Abstraction of informed virtual geographic environments

Mehdi MEKNI*

Eau Terre Environnement Center, Institut National de Recherche Scientifique (INRS), Quebec City, Canada G1K 9A9

(Received 13 January 2012; final version received 28 June 2012)

We propose a novel method for the automated generation of virtual geographic environments that allows using geographic information system data to build what we call informed virtual geographic environment (IVGE). The description of an IVGE integrates semantic information expressed using conceptual graphs, a standard knowledge representation technique. In addition, we propose an abstraction process that uses geometric, topologic, and semantic characteristics of geographic features to build a hierarchical graph-based structure describing this IVGE. Our IVGE model enables the support of large-scale and complex geographic environment modeling for multiagent geo-simulations in which the agents are situated and with which they interact.

Keywords: GIS; informed virtual geographic environment (IVGE); abstraction process; semantic information

1. Introduction

During the last decade, the multiagent geo-simulation (MAGS) approach has attracted growing interest from researchers and practitioners to simulate phenomena in a variety of domains including traffic simulation, crowd simulation, urban dynamics, and changes in land use and coverage, to name a few examples (1). Such approaches are used to study phenomena (i.e. car traffic, mobile robots, sensor deployment, crowd behaviors, etc.) involving a large number of simulated actors (implemented as software agents) of various kinds evolving in, and interacting with, an explicit description of the geographic environment called virtual geographic environment (VGE).

A critical step towards the development of MAGS is the creation of a VGE, using appropriate representations of the geographic space and of the objects contained in it, in order to efficiently support the agents’ situated reasoning. Since a geographic environment may be complex and large-scale, the creation of a VGE is difficult and requires large quantities of geometrical data originating from environmental characteristics (terrain elevation, location of objects and agents, etc.) as well as semantic information that qualifies space (building, road, park, etc.).

In order to yield realistic MAGSs, a VGE must precisely represent the geometrical information which corresponds to geographic features. It must also integrate several semantic notions about various geographic features. To this end, we propose to enrich the VGE data structure with semantic information that is associated with the geographic features. Moreover, we propose to abstract this semantically enriched and geometrically precise VGE description in order to enable large-scale and complex geographic environments modeling.

Virtual environments and spatial representations have been used in several application domains. For example, Farenc et al. proposed a virtual scene for virtual humans, representing a part of a city for graphic animation purposes (3). Paris et al. proposed a modeling system which is able to produce a multilevel database for virtual urban environments devoted to driving simulations (4). More recently, Shao et al. proposed a virtual environment representing New York City’s Pennsylvania train station populated by autonomous virtual pedestrians in order to simulate the movement of people (3). Paris et al. also proposed a virtual environment representing a train station populated by autonomous virtual passengers, in order to characterize the levels of services inside exchange areas (4). However, since the focus of these approaches is computer animation and virtual reality, the virtual environment usually plays the role of a simple background scene in which agents mainly deal with geometric characteristics. Indeed, the description of the virtual environment is often limited to the geometric level, though it should also contain topological and semantic information for other types of applications using advanced agent-based simulations.

Current virtual environment models do not support large-scale and complex geographic environments and fail to capture real world physical environments’ characteristics. When dealing with large-scale and complex geographic environments, the spatial subdivision which
can be either exact or approximate produces a large number of cells (5). The topologic approach allows representation of such a spatial subdivision using a graph structure and takes advantage of efficient algorithms provided by the graph theory (4). However, the graph size may still remain large when dealing with geographic environments with dense geographic features (5). Moreover, geographic features with curved geometries (Figure 1) produce a large number of triangles since they are initially represented by a large number of segments.

Environment abstraction is a process to improve organization of the information obtained at the time of spatial subdivision of the geographic environment. The unification process is addressed principally in two ways: (1) a pure topological (7) unification which associates the subdivision cells according to their number of connections and (2) a more conceptual unification which introduces a semantic definition of the environment, like the informed hierarchical topologic (IHT)-graph structure (8). Lamarche and Donikian proposed a topologic abstraction approach that assigns to each node of the graph resulting from spatial decomposition, a topological qualification according to the number of connected edges given by its arity (7). The topologic abstraction algorithm aims to generate an abstraction tree by merging interconnected cells while trying to preserve topological properties (7). When merging several cells into a single cell, the composition of cells is stored in a graph structure to generate the abstraction tree.

The topologic abstraction proposed by Lamarche and Donikian reduces the size of the graph that represents the spatial subdivision (7). The grouping process relies on the topological properties of the cells and the resulting graph contains fewer nodes and preserves the topologic and geometric characteristics of the geographic environment. However, the topological characteristics are not sufficient to abstract a virtual environment when dealing with a large-scale and complex environment involving areas with various qualifications (buildings, roads, parks, sidewalks, etc.).

In this paper, we present a novel approach that addresses these challenges toward the creation of such a semantically enriched and geometrically accurate VGE, which we call an informed virtual geographic environments (IVGEs). We also detail our abstraction technique to support large-scale and complex geographic environments. The rest of the paper is organized as follows: Section 2 introduces our IVGE computation model. Section 3 presents the proposed abstraction approach which is composed of the three processes: (1) geometric abstraction; (2) topologic abstraction; and (3) semantic abstraction. Section 4 discusses the proposed abstraction approach. Finally, Section 5 concludes and presents the future perspectives on this work.

Not much research has been done on semantic integration in the description of a virtual environment. The computer animation and behavioral animation research fields provide a few attempts to integrate the semantic information in order to assist agents interacting with their environments. Semantic information has been used for different purposes, including the simulation of inhabited cities (3), computer animation (6), and simulation of virtual humans (4,5,7,8). Farenc has first used the notion of informed environments (2). She defined informed environments as a database which represents urban environments with semantic information representing urban knowledge (2). An informed environment is thus characterized as a place where information (semantic and geometrical) is dense, and can be structured and organized using rules (2). Building an informed environment as presented by Farenc consists of adding a semantic layer onto a core corresponding to classical scene (a set of graphical objects) modeled using graphical software for computer animation purposes (2). Despite multiple designs and implementations of virtual environment frameworks and systems, the creation of geometrically accurate and semantically enriched geographic content is still an open issue. Indeed, research has focused almost exclusively on the geometric and topologic characteristics of the VGE. However, the structure of the virtual environment description, the optimization of this description to support large-scale and complex geographic environments, the meaning of the geographic features contained in the environment as well as the ways to interact with them have received less attention.

2. Computation of IVGE

In this section, we briefly present our automated approach to compute the IVGE data using vector geographic information system (GIS) data. This approach is based on four stages: input data selection, spatial decomposition, map unification, and finally the generation of the informed topologic graph (ITG) (9).

2.1. GIS input data selection

The first step of our approach consists of selecting the different vector data sets which are used to build the IVGE. The input data can be organized into two categories. First, elevation layers contain geographical marks indicating absolute terrain elevations. Second, semantic
layers are used to qualify various types of data in space. Each layer indicates the physical or virtual limits of a given set of features with identical semantics in the geographic environment, such as roads or buildings.

2.2. Spatial decomposition

The second step consists of obtaining an exact spatial decomposition of the input data into cells. First, an elevation map is computed using the constrained Delaunay triangulation (CDT) technique. All the elevation points of the layers are injected into a 2D triangulation, the elevation being considered as an attribute of each node. Second, a merged semantics map is computed, corresponding to a constrained triangulation of the semantic layers.

Indeed, each segment of a semantic layer is injected as a constraint which keeps track of the original semantic data by using an additional attribute for each semantic layer.

2.3. Map unification

The third step to obtain our IVGE consists of unifying the two maps previously obtained. This phase can be depicted as mapping the 2D merged semantic map onto the 2:5D elevation map in order to obtain the final 2:5D elevated merged semantics map. First, preprocessing is carried out on the merged semantics map in order to preserve the elevation precision inside the unified map. Indeed, all the points of the elevation map are injected into the merged semantics triangulation, creating new triangles. Then, a second process elevates the merged semantics map.

2.4. Informed topologic graph

The resulting unified map now contains all the semantic informations of the input layers, along with the elevation information. This map can be used as an ITG, where each node corresponds to the map’s triangles and each arc corresponds to the adjacency relations between these triangles. Then, common graph algorithms can be applied to this topological graph, and graph traversal algorithms in particular.

3. Abstraction of IVGE

In this section, we describe the abstraction process which optimizes the description of the IVGE. Section 3.1 presents the first enhancement related to the qualification of terrain. We propose a novel approach for information extrapolation using a one-time spatial reasoning process based on a geometric abstraction. This approach can be used to _x input elevation errors, as well as to create new qualitative data relative to elevation variations. These data are stored as additional semantics bound to the graph nodes, which can subsequently be used for spatial reasoning. Section 3.2 introduces the second enhancement which optimizes the size of the informed graph structure using a topological abstraction process.

This process aims at building a hierarchical topologic graph structure in order to deal with large-scale VGEs. Section 3.3 details the third enhancement technique which propagates qualitative input information from the arcs of the graph to the nodes, which allows deduction of the internal parts of features such as buildings or roads in addition to their boundaries. Moreover, this technique uses conceptual graphs (CG) (10), a standard formalism for the representation of semantic information. Figure 2 illustrates the abstracted IVGE generation model.

3.1. Geometric abstraction

Spatial decomposition subdivides the environment into convex cells. Such cells encapsulate various quantitative geometric data which are suitable for precise computations. Since geographic environments are seldom flat, it is important to consider the terrain’s elevation and shape. While elevation data are stored in a quantitative way suitable for exact calculations, spatial reasoning often needs to manipulate qualitative information. Indeed, when considering a slope, it is obviously simpler and faster to qualify it using an attribute with ordinal values such as gentle and steep rather than using numerical values. However, when dealing with large-scale geographic environments, handling the terrain’s elevation, including its light variations, may be a complex task. To this end, we propose an abstraction process that uses geometric data to extract the average terrain’s elevation information from spatial areas. The objectives of this geometric abstraction are threefold. First, it aims to reduce the amount of data used to describe the environment. Second, it helps for the detection of anomalies, deviations, and aberrations in elevation data. Third, the geometric abstraction enhances the environment description by integrating qualitative information characterizing the terrain shape. In this section, we first present the

Figure 2. The IVGE global architecture of IVGE generation including the environment abstraction process.
algorithm which computes the geometric abstraction. Then, we describe two processes which use the geometric abstraction, namely Filtering elevation anomalies and Extracting elevation semantics.

3.1.1. Geometric abstraction algorithm

As presented in the previous chapter, the geographic environment is subdivided into cells of different shapes and sizes. The algorithm takes advantage of the graph structure obtained from the IVGE extraction process. A cell corresponds to a node in the topological graph. A node represents a triangle generated by the CDT spatial decomposition technique. A cell is characterized by its boundaries, its neighboring cells, its surface as well as its normal vector which is a vector perpendicular to its plane. Now, we introduce the notion of a group, which is a collection of adjacent cells. The grouping strategy is based on a coplanarity criterion assessed by computing the difference between the normal vectors of two neighboring cells or groups of cells. Since a group is basically composed of adjacent cells, it is obvious to characterize a group by its boundaries, its neighboring groups, its surface, as well as its normal vector. However, the normal vector of a group must rely on an interpretation of the normal vectors of its composing cells. In order to compute the normal vector of a group, we adopt the area-weight normal vector which takes care of the unit normal vectors of its composing cells, as well as their respective surfaces.

Let \( S_c \) denotes the surface of a cell \( c \) and \( N_c \) be its unit normal vector, the normalized area-weight normal vector \( \vec{N}_G \) of a group \( G \) is computed as follows:

\[
\vec{N}_G = \frac{\sum_{c \in G} (S_c \cdot \vec{N}_c)}{\sum_{c \in G} S_c} \tag{1}
\]

The geometric abstraction algorithm uses two input parameters: (1) a set of starting cells that act as access points to the graph structure and (2) a gradient parameter that corresponds to the maximal allowed difference between cells’ inclinations. Indeed, two adjacent cells are considered coplanar and hence grouped, when the angle between their normal vectors is inferior or equal to the gradient. The algorithm stops when the list of starting cells is empty. The recursive geometric abstraction algorithm is composed of five steps:

Step 1: For each cell \( c \) of the starting cells, remove \( c \) from starting cells, if \( c \) belongs to a group which is tagged “processed” then proceed to step 1, else create a new group \( G \) and proceed to step 2.

Step 2: For each neighboring group \( n \) of \( G \), depending if the neighbor has already been processed, proceed to step 4, else proceed to step 3.

Step 3: Add \( n \) to starting cells. Proceed to step 1.

Step 4: If angle \((\vec{N}_G, \vec{N}_n)\) ≤ gradient then proceed to step 5. Otherwise proceed to step 6.

Step 5: Merge \( n \) in group \( G \), and then compute \( \vec{N}_n \) using Equation (1). If the merge is validated, then \( G \) is tagged “processed.” Proceed to step 1 again for \( G \).

Step 6: If \( n \) is an unprocessed cell, create a new group \( G \) with \( n \) and proceed to step 2.

The terrain’s elevation, which characterizes each group, is still quantitative data described using area-weighted normal vectors. Such quantitative data are too precise to be used by qualitative spatial reasoning. Hence, a qualification process would greatly simplify spatial reasoning mechanisms. The geometric abstraction allows improving IVGE by qualifying the terrain’s elevation using semantics and integrating such semantics in the description of the geographic environment.

In order to highlight the outcomes of our approach, let analyze the computation cost of our geometric abstraction process. Let \( G(V,E) \) be the graph representing the virtual environment, corresponding to cells produced by the spatial decomposition process. Let \( V \) corresponds to the set of vertices and \( E \) corresponds to the set of edges at level 0. Let \( |V| = N \) be the number of nodes of the graph \( G \). Assuming that there are \( n \) starting nodes, where each starting node will finally traverse \( e_1, e_2, \ldots, e_n \) edges, respectively. The computational complexity of the algorithm is given by Equation (2):

\[
C = O\left(N^* \sum_{i=0}^{n} e_i\right) \tag{2}
\]

For existing parallel graph traversal algorithms with \( p \) processors, due to the difficulty of ensuring effective load balancing, we present them optimistically, assuming the following optimal load balancing scenario: each processor takes care of \( N/p \) source nodes. The computational complexity of our algorithm, where the upper bound of processing units is \( N \), becomes as illustrated by Equation (3):

\[
C = O(N/P \max(e_i)) \tag{3}
\]

3.1.2. Qualification of terrain shape

The geometric abstraction algorithm computes quantitative geometric data describing the terrain’s inclination. Such data are stored as numerical values allowing accurate characterization of the terrains’ elevations. However, handling and exploiting quantitative data are a complex task as the volume of values may be too large and consequently, difficult to transcribe and analyze. Therefore, we propose to interpret the quantitative data for the terrain’s inclination by qualifying areas’ elevations. Semantic labels, which are called semantics elevation, are associated with quantitative intervals of values that represent the terrain’s elevation. In order to obtain elevation semantic, we propose a process in two steps that takes
advantage of the geometric abstraction: (1) discretize the angle \( \alpha \) between the weighted normal vector \( \vec{N}_g \) of a group \( g \) and the horizontal plane and (2) assign to each discrete value the semantic information to qualify it. The discretization process can be done in two ways: a customized or automated approach.

The customized approach qualifies the terrain’s elevation by requiring the user to provide a complete specification of the discretization. Indeed, a list of angle intervals as well as their associated semantic attributes must be specified. The algorithm iterates over the groups obtained by the geometric abstraction process. For each group \( G \), it retrieves the terrain inclination value \( I \). Then, this process checks the list of angle intervals’ bounds and determines to which one belongs the inclination value \( I \). Finally, the customized discretization extracts the semantic elevation from the selected elevation interval and assigns it to the group \( G \). For example, let us consider the following inclination interval and the associated semantic elevations: \( ([10^\circ, 20^\circ], \text{light slope}) \) and \( ([20^\circ, 25^\circ], \text{steep slope}) \). Such a customized specification associates the semantic elevation “light slope” with inclination values included in the interval \([10^\circ, 20^\circ]\) and the semantic elevation “steep slope” with inclination values included in the interval \([20^\circ, 25^\circ]\).

The automated approach only relies on a list of semantic elevations representing the elevation qualifications. Let \( N \) be the number of elements of this list and \( T \) be the total number of groups obtained by the geometric abstraction algorithm. First, the automated discretization orders groups, using their terrain inclination. Then, it iterates over these ordered groups and uniformly associates a new semantic elevation from the semantics set, each \( T/N \) processed groups. For example, let us consider the following semantic elevations: \{light, medium, and steep\}. Given an ordered set \( S \) of groups as follows:

\[ S = \{ G_i | i \in \{1, 2, \ldots, 6\} \} \]

with the following terrain’s inclination values, respectively: \{5\(^\circ\), 10\(^\circ\), 15\(^\circ\), 20\(^\circ\), 25\(^\circ\), and 30\(^\circ\)\}. For every two groups (as \( T=6 \) and \( N=3 \), \( T/N=2 \)), the automated discretization assigns a new semantic elevation.

Compare these two discretization approaches. On the one hand, the customized discretization process allows users to freely specify the qualification of the terrain’s elevations. Such qualifications are used by situated agents for spatial reasoning purposes. However, qualifications resulting from such a flexible approach rely deeply on the correctness of the interval bounds’ values provided by users. Therefore, the customized discretization method requires that users have a good knowledge of the terrain characteristics in order to guarantee a valid specification of inclination intervals. On the other hand, the automated discretization process is also able to qualify groups’ elevations but without the need to specify elevation intervals’ bounds. Such a qualification usually produces a visually uniformed semantic assignment. This method also guarantees that all the specified semantic attributes will be assigned to the groups without a prior knowledge of the environment characteristics (Figure 3).

3.1.3. Improving the geometric abstraction

The geometric abstraction algorithm produces groups that are built based on their terrain’s inclination characteristics. Thanks to the extraction of elevation semantics, terrain’s inclination is qualified using semantic attributes and associated with groups and with their cells. Depending on the classification intervals, adjacent groups with different area-weighted normal vectors may obtain the same elevation semantic. In order to improve the results provided by the geometric abstraction, we propose a process that merges adjacent groups which share the same semantic elevation (Figure 4). This process starts by iterating over groups. Then, every time it finds a set of groups sharing an identical semantic elevation, it creates a new group. Next, cells composing the adjacent groups are registered as members of the new group. Finally, the area-weighted normal vector is computed for the new group. Hence, this process guarantees that every group is only surrounded by groups which have different semantic elevations.

3.2. Topological abstraction

We presented our work on the generation of IVGEs using an exact spatial decomposition scheme which subdivides the environment into convex cells organized in a topological graph structure. However, inside large-scale

![Figure 3](image-url)

Figure 3. Profile section of anomalous isolated groups adjusted to the average elevation of the surrounding ones (a) identifying elevation anomalies. Two isolated groups and angles (1 and 2) are resulting from the difference between the area-weighted normal vectors and (b) fixing elevation anomalies. Nodes in isolated groups are adjusted to the average elevation level.
and complex geographic environments (such as a city for example), such topological graphs can become very large. The size of such a topological graph has a direct effect on computation time for path finding. In order to optimize the performance of path computation, we need to reduce the size of the topological graph representing the IVGE. The aim of the topological abstraction is to provide a compact representation of the topological graph that is suitable for situated reasoning and enables fast path planning. However, in contrast to the geometric abstraction which only enhances the description of the IVGE with terrain semantics, the topological abstraction extends the topological graph with new layers. In each layer (except for the initial layer which is called level 0), a node corresponds to a single or a group of nodes in the immediate lower level (Figure 5). The topological abstraction simplifies the IVGE description by combining cells (triangles) in order to obtain convex groups of cells. Such a hierarchical structure evolves the concept of the hierarchical topologic graph in which cells are fused into groups and edges are abstracted in boundaries. To do so, convex hulls are computed for every node of the topological graph. Then, the coverage ratio of the convex hull is evaluated as the surface of the hull divided by the actual surface of the node. The topological abstraction finally performs groupings of a set of connected nodes if and only if the group ratio is equal or close to one depending on the problem domain. Let $C$ be the convexity rate and $\text{CH}(g_i)$ be the convex hull of the polygon corresponding to $g_i$. $C$ is computed as follows:

$$ C(G) = \frac{\text{Surface}(G)}{\text{Surface}(\text{CH}(G))}, \quad 0 \leq C(G) \leq 1 $$

Thus, the convex property of groups needs to be preserved after the topological abstraction. This ensures that an entity can move freely inside a given cell (or group of cells), and that there exists a straight path linking edges belonging to the same cell (or group of cells).

Figure 5 illustrates an example of the topological abstraction process and the way it reduces the number of cells representing the environment. In Figure 6(a), we present the initial vector format GIS data of a complex building. Figure 6(b) depicts the initial exact spatial decomposition which yields 63 triangular cells. Figure 6(c) presents 28 convex polygons generated by the topological abstraction algorithm. The optimization rate of the number of cells representing the environment is around 55%.

To conclude, we proposed in this section two approaches aiming at enhancing the description of the IVGE. The first approach permits to qualify the terrain’s elevation using semantic elevation which is integrated in the IVGE. The second approach aims at simplifying large informed graphs corresponding to large-scale and complex geographic environments. This approach reduces the number of convex cells by overlaying the informed graph with a topologically abstracted graph produced by a topological abstraction. The resulting IVGE is hence based on a hierarchical graph whose lowest level corresponds to the informed graph initially produced by the spatial decomposition. In the following section, we show how we use a well-known knowledge representation formalism to represent the semantic information. Using this formalism, we further enhance the IVGE description with respect to both agents’ and environments’ characteristics.
3.3. Semantic abstraction

Two kinds of information can be stored in the description of an IVGE. Quantitative data are stored as numerical values which are generally used to depict geometric properties (like a path’s width of 2 m) or statistical values (like a density of 2.5 persons per square meter). Qualitative data are introduced as identifiers which can range from a word with a given semantics, called a label, to a reference to an external database or to a specific knowledge representation. Such semantic information can be used to qualify an area (like a road or a building) or to interpret a quantitative value (like a narrow passage or a crowded place). An advantage of interpreting quantitative data is reducing a potentially infinite set of inputs to a discrete set of values, which is particularly useful to condense information in successive abstraction levels for reasoning purposes. Further, the semantic information enhances the description of the IVGE, which in turn extends the agents’ knowledge about their environment. However, the integration of the semantic information raises the issue of its representation.

Therefore, we need a standard formalism that allows for precisely representing the semantic information, which qualifies space and is computationally tractable in order to be exploited by the spatial reasoning algorithms used by agents.

Several knowledge representation techniques can be used to structure semantic information and represent knowledge in general such as frames (12), rules (13) (also called If-Then rules), tagging (14), and semantic networks (10) originating from theories of human information processing. Since knowledge is used to achieve intelligent behavior, the fundamental goal of knowledge representation is to represent knowledge in a manner that facilitates inference (i.e. drawing conclusions) from knowledge. In order to select a knowledge representation (and a knowledge representation system to logically interpret sentences in order to derive inferences from them), we have to consider the expressivity of the knowledge representation. The more expressive a knowledge representation technique is, the easier (and more compact) we can describe and qualify geographic features which characterize IVGE. Various artificial languages and notations have been proposed to represent knowledge. They are typically based on logic and mathematics, and can be easily parsed for machine processing. However, Sowa’s CG (10) are widely considered an advanced standard logical notation for logic based on existential graphs proposed by Charles Sanders Peirce and on semantic networks.

Syntactically, a CG is a network of concept nodes linked by relation nodes. Concept nodes are represented by the notation (concept type: concept instance) and relation nodes by (relationship-name). A concept instance can be either a value, a set of values, or even a CG. The formalism can be represented in either graphical or character-based notations. In the graphical notation, concepts are represented by rectangles, relations by circles, and the links between concept nodes and relation nodes by arrows. The most abstract concept type is called the universal type (or simply universal) denoted by the symbol ∙.

A MAGS usually involves a large number of situated agents of different types (human, animal, static, mobile, etc.) performing various actions (moving, perceiving, etc.) in virtual geographic spaces of various extents. Using CGs greatly simplifies the representation of complex situated interactions occurring at different locations and involving various agents of different types. In order to create models for MAGS, we consider three fundamental abstract concepts: (1) agents; (2) actions; and (3) locations. Taking advantage of the abstraction capabilities of the CGs formalism through the concept type lattice (CTL): concept types are organized in a hierarchy according to levels of generality. However, this hierarchy is not a tree, since some concept types may have more than one immediate supertype [?] instead of representing different situated interactions of various agents in distinct locations, we are able to represent abstract actions performed by agent archetypes in abstract locations. However, we first need to specify and characterize each of the abstract concepts. The CTL enables specialization of each abstract concept to represent situated behaviors such as path planning for agents in space. Figure 7 presents the first level of the CTL refining the agent, action, and location concepts. Figure 8(a)–(c) presents the expansion of the CTL presented in Figure 7. Figure 8(a) illustrates some situated actions that can be performed by agents in the IVGE such as sailing for maritime vehicles, rolling for terrestrial vehicles, walking for humans, and accessing for humans to enter or exit buildings (we assume that buildings are not navigable.
locations from the perspective of outdoor navigation). Figure 8(b) depicts how the location concept may be specialized into Navigable and Not Navigable concepts. The Navigable concept may also be specialized into Terrestrial Vehicle Navigable, Pedestrian Navigable, Marine Vehicle Navigable, and Bike Navigable which are dedicated navigable areas with respect to agent archetypes and environmental characteristics as specified by the elementary semantics. Figure 8(c) illustrates a few agent archetypes that are relevant to our geo-simulation including pedestrians, cars, trucks, and bikes.

In order to show how powerful such a representation may be, let us consider the following example. We want to build a MAGS simulating the navigation of three human agents (a man, a woman, and a child), two bike riders (a man and a woman), and three vehicles (a car, a bus, and a boat) in a coastal city. The navigation behaviors of these different agent archetypes must respect the following constraints (or rules): (1) pedestrian agents can only move on sidewalks, on pedestrian streets, and eventually on crosswalks if needed; (2) vehicles can move on roads and highways; (3) boats sail on the river and stop
at the harbor port; and (4) bikes move on bikeways, roads, and streets but not on pedestrian streets. Using standard programming languages, it might be difficult to represent or develop the functions related to such simple navigation rules which take into account both the agents’ and the locations’ characteristics. However, the representation of these navigation rules becomes an easy task when using CGs and our defined CTL. Here are their expressions in CGs:

\[ \text{[PEDESTRIAN:} \text{p}] \leftarrow (\text{agtnt}) \rightarrow [\text{WALK:} \text{w1}] \leftarrow (\text{loc}) \rightarrow [\text{PEDESTRIAN NAVIGABLE:} \text{pn}] \]

\[ \text{[VEHICLE:} \text{v}] \leftarrow (\text{agtnt}) \rightarrow [\text{ROLL:} \text{r1}] \leftarrow (\text{loc}) \rightarrow [\text{TERRESTRIAL NAVIGABLE:tn}] \]

The arrows indicate the expected direction for reading the graph. For instance, the first example may be read: an agent \( p \) which is a “pedestrian” walks on a location \( pn \) which is “pedestrian navigable”. Since this expression involves the concepts pedestrian, walk and pedestrian navigable, this rule remains valid for every subtype of these concepts. Therefore, thanks to CGs and the CTL, there is no need to specify the navigation rules for men, women, and children if they act as pedestrians in locations such as pedestrian streets, sidewalks, or crosswalk. Indeed, these agent archetypes are subtypes of the pedestrian concept and pedestrian streets, sidewalks, and crosswalks are subtypes of the pedestrian navigable concept. To conclude, CGs offer a powerful formalism to easily describe different concepts involved in MAGS including agents, actions, and environments.

4. Discussion

Thomas and Donikian proposed an IHT (7) graph representing a part of the city of Renne (France) for human behavior animation purposes. This graph is composed of three layers: (1) the basic topological layer which contains real urban objects modeled as simple spaces such as buildings and road sections; (2) the compositesSpace layer which is composed of simple spaces or composite spaces of lesser importance; and (3) the local area layer which is the highest level of the IHT-graph and which is composed of composite spaces. This hierarchical urban model allows manual abstraction of buildings into blocks and road-sections and crossings into roads. The abstraction process is done by the user which constrains and considerably limits its application to real world large-scale and complex geographic environments. Thomas’s approach relies on a predefined decomposition of the virtual environment which is dedicated to urban environments. This decomposition is application-dependent (urban environments) and does not take into account the topologic and the geometric characteristics of the environment.

In contrast with the Thomas and Donikian (8) and Lamarche and Donikian (7) approaches, our abstraction technique optimizes the representation of the geographic environment while taking into account the geometric, topologic, and semantic characteristics of the geographic environment. This abstraction approach relies on an exact space decomposition technique (CDT) in order to preserve the geometric and topologic characteristics of the geographic environment rather than on predefined space decomposition. It also integrates semantic information associated with GIS data in order to enrich the description of the IVGE.

Embedding the information directly in the environment allows the support of agents’ spatial reasoning capabilities. However, the preparation of the fully augmented geometric model is very time consuming and difficult due to the sheer amount of data. For example, a typical model of a city quarter as used by Farenc can contain several thousands of primitives of many types (such as polygons modeling sidewalk pieces, benches, trees, bus stops, etc.). Moreover, Farenc built the urban environment using data provided by computer assisted graphic design systems since the purpose of the simulation is computer animation. However, when building VGEs representing large-scale and complex geographic environments based on reliable GIS data, Farenc’s approach cannot be used since it is dedicated to exclusively represent urban environments. Therefore, manual hierarchical space partitioning as proposed by Farenc is not feasible when dealing with geometrically complex environments. Moreover, the data structure of the urban environment’s description as proposed by Farenc needs to be enhanced in order to manage a large amount of geometric and topologic data. Finally, the hierarchical structure should be built using the geographic environment’s characteristics rather than being defined a priori as Farenc proposed. The work done toward representation of semantic information in virtual environments has been mostly carried out at a geometric level (15). Virtual environments are usually created as computer graphics applications, with minimal consideration given to the semantic information (16). In addition, semantic information is used in an ad hoc way without any standard formalism. There is a gap between geometry and semantic information in current VGE models. Since we believe that semantic information integration into a VGE’s description is by nature a knowledge representation problem, suitable and standard knowledge representation formalism has been proposed to integrate semantic information in the VGE’s description.

5. Conclusion and future work

In this paper, we introduced our IVGE model which automatically builds semantically enriched and geometrically accurate description of IVGEs. We also proposed an abstraction approach of the IVGE’s description in order to support large-scale and complex geographic environments. First, we described a geometric abstraction process which enriches the IVGE description with terrain semantics. Moreover, the geometric abstraction process
helps to detect and filter elevation anomalies and qualifies the terrain shape, specifically slope. Second, we detailed a topologic abstraction that builds a hierarchical topologic graph in order to deal with large-scale VGEs. This hierarchical structure reduces the size of the topological graph representing the IVGE. Third, we showed how the semantic abstraction process enhances the hierarchical topological graph using the CTL in order to build different views of the IVGE. We are currently working on the leverage of our enhanced IVGE model to support hierarchical path planning algorithms which take into account both the abstracted description of the IVGE and the agent type’s characteristics.

Acknowledgements
The author thanks Dr Bernard Moulin, Laval University, Dr Phil Graniero, University of Windsor, and Dr Normand Bergeron, Institute for their advice during this research. Supported by the Canadian Network of Centers of Excellence in Geomatics (GEOIDE); the Natural Sciences and Engineering Research Council (NSERC) of Canada; and the Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT).

Notes on contributor
Mehdi Mekni received his BE (2000) degree from the department of information sciences at the University of Tunis (Tunisia) and his PhD (2010) and MSc (2006) degrees from the department of computer sciences at Laval University (Quebec City, Quebec, Canada). He is now a postdoctoral fellow at Institut national de la Recherche Scientifique (INRS). His main interests include multiagent systems, agent-based geosimulation, and application of AI techniques on modeling and simulation of software agents.

References
(1) Benenson, I.; Torrens, P. Geosimulation: Automata-Based Modeling of Urban Phenomena; Wiley: Chichester, 2004.
(2) Farenc, N.; Boulic, R.; Thalmann, D. An Informed Environment Dedicated to the Simulation of Virtual Humans in Urban Context. In Computer Graphics Forum (Eurographics ’99); Brunet, P.; Scopigno, R., Eds.; The Eurographics Association and Blackwell Publishers: Oxford, 1999.
(3) Shao, W.; Terzopoulos, D. Environmental Modeling for Autonomous Virtual Pedestrians: Digital Human Modeling for Design and Engineering Symposium: Iowa, 2005.
(4) Paris, S.; Donikian, S.; Bonvalet, N. Environmental Abstraction and Path Planning Techniques for Realistic Crowd Simulation. Comput. Animat. Virtual Worlds. 2006, 17, 325–335.
(5) Fehler, M.; Klugl, F.; Neumann, M. Landscape Abstractions for Agent-Based Biodiversity Simulation. The Fourth International Joint Conference on Autonomous Agents & Multiagent Systems, Utrecht University, The Netherlands, 2005.
(6) Kallmann, M. Object Interaction in Real-Time Virtual Environments. Ecole Polytechnique Fdrale de Lausanne, 2001.
(7) Lamarche, F.; Donikian, S.; Crowds of Virtual Humans: A New Approach for Real Time Navigation in Complex and Structured Environments. Computer Graphics Forum, Eurographics ’04, Grenoble, France.
(8) Thomas, R.; Donikian, S.; A Model of Hierarchical Cognitive Map and Human Memory Designed for Reactive and Planned Navigation. Proceedings of the 4th International Space Syntax Symposium, Londres, 2003.
(9) Mekni, M.; Moulin, B.; Holonic Modelling of Large Scale Geographic Environments. HOLOMAS’09: 4th International Conference on Industrial Applications of Holonic and Multi-Agent Systems, Linz, Austria, 2007.
(10) Sowa, J. F. Knowledge Representation: Logical, Philosophical, and Computational Foundations. Course Technology, 1st edition, August 1999.
(11) Chen, S.; Wu, J. A Geometric Interpretation of Weighted Normal Vectors and its Improvements. Proceedings of the International Conference on Computer Graphics, Imaging and Vision: New Trends, Beijing, 2005.
(12) Minsky, M. Framework for Representing Knowledge. MIT AI Laboratory, 1974. http://hdl.handle.net/1721.1/6089.
(13) Lewis, J.; Skarek, P.; Varga, L. A Rule-based Consultant for Accelerator Beam Scheduling used in the Cernps Complex. ICACLEPCS’95: International Conference on Accelerator and Large Experimental Physics Control Systems, Chicago, IL, 1995.
(14) Varlan, L. Knowledge representation in the context of e-business applications. BRAND. Broad Research in Accounting, Negotiation, and Distribution, 1 (1), 1–4, September 2010.
(15) Martinez, A. Semantics for Virtual Humans. Ecole Polytechnique Fdrale de Lausanne, 2009.
(16) Alonso, M. Semantic Virtual Environments. Ecole Polytechnique Federale de Lausanne, Swiss, 2005.