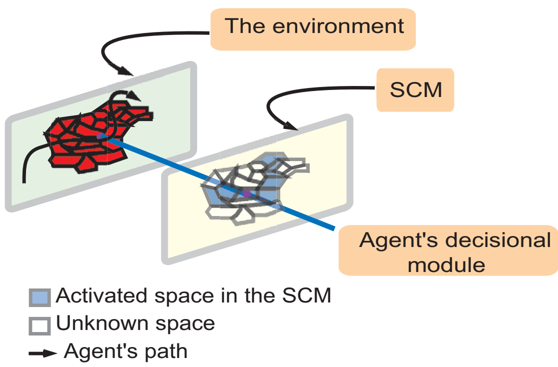

Fig. 5. The Spatial Cohnitive Map acting as a filter on the environment

Fig.5 shows a sample of simple spaces taken from an environment and the way the SCM can act as a filter. An important concept of our model is the **Activation**. Once the agent has visited a space of the environment, the corresponding filter simple space is activated is the SCM. A **filter simple space** is a "light version" of a simple space. It does not contain the geometrical, topological nor semantic information stored in the simple space, but only a reference to its corresponding space in the environment (we will see in the next section that filter spaces are endowed with memory parameters). The activation of a filter simple space implies that the agent's decisional module can obtain all information from the corresponding simple space in the IHT-graph. We mentioned that the SCM has a structure very similar to the one of the environment (IHT-Graph) that we will call **Filter IHT-Graph**. It is a hierarchical topological graph containing:

- **The Filter Simple Space Layer (FSSL)**: this layer is a topological graph containing all the filter simple spaces corresponding to the simple spaces of the environment already visited by the agent.
- **The Filter Global Area Layer (FGAL)**: this layer is the one corresponding to the global area layer of the IHT-graph of the environment. The activation of a global area is a bit more tricky than the simple space one: a filter global area **(FGA)** can only be activated if the kernel of the corresponding global area has been visited. Then, the ring of the FGA, will contain all the filter simple spaces already visited by the agent.
- **The Local Space Layer (LSL)**: this layer does not have any corresponding one in the IHT-Graph. We added the concept of local space in our architecture to model a local vision of the space surrounding the agent. As we will see below, local spaces are employed to determine visual landmarks which take part in the planning and navigation processes of the agent. A local space is a set of simple spaces, that lye in the agent's visual field.

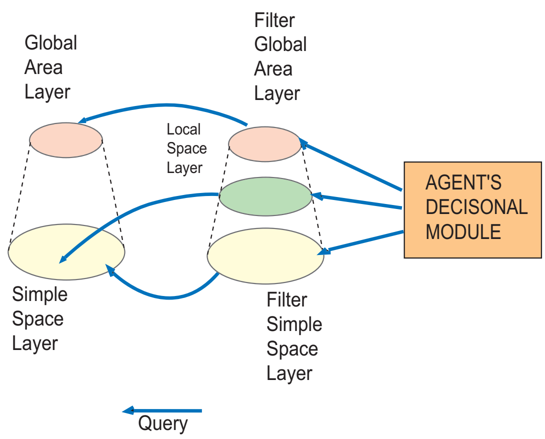

Fig. 6. Querying the environment through the Spatial Cognitive Map.

Fig.6 illustrates the query process through the SCM. When the agent wants to access information about a distant simple space or global area, it has to query the SCM first. If the filter object is activated in the SCM, access to the IHT-graph is allowed and the information is transferred to the agent's decisional module.

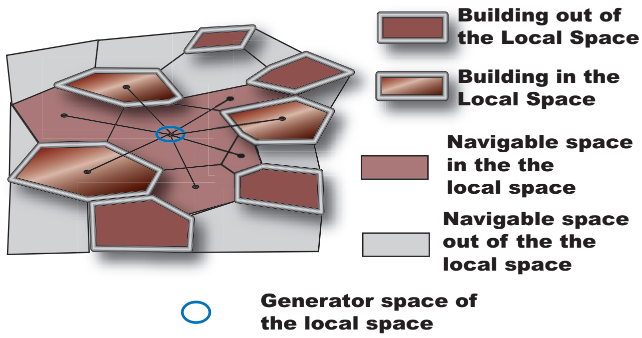

Fig. 7. An example of local space

Local Spaces are set of simple spaces that model a specific vision of the environment an agent can have (cf figure 7). Simple spaces contained in a local space are spaces which can be seen by the agent from a particular space called the **Generator** of the local space. The set of perceived spaces is called the **Surface** of the local space. All spaces that can be observed from the generator within the bounds of the agent's visual field (which the user sets at the start of the simulation), represent all the possible points of view one can have from a generator. A local space can be seen as a model of isovist [28]. The only problem in building a local space is the choice of the generator space: a generator is a simple space whose saliency value is above a saliency threshold set at the start of the simulation and is corresponding to the degree to which the agent can be impressed by its environment. Another constraint of the generator selection is given by the **agent's perceptive modes**. We adapted the Chopra and Badler [4] attentional mode, to calibrate the perception of our agents. It consists of three modes: exogenous, endogenous and passive. In the exogenous mode, the attention is spread over a high number of things, exceptional and peripheral events are noticed (The attention is high). In the endogenous mode, the agent is supposed to be thoughtful and focused on a plan to execute (It is not prone to pay attention to its environment). In the passive mode, the perception is attracted by highly contrasted and salient zones but the attention is quite low. We can order the perception modes relatively to the quality of the agent's attention they imply, from the strongest to the weakest: $Exogenous \Rightarrow Passive \Rightarrow Endogenous$. Then we can state that, a generator can only be selected if the agent perception is in a *passive* or an *exogenous* mode: we suppose that an agent in *endogenous* mode is too thoughtful to perceive the importance of a potential generator. Once a generator is perceived, its surface is computed, and the local space is added to local space layer. As described in the next section, a local space allows to identify visual landmarks. In the contrary of a global area that gathers spaces linked by a common identity given by the kernel of the GA, but which are not visible, a local space regroups spaces that are partially mutually visible. The underlying idea of global areas is identity, whereas local spaces relies on visibility.

5 The memory model

The memory model of our system is described in detail in [27]. In this section, we will sketch shortly the way it works. The memory model uses the Filter IHT-Graph as the static part of the agent's long-term memory. It can be seen as a blend of the works of Yeap [13], Atkinson and Schiffrin [3] and Lieury[18], that we have adapted to the constraint of our architecture. Each element of the IHT-Graph is endowed with a **recall** and a **recognition** parameter. Fig.8 shows the way object retrieving works. We have been inspired by the Anderson activation and retrieval model [1]. The navigation module will ask for a specific space to be retrieved in the SCM. Then rehearsal of recognition or recall tests will be triggered. The agent is endowed with a recall threshold and a recognition one. The memory values of the object to be retrieved must be above the corresponding threshold. If they are not, the threshold is weakened and a new test is run. The number of tests depends on the agent's profile, defined at the start of the simulation.

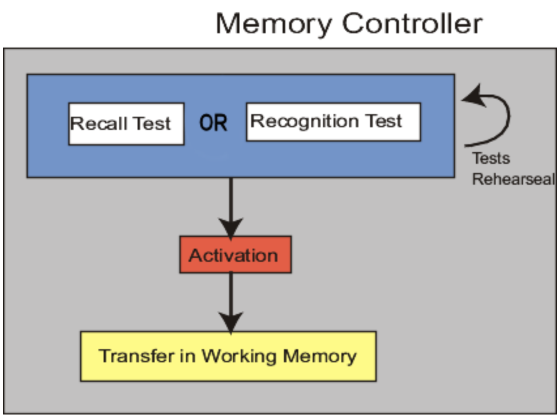

Fig. 8. The retrieving process

The Landgraph, which is a graph linking all the landmarks the agent knows, is our associative memory model. A landmark is the generator of a local space, as a local space represents a *memory image* in the Gillund and Schiffrin theory [11]. Spaces around the landmark are linked to it (Fig.9), and the edges linking them are endowed with memory parameters, so that it is possible to have an associative chain of landmarks symbolizing an abstract path in the environment.

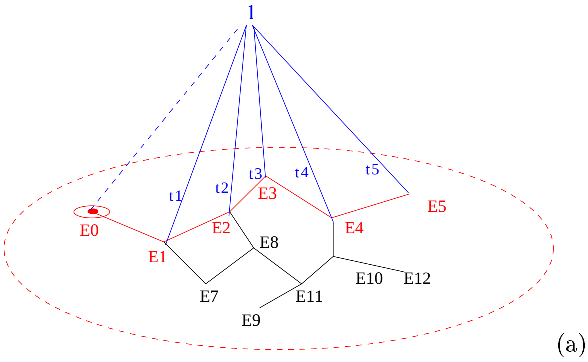

Fig. 9. A landmark l connected to its neighbour simple spaces

At last, the working memory (Fig.10) manages the short-term processes of our memory system. It can store subgraphs taken from FSSL, LSL, and FGAL, which can be viewed as Miller chunks [20]. The elements inside the WM are ordered by the navigation order, which is a temporal order as presented in the next section.

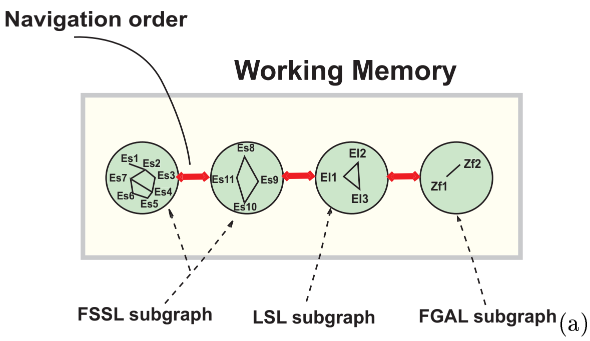

Fig. 10. A working memory example

6 The navigation

The environment, the spatial cognitive map and the memory system, described in the previous sections, allow to design the navigation system. The central idea of this system is that a human being navigates through an urban environment, using a plan it previously made. This plan contains a very small number of elements, due to the limiting properties of the working memory. Therefore many planning phases will occur in the process of navigation, we name this type of navigation the **planned navigation**. Our model of planned navigation shows many similarities with the one developed by Wiener and Mallot [30], in the sense that it generates a plan based on several levels of space abstraction, and that the plan evolves during the navigation. However, it is sometimes impossible for the agent to plan a path correctly, then it will have to navigate following an immediate heuristic [16, 30, 15], until it recognizes a space allowing to replan its path to the next place. That is what we call the **reactive navigation**. Because of the very high number of subcases in the algorithm, depending on the number of heuristics, we will not explain in detail how reactive navigation works in our system, but we will simply show how it is triggered during the planning and the navigation process.

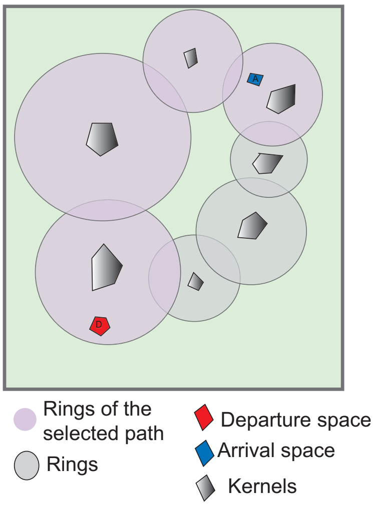

Fig. 11. Global areas selections

First, we explain the planned navigation algorithm. The first step is to run the planning algorithm. This algorithm generates a **navigation plan**. A navigation is composed of a succession of **Beacons**. The beacons are global areas kernels and Landmarks taken from the SCM of the agent. It can be either **complete** or **partial**. Given a starting point and a destination one, a complete plan contains all the information for the agent to navigate between these two points. A partial plan is produced when the agent cannot establish a continuous chain of beacons between the two points. Complete navigation plan triggers a planned navigation whereas partial navigation plan triggers a reactive navigation.

The first step in elaborating a plan is to compute a path of global areas. A continuous succession of recalled filter global areas is searched in the FGAL of the IHT-graph as shown in Fig.11. This simulates the fact that a human subject will first identify the different major zones involved in the path computation.

The FGA selection criterion is the minimum number of GA employed to make a path. GAs with high recall parameter values are privileged.

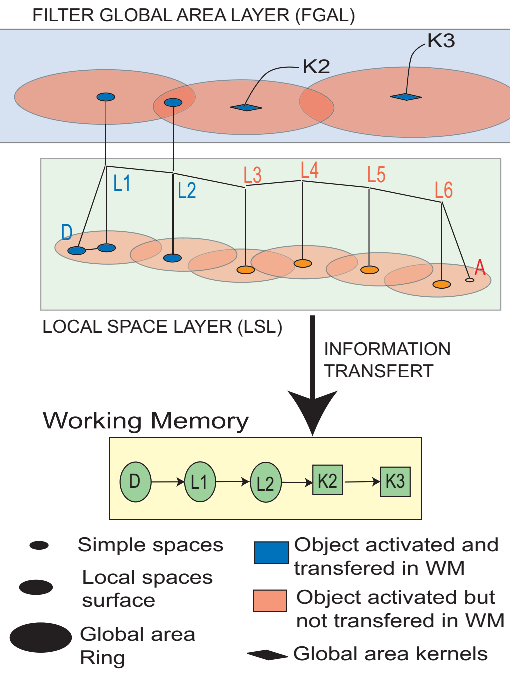

Fig. 12. a plan elaboration from the departure point D to the arrival point A

Fig.12 shows how a complete plan is computed. Three global areas are selected from the FGAL. Then the LSL is searched to find a chain of landmarks joining the point D to the second zone. Landmarks with highest recall values will be selected. Then the information selected is transferred by the memory controller in the working memory. The planned navigation is executed until the agent reaches L2. The way landmarks are linked together by the means of the surface of local spaces is described in detail in [26]. Once in L2, the agent searches local spaces of the second global area for landmarks to reach the third and last GA, and the planned navigation is started again.

From now the only recall parameters were used to retrieve information in the cognitive map. Recognition parameter values are higher than recall ones. So, if during the navigation process the agent recognizes a space that is susceptible to be the starting point of a shortest path, a re-planning process will be executed, exactly like the one described above.

During the computing of the navigation plan, due to the uncompleteness of the cognitive map, it is possible that the system reaches a configuration where it is unable to obtain a chain of global area or landmarks between the origin

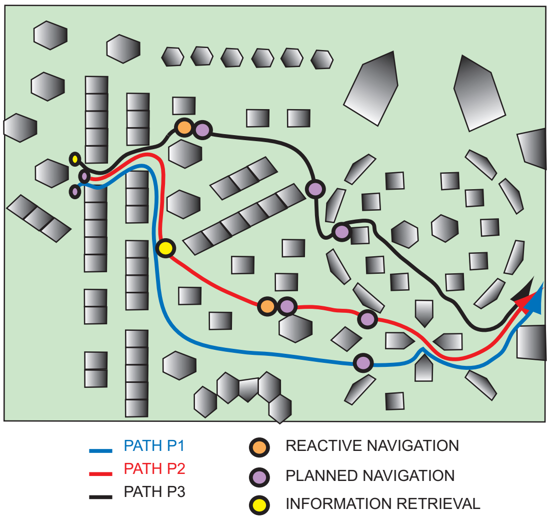

Fig. 13. Three different paths taken by the agent, depending on the state of its memory

point and the destination one. In that case, the reactive navigation algorithm is executed. Its principle is to make the agent wander in the environment, following various heuristics like "minimum deviation angles" [5], "the maximum number of borders" [16], etc... until a space is recognized, which will eventually trigger the recall of neighbour spaces. A plan is elaborated using the newly recalled and recognized spaces, until the agent loses itself again or reaches its destination. Fig.13 illustrates three different paths taken by an agent to reach a destination point from an origin point in a part of the sample environment, depending on the state of its SCM and Memory. The figure shows that many planning phases can occur in the navigation process. In some phases, the agent could not recall enough elements to create a navigation plan and is forced to switch into a reactive navigation mode.

Fig.14 shows an autonomous character travelling inside a virtual environment and illustrates the evolution of the spatial cognitive map during its navigation process. The white circle illustrates the location of the agent in the map. The different colors given to spaces of the Filter Simple Space Layer are used to illustrate the value of the recall parameter of the corresponding area. A linear interpolation between red (higher value of the parameter) and blue (lower value) colors is used.

7 Conclusion

In conclusion, we elaborated our navigation architecture which complete the classical *perception-decision-action* architecture. We presented how it was necessary to endow the agent with such structures, to allow it to exhibit navigation

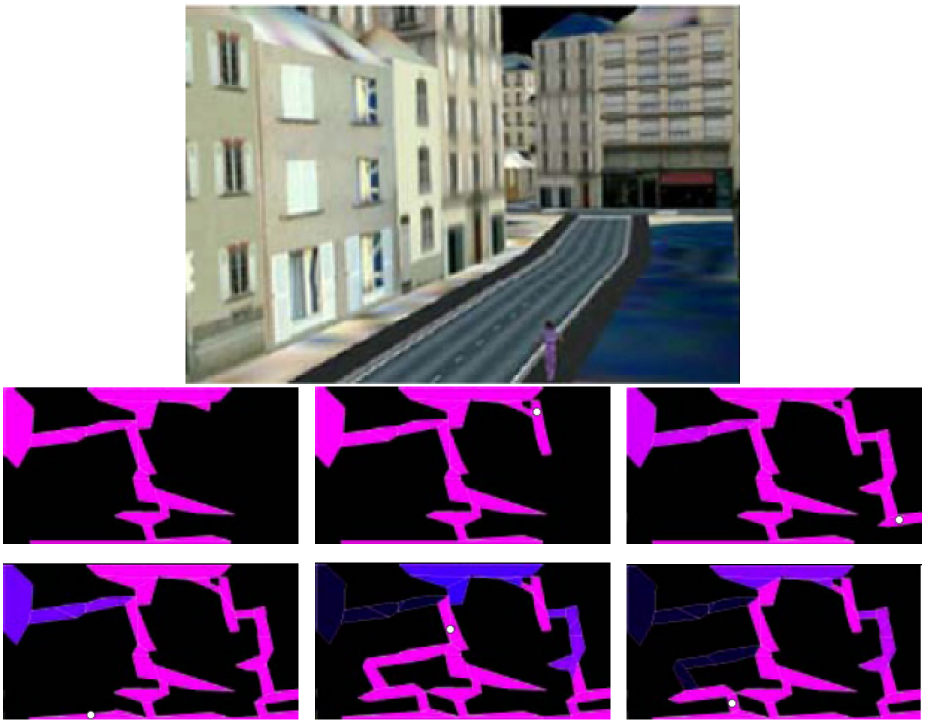

Fig. 14. Cognitive map (F-SSL) evolution during the navigation activity.

behaviors which are not standardized, due to the fact that each agent of the simulation plans its route using the personal and incomplete information taken from its SCM, which gained along the simulation. The planing process is a multi-level planning. An agent with a weak knowledge of the environment will still be able to compute a path, and will eventually reach its destination, in a different way than an agent endowed with a better knowledge of the environment. Furthermore, the fact that this information can not be available at any time due to the constraints implied by the agent's memory, allows us to use different navigation modes (planned and reactive ones), and to multiply the type of paths exhibited by the agent during the simulation. We built our model using theoretical results from the cognitive science field dedicated to human navigation, so most of our construct are cognitively justified but not validated. The model is highly parameterizable (for sake of simplicity, we have only given few of the control parameters of the model) so it would be an interesting future work to calibrate results of our model comparing to studies involving human subjects in a large scale environment.

References

1. J.R. Anderson and L.J. Schooler. Reflections of the environment in memory. *Psychological science*, 2:396–408, 1991.
2. P. Arthur and R. Passini. *Wayfinding: People, Signs and Architecture.* McGraw-Hill Ryerson and Yohimbine, Toronto, 1992.
3. R.C. Atkinson and R.M. Shiffrin. Human memory: A proposed system and its control processes. In K.W. Spence and J.T. Spence, editors, *The Psychology of Learning and Motivation*, volume 2, pages 89–95. New York: Academic Press, 1968.
4. S. Chopra-Khullar and Norman Badler. Where to look? automating attending behaviors of virtual human characters. *Autonomous Agents and Multi-Agent Systems*, 4(1/2):9–23, 2001.
5. Ruth Conroy. *Spatial Navigation In Immersive Virtual Environments.* Philosophy and architecture, The Faculty of the Built Environment, London, January 2001.
6. Nicolas Courty, Éric Marchand, and Bruno Arnaldi. A new application for saliency maps: synthetic vision of autonomous actors. In *International Conference on Image Processing, ICIP (3)*, pages 1065–1068, 2003.
7. Valerio Cutini. Lines and squares: towards a configurational approach to the morphology of open spaces. In *4th International Space Syntax Symposium*, volume 2, pages 49.1–49.14, London, 2003.
8. M. Duckham and L. Kulik. Simplest paths: Automated route selection for navigation. In W. Kuhn, M.F. Worboys, and S. Timpf, editors, *Spatial Information Theory. Foundations of Geographic Information Science, International Conference, COSIT'03*, volume 2825 of *Lecture Notes in Computer Science*, pages 169–185. Springer-Verlag, 2003.
9. N. Farenc, R. Boulic, and D. Thalmann. An informed environment dedicated to the simulation of virtual humans in urban context. In P. Brunet and R. Scopigno, editors, *EUROGRAPHICS'99*, pages 309–318. Blackwell, September 1999.
10. James J. Gibson. *The ecological approach to visual perception.* Lawrence Erlblaum Associates, Hillsdale, USA, 1979.
11. G. Gillund and R.M. Schiffrin. A retrievial model for both recognition and recall. *Psychological Review*, 91(1):1–66, January 1984.
12. H.H. Hochmair and V. Karlsson. Investigation of preference between the least-angle strategy and the initial segement strategy for route selection in unknown environments. In C. Freksa et al., editor, *Spatial Cognition IV, LNAI 3343*, pages 79–97. Springer-Verlag, 2005.
13. M.E. Jefferies and W.K. Yeap. The utility of global representation in a cognitive map. In LNCS 2205, editor, *COSIT'01*, pages 233–246, 2001.
14. M.E. Kallmann and D. Thalmann. A behavioral interface to simulate agent-object interactions in real-time. In *Computer Animation'99*, pages 138–146, Geneva, Switzerland, 1999. IEEE Computer Society Press.
15. D. Kettani and B. Moulin. A spatial model based on the notions of spatial conceptual map and object's influence areas. In *Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, International Conference, COSIT'99*, volume 1661 of *Lecture Notes in Computer Science*, pages 401–416, Berlin, 1999. Springer-Verlag.
16. B.J. Kuipers and Y.T. Byun. A qualitative approach to robot exploration and map-learning. In *Proc. of Workshop on Spatial Reasoning and multi-sensor fusio*, pages 390–404, Los Alamitos, CA, USA, 1987. IEEE Computer Society Press.

17. F. Lamarche and S. Donikian. Crowd of virtual humans: a new approach for real time navigation in complex and structured environments. *Computer Graphics Forum*, 23(3):509–518, 2004.
18. Alain Lieury. *La mémoire*. Mardaga, 4eme edition, 1992.
19. Kevin Lynch. *The image of the city*. MIT Press, Cambridge, Massachusetts, 1960.
20. Georges Miller. The magical number seven, plus and minus two: some limits of our capacity for processing information. *Psychological Review*, 63:81–97, 1956.
21. Alan Penn. Space syntax and spatial cognition: Or, why the axial line? *Environment and Behaviour*, 35:30–65, 2003.
22. C. Peters and C. O'Sullivan. Synthetic vision and memory for autonomous virtual humans. *Computer Graphics forum*, 21(4):743–752, 2002.
23. Soraia Raupp-Musse. *Human crowd modeling with various levels of behaviour control*. PhD thesis, EPFL, Lausanne, Switzerland, 2000.
24. Nadia Magnenat Thalmann, Daniel Thalmann, and Olivier Renault. A vision-based approach to behavioral animation. *Journal of Visualization and Computer Animation*, 1(1):18–21, 1990.
25. G. Thomas and S. Donikian. Modelling virtual cities dedicated to behavioural animation. *Computer Graphics Forum*, 19(3):71–80, 2000.
26. R. Thomas. *Modèle de mémoire et de carte cognitive spatiales : application à la navigation du piéton en environnement urbain*. PhD thesis, University of Rennes 1, February 2005.
27. R. Thomas and S. Donikian. A model of hierarchical cognitive map and human memory designed for reactive and planned navigation. In *4th International Space Syntax Symposium*, volume 2, pages 72.1–72.18, London, 2003. University Colledge.
28. A. Turner, M. Doxa, D. OSullivan, and A. Penn. From isovists to visibility graphs: a methodology for the analysis architectural space. *Environment and Planning B: Planning and Design*, 28:103–121, 2001.
29. Barbara Tversky. Structures of mental spaces. In *Proceedings of the 3th international space syntax symposium*, pages 12.1–12.5, Atlanta, 2001.
30. J.M. Wiener and H.A. Mallot. "fine-to-coarse" route planning and navigation in regionalized environments. volume 3, pages 331–358, 2003.