Distributed Belief Revision and Environmental Decision Support

Benedita Malheiro†and Eugénio Oliveira‡

†Dept. of Electrical Engineering,

ISEP, Rua de S. Tomé, 4200 Porto, Portugal

Email: mbnm@garfield.fe.up.pt

WWW URL: http://www.up.pt/~bene

‡Dept. of Electrical Engineering and Computers,

FEUP, Rua dos Bragas, 4099 Porto CODEX, Portugal

Email: eco@garfield.fe.up.pt

WWW URL: http://garfield.fe.up.pt:8001/~eol/eco.html

Abstract

This article discusses the development of an Intelligent Distributed Environmental Decision Support System, built upon the association of a Multi-agent Belief Revision System with a Geographical Information System (GIS). The inherent multidisciplinary features of the involved expertises in the field of environmental management, the need to define clear policies that allow the synthesis of divergent perspectives, its systematic application, and the reduction of the costs and time that result from this integration, are the main reasons that motivate the proposal of this project.

This paper is organised in two parts: in the first part we present and discuss the developed *Distributed Belief Revision Test-bed - DiBeRT*; in the second part we analyse its application to the environmental decision support domain, with special emphasis on the interface with a GIS.

1 Introduction

The decision activity in the field of environmental management is highly complex and involves a great number of contradictory interests (socio-economic, ecological, etc.). Environmental problems often result from the distributed and uncoordinated land use management practices of individual decision-makers that, when taken together, cause significant environmental impacts. To develop feasible and politically acceptable solutions to such problems it is often necessary to foster compromise and consensus among a diverse set of special interest groups who possess overlapping sets of objectives; some quantifiable, some not. The union of these sets forms a criteria space that constrains the set of feasible solutions that may be adopted by resource managers [Bennett, 1995]. Since decision processes and methods vary among individuals, group decision support tools must not only enable group members to specify their preferences, but they must also enable them to highlight differences and similarities among alternative scenarios and resolve conflicts that will inevitably arise [NCGIA, 1995].

This activity presents a number of well defined features: (i) is distributed over the group of advisors; (ii) each indivual performs autonomously his share of problem solving; (iii) the existing dependency between expertise domains establishes cooperative links among the members of the technical committee; and (iv) lacks a systematic methodology to accommodate divergent opinions. In view of this scenario, we propose the modelling of the behaviour of the environmental management team as an autonomous cooperative multi-agent system with consistency maintenance capabilities. Finally, the integration of such a multi-agent system with a GIS will provide a common systemic framework for the environmental management activity, with clear advantages (inferior costs, reduction of the turnaround time, well defined criteria for consistency maintenance, etc.).

This paper is structured in two parts: in the first one, the developed distributed belief revision system is presented, and the most relevant aspects of distributed consistency maintenance are discussed; the second part is dedicated to the analysis of the application of the distributed consistency maintenance system to the environmental management domain and to the current state of the project.

2 Distributed Belief Revision

Reasoning while maintaining the knowledge base's consistency in a distributed system presents many challenges (see [Huhns, 1991]). In a multi-agent system two perspectives regarding the consistency of the available information coexist: the global system perspective and the local or partial view of the system distributed components. The functionalities that the system exhibits at a higher level emerge from the conjunction of the individual elements functionalities. From the point of view of the individual agents, the local representation of external propositions as well as the belief revision of the internal and external propositions are crucial issues. From the multi-agent system perspective, the consistency among different perspectives of the various agents, the management of the multiple existing contexts, and the amount and size of the inter-agent messages are essential aspects.

Typically, a consistent context is one in which the simultaneous occurrence of a proposition and its negation is not allowed. The notion of consistency used through out this paper is one where the simultaneous occurrence of a proposition and its negation is not permitted, a proposition is not allowed to hold distinct belief status, and semantically contradictory propositions are forbidden.

In a distributed system diverse levels of consistency occur: (i) when the sets of propositions of each one of the various agents are internally consistent the system is referred as *locally consistent*; (ii) when the reunion of these sets is consistent the system is classified as *globally consistent*. The selection of the adequate consistency level becomes one of the more important aspects of the design of a distributed system with consistency maintenance.
The adoption of a distributed system consistency policy has to take into account the system's type of control (centralised, decentralised), the characteristics of the tasks to be performed (tasks with hard or soft time constraints; granularity of the tasks), and the problem to be solved (some problems impose automatically the required consistency level).

2.1 Distributed Belief Revision with DiBeRT

The *Distributed Belief Revision Test-bed (DiBeRT)* developed is intended to study and model inherently distributed systems, with decentralised control, in which the available information is incomplete and dynamic and the time factor is relevant. A presentation and discussion of the architecture, the used knowledge representation, the multiple contexts management mechanism, the belief revision methodology, the inclusion and representation of external propositions, the applied belief status synthesis criteria, and the sharing of relevant inconsistencies follows.

2.1.1 Architecture

The adopted multi-agent architecture is based on the architectural model proposed by the Esprit ARCHON project [Wittig, 1992]. The agents have a double layer architecture (Fig. 1): the cooperation layer (CL) and the intelligent system (IS) layer. While

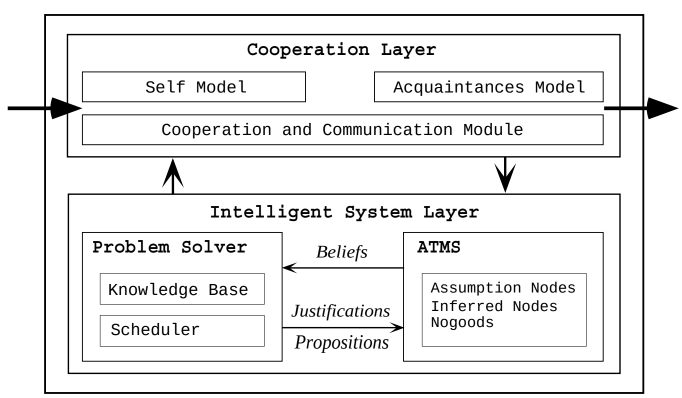

Figure 1: Agent Architecture

the latter contains the agent's domain knowledge based system, the first, holds the functionalities needed for the establishment of the inter-agent cooperative actions.

The CL contains a model of the agent - the Self Model, as well as a model of its acquaintances - the Acquaintances Model. Based on these models, the CL determines when and what type of cooperative action to start, and guarantees that the data sent is relevant to the activity of the recipients through the use of a direct message passing mechanism.
In DiBeRT, the IS is a belief revision system composed of two modules: the problem solver and the assumption based truth maintenance system (ATMS) [de Kleer,1986]. The latter is responsible for maintaining the agent's IS knowledge base free of inconsistencies.

2.1.2 Knowledge Representation in the IS

The IS knowledge base is a production rule system. The rules and propositions of each agent can be of several types. There are *inference rules* and *inconsistency detection rules* represented through the following structure:

$rule(Rule\_Id, Dep\_Lev, Type, Agent\_Id, Antecedents\_List, Consequent)$
$Rule\_Id$ - The rule's identification;
Dep_Lev - The rule's dependency level;
$Type$ - The rule's type ($Type \epsilon \{inference, inconsistency\_detection\}$);
$Agent\_Id$ - The identification of agent owner of the rule;
$Antecedents\_List$ - The list of the rule premises;
$Consequent$ - The rule conclusion.

The propositions are of the type

$prop(Attribute, Value)$
$prop(Concept, Attribute, Value)$

and can represent *axioms*, *assumptions* or *ordinary inferred propositions*.
The rules and propositions are internally represented at the assumption based truth maintenance system (ATMS) as arbitrary identifiers called *nodes*. The node structure has the following fields:

$node(Node\_Id, Type, Label, Scope, Agent\_Id, Status)$
$Node\_Id$ - The node's identifier;
$Type$ - The node's type ($Type \epsilon \{axiom, assumption, ordinary\}$);
$Label$ - The node's set of support assumptions;
$Scope$ - The scope of the node ($Scope \epsilon \{private, shared\}$);
$Agent\_Id$ - The identification of the agent owner of the node;
$Status$ - The node's belief status ($Status \epsilon \{believed, unbelieved\}$).

2.1.3 Knowledge Representation in the CL

The CL is responsible for the interface between the IS and the remaining agents of the multi-agent system, establishing, automatically, cooperative actions whenever the IS:

(i) needs help - it asks for external help to accomplish the undergoing tasks (task sharing); (ii) provides voluntary help - it sends, voluntarily, results relevant to the problem solving activity of the recipients (result sharing); (iii) performs belief revision over shared propositions - it automatically resends to every agent with whom the revised proposition is shared the updated belief status [Malheiro, 1995a].
The agent that starts a cooperative action is called organiser and the agent that responds to the *organiser agent* is named the *respondent agent*.
According to the different kinds and possible stages of the cooperative actions a high level communication protocol with the following primitives was developed:

$primitive(requests, Organiser, Respondent, Data)$ - the organiser agent requests the respondent agent for help in obtaining the specified data;

$primitive(queries, Organiser, Respondent, Data)$ - the organiser agent queries the respondent agent about the belief status of the data specified;

$primitive(assigns, Organiser, Respondent, Data)$ - the organiser agent sends the respondent agent the updated belief status of some proposition(s) they share;

$primitive(answers, Respondent, Organiser, Data)$ - the respondent agent answers the organiser agent by sending the requested data;

$primitive(replies, Respondent, Organiser, Data)$ - the respondent agent answers the organiser agent question concerning the specified data belief status;

$primitive(explains, Respondent, Organiser, Data)$ - the respondent agent explains to the organiser agent its foundations for the specified beliefs.

The contents of the $Data$ field are (proposition, belief status) pairs:

$$Data = [(Node - Id_1, Status_1), \ldots, (Node - Id_n, Status_n)]$$

There are other primitives used by the launcher agent for controlling the activity of the community:

$primitive(starts, Organiser, Respondent, Data)$ - the organiser agent (the launcher) commands the respondent agent to start its activity;

$primitive(halts, Organiser, Respondent, Data)$ - the organiser agent (the launcher) orders the respondent agent to cease its activity.

The implemented inter-agent communication is asynchronous and is based on UNIX operating system sockets.

2.1.4 Multiple Context Management

The DiBeRT prototype was meant as a modelling tool for inherently distributed systems in which the available information is incomplete and dynamic. In order to accommodate as fast as possible the perceived or communicated world changes DiBeRT's main inference mechanism is forward chaining or data-driven. This operative mode has some disadvantages, namely, the possibility of being easily distracted from the main goal - staying absorbed in processing data which is not relevant to the current system's focus. To allow meta-control over the system's activity, a multiple context management mechanism was designed and implemented. This control is achieved through the classification of the knowledge domain into sub-domains. A sub-domain is a pre-defined set of rules and propositions that can be distributed over a group of agents. The control is performed through the enabling of the relevant sub-domains and the disabling of the less interesting sub-domains. This mechanism is based on the attribution of belief status to the rules of the system according to the current focus of the system: (i) the rules that belong to the relevant sub-domains are believed - represented by assumption nodes, and thus ready to be triggered; (ii) the rules that belong to the irrelevant sub-domains are not believed - represented by ordinary nodes without valid foundations, and thus disabled. The change of focus of interest is specified by the User.
This meta-control uses, exclusively, the already available ATMS functionalities.

2.1.5 Belief Revision

The belief revision task is based on the prior classification of the propositions. An agent's group of propositions is divided into two sets: (i) the set of *private propositions* - propositions only used by this agent; and (ii) the set of *shared propositions* - propositions that are shared with some acquaintance. The belief revision of the private propositions is automatically performed by the local agent's ATMS. The belief revision of the shared propositions is accomplished by the shared propositions' owner agent. An agent, upon revising the belief status of a shared proposition, immediately communicates its updated belief status to every recipient with whom it is shared, guaranteeing the system's physical consistency (see [Mason, 1994]).

Inclusion of External Beliefs The decision of how and when to include external beliefs in an agent's knowledge base is fundamental to the characterisation of a distributed belief revision system [Malheiro, 1996]. The DiBeRT agents act in "good faith" and exchange messages using the direct message passing mechanism, thus guaranteeing, that the information received by a recipient agent is, not only, relevant for its activity, but also, truthful from the sender's perspective. A wide range of different methodologies for the inclusion of incoming beliefs can be adopted by the recipient agents: unconditional acceptance, conditional acceptance, rejection, etc.. From this spectra DiBeRT has chosen two policies for the local inclusion of communicated beliefs:

local consistency of the shared propositions - the local beliefs prevail over the communicated beliefs, i.e., the adoption of an external belief is conditioned by the existence or absence of the belief in the agent's local knowledge base. A previously incorporated external proposition is abandoned as soon as the agent infers it by itself. In the absence of a locally deduced belief, a shared proposition for which there are several external belief status is represented by as many nodes as there are external beliefs;

global consistency of the shared propositions - every communicated belief is unconditionally added to the local knowledge base of the recipient agent. A shared proposition owned by different agents has a multiple node representation: is represented by as many nodes as there are agents with beliefs concerning the shared proposition.

While the first consistency methodology for the shared propositions is an instance of a conditional acceptance - a communicated belief is included, if and only if, there is no local belief concerning the shared proposition, the second, is an example of unconditional acceptance - a received external belief is always added to the local knowledge base.

Local Representation of the External Propositions The actual representation of the adopted external propositions is based on the available ATMS functionalities, and depends on the belief status of the external propositions:

an externally believed proposition - is locally represented by an assumption node;

an externally unbelieved proposition - is locally represented by an ordinary node without valid foundations (the contents of the node's label is the empty set).

The accommodation of external propositions' belief status update from:

believed to unbelieved - corresponds, locally, to the removal of the previous assumption node representation, and to the creation of a new ordinary node without valid foundations to hold the external belief;

unbelieved to believed - corresponds, locally, to the removal of the prior ordinary node with empty label representation, and to the creation of a new assumption node in agreement with the external belief update.

Belief Status Synthesis Criteria Upon accepting a set of external beliefs, an agent may find itself with conflicting belief status for the same proposition. In such circumstances which belief status to adopt? The synthesis of the different belief status attributed by the involved agents to the same shared proposition can be performed using diverse criteria, such as unanimity, majority, negotiation, and many others.
In DiBeRT two distinct synthesis criteria were implemented guaranteeing the assignment of an unique belief status to every shared proposition:

the disjunctive (OR) synthesis criterion - a shared proposition is believed as long as there is some agent where it is believed;

the conjunctive (AND) synthesis criterion - a shared proposition is believed, if and only if, it is believed by every agent that share the proposition.

These two synthesis criteria reflect different levels of demand: in the case of the OR synthesis, the belief in a shared proposition by one of the involved agents is enough to make it believed by the system, while, in the case of the AND synthesis, only the consensus among the involved agents makes a shared proposition believed by the system. Although, at first sight, the OR criterion may seem too relaxed, it holds a justifiable explanation. The meaning of the believed/unbelieved belief status is considerably different from the Boolean true/false attributions. In a system based on assumptions, every proposition is believed as long as it holds valid foundations, and ceases to be believed when there is a lack of valid reasons for believing it. In this context, the application of the OR criterion is consistent with the methodology used in these systems to attribute belief status, i.e., a proposition is believed by the system as long as there is, at least, one valid reason for its belief.
The AND synthesis criterion represents a higher consistency demand, well suited for problems where the generalised consensus is a must.
The interpretation of the two adopted synthesis criteria is similar to the notions behind the necessary truth and possible truth operators of the standard Modal Logic. The necessary truth operator translates the idea of truth in every accessible contexts (worlds), and the possible truth operator conveys the idea of truth in at least one accessible context.

2.1.6 Belief Sharing Format

The information exchanged among agents is, as was previously explained, mainly composed of beliefs. Supposing the code is optimised, the total execution time can only be reduced through the minimisation of the time and amount of beliefs exchanged. Typically, and since the beliefs are internally represented by nodes, a structure very similar to the node structure (proposition + label) should be expected. The adoption of the node-like structure would imply, not only, the future update messages of the proposition and its label, as well as, the update messages regarding the assumptions contained in the label.
In an effort to reduce the number and size of inter-agent messages, the structure selected for the exchange of beliefs was reduced to the (proposition, belief status) pair. Since this format does not include the label, not only, is obviously shorter, but also, avoids the subsequent updating messages of the assumptions contained in the label. However, the reduced belief exchange format no longer allows the receiver agent to verify if there are inconsistencies between its local beliefs and the foundations of the incoming belief.

2.1.7 Sharing of Relevant Inconsistencies

This limitation imposed by the selected reduced belief exchange format led to the sharing of detected inconsistencies (invalid environments or nogoods) among the agents that share propositions. The intention is to guarantee that the shared propositions hold valid foundations at all times.

An agent upon the detection of a local inconsistency, removes it from every valid context, registers it in its ATMS, and, inspects the nogood. If the nogood affects any shared beliefs, the agent immediately sends it to the concerned agents. The receiver removes the communicated nogood from its contexts and records it in its ATMS.

The sharing of the inconsistencies relevant to the activity of the agents results in the maintenance of valid foundations for the shared beliefs.

2.1.8 DiBeRT functionalities

In DiBeRT, the user is asked, at launch time, to select from the available set of agents the sub-set to be run, the synthesis criterion to be applied, and the level of consistency desired. The user choses one from the following four available distributed consistency modes for execution:

shared beliefs local consistency and conjunctive belief status synthesis criterion;

shared beliefs local consistency and conjunctive belief status synthesis criterion;

shared beliefs global consistency and conjunctive belief status synthesis criterion;

shared beliefs global consistency and disjunctive belief status synthesis criterion.

The private beliefs consistency level is unique: they are always locally consistent.

After launching the community of multi-agents, the interaction between DiBeRT and the User is performed by a specialised agent called User Interface Agent. The User Interface Agent architecture is identical to the remaining system agents, being the IS role played by User.

This interface allows, during runtime: (i) the addition of new assumptions; (ii) the multiple contexts management; (ii) the attribution of specific belief status; (iv) the querying of the system about any beliefs.

3 Distributed Environmental Decision Support

So far, the DiBeRT prototype has been tested with simplified problems based on simulated data. The need to evaluate its performance and adequacy in face of a real world problem solving suggested its application to the environmental decision support field. This second part is focused on the analysis of the diverse aspects of the association of DiBeRT with a GIS.

3.1 Geographical Information Systems

The main caracteristics of a GIS are the storing, processing and analysis capabilities of the spatial and alphanumeric data representing a geographic area (see [Burrough, 1992]). On one hand, a GIS constitutes a resourceful geographic data bank, encapsulating the capabilities of a relational database management system (RDBMS) together with a spatial database, allowing the simultaneous representation of inter-related graphic entities and respective alphanumeric attributes (Fig. 2). On the other hand, it contains a large

Id	Height	...
C_1	10000	
...		
C_4	200	

Id	Soil Cover	Area (km^2)
A_1	forestry	2100
...		
A_6	urban	350

Id	Unemployment	...
A_1	>10%	
...		
A_4	<4%	

Id	Prot. Species	Area (km^2)
A_1	red fox	900
A_2	ilex	1200
A_3	salamandra	800

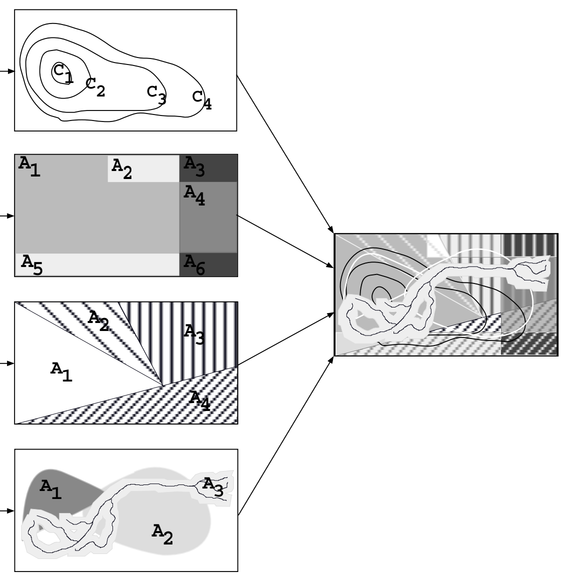

Figure 2: GIS Architecture

variety of geographical data processing and handling functions to be used on the stored data, to produce its classification and analysis.
The spatial database is structured in layers, where each layer can contain one of two different spatial data representations: raster or vector data. Raster data is based on a grid cell matrix representation, while vector data approach is based on the graphic entity concept. Raster data is particularly well suited for satellite imagery and aerial photography. Vector data is best tailored for the representation of themes extracted from pre-existing maps (roads, streams, power lines, county borders, etc.) or resulting from the processing and classification of raster data (e.g. contour lines extracted from a digital elevation terrain model). Typically, each spatial data layer contains a specific theme (vegetation, digital elevation terrain model, streams, demographic census, etc.) and is geo-referenced, permitting the simultaneous consultation and overlaying of the desired thematic layers. The vector data is made of sets of geo-referenced space points, which, define, recursively, more complex graphic entities (Fig. 3): a line is an ordered list of points, and polygon is an ordered list of lines. Associated with every graphic entity (point, line or polygon) exists an unique identifier, the relational database key to access

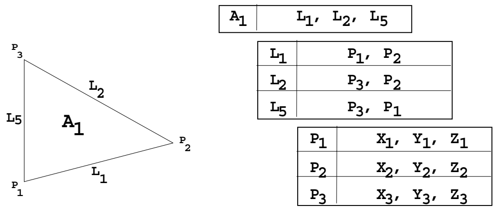

Figure 3: Vector Data Organisation

the entity's existing alphanumeric attributes (vegetation type, area, altitude, etc.). Finally, the multitude of raster and vector data handling, processing and classification functions, the automatic availability of the generated results, the constant data accessibility, and the countless possible studies that can be performed, make GISs powerful integrated geographic data tools.

3.2 Multi-agent Systems and Environmental Decision Support

Distribution over different domains and expertises exhibited in the environmental management field led, naturally, to the idea of its modelling trough a multi-agent system. The environmental management task force (see for example Fig. 4) can be viewed as

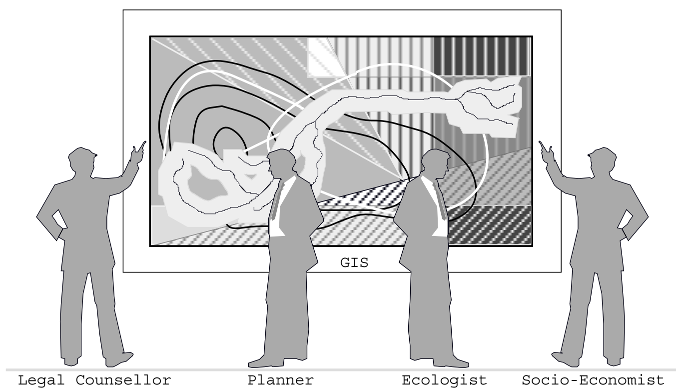

Figure 4: Example of an Environmental Management Team

a community of autonomous cooperative agents. The modelling of the experts cooperative activity through a distributed belief revision system is very adequate.
The goal is to reproduce, in an integrated framework, the behaviour of the group of technical advisors, while trying to reach a conclusion.
The advantages of the integration of the advisors activity with the GIS, when compared with the stand alone operative mode, include: (i) the definition and systematic application of adequate synthesis criteria whenever conflicts arise, and (ii) the explicitation of the existing inter-agent dependency relations and domain overlap. A better understanding, clarification and optimisation of the current technical committee operative mode

results in obvious benefits for the overall environmental decision support task (total time and cost reduction).
The distributed nature of the problem, the dynamic and incomplete features of the information, and the variety of expertise involved provide an adequate real world setting for the testing of the DiBeRT.

3.3 Development of a Distributed Environmental Decision Support System

The development of a distributed environmental decision support system prototype presents two main aspects:

the specification of the generic interface between both applications - (i) the detailed analysis of the GIS functionalities (expected inputs and produced outputs), and (ii) the definition of the interface between the two applications;

the modelling of the application domain - (i) the knowledge elicitation phase to build each agent knowledge base, and (ii) the acquisition and introduction in the GIS of the data concerning the study area.

The specification of the interface between the GIS and DiBeRT is fundamental. It has to allow the agents to: (i) consult/query any spatial or alphanumeric data stored; (ii) to create and store new spatial and alphanumeric data; (iii) to alter/remove any spatial or alphanumeric data stored; (iv) to invoke the available GIS data processing procedures. Two approaches have been considered to act as interface:

the interface software agent approach - the agent collects every request made to the GIS, translates it into GIS inputs, submits it to the GIS, and, finally, sends the produced outputs back to DIBeRT.

the DiBeRT agent approach - the agent's Intelligent System Layer (see subsection 2.2) corresponds to the GIS (domain knowledge database), and the Cooperation Layer constitutes the interface between the IS and the rest of the community. The translation of the requests presented to the GIS into GIS commands will be executed by a Convergence Layer (CvL), according to ARCHON architecture proposal (see [Wittig, 1992]), turning DIBeRT into a heterogeneous multi-agent system.

3.4 Proposed Interface Specification

After a careful analysis of both proposals (the implementation of an interface software agent or the modelling of the GIS as a DiBeRT agent) the DiBeRT agent proposal was selected (Fig. 5). Not only it interfaces both naturally and automatically with the DiBeRT prototype, but, it will also constitute another heterogeneous application of the ARCHON architecture. In this case, the specification of the interface includes:

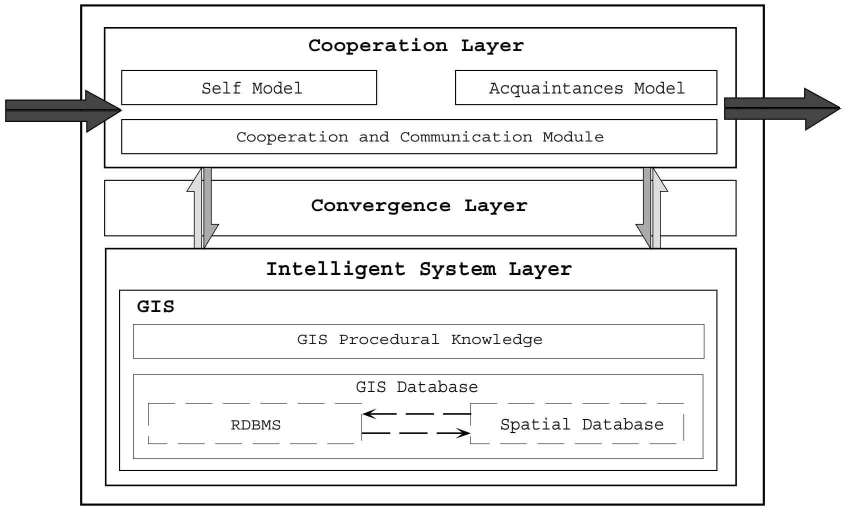

Figure 5: GIS Agent Architecture

- ***the CL of the GIS agent*** - the CL contains the Self Model, the Acquaintances Model, and a Cooperation and Communication Module. The Acquaintances Model is built at launch time, and enumerates, not only, the acquaintances and situations in which help can be provided to each specific agent, but also, when each specific agent is able to voluntary supply help to others. The Self Model describes the tasks each agent knows how to perform, as well as the knowledge it is capable of inferring. In the case of the GIS agent, the Self Model has to specify:

    1. the concepts and attributes existing within the GIS - suppose that every graphic entity of a vector data layer l has n different attributes in a RDBMS table. It is represented in the Self Model by the structure of the type ($Layer_l$, [$Attribute_1$, $\cdots$, $Attribute_n$]). If the theme of layer l is vegetation, and the attributes for the vegetation layer include, among others, $type$ and $area$ this structure may hold the following data ($vegetation$, [$type$, ..., $area$]). There will be one such structure for each existing vector data layer with associated attributes; Raster data will be represented accordingly - every raster layer represents an attribute (e.g. $soilcovertype$) where each pixel value corresponds to a known soil cover type (e.g. $1 - grass, 2 - corn, \ldots,$ etc.) and constitutes the value of the attribute for that pixel.

    2. the high level GIS functionalities relevant to the application domain, and their decomposition in more elementary GIS procedures.

- ***the CvL of the GIS agent*** - the convergence layer (CvL) role is to act as a translator between the CL and the IS, converting the CL requests into GIS commands. It is highly dependent of the selected GIS and it contains the necessary knowledge to act as an intelligent translator.

- ***the IS of the GIS agent*** - the IS is the actual GIS: contains the domain knowledge base (a model of geographic area is in the spatial and alphanumeric databases), as well as a set of procedures to transform, analyse, and extract the necessary data.

4 Conclusion

The motivation behind the Inteligent Distributed Environmental Decision Support System was, not only, the intrinsic appeal and importance of the nature of the problem domain, but also, its particular suitability for the evaluation and testing the distributed belief revision methodologies developed for the DiBeRT prototype [Malheiro, 1996]. The application domain setting is naturally distributed over a set of domain experts and the exchanged data is dynamic and incomplete, making, therefore, the belief revision activity essential to perform opinion synthesis and to guarantee knowledge consistency. Finally, the selected application domain is sufficiently complex to provide a large set of challenging and motivating problems.
Although the development of the system is still in an early stage the presentation of the ideas behind the project seemed interesting enough to motivate this paper. Currently, the project is in the design and specification phase of the Cooperation and Convergence Layers for the GIS agent.

References

[Bennett, 1995] D. A. Bennett, M. P. Armstrong, and G. A. Wade, ”Agent Mediated Consensus-Building for Environmental Problems: A Genetic Algorithm Approach”, Proceedings of the 3rd International Conference/Workshop on Integrating GIS and Environmental Modeling, 1996.

[Burrough, 1992] P. A. Burrough, ”Principles of Geographical Information Systems for Land Resources Assessment”, Monographs on Soil and Resources Survey N. 12, Oxford Science Publications, 1992.

[de Kleer,1986] J. de Kleer, ”An Assumption-based TMS”, Artificial Intelligence, 28 (2), 1986.

[Huhns, 1991] M. N. Huhns and D. M. Bridgeland, ”Multi-Agent Truth Maintenance”, IEEE Transactions on Systems, Man and Cybernetics 21 (6), 1991.

[Malheiro, 1995a] B. Malheiro e E. Oliveira, ”Improving Assumption based Distributed Belief Revision”, 5th Scandinavian Conference on Artificial Intelligence - SCAI95, Frontiers in Artificial Intelligence and Applications, Eds. A. Aamodt and J. Komorowski, IOS Press, 1995.

[Malheiro, 1995b] B. Malheiro e E. Oliveira, ”An Intelligent Distributed System for Environmental Management”, Environmental Informatics, Methodology and Applications of Environmental Information Processing, Eds. Nicholas M. Avouris and Bernd Page, Computer and Information Science, Vol. 6, Kluwer Academic Publishers, 1995.

[Malheiro, 1996] B. Malheiro e E. Oliveira, Consistency and Context Management in a Multi-Agent Belief Revision Testbed, Agent Theories, Architectures, and Languages, Eds. Michael Wooldridge, Joerg P. Muller and Milind Tambe, Springer-Verlag, Lecture Notes in Artificial Intelligence, Vol: 1037, 1996.

[Mason, 1994] C. Mason, ”ROO: A Distributed AI Toolkit For Belief Based Reasoning Agents”, Proc. of the 2nd International Working Conference on Cooperating Knowledge Based Systems, Keele, UK, 1994.

[NCGIA, 1995] NCGIA Iniciative 17 Collaborative Spatial Decision-Making, Appendix C - Initiative Proposal to NCGIA Board, http://www.ncgia.ucsb.edu/research/i17/appendC.html.

[Wittig, 1992] T. Wittig (ed.), ”ARCHON: An Architecture for Cooperative Multi-Agent Systems”, Ellis Horwood, 1992.