

A Multi-Agent Geo-Simulation Model for Intelligent Sensor Web Spatial Coverage Management Using IVGE

Mehdi Mekni

Abstract—A Sensor Web (SW) consists of a large collection of small nodes providing collaborative and distributed sensing abilities in unpredictable environments. Nodes composing such a SW are characterized by resource restrictions, especially energy, processing power, and communication capacities. A sensor web can be thought of as a spatially and functionally distributed complex system evolving in and interacting with a geographic environment. So far, the majority of the currently deployed SWs has been mainly used for prototyping purposes. These SWs operate without considering a management scheme and do not take into account the geographic environment characteristics in which they are deployed. Multi-Agent Geo-Simulation (MAGS) is a recent modeling and simulation paradigm which provides a flexible approach that can be used to analyze complex systems such as SW in large-scale and complex georeferenced environments. In this paper, we propose to use a MAGS approach to support SW management. Moreover, we present Sensor-MAGS, a multi-agent geo-simulation system which manages sensor nodes using Informed Virtual Geographic Environments (IVGE). This system is applied in the context of a water resource monitoring project.

Keywords—Sensor Web, Multi-Agent Systems, Geo-Simulation, Virtual Geographic Environment, Sensor Web Management.

I. INTRODUCTION

RECENT advances in wireless communications and sensing technologies have enabled the development of low-cost, low-power, multi-functional sensor nodes that are small in size and which communicate over short distances [3]. Sensor Webs (SW) are distributed network systems composed of hundreds of such sensor nodes [12]. New capabilities such as micro-sensing and in-situ sensing as well as the wireless connection of these nodes open new possibilities for applications in various domains such as military, environment, health, home, space exploration, chemical processing and disaster relief [2]. The low per-node cost and the shrinking

size of microprocessors in addition to the enhancement of their computation capacities, while decreasing their energy consumption, will allow dense distribution of these wireless networks of sensors and actuators [25].

SW can be thought of as a macro-instrument concept that allows for the spatio-temporal understanding of phenomena which take place in geographic environments through the coordinated efforts of a large number of sensing nodes of different types [12]. However, once SW are deployed, the management of such complex systems is a real challenge because of their limited energy, communication, and processing capabilities [3]. In spite of the intensity of the research carried out in this field, SW are still in their infancy [2]. The majority of currently deployed sensor webs are mainly used for prototyping purposes [12]. In many cases, it is impractical to experiment on real sensor web systems for several reasons. First, a particular hardware platform, while theoretically possible, may not yet be manufactured because its fabrication may be constrained by technical or design limits [38]. Second, even if the hardware platform exists, it may be prohibitively expensive for experimentation [3]. For example, applications developed for research purposes may require hundreds or thousands of nodes in order to accurately monitor a natural phenomenon [2]. With current sensor nodes costing up to hundreds of dollars, evaluating such research could cost tens of thousands dollars [2]. Third, even if it is practical to evaluate research on the real hardware platform, it may not be practical to experiment in an appropriate environment [12]. An example of this are sensor webs which operate on glaciers, remote wildlife habitats, volcanoes, and other environments where in-situ sensing techniques are required and with which it is expensive or dangerous to experiment [1]. Therefore, there is a need for an efficient modeling paradigm which allows us to analyze SW using actors representing sensor nodes evolving in and interacting with a representation of their geographic environment.

One solution to the problem of SW analysis is Multi-Agent Geo-Simulation (MAGS). MAGS is a modeling and simulation paradigm which aims to study phenomena in a

Manuscript received August 6th, 2010.

Mehdi Mekni is a postdoctoral fellow at Sherbrooke University, Department of Computer Science Sherbrooke, QC, J1J 4E1, CANADA (phone: 001-418-262-7601; fax: 001-418-840-5105; e-mail: mehdi.mekni@fujitsu.ca.com).

variety of domains involving a large number of heterogeneous actors (implemented as software agents) evolving in, and interacting with, a Virtual representation of the Geographic Environment (VGE) [26, 28]. Most of the current SW simulation platforms that we analyzed and assessed lack an explicit representation of the geographic environment [27]. A SW is deployed in a spatial environment, and ignoring the characteristics of this environment would greatly decrease the quality of SW simulations. A critical step towards the development of an efficient SW simulation tool is the creation of a VGE, using appropriate representations of the geographic space and of the sensors evolving in it, in order to efficiently support the sensors' situated reasoning [24]. Since a geographic environment may be complex and of large scale, the creation of a VGE is difficult and needs large quantities of geometrical data originating from the environment characteristics (terrain elevation, location of objects and sensors, etc.) as well as semantic information that qualifies space (trees, buildings, rivers, etc.). In order to yield realistic SW simulations, 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 with semantic information that is associated with the geographic features. A number of challenges arise when creating such a semantically-enriched and geometrically-accurate representation of a VGE, among which we mention [33]: 1) automatically creating an accurate geometric representation of a 3D VGE; 2) automatically integrating the geometric representation with several types of semantic information; 3) making use of this representation in "situated reasoning" algorithms. Examples of such algorithms include sensor placement which aims to support sensors' deployment in order to optimize the communication between sensor nodes and with base stations, with respect to the environment's characteristics (obstacles, land cover, terrain, etc.).

In this paper, we present Sensor-MAGS, a Multi-Agent Geo-Simulation (MAGS) architecture for the management of sensor webs. While most works in the sensor network domain address issues such as networking, communication protocols, and energy efficiency which do not really depend on geographical constraints, we are interested in large sensor webs which are highly constrained by the geographical environment. Indeed, we use a MAGS approach with the purpose of analyzing potential interactions between sensors, as previous works do, but also with the geographic environment in which the sensor web is deployed. We thus aim at showing how a MAGS approach can solve complex problems such as intelligent management in large sensor webs.

Consider a water resource monitoring project scenario which uses sensor webs (Figure 1). This project is a distributed system that spans a large geographic area to monitor, model and forecast physical processes, such as environmental pollution and flooding. It includes several management goals and complex challenges. A deployed sensor node has several

sensors, communication hardware, and a Measurement Agent (MA) controlling the station's behavior. Sensors take their measurements and the MA forwards them to the data store or application user according to a predefined schedule which depends on the sensitivity of their positions. For example, in Figure 1, the sensor node MA2 may sleep most of the time and be triggered when water flow in its location reaches a critical level. According to the activity of the factory, the sensor node MA1 increases the frequency of its measurements in order to analyze the water quality in the soil which may be polluted by the factory's wastes.

When rain falls in abundance, the sensors MA10, MA4, MA11, and MA12 must increase the frequency of their measurements, assess the level of water in the big lake, and notify the authorities of any likely inundation. Due to flood risks in rough conditions and soil contamination, we need a sensor web that supervises the flow and levels of water in sensitive places (e.g. areas A, B, and C delimited by dashed circles in Figure 1) and adapts to changes on the basis of the collected data. This self-adaptation may be carried out by intelligent sensor nodes which collect, analyze, and share their data and coordinate their tasks in order to monitor water resources (Figure 2)). Current sensor web solutions are unable to anticipate the evolution of such kinds of situations and adjust the priorities of their processing. Moreover, sensor nodes must be placed with respect to the environment characteristics in order to ensure an optimized communication between sensor nodes and with base stations. To meet this end, we propose a multi-agent geo-simulation approach.

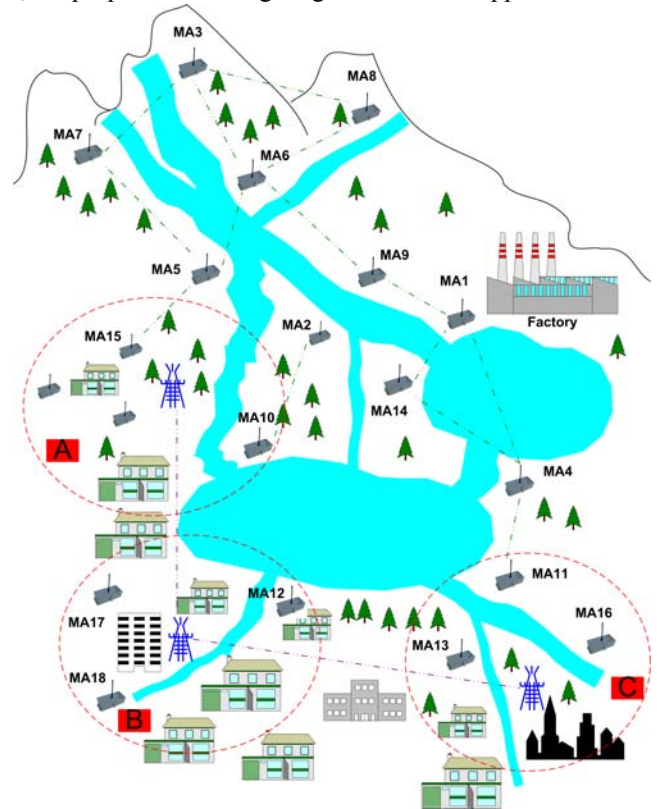


Fig.1 A water resource monitoring scenario. MA- (Measurement Agent) operating on a sensor node [25].

In the reminder of this paper, Section II presents fundamental notions related to the multi-agent geo-simulation approach. Next, Section III provides an overview of the sensor network management domain. Section IV provides an overview of our multi-agent geo-simulation methodology for sensor web management. Section V describes the Sensor-MAGS system's main architectural features. Afterwards, Section VI details the geometrically-accurate and semantically-enhanced virtual geographic environment in which sensor agents evolve. Next, Section VII points out the way we take advantage of Sensor-MAGS to support sensor placement in informed VGEs. We also highlight the advanced visualization and spatial analysis capabilities of Sensor-MAGS's viewer. Finally, in Sections VIII and IX we discuss our proposal and conclude with future works.

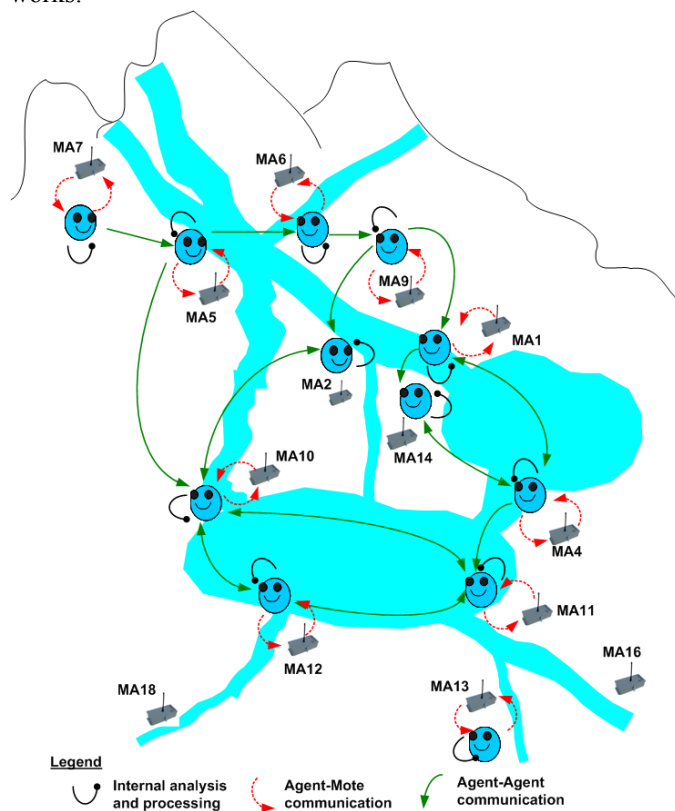


Fig.2 The virtual sensor web using Multi-Agent Systems [28].

II. MULTI-AGENT GEO-SIMULATION

A fundamental idea underlying our approach is to move the most intensive processing out of the Physical Sensor Web (PSW) into a parallel Virtual Sensor Web (VSW) operating on a base station or a remote server (Figure 2). The objective is to reproduce, in a realistic manner, the real world in a virtual environment. Indeed, in this virtual environment, which imposes no limits on data processing, energy consumption and communication capabilities, it is possible to create a system for the management of the physical sensor web. In order to faithfully mimic the physical sensor web deployed in the area of interest, we need to simulate, in a realistic way, the physical sensor nodes as well as the geographic environment where

they are located. We propose to use software agents for the virtual representation of the physical sensor web. An agent is a program with domain knowledge, goals and actions [41]. An agent can observe and sense its environment as well as affect it. Agents' capabilities may include (quasi-) autonomy, perception, reasoning, assessing, understanding, learning, goal processing, and goal-directed knowledge processing [40]. The reproduction of the geographic environment in which physical sensor nodes are deployed should be based on reliable data obtained from Geographic Information Systems (GIS). The concept of Multi-Agent Geo-Simulation (MAGS) evolves from such type of simulations involving software agents immersed in a virtual geographic environment.

MAGS has attracted a growing interest from researchers and practitioners to simulate various phenomena in a variety of domains including traffic simulation, crowd simulation, urban dynamics, and changes of land use and cover, to name a few [6]. Such approaches are used to study various phenomena (i.e. car traffic, crowd behaviors, etc.) involving a large number of simulated actors (implemented as software agents) evolving in, and interacting with, an explicit description of the geographic environment called Virtual Geographic Environment (VGE). MAGS is a useful approach to integrate the spatial dimension in models involving different kinds of interactions (economic, political, social, etc.) [13]. From this perspective, the Geographic Information System (GIS) plays an important role in the development of geo-simulation models. MAGS can be thought of as a coupling of two technologies: the Multi-Agent Systems (MAS) and the Geographic Information Systems [6]. Based on the MAS technology, the simulated entities are represented by software agents that can behave and make decisions autonomously. They can interact with other agents and with a virtual representation of the actual geographic environment. They may be reactive, proactive, stationary or mobile, social or cognitive [6]. These agents evolve and interact with their VGE.

An accurate VGE requires the use of reliable GIS data. GIS data are usually available in either raster or vector formats [32]. The raster format subdivides the space into regular square cells, each associated with an attribute related to the space. In contrast, the vector format describes geographic information using unconstrained geometric shapes, and generally associates one qualitative object with each shape. Such data are usually exploited by a VGE in two ways [17]: the approximate geometric subdivision and the exact geometric subdivision methods. The approximate geometric subdivision method is the direct mapping of the raster format, but it can also be applied to the vector format (Figure 3(c)). This discrete representation can be used to merge multiple semantic data [42], the locations where these data are stored being predefined by the grid cells. The main drawback of the grid method is related to a loss in spatial precision [4], making it difficult to accurately position any information which is not aligned with the subdivision. Another drawback arises when trying to precisely represent large environments using a grid: the number of cells tends to increase dramatically, which makes the environment description very costly. The grid-

based method is mainly used for overlay and animation purposes [37] because of the fast data access it provides.

The second method, called exact geometric subdivision, consists in subdividing the environment in convex cells using the vector format as an input. The convex cells can be generated by several algorithms, among which the most popular is the Constrained Delaunay Triangulation (CDT) [21]. The CDT produces triangles while keeping the original geometry segments which are named constraints (Figure 3(b)). The first advantage of the exact subdivision method is to preserve the geometry of the input data, allowing accurately manipulating and visualising the environment at different scales. Another advantage of this approach is that the number of produced cells only depends on the complexity of the input shapes, but not on the environment's size and scale as is the case with the grid method. The main drawback of this approach is the difficulty to merge multiple semantic data for overlapping shapes. Moreover, this method is generally used to represent planar environments because the CDT can only handle 2D geometries. This method tends to be used for micro-scale simulations centered on individuals where motion accuracy is essential [22].

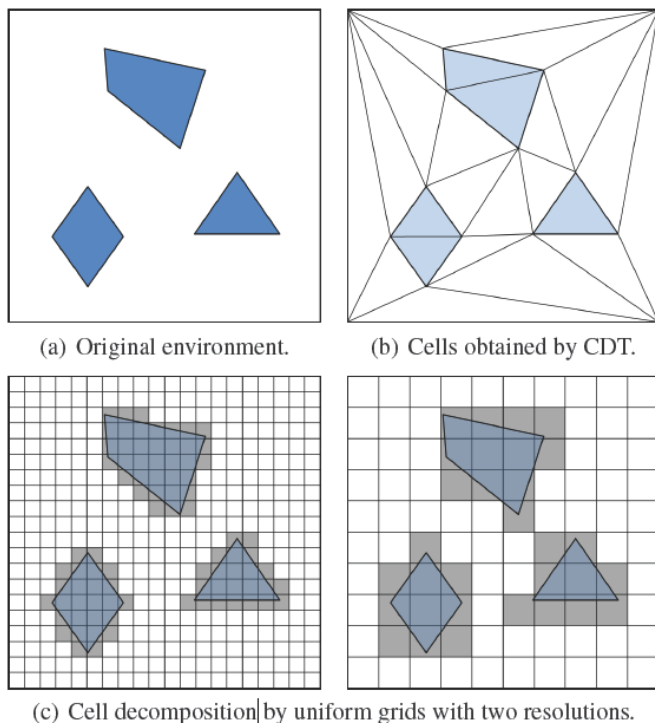


Fig.3 The two common cell decomposition techniques used to represent environments.

III. SENSOR NETWORK MANAGE

According to our literature review, architectures for the management of sensor webs involving the Multi-Agent Geo-Simulation (MAGS) paradigm do not exist. However, a few research projects have attempted to integrate the agent paradigm into sensor web architectures such as IrisNet [16], Abacus [5], Biswas and Phoha's architecture [7], and SWAP [29]. Most of these architectures identify the need for

distributed data collection and processing, and propose layered architectures to achieve this. In Abacus different agents in the processing layer detect and report alert conditions to a higher layer interacting with users [5]. IrisNet uses agents such as Sensor Agents (SA) and Sensor Organisers (SO) to collect and analyze data from sensors to answer specific classes of queries [16]. Biswas and Phoha's approach uses agents in the service layer to analyse data from sensors and transfer it to the application layer [7]. All these approaches deal with data collection by providing a distributed infrastructure for publishing, discovering and accessing sensor resources. They also address the challenge of data fusion, to some extent, and aim to provide end-users with the information they need. These approaches share a common objective through the use of the agent-paradigm which is the distribution of tasks. However, these applications do not take complete advantage of the multi-agent systems approach. Indeed, they use reactive agents which are efficient for alerting purposes, but are neither able to perform situated behaviors nor autonomous decision-making. On the one hand, situated behaviors include performing spatial reasoning and taking advantage of the virtual environment's description where sensor agents are located. On the other hand, autonomous decision-making includes managing sensor nodes in order to efficiently cover the area of interest while taking into account their limited capabilities as well as local spatial characteristics. We think that, in order to achieve intelligent and autonomous, it is essential to use a multi-agent geo-simulation approach in which agents are endowed with advanced capabilities such as perception, navigation, memory, and knowledge management. As the abovementioned architectures do not address the more challenging sensor web management issues, we propose the Sensor-MAGS System, a multi-agent geo-simulation platform for the intelligent management of sensor webs.

IV. SENSOR-MAGS: AN INTELLIGENT SENSOR WEB MANAGEMENT SYSTEM

In this section, we first introduce the concept of sensor web management. Next, we present the approach we propose to support the intelligent management of sensor webs. Finally, we describe the Sensor-MAGS system's main architectural features.

A. Sensor Web Management

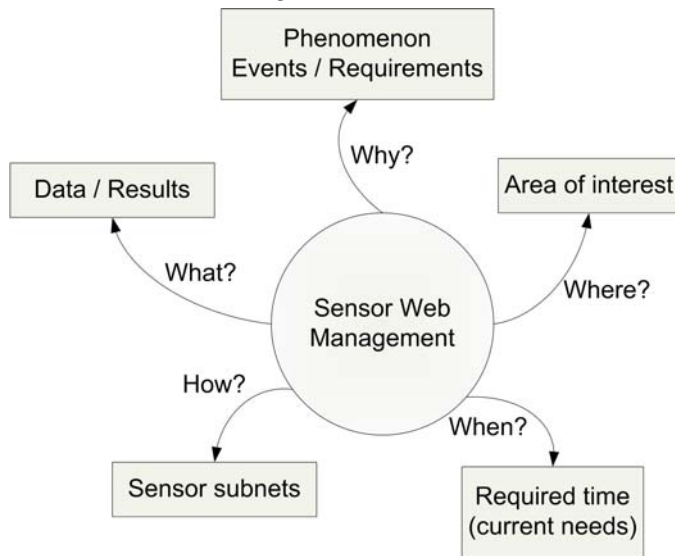


Fig.4 Sensor web management objectives [20].

Sensor web management, as defined in [30], is “to manage, co-ordinate and integrate sensor nodes to accomplish specific and often dynamic sensing mission objectives”. The term “manage” means the control over the sensors. The term “co-ordinate” outlines the load balancing of tasks assigned to sensors with respect to their limited capabilities. Finally, the term “integrate” highlights the organization of the sensor into a coherent and structured architecture.

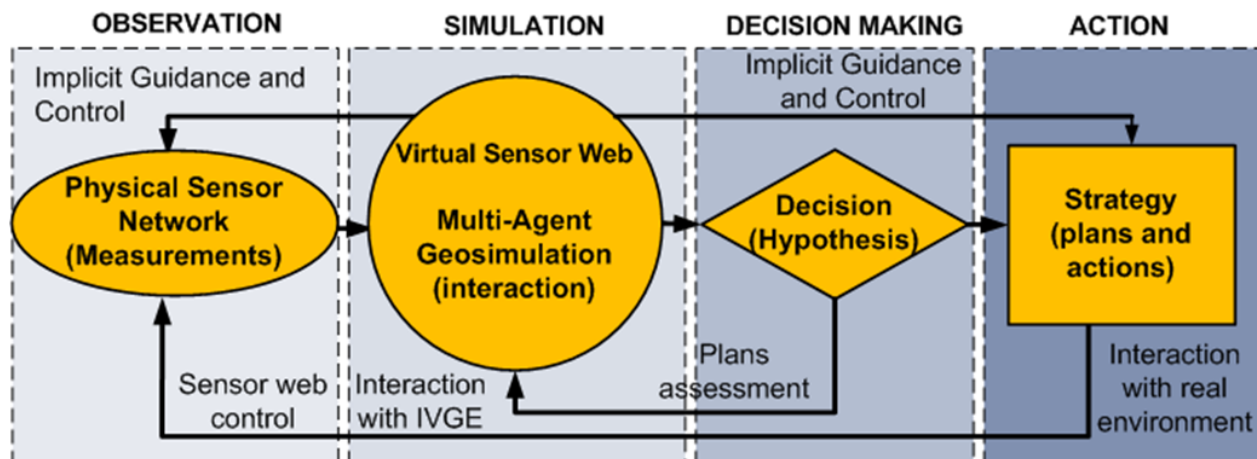


Fig.5 Our MAGS approach for the management of sensor webs based on the Observe-Orient-Decide-Act (OODA) loop.

Interpreting the collected data as well as managing the sensing resources has historically been done manually. However, this task has grown difficult, if not impossible, due to the complex functionalities of modern sensory systems. Current initiatives seek to automate the process of data interpretation and sensor management. Data and information fusion is a concept whereby data from various sources are integrated and interpreted to allow accurate inferences about the environment. The idea is that fusing data from numerous sources provides a better picture of the environment than inferring from sensor measurements independently. The concept of sensor web management is deeply related to the data fusion research field. Its aim is to utilize the sensing resources in a manner that synergistically improves the process of data acquisition and ultimately enhances the perception and the comprehension of the situation of interest. Sensor web management also concerns the control and coordination of limited sensing resources in order to collect the most complete and accurate data from a dynamic scene. As such, sensor management is a key enabler of environment

monitoring applications. Sensor web management may be thought of as closing the loop on the fusion process, whereby sensing resources are actively managed to improve the quality of inferences made about the environment. Therefore, there is a need for an architecture which is able to aggregate sensor nodes into a manageable organization. This architecture must take into account the characteristics of the geographic environment where the sensor web is deployed. Ultimately, this architecture must provide a context which governs the sensors' interactions while achieving their goals including sensing, resource optimization, data dissemination, information fusion, and decision making.

A. Our Approach

We propose to use a computer simulation approach in order to support sensor web management. In this context, the simulation is actually a copy of the real system on which experiments (or scenarios) can be run to evaluate various strategies. However, creating a computer simulation in order to support sensor web management raises a number of

challenges among which we mention: 1) collecting observations and measurements performed by physical sensor nodes deployed in the field; 2) simulation of the interactions occurring between sensor nodes and with their geographic environment; 3) an overall understanding of the situation in the field including the observed phenomena and the sensor web in order to monitor its evolution.; 4) development of strategies to adequately intervene and better control the sensor web; and 5) comparing alternative intervention scenarios and anticipating the consequences of these interventions. Figure 5 illustrates the Observe-Orient-Decide-Act (OODA) methodology of the proposed solution [39]. As depicted, the physical sensor web corresponds to the Observation phase. These observations refer to measurements obtained in the field (sensed data). The next phase, Simulation, corresponds to our multiagent geo-simulation infrastructure. This phase combines the sensed data of the Observation phase with the spatial information retrieved from GIS in order to carry out simulations. The Decision-Making phase analyses the simulations' results and proposes plans and strategies aim to improve the sensor web performance. It also may assess the efficiency of plans and strategies involving the simulation phase. Finally, the Action phase corresponds to assigning scheduled tasks to the physical sensor nodes.

B. System Architecture

In this section, we introduce a layered architecture to support sensor web management. This solution is based on a multiagent system and uses the geo-simulation approach in order to carry out simulations in virtual geographic environments. Such simulations aim to evaluate various management strategies for sensor webs. Since the management of large-scale sensor webs remains a critical task which should be controlled by human administrators, our solution proposes different strategies in order to support decision makers. Sensor-MAGS can be thought of as a layered architecture as illustrated in Figure 6 (rectangles on the right hand side). This architecture is inspired by the layered simulation model proposed in [34, 35]. In what follows, we briefly present these layers.

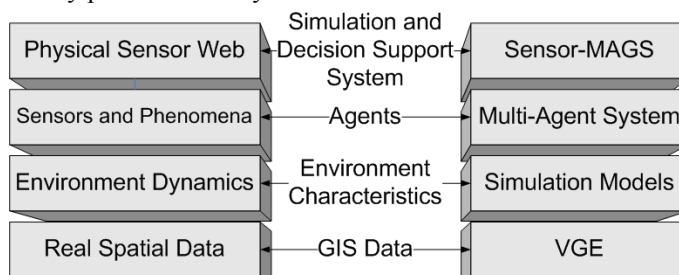


Fig.6 The layered architecture of our multi-agent geo-simulation platform for the management of sensor webs.

The first layer is the software platform which is in charge of reproducing the real geographic world in the Virtual Geographic Environment (VGE). A GIS is essential to reproduce real spatial data in the simulation environment. In

addition, human users need a visual tool to monitor the geographic environment. It is thus necessary to transform GIS data into a simulation software platform which is visual for human users and navigable for software agents. Therefore, the Sensor-MAGS system integrates geographic information such as elevation, ground cover, land use, etc. This information is relevant to the simulation of sensors' activities in the field.

The second layer is responsible for modeling the dynamic factors influencing the real world. External factors such as atmospheric phenomena and weather conditions make the environment more dynamic and unpredictable, which requires data that are not provided by a GIS. Complementary models (such as water contamination, forest fire) need to be used to simulate this dynamism. Data used by these models should first be captured from the real world and then be continuously updated and transferred to the VGE. The model can thus provide a reliable and progressive simulation of the environment. For example, in the context of a water resource monitoring project, there is a need to simulate dynamic data related to flood phenomena.

The third layer corresponds to the multi-agent layer. It represents actors which perform actions in the world. There is a need for software agents situated in the terrain. Concretely, these could be sensors or electronic devices which sense their neighborhood and in which agents are embedded. We also need agents in the VGE. Agents within the real world may then communicate with agents within the VGE, which guarantees a better coherency between data collected from both the real world and the VGE. Each actor should have a software agent as a representative within the VGE. In our example, an actor may be a sensor node or a group of nodes. Such an agent interacts with the physical sensor nodes via an interface and communicates with its representative in the VGE via remote messages. Before they can act (navigate, perceive, etc.) within the VGE, software agents need to be coherently linked to real world actors.

The fourth layer represents the functionalities which are domain-specific. Actually, the three previous layers provide a foundation for applications that support decision makers in understanding the situation in the field, monitoring its evolution, developing and comparing different strategies to adequately intervene. For our example, the goal of the simulation of the sensor web is to manage and control its activities and to propose recommended actions aiming at achieving efficient in-situ sensing while optimizing resources.

V. SIMULATION PLATFORM

In Figure 7, we present an overview of our multi-agent geo-simulation platform which is fed by two data flows. On the one hand, the simulation system gets data collected by the physical sensor web (right hand side of Figure 7), including data related to the observed phenomenon in the area of interest (measurements) and control data related to the status of the sensor nodes (level of energy, position, etc.). On the other hand, the application's administrators provide the simulation

system with information (left hand side of Figure 7) describing sensor web management scenarios.

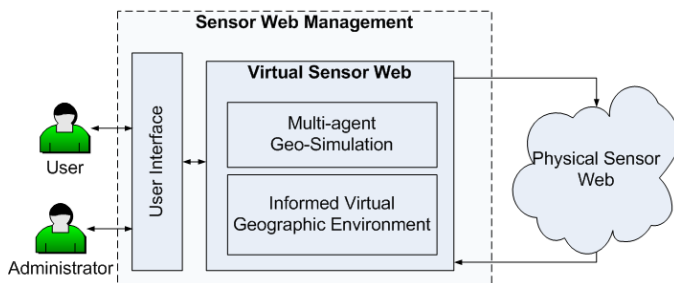


Fig.7 Conceptual architecture of our sensor web management platform.

Basically, a sensor web management scenario is a plan or a strategy proposed by the application's administrators whose objective is to respond to a critical situation in the field or to assess an intervention plan given a sensor web's characteristics. However, before applying the plan or the strategy, the application's administrators need to assess its adequacy and effectiveness. The sensor webs that we aim to manage are composed of a large number of heterogeneous sensor nodes of different types and with different capabilities, spatially distributed in geographic environments of various extents. It is rather complex or even infeasible to analytically model such a sensor web and it usually leads to an oversimplified analysis in which people have limited confidence. Indeed, most works only consider homogeneous sensor webs (sensors with identical capabilities) evolving in a homogeneous and static geographic environment.

Such assumptions considerably reduce the capacity to handle large-scale and complex real world geographic environments. Besides, the effective deployment of sensor webs for prototyping purposes implies a huge effort [36]. In order to reduce the need for a real physical sensor web in early design stages, we propose to model the physical sensor web using a sensor web simulator. Thus, the conceptual architecture is modified as shown in Figure 8. We propose to use OMNET++ [31] in order to simulate the physical sensor web. We assessed the capabilities of Omnet++ in [27] and find it suitable to the requirements of the water resource monitoring project.

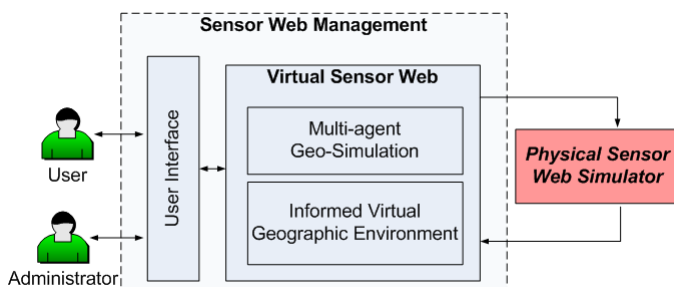


Fig.8 Conceptual Architecture of the Sensor Web Management Platform Based on a Sensor Web Simulator.

Sensor webs are deployed for data acquisition purposes within a region of interest. Therefore, the spatial dimension is a key part of any sensor web management system. In order to

provide sensor nodes with spatial perception, navigation, and reasoning capabilities we propose to augment their knowledge using a virtual geographic environment. A virtual geographic environment is an explicit accurate spatial representation of the real world in which agents representing sensor nodes evolve and with which they interact. Such a virtual geographic environment is built using reliable GIS data. It is decomposed into cells which include some semantic information in order to enhance data extractions. Such cells are abstracted using graph theory techniques in order to accelerate data access. As a result, the virtual geographic environment may encompass levels that are of various conceptual abstractions. Thus, the spatial environment can be analyzed from various perspectives according to the proposed level of abstraction [25].

VI. FROM GIS DATA TO IVGE

In this section, we present our automated approach to compute the IVGE data directly from vector GIS data. Figure 9) depicts the four stages which compose our approach: Input data selection, spatial decomposition, maps unification, and finally the informed graph generation. Moreover, in subsection VI-E, we propose a topologic abstraction technique which aims at enhancing the IVGE description when dealing with large scale and complex geographic environments.

A. Input data selection

The first step of our approach is the only one requiring human intervention. It consists in selecting the different vector data sets which are used to build the IVGE. The only restriction concerning these data sets is that they must respect the same scale. The input data can be organized into two categories. First, elevation layers containing the geographical marks that indicate absolute terrain elevations. We only consider 2:5D IVGE in which a given coordinate cannot have two different elevations, making it impossible to represent tunnels for example. This defined, multiple elevation layers can then be specified, and the model merges them automatically. 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 and buildings. The limits can overlap between two layers, and our model can merge the information.

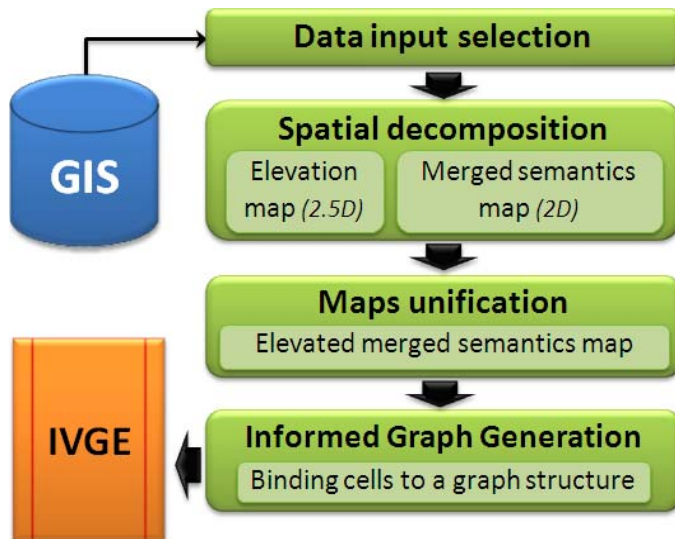


Fig.9 The four stages to obtain an IVGE from GIS data. All the stages are automatic but the first.

B. Spatial Decomposition

The second step consists of obtaining an exact spatial decomposition of the input data into cells. This process is entirely automatic, using Delaunay's triangulation, and can be divided into two parts in relation to the previous phase. First, an elevation map is computed, corresponding to the triangulation of the elevation layers. All the elevation points of the layers are injected into a 2D triangulation, the elevation being considered as an additional datum. This process produces an environment subdivision composed of connected triangles (Figure 11(a)). Such a subdivision provides information about coplanar areas: the elevation of any point inside a triangle can be deduced thanks to the elevation of the three original points. 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 adding additional attributes. The obtained map is then a constrained triangulation merging all input semantics (Figure 11(b)): each constraint represents as many semantics as the number of input layers containing it.

C. Maps unification

The third step to obtain our IVGE data consists of unifying the two maps previously obtained. This phase can be depicted as the mapping of the 2D merged semantic map (Figure 11(b)) onto the 2.5D elevation map (Figure 11(a)) in order to obtain the final 2.5D elevated merged semantics map (Figure 11(c)). First, a 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 in the merged semantics triangulation, creating new triangles. This first process can be dropped if the elevation precision is not important. Then, a second process elevates the merged semantics map. The elevation of each merged semantics point P is computed by retrieving the corresponding triangle T inside the elevation map, i.e. the triangle whose 2D

projection contains the coordinates of P . Once T is obtained, the elevation is simply computed by projecting P on the plane defined by T using the Z axis. When P is outside the convex hull of the elevation map, then no triangle can be found and the elevation cannot be directly deduced. In this case, we use the average height of the points of the convex hull which are visible from P .

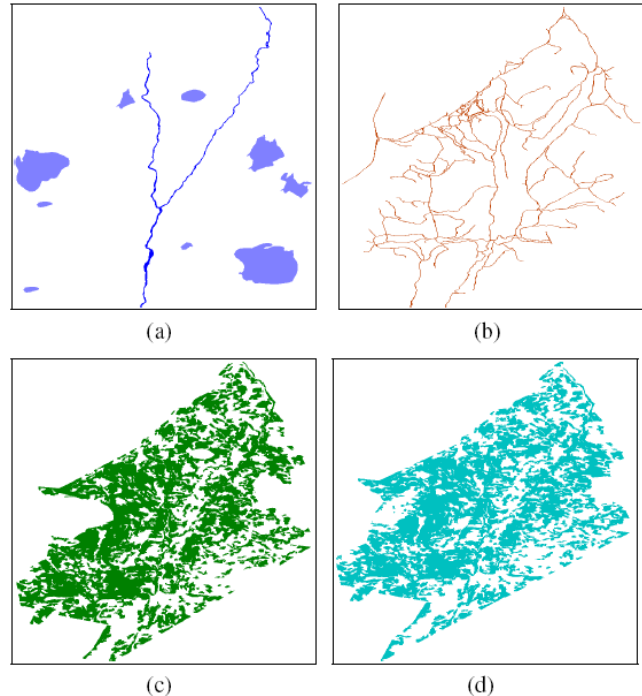


Fig.10 Various geometric and semantic layers related to the Montmorency experimental forest, St.Lawrence Region, Quebec, Canada: (a) water resources including rivers, water streams, and lakes; (b) road network; (c) and (d) two types of vegetation characterizing the landcover

D. Informed graph generation

The resulting unified map now contains all the semantic information of the input layers, along with the elevation information. This map can be used as a topological graph in which each node corresponds to the map's triangles and each arc to the adjacency relations between these triangles. Then, common graph algorithms can be applied to this topological graph, especially graph traversal ones. One of these algorithms retrieves the node, and so the triangle, corresponding to given 2D coordinates. Once this node is obtained, it is possible to extract the data corresponding to the position, such as the elevation, and the semantic information. Many other algorithms can be applied, such as path planning and graph abstraction, but they are out of the scope of this paper and will not be detailed here.

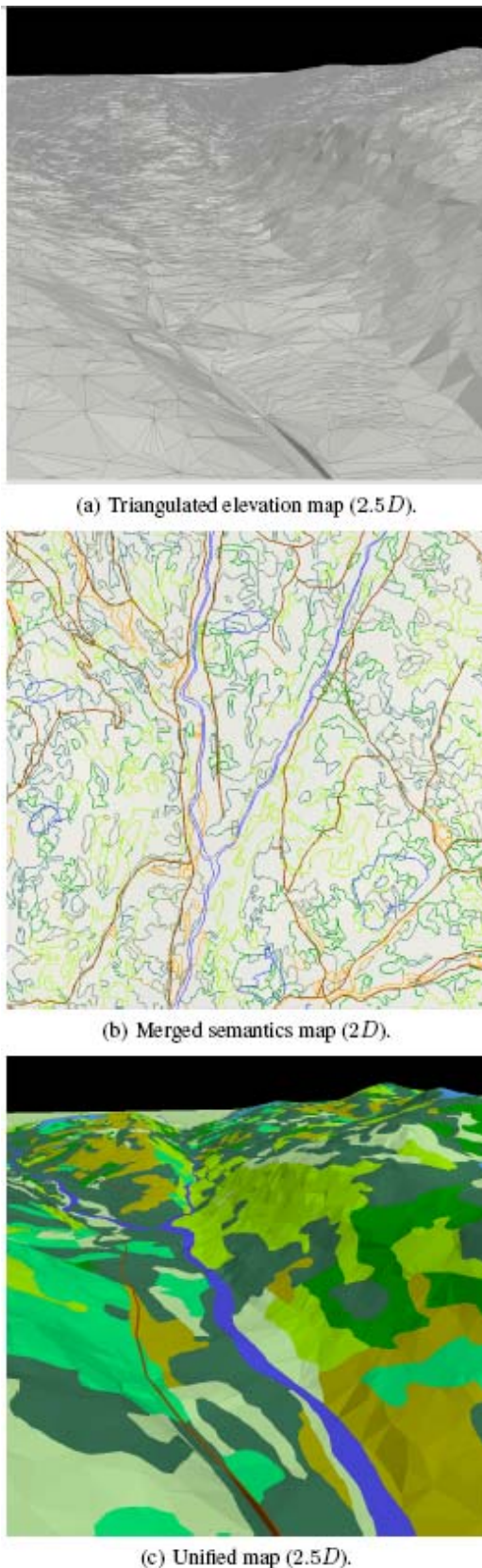


Fig.11 The two processed maps (a, b) and the unified map (c). The semantic colours are the same as in figure 10.

A. Topologic Abstraction

When dealing with large scale and complex geographic environments such as the geographic environment where we

will deploy the sensor web in the context of a water resource monitoring project, the topologic graph becomes very large. The size of a topologic graph has a direct impact on the computation time of sensors' spatial reasoning processes. In order to optimize such a computation time, we need to reduce the size of the topologic graph representing the IVGE. The aim of the topologic abstraction is to provide a compact representation of the topologic graph suitable for situated reasoning of sensor nodes. To this end, the topologic abstraction process extends the topologic graph with new layers. In each layer (except for the initial layer which is called level 0), a node corresponds to a group of nodes of the immediate lower level (Figure 12). The topologic 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 Hierarchical Topologic Graph in which cells are fused in groups and edges are abstracted in boundaries (Figure 12).

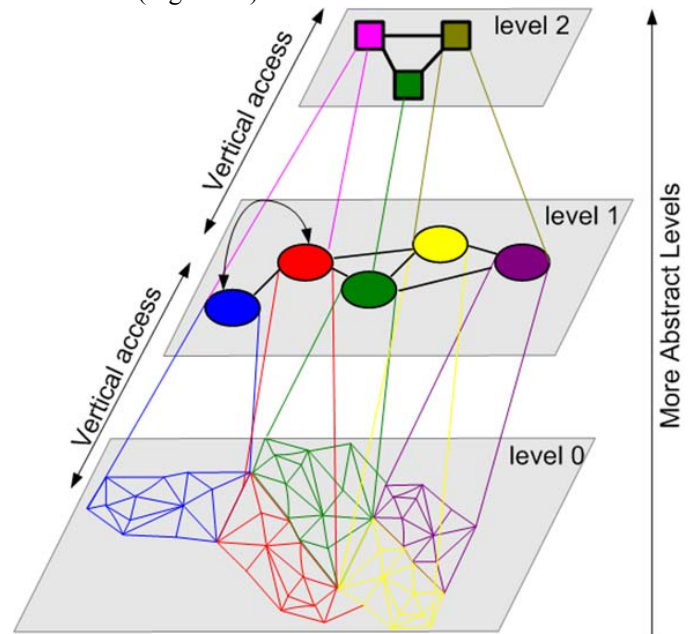


Fig.12 The topologic graph extraction from space decomposition and extension into different levels using the topologic abstraction.

To do so, convex hulls are computed for every node of the topologic 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 topologic abstraction finally performs groupings of a set of connected nodes if and only if the group ratio is close to one. Let C be the convexity rate and $CH(G)$ be the convex hull of the polygon corresponding to the group G . C is computed as follows:

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

The convex property of groups needs to be preserved after the topologic 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 13 illustrates the topologic abstraction process and the way it reduces the number of cells representing the environments. Figure 13(a) depicts the initial exact spatial decomposition of a geographic environment which yields 63 triangular cells. Figure 13(b) presents 2 convex polygons generated by the topologic abstraction algorithm. The optimization rate of the number of cells representing the environment is around 55%. To conclude, we proposed in this section an automated approach for the generation of geometrically-accurate and semantically-enhanced IVGEs. We also proposed an approach which aims to optimize the structure of the IVGE description when dealing with 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 the topologic 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. The Montmorency experimental forest, St. Lawrence Region, Quebec, Canada, in which we envision to deploy the physical sensor web in the context of the water resource monitoring project is characterized by its large scale and by the complexity of the geographic features that it encompasses (vegetation covers of various types and extents, roads, rivers, lakes). Indeed, as we will detail in Section VII, this geographic region extends over an area of almost 87km². The IVGE generation as well as the topologic abstraction process constitute a fundamental step to develop a system that will help sensor web's administrators to visualize, localize and analyze the sensors' placement with

respect to the characteristics of this large-scale geographic environment.

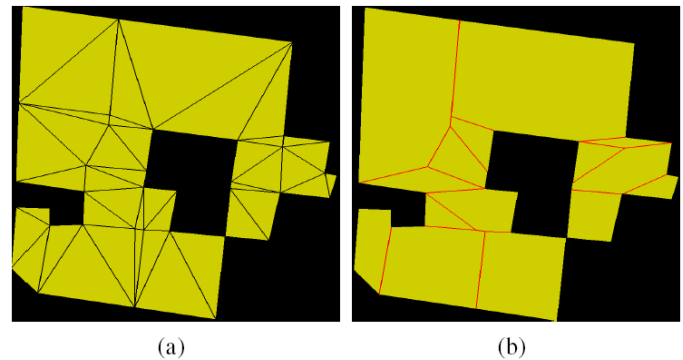


Fig. 13 Illustration of the topologic abstraction process with a strict convex property ($C(G) = 1$); (a) the exact space decomposition using CDT techniques (63 triangular cells); (b) the topologic abstraction (28 convex polygons).

VII. RESULTS

In this section, we present some implementation characteristics of our Sensor-MAGS platform. First, we introduce Sensor-MAGS Viewer software that we developed in order to automatically generate geometrically-accurate and semantically-enhanced virtual geographic environments and to inspect the environment partitioning in 2D and 3D. Next, we demonstrate how we take advantage of the semantically-enriched description of the IVGE in order to support sensor placement, taking into account the environmental characteristics.

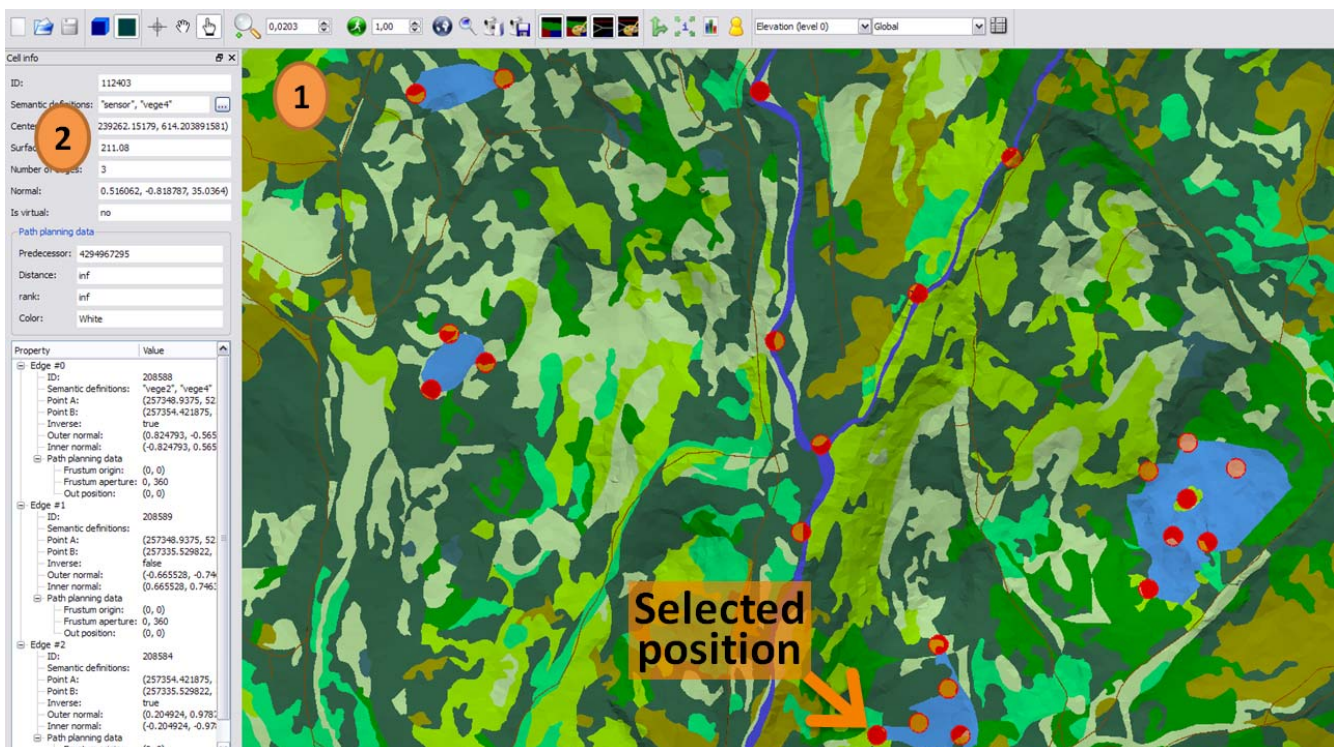


Fig-14 2D-IVGE visualisation of the GIS data. (1) unified map, (2) selected position's information pointing out a sensor coverage field and including geometric and semantic data. Circles represent sensors deployed by the administrators of Sensor-MAGS for water resource monitoring purposes.

A. Sensor-MAGS viewer

The proposed environment extraction method is used to create an accurate semantically-enhanced virtual geographic environment (IVGE) and provides the advantages of the semantic merging of grids along with the accuracy of vector data layers. Thanks to the automatic extraction method that we propose, our system handles the IVGE construction directly from a specified set of input vector GIS files. This set of GIS data files accurately describes the geographic areas of the Montmorency experimental forest, in which we envision to deploy the sensor web in the context of the water resource monitoring project. The performance of the extraction process is very good, processing such as area described with one elevation map and ten semantic layers, in less than five seconds on a standard computer (Intel Core 2 Duo processor 2.13Ghz, 1GB RAM). The obtained unified map approximately contains 478; 000 triangles covering an area of 87km². The time needed to obtain the triangle corresponding to a given coordinate is negligible (less than 10⁻⁴seconds). Moreover, the geometric abstraction produces approximately 119; 000 groups of cells in 2.8 seconds. Regarding the topologic abstraction, we tested our model using two topologic abstraction levels. The first level fully respects the convexity (i.e. convex ratio = 1) and produces around 332; 000 convex groups of cells, which corresponds to a reduction of almost 30:5%. The second level performs groupings of adjacent convex groups if and only if the convex ratio is better than 0:85 (i.e. convex ratio \leq 0:85). The second level yields approximately 149; 000 cells which represent 68:8% of the total number of convex cells produced by the spatial decomposition. The Sensor-MAGS Viewer tool provides two visualization modes for the computed data. First, a 3D view (as shown in Figure 11(c)) allows the sensor web's administrators to freely navigate in the virtual environment in order to select the appropriate location for sensor placement. Second, we propose an upper view with orthogonal projection to represent the GIS data as a standard map. In this view the user can scroll and zoom the map (1 in Figure 14), and accurately view any portion of the environment at any scale. Additionally, one can select a position in the environment in order to retrieve the corresponding data (2 in Figure 14), such as the underlying geometric information, the corresponding height and the associated semantics, and the attributes of the sensor node which is deployed at this location.

B. Sensor placement in the IVGE

In the context of our water resource monitoring project, sensor placement must take into account the nature of the terrain. For example, for radio signal propagation we must consider several features including the landcover and obstacles such as trees and foliage in the line of sight, as well

as uneven surfaces and elevations for hilly terrains. Indeed, the radio signal may be attenuated or degraded (called signal fade) due to absorption or scattering of the signal energy by natural or human-made obstructions along the line-of-sight path. The radio waves may also be reflected or refracted off nearby obstructions (i.e. multi-pathing), creating destructive interference and difficult signal processing. Strong signal fade can cause the effective transmission range to be less than the calculated radio horizon. Radio signal fading and interference interactions are highly variable and site-dependent, and are therefore very difficult to fully predict and avoid in anticipation. In the IVGE model, the geographic environment where the sensor web will be deployed is accurately described with respect to terrain and landscape features. This IVGE is populated with sensor agents (software agents representing physical sensors) which have the capability to interact with their virtual geographic environment. The sensor web's administrators can easily add, remove, and place sensors in the IVGE as shown in Figure 11(c). In contrast with physical sensor nodes, sensor agents have an enhanced knowledge about their environment's characteristics. Hence, with respect to the environment's characteristics including elevation, vegetation density and foliage, and location of water resources, each sensor agent computes the radio signal propagation and attenuation. We use the propagation model presented in [15]. Indeed, the environmental conditions (landcover, vegetation, terrain) affect electromagnetic waves propagation, reducing its power. When a transmitter Tx is placed at a distance d from a radio receiver Rx, the later will sense the electromagnetic power that is a function of the distance and the absorption factor (a) of the medium [14, 15]. Figure 15 depicts the quality of sensors' communication with respect to the characteristics of the Montmorency experimental forest. Three communication patterns are considered for sensor agents: non constrained communication, partially constrained communication, and highly constrained communication. Non-constrained sensors' communications are observed in flat areas, rivers surfaces, and terrain with very low vegetation (heights below 20cm, e.g. grass or branches lying on the earth). Partially constrained sensors' communications are observed in slopping terrains, and landcover with bushes (heights below 80cm). Finally, highly constrained sensors' communications are observed in steep slopes and areas where vegetation is high (heights over 80cm). Hence, the sensor web's administrators can freely add, move, or remove sensors from the Sensor-MAGS viewer. They can also analyze the sensor's communication quality, illustrated by specific colors as shown in Figure 15, given the environment's characteristics of the location where this sensor is situated. Finally, the sensor web's administrators can investigate and assess various sensor web deployment strategies using the Sensor-MAGS viewer before effectively

deploy the physical sensor web.

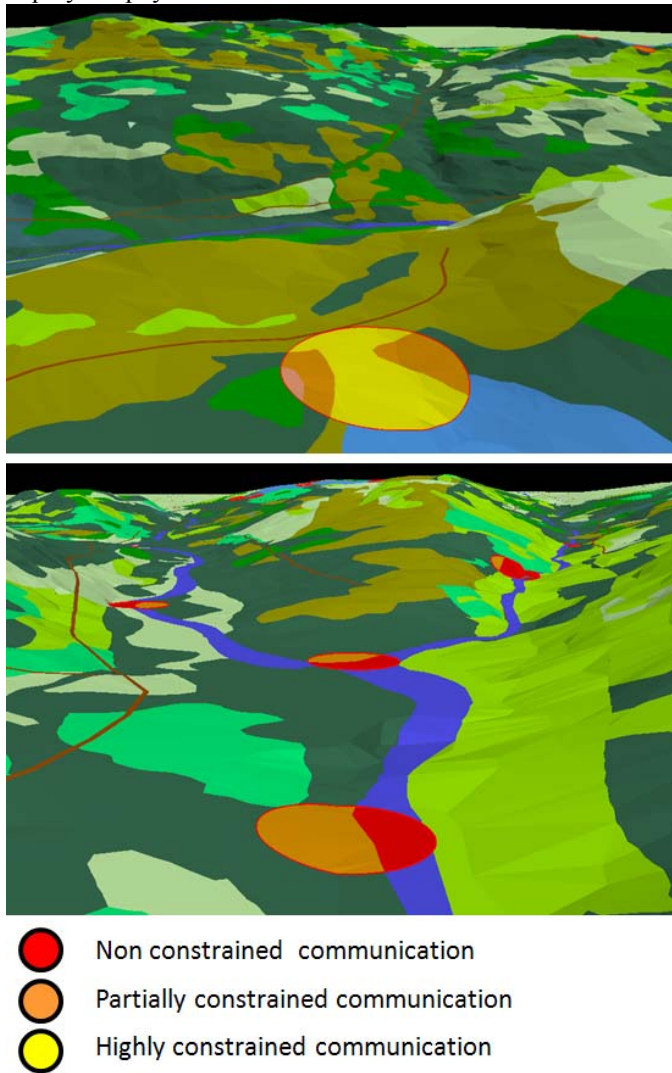


Fig.12 Illustration of the quality of sensors' communication with respect to the environment's characteristics including landcover and terrain conditions.

VIII. DISCUSSION

As sensors are used in greater numbers for field operation, efficient deployment strategies become increasingly important. Related work on self-deployment for mobile sensors using the notion of potential fields is presented in [19]. However, self-deployment does not provide a solution in the case of static sensors that need to be deployed in a specific configuration for applications such as our water resource monitoring project. A related problem in wireless sensor networks is spatial localization [8]. Localization is particularly important when sensors are not deployed deterministically, as for example, when sensors are thrown from airplanes in a battlefield or when underwater sensors drift under the current's action. A number of techniques for both fine- and coarse-grained localization have been proposed recently [9, 18]. The problem of determining the coverage provided by a given placement of sensors has also been discussed in the literature [23]. Sensor placement on two- and three-

dimensional grids has been formulated as a combinatorial optimization problem, and solved using integer linear programming [10, 11]. This approach suffers from three main drawbacks. First, computational complexity makes the approach infeasible for large problem instances such as the experimental Montmorency forest. Second, the grid coverage approach relies on ideal sensor detection and communication, i.e. a sensor is expected to yield a boolean (true/false) detection or communication outcome in every case. There is inherent uncertainty associated with sensor readings; hence sensor detections must be modeled probabilistically. Third, the grid-based approach which is used to represent the geographic environment fails to capture the geometric complexity of the real space and do not take into account the semantic information associated with various geographic features (river, lake, road, vegetation). The approach and tools we propose can possibly solve these problems as the models are refined.

IX. CONCLUSION, AND FUTURE WORKS

As a result of new advances in wireless communications, sensing technologies, and application design in SW, the development of a SW management system is becoming necessary and possible. Because of the significant differences between traditional networks and SW, a different management solution is required. In contrast to classic sensor network management systems which address network-health monitoring, fault-detection, traffic management, congestion avoidance, power management, and resource management, we focus on the spatial interactions between sensor nodes and their geographic environment. Recent research works have started to integrate artificial intelligence techniques in sensor webs in order to improve their autonomy and capabilities. However, these techniques are mainly applied at the sensor level (micro-level), which is still suffering from its limited capabilities. Moreover, most research works used artificial intelligence techniques for distributed purposes. In this article, we described an intelligent SW management framework based on the MAGS approach which takes into account the geographic environment characteristics. Our Sensor-MAGS framework aims at gaining better insight in intelligent sensor web management by analyzing the interaction mechanisms between sensors and their geographic environment. None of the existing sensor web management systems provides a fully integrated representation of the geographic environment as we propose. However, the development of general-purpose intelligent SW management is a challenging problem and remains a largely unexplored area for SW.

Given the constraints inherent to the water resource monitoring project, certain aspects of Sensor-MAGS have been simplified or not fully addressed in this design. In particular, we focused on the role of the IVGE to support spatial decision making of sensor nodes, but we did not discuss the synchronization of the physical (real) and virtual (simulation) sensor webs. It is important to note however that

this issue can be addressed within the Sensor-MAGS design proposed here and will be presented in forthcoming papers. In addition, the most challenging part of the proposed multi-agent geo-simulation approach corresponds to the mirroring of the physical sensor web using an agent-based architecture. Indeed, the original aspect of the proposed architecture is its capacity to go beyond the physical limits of the sensor nodes in order to manage sensor webs. Our goal is to provide resource constrained sensor nodes with agents' capabilities in order to improve the sensing and monitoring tasks of sensor webs. Besides, sensor webs are geographically distributed systems in which the spatial dimension is of fundamental importance. Hence, being able to access reliable GIS data helps assessing the spatial distribution of sensor nodes. It also allows guiding the geo-simulation process by immersing software agents in virtual geographic environments. Currently, we are working on the implementation of the behavioral models for sensor agents and their integration within the IVGE with the goal of demonstrating the applicability of MAGS to SW management.

REFERENCES

- [1] K. Akkaya and M. F. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, 2005.
- [2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *IEEE Communication Magazine*, 40(8):102–114, 2002.
- [3] I. F. Akyildiz, M. C. Vuran, and O. B. Akan. On exploiting spatial and temporal correlation in wireless sensor networks. In *WiOpt'04: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 71–80, 2004.
- [4] R. Andersen, J. L. Berrou, and A. Gerodimos. On some limitations of grid-based (ca) pedestrian simulation models. In *First International Workshop on Crowd Simulation (V-Crowds'05)*. VRLab, EPFL, 2005.
- [6] N. Athanasiadis, M. Milis, M. P.A., and S. Michaelides. Abacus: A multi-agent system for meteorological radar data management and decision support. In E. Jakeman & D.A. Swayne, editor, *Sixth Int'l Symposium on Environmental Software Systems (ISESS-05)*, Sesimbra, Portugal, 2005.
- [7] I. Benenson and P. Torrens. *Geo-simulation : Automata-based modeling of urban phenomena*. Wiley. Wiley, 2004. [7] P. K. Biswas and S. Phoha. A middleware-driven architecture for information dissemination in distributed sensor networks. In *Sensor Networks and Information Processing Conference*, pages 605–610, 2004.
- [8] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, October 2000.
- [9] N. Bulusu, J. Heidemann, and D. Estrin. Adaptive beacon placement. In *Twenty-first International Conference on Distributed Computing Systems (ICDCS-21)*. University of California, Los Angeles, IEEE Computer Society, April 2001.
- [10] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Coding theory framework for target location in distributed sensor networks. In *International Symposium on Information Technology: Coding and Computing*, 2001, pages 130–134, 2001.
- [11] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Grid coverage for surveillance and target location in distributed sensor networks. *IEEE Transactions on Computers*, 51:1448–1453, 2002.
- [12] K. Delin. The sensor web: A macro-instrument for coordinated sensing. *Sensors*, 2(1):270–285, 2002.
- [13] S. W. Frye and D. J. Mandl. Autonomous mission operations systems. *Sensor Webs*, pages 13–16, 2006.
- [14] J. Giacomini and F. Vasconcelos. Wireless sensor network as a measurement tool in precision agriculture. pages 17–22, Rio de Janeiro, Brazil, September 2006.
- [15] J. Giacomini, F. Vasconcelos, and E. da Silva. Estimating vegetation water content with wireless sensor network communication signals. pages 1–5, May 2007.
- [16] P. B. Gibbons, B. Karp, Y. Ke, S. Nath, and S. Srinivasan. Irisnet: an architecture for a worldwide sensor web. *Pervasive Computing, IEEE*, 2(4):22–33, 2003. 1536-1268.
- [17] J. Gong and L. Hui. Virtual geographical environments: Concept, design, and applications. In *International Symposium on Digital Earth (ISDE)*, pages 369–375, Beijing, China, November 29 - December 2 1999.
- [18] J. Heidemann and N. Bulusu. Using geospatial information in sensor networks. In *Proceedings of the Workshop on Intersections between Geospatial Information and Information Technology*, Arlington, VA, USA, October 2001. National Research Council.
- [19] A. Howard, M. J. Matarić, and G. S. Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Auton. Robots*, 13(2):113–126, 2002.
- [20] N. Jabeur, J. D. McCarthy, X. Xing, and P. A. Graniero. A knowledge-oriented meta-framework for integrating sensor network infrastructures. *Computers & Geosciences.*, 35(4):809–819, 2009.
- [21] M. Kallmann, H. Bieri, and D. Thalmann. Fully dynamic constrained delaunay triangulations. *Geometric Modelling for Scientific Visualization*, 2003.
- [22] F. Lamarche and S. Donikian. Crowds of virtual humans: a new approach for real time navigation in complex and structured environments. *Computer Graphics Forum, Eurographics' 04*, 2004.
- [23] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *IEEE INFOCOM*, pages 1380–1387, 2001.
- [24] M. Mekni, N. Jabeur, and B. Moulin. A generic model for situated holonic multi-agent systems: A case study on sensor web management. In *MDAI 2008: Proceedings of 5th International Conference on Modelling Decisions for Artificial Intelligence*, pages 35–46, Sabadell, Spain, October 2008.
- [25] M. Mekni and B. Moulin. Agent-based clustering for sensor webs self-organization. In A. Vukovic, editor, *Proceedings of the Eighth IASTED International Conference Wireless and Optical Communications (WOC 2008)*, pages 89–94, Quebec City, Quebec, Canada, 26-28 May 2008.
- [26] M. Mekni and B. Moulin. A multi-agent geo-simulation approach for sensor web management. In *SensorComm'08: Proceedings of the Second International Conference on Sensor Technologies and Applications*, volume 0, pages 129–134, Agay, France, 2008. IEEE Computer Society.
- [27] M. Mekni and B. Moulin. A survey on sensor webs simulation tools. In *SensorComm'08: Proceedings of the Second International Conference on Sensor Technologies and Applications*, volume 0, pages 574–579, Agay, France, 2008. IEEE Computer Society.
- [28] M. Mekni and B. Moulin. Using multi-agent geo-simulation techniques for intelligent sensor web management. In *CSTST'08: Proceedings of the Fifth International Conference on Soft Computing as Transdisciplinary Science and Technology*, volume 0, pages 669–674, Paris, France, 27-31 September 2008. IEEE Computer Society.
- [29] D. Moodley and I. Simonis. New architecture for the sensor web: the swap framework. In *ISWC 2006: 5th International semantic web conference*, page 17, Athens, GA, USA, 2006.
- [30] G. Ng and K. Ng. Sensor management - what, why and how. *Information Fusion*, 1(9):67–75, 2000.
- [31] OMNeT++. <http://www.omnetpp.org/>.
- [32] Open Source Geospatial Foundation (OSGeo). GDAL-OGR: Geospatial Data Abstraction Library / Simple Features Library Software, Aug. 2008. <http://www.gdal.org>.
- [33] S. Paris, M. Mekni, and B. Moulin. Informed virtual geographic environments: an accurate topological approach. In *The International Conference on Advanced Geographic Information Systems & Web Services (GEOWS)*, pages 1–6. IEEE Computer Society Press, 2009.
- [34] N. Sahli, M. Mekni, and B. Moulin. A multi-agent geo-simulation approach for the identification of risky areas for trains. In *5th Workshop on Agents in Traffic and Transportation @ Autonomous Agents and Multiagent Systems (AAMAS 2008)*, 2008.
- [35] N. Sahli and B. Moulin. Agent-based geo-simulation to support human planning and spatial cognition. In *Multi-Agent-Based Simulation VI*, pages 115–132. 2006.

- [36] P. Sung, A. Savvides, and M. B. Srivastava. Simulating networks of wireless sensors. In *Simulation Conference, 2001. Proceedings of the Winter, volume 2*, pages 1330–1338 vol.2, 2001.
- [37] F. Tecchia, C. Loscos, and Y. Chrysanthou. Visualizing crowds in real-time. *Computer Graphics Forum*, 21(4):753–765, 2002.
- [38] M. Tubaishat and S. Madria. Sensor networks: an overview. *Potentials, IEEE*, 22(2):20–23, 2003. 0278-6648.
- [39] D. G. Ullman. *Making Robust Decisions: Decision Management For Technical, Business, and Service Teams*. Trafford Publishing, Victoria, BC, Canada, first edition, 2006.
- [40] G. Weiss. *Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence*. MA: MIT Press., Cambridge, 1999.
- [41] M. Wooldridge. *Introduction to Multiagent Systems*. John Wiley & Sons, Inc., London, UK, 2001.
- [42] J. Zhu, J. Gong, H. Lin, W. Li, J. Zhang, and X. Wu. Spatial analysis services in virtual geographic environment based on grid technologies. *MIPPR 2005: Geospatial Information*.