Expert Systems with Applications 39 (2012) 3469–3483

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

# Evaluation of environmental impact upon human health with DeciMaS framework

Marina V. Sokolova [a,b], Antonio Fernández-Caballero [a,c,*]

[a] Instituto de Investigación en Informática de Albacete (I3A), Universidad de Castilla-La Mancha, 02071 Albacete, Spain
[b] Kursk State Technical University, ul.50 Let Oktyabrya, Kursk 305040, Russia
[c] Departamento de Sistemas Informáticos, Escuela de Ingenieros Industriales de Albacete, Universidad de Castilla-La Mancha, 02071 Albacete, Spain

ARTICLE INFO

*Keywords:*
Environmental impact
Decision support systems
Complex systems
Interdisciplinary approach

ABSTRACT

The article is dedicated to the problem of decision making in complex systems. Application of a novel interdisciplinary approach, which widely use intelligent agents is offered. The principal ideas of the novel approach are embodied into the DeciMaS framework, that offers a logical set of stages oriented to creation of decision support systems for complex problem management. The components of the DeciMaS framework and the way in which they are organized are introduced. Design and implementation of the system are discussed. The article demonstrates how the initial information is transformed into knowledge. Impact assessment upon human health evaluation is the case study, which is resolved by DeciMas framework. It includes creation of the meta-ontology. In addition, a multi-agent architecture for a decision support system is introduced. The sequence of the steps for the DeciMaS framework design with Prometheus Development Kit and its implementation with JACK Development Environment are presented as well. Finally, data and experiment results of data modeling, simulation, impact assessment, and decision generation are discussed.



## 1. Introduction

Human activity increases constantly, and both the scale and speed of human influence on the natural, social, economic, and other processes has grown significantly. Therefore, now it is impossible not to take into account as one of the driving forces in the "human–nature–technology" arena. Furthermore, not only must the man-made activity be taken into account, but also the correspondent interactions and feedbacks, and this portfolio of emerging hazards compose the complex systems (CS), which are the object of this research. The science of today has produced significant results in modeling and control over man-made technical systems. Notwithstanding, effective managing of natural complex phenomena often lies beyond our reach.

The majority of real-life problems related to sustainable development and environment, can be classified as complex composite ones, and, as a result, they possess particular characteristics. Complex systems interweave with numerous social, technological, and natural processes, and have connections with various institutions that impedes making mono-solutions and/or mono-approaches. These systems are characterized by high complexity and great number of interacting components. Here difficulties have already appeared during the creation of an abstract model of a complex system, because of the great number of decisions to be made regarding its design. This is why they require interdisciplinary approaches for their study. Non-traditional tools from different domains can be highly effective in the case of CS, providing novel ways to generate decisions and find solutions (Sokolova & Fernández-Caballero, 2009).

The multi-agent approach (Purvis et al., 2003) is an excellent technique that can help to reduce the complexity of a system by creating modular components, which solve private subtasks that together achieve common goals (Weiss, 2000). Every agent utilizes the most effective technique for solving the subtask and does not apply a general approach, which is often acceptable for the system in the whole, but not optimal for a concrete subtask (Sokolova & Fernández-Caballero, 2009). Modern "decision support systems" (DSS) and "expert systems" (ES) are commonly based on intelligent agents, and the concepts of DSS as well those of ES have also been recently modified (Cornelius, 1997; Lussier et al., 2007). For example, some recent academic reports present examples of agent-based DSS for home and hospital care, pre-hospital emergency care and health monitoring and surveillance (Annicchiarico, Cortés, & Urdiales, 2008).

In Athanasiadis and Mitkas (2004) the outcomes of the construction and usage of an agent-based environmental monitoring system are presented. It is aimed to provide measurements of meteorological information and air pollution, to analyze them

* Corresponding author at: Instituto de Investigación en Informática de Albacete (I3A), Universidad de Castilla-La Mancha, 02071 Albacete, Spain. Tel.: +34 967599200; fax: +34 967 599224.

*E-mail addresses:* marina.v.sokolova@gmail.com (M.V. Sokolova), caballer@dsi.uclm.es (A. Fernández-Caballero).

0957-4174/$ - see front matter 
doi:10.1016/j.eswa.2011.09.037

and to generate alarm signals. The system is created by means of the intelligent platform "Agent Academy". The system has a three-leveled organizational structure where data preprocessing, its manipulation and distribution are carried out. The necessary steps for data transformation are executed by the following types of intelligent agents: diagnosis agents, alarm agents, database agents and distribution agents. In another article (Athanasiadis & Mitkas, 2005), the authors report about the application of the agent paradigm for the evaluation of socially-oriented advertising campaigns aimed to affect consumers' behavior. The authors create social communication models to simulate a public response to mass-media influence and introduce a social grid populated with autonomous consumer agents.

In the publication dedicated to working out an interdisciplinary approach to solving conflicts in water domain (Nastar & Wallman, 2009), the authors bring together the problem of managing a complex system. The problem consists a complex application domain (water resources) (A) as well as a wide range of decision makers, experts and other personnel (B). In this work the authors attempt to deal with this composite system and discuss the creation of a conceptual framework, which would be able to solve possible conflicts in system A and simultaneously solve problems in the application domain. The framework presented in Rotmans (2006) is based on the application of the agency paradigm to the Integrated Sustainability Assessment (ISA) cycle. The ambition of ISA is to provide an international scientific society with a general framework, which would include a variety of assessment tools and methods. Then, the author proposes a two-track strategy: intent to use the current portfolio of ISA tools as efficiently and effectively as possible, and contribute to the generation of new ISA tools.

Another work (Jakeman & Letcher, 2003) presents a study dedicated to application of an integrated assessment approach to catchment management. In Letcher, Croke, and Jakeman (2007) a network-based approach to the case of water resource allocation and management is applied. The proposed framework uses a nodal network structure (Letcher et al., 2007), which changes its form depending on the type of decision being made. The framework produces scenarios in the form of "what if" statements related to policy and management of the case of study. The framework presented in Jakeman and Letcher (2003) is centered on the need for improved techniques of uncertainty and sensitivity analysis that can be taken as a measure of confidence for ranking decisions and making a choice. An approach for the integrated assessment of both technical and valuation uncertainties during decision making, based on life cycle assessment has also been presented (Basson & Petrie, 2007). The approach is built on three conditions: placing appropriate bounds on particular aspects, non-overlapping alternatives, and conducting a sensitivity analysis for valuating uncertainties.

To summarize, it is possible to classify the following types of environmental information systems (EIS) (e.g. Garrido & Requena, 2011; Liu & Lai, 2009):

1. *Local systems*. This type of EIS is a kind of "island solution", which is dedicated to the evaluation or assessment of a few parameters or indicators, in other words, these systems are designed to solve a specific problem. For example, one could find a system that provides a specific assessment of parameters for a specific case study or for a limited area. Domain ontologies for such systems are limited, although they may suffer from possible heterogeneity. As a rule, such systems are effective when working within the application domain but are sensitive to any unforeseen changes.
2. *Multi-functional systems*. These systems provide multiple analysis of input information, can be based upon hybrid techniques, and possess tools and methods of data pre- and post-processing, modeling and simulation. Multi-functional systems are less sensitive to changes in the application domain as they possess tools to manage uncertainty and heterogeneity.
3. *Methodologies/frameworks of EIS development*. Frameworks support all the stages of EIS life cycle, starting with the initial system planning. They include system analysis and domain (problem) analysis phases, and then assist and provide EIS design, coding, testing, implementation, deployment and maintenance. In this case, the consolidated cooperation of specialists from various domains with various backgrounds is necessary. Methodologies/frameworks are based upon interdisciplinary approaches and system analysis.

The review of the current state of art in the area of complex systems modeling, agent-oriented methodologies, and decision making approaches allows us to make some conclusions. It is not possible to create a unified methodology for design of decision support systems for complex domains, as decision support systems perform better results when oriented to limited and determined domains. This goal is extremely difficult to achieve because we have to take into account the similarities shared by diverse CS without losing their specific features. However there are many solutions and multi-function tools, which can be successfully used. The problem lies in the existence of a great number of overlapping approaches and methodologies which demonstrate successful results, but nevertheless fail to meet the needs of decision makers for an integrated methodology which supports decision making in complex domains. However, two solutions can be proposed: first, bring together existing methods for decision support systems creation within a more coherent system; second, provide an interdisciplinary flexible methodology for complex, systemic domains and policies.

## 2. Multi-agent system creation with DeciMaS framework

The purpose of the "Agent Based Framework for **Deci**sion **Ma**king in Complex **S**ystems" (DeciMaS) is to provide and to facilitate complex systems analysis, simulation, and their comprehension and management. From this standpoint, the principles of the system approach are implemented in this framework. The overall approach used in the DeciMaS framework is straightforward. The system is decomposed into subsystems, and intelligent agents are used to study them. Next, the obtained fragments of knowledge are pooled together and general patterns of the system behavioral tendencies are produced (Sokolova & Fernández-Caballero, 2008, 2009).

The framework consists of the following three principal phases:

1. **Preliminary domain and system analysis.** This is the initial and preparatory phase where an analyst, in collaboration with experts, studies the domain of interest, extracts entities and discovers its properties and relations. Then, s/he states the main and supplemental goals of the research, and the possible scenarios and functions of the system. During this exploration analysis, the analyst researches the following questions: *what* the system has to do and *how* it has to do it. As a result of this collaboration the meta-ontology and the knowledge base appear. This phase is supported by the Protégé Knowledge Editor, which implements the meta-ontology.
2. **System design and coding.** The active "element" of this phase is a developer, who implements the agent-based system and prepares it for further usage. As support at this phase, the Prometheus Design Kit, which is used to design the multi-agent system, and the JACK Development Environment software tools are used (Winikoff, 2005). Once the coding has finished and the system has been tested, the second phase of the DeciMaS is concluded.

3. **Simulation and Decision making.** This is the last phase of the DeciMaS framework and it has a very special mission. During this phase, the final user, a decision maker, can interact with the system. This interaction consists of constructing solutions and policies, and estimating consequences of possible actions on the basis of simulation models.

The overall view on the support of the principal phases of the development process within the DeciMaS framework is provided in Fig. 1. The figure also illustrates how the DeciMaS phases correspond to lifecycle stages (Sokolova & Fernández-Caballero, 2008). The five standard stages of the information system lifecycle include:

- "Domain and System Requirements Analysis".
- "Design".
- "Implementation".
- "Verification".
- "Maintenance".

The lifecycle stages correspond to the DeciMaS framework phases:

- "Preliminary domain and system analysis" corresponds to "Domain and System Requirements Analysis" stage.
- "System design and coding" integrates "Design", "Implementation","Verification", and "Maintenance" stages.
- "Simulation and Decision making" supports and facilitates the further usage of the system.

DeciMaS supports the general standard flow of steps for information system lifecycle. It makes the framework useful for application across a wide range of complex domains. The possibility of the DeciMaS to be easily adapted to any domain of interest should be noted. The framework is organized in such a way that the change of domain is realized during the first stage of the DeciMas framework, but all the further procedures of data mining and decision generation are completed in a similar way for various domains. This characteristic adds flexibility to the DeciMaS and widens the areas of its application. Moreover, usage of agent teams let to distribute, control, and synchronize the workflows within the system, which are supervised and organized by the team leader agents to manage autonomous knowledge discovery. Furthermore, DeciMaS provides interdisciplinary approach using heterogeneous agent teams, which possess methods from various disciplines. Additionally, the DeciMaS framework uses known terminology, and integrates tools and methods from various disciplines, making good use of their strong sides. This facilitates the usage of the DeciMaS framework by non-scientific users.

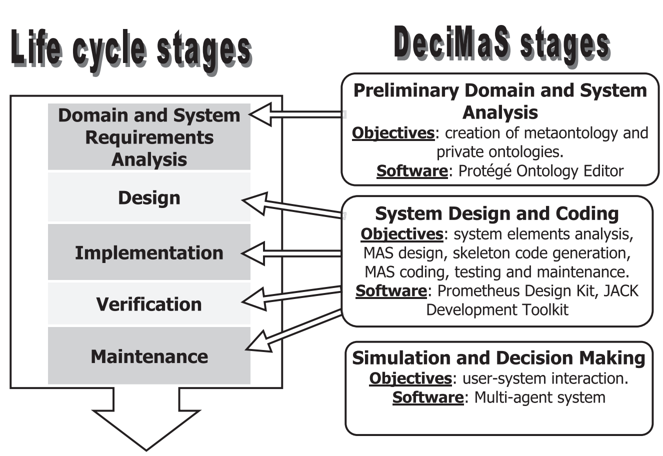


**Fig. 1.** The correspondence between software life cycle and the DeciMaS framework stages.

## 3. Environmental impact assessment with DeciMaS

### 3.1. Statement of a problem

Environment is a clear example of a complex domain, composed of numerous self-organized subsystems. If interactions of humans within the environment are studied, the level of complexity of such a system greatly increases (Briggs, 2008; Lux & Matthews, 2007; Miller et al., 2007; Sokolova, Fernández-Caballero, & Gómez, 2010). It is a fact that the environment affects human health. Climate changes together with growing anthropogenic impact intensify interactions within the "environmental pollution–human health" system. The link between the sustainable development and public health is obvious and does not to be emphasized. Direct and indirect routes by which energy sources, which may affect human health, are presented in Gohlke, Hrynkow, and Portier (2008). In accordance with this reference, direct routes include air pollutants, oil and nuclear energy sources, and indirect routes count water contamination, contamination by heavy metals, climate change and social factors. Humans are affected by this global imbalance, and react with direct and indirect health problems, some example include "excessive heat-related illnesses, vector- and waterborne diseases, increased exposure to environmental toxins, exacerbation of cardiovascular and respiratory diseases due to declining air quality, and mental health stress. Vulnerability to these health risks will increase as elderly and urban populations increase and are less able to adapt to climate change. In addition, the level of vulnerability to certain health problems vary by location. As a result, strategies to address climate change must include health as a strategic component on a regional level. Improving health while addressing climate change will contribute to public health infrastructure today, while reducing the negative consequences of a changing climate for future generations" (Campbell-Lendrun & Bertollini, 2009).

Unfortunately, decision making in environmental health is not a simple task. Indeed, retrospective environmental data are skewed by noise, gaps and outliers, as well as measurement errors. Working with public health information adds restrictions caused by the methodologies of data measurement, the standards currently in use, data availability, and so on. In recent years, it has been proven that it is essential to use products and energy life cycle indicators in order to assess the ecological impact. The idea was developed and fixed in International Standard ISO 14031 "Environmental Management–Environmental Performance Evaluation – Guidelines", which certifies the usage of indirect indicators (ISO, 1999). In case of environmental impact assessment, all the advantages of intelligent agents become crucial. Environmental impact is an indicator, which enables evaluation of the caused by environmental pollution and harmful for human health effects. Environmental pollution, a factor with dominant and obvious influence, causes direct and latent harm, which must be evaluated and simulated in order to create a set of preventive health-preserving solutions.

### 3.2. Preliminary domain and system analysis

#### 3.2.1. Creation of meta-ontology

The structure shown in Fig. 2 is proposed as a framework for meta-ontological MAS design. It is obtained as a result of private ontologies mapping, and is pooled by their common use and execution. Existing relations between concepts, their properties and the ontological semantics make mapping possible. The shared

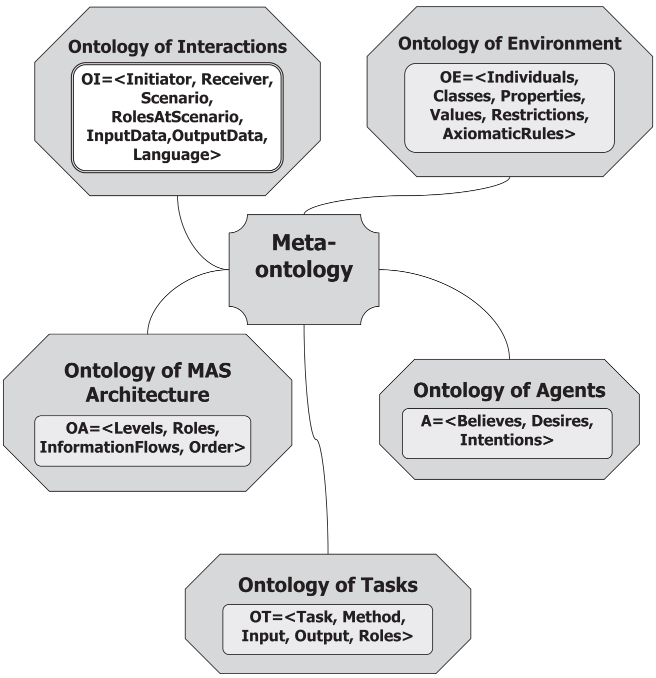


**Fig. 2.** The meta-ontology model, which is used in DeciMaS.

ontological dimension, filled with the data, provides agents with correct addressing of proper concepts and synchronizes the MAS functionality.

The problems that appear at this stage are mostly associated with data heterogeneity. Indeed, in many cases data might be stored in different sources, represented by various identifiers and measured unequally. These procedures can be solved by different methods. Meta-ontology consists of five private ontologies and includes the following models: domain of interest, aims and tasks, agents, interaction, and environment.

1. Ontology of Environment.
2. Ontology of MAS Architecture.
3. Ontology of Agents.
4. Ontology of Interactions.
5. Ontology of Tasks.

The Protégé Ontology Editor was used to design and implement these ontologies http://www.protege.stanford.edu/ (Sokolova & Fernández-Caballero, 2007).

*3.2.2. Private ontologies*

Initially, the Ontology of Environment was created. For the current case of the study, the Ontology of Environment is able to describe the hierarchy of the "environment–human health" system, which includes the following entities:

- To describe the "human health" concept:
  - Morbidity by classes of diseases or external reasons.
  - Endogenous and exogenous morbidity.
- To describe the "environmental impact" concept:
  - Air pollutants.
  - Pollutants of soil.
  - Potable water pollutants.
  - Noise contamination.
  - Usage of energy.
  - Wastes and its structure.

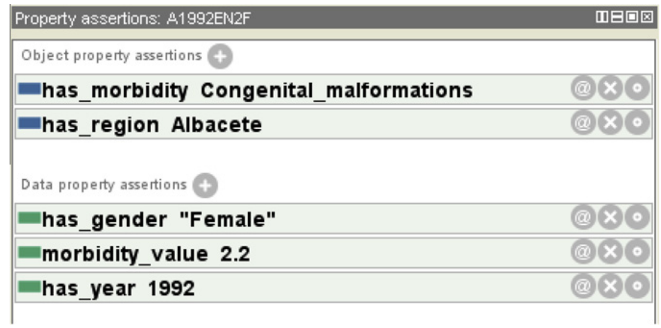


**Fig. 3.** An example of an individual from the *individuals_morbidity* class, its properties and values.

These entities are arranged into classes. Each class has its properties, values, restrictions and axiomatic rules. Using the Protégé Knowledge Editor, properties were divided into "object properties" and "data properties". The Ontology of Environment contain individuals from the two classes: the *individuals_pollution* and *individuals_morbidity* subclasses. These classes were created because "morbidity" and "pollution" individuals have different properties. Fig. 3 illustrates the individual, which belongs to the *individuals_morbidity* class.

This class has two object properties: "has_morbidity" and "has_region". These properties contain other objects as values. The first one contains the "Congenital malformations" individual from the *Morbidity* class as a value, and the last one contains the "Albacete" value from the *Region* class. The three other properties belong to "data property" type. Next, Fig. 4 exemplifies an individual from the *individuals_pollution* class. This class has two object properties and two data properties. The object properties are: "has_pollution" and "has_region". The data properties are "pollution_value" and "has_value".

The ontology for MAS architecture is stated as:

$$OA = \langle Levels, Roles, InformationFlows, Orders \rangle \quad (1)$$

where *Levels* correspond to logical levels of the MAS:

- *Roles* is a set of determined roles.
- *InformationFlows* is a set of the corresponding input and output information, represented by protocols.
- *Order* determine the sequence of execution for every role.

The Task ontology is represented by the following components:

$$OT = \langle Task, Method, Input, Output, Role \rangle \quad (2)$$

where

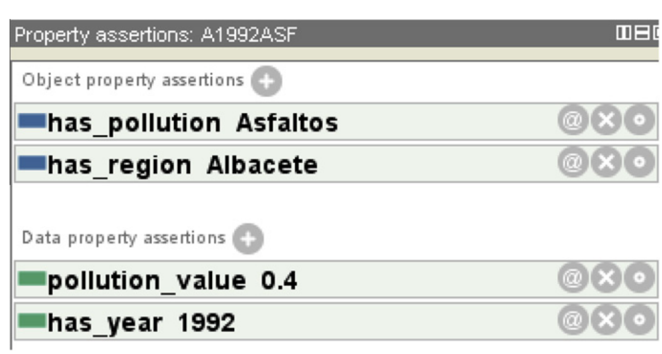


**Fig. 4.** An example of an individual from the *individuals_pollution* class, its properties and values.

- *Task* is a set of tasks to be solved in the MAS.
- *Method* is a set of activities related to the concrete task.
- *Input* and *Output* are input and output data flows.
- *Role* is a set of roles.

The Agent ontology consists of the following components:

$$Agent = \langle Beliefs,\ Desires,\ Intentions \rangle \tag{3}$$

where:

- *Beliefs* are usually represented as facts or in the form of information files, databases, and correspond to the information that the agent has about its environment.
- *Desires* are actions or goals that the agent wants to achieve, and
- *Intentions* are the desires that the agent chooses under the given circumstances.

The interactions between agents is represented by the components of the Interaction ontology and include an initiator and a receiver, a scenario and the roles taken by the interacting agents,the input and output information and a common communication language. The ontology is set up as:

$$OI = \langle Initiator, Receiver, Scenario, Roles, InData, OutData, Language \rangle \tag{4}$$

where

- *Initiator* and *Receiver* are roles, which are assigned to split the information and deliver it to the proper agents.
- *Scenario* corresponds to a protocol.
- *Roles* is a set of roles that the agents play during the interaction.
- *InData* and *OutData* are represented by informational resources, read and created, respectively.
- *Language* determines the communication language.

### 3.3. System design and coding

#### 3.3.1. Logical levels of the decision support system

The DeciMaS framework consists of three phases, which are reflected in the architecture of the agent-based decision support system (ADSS), as it is logically and functionally divided into three layers: the first is dedicated to meta-data creation (information fusion), the second is aimed at knowledge discovery (data mining), and the third layer provides real-time generation of alternative scenarios for decision making (Sokolova & Fernández-Caballero, 2007, 2009). The levels do not have strongly fixed boundaries, because the agents' spheres of competence can overlap and complement each other.

The first layer uses expert knowledge to create meta-ontology and to extract relevant data from external data sources, and then clear and preprocess it. The second logical level is completely based on autonomous agents, which decide how to analyze data and use their abilities to do so. Thus, the aim of the second logical level is to discover the knowledge in the form of models, dependencies and associations from the pre-processed information, which comes from the previous logical layer. The third level of the system is dedicated to decision generation. Both the decision making mechanisms and the human–computer interactions are important here. The system works in a cooperative manner, and it allows decision makers to modify, refine or complete the decision suggestions, providing them to the system and validating them.

#### 3.3.2. Goals and scenarios

The principal goals of the proposed ADSS follow the logical sequence of the main stages of the DeciMaS framework. For this reason, the process of the system design starts with identification of general goals that are divided into subgoals and then refined. Fig. 5 shows the goal tree specified for the system, and where the logical systems' layers are marked.

The final goal is *Create recommendation* and it is achieved as a result of three parallel goals *Make forecast*, *Make sensitivity analysis*

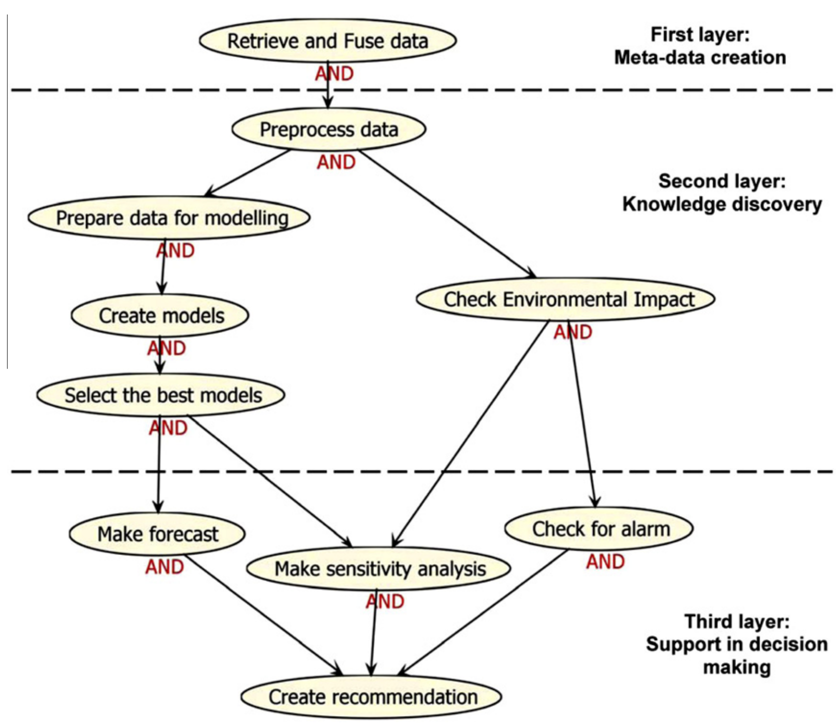


**Fig. 5.** The goals tree and logical layers of the proposed multi-agent system.

and *Check for alarm*. The goals *Make forecast* and *Make sensitivity analysis* use parts of the same knowledge. Both sensitivity analysis and forecasting are based on models received as a result of *Create models* and *Select the best models* goals. The goal *Check environmental impact* is independent from the other goals of the second logical layer, although its outcomes are used for making recommendations on the third logical layer. The goal *Preprocess data* is the initial goal for all the data mining procedures. This fact is illustrated (see Fig. 5) as the goal *Preprocess data* inherits meta-data from the first layer and from the goal *Retrieve and Fuse data*.

Five global scenarios were created, which are:

1. `Retrieve and fuse data`, which suggests collaboration with the actor EXPERT in order to find relevant data storages and extract information from these data sources.
2. `Preprocess data`, which carries out data cleaning activities.
3. `Create models`, which contains a set of activities for modeling, based on data mining procedures.
4. `Check environmental impact`, which evaluates impact caused by environmental contamination on human health.
5. `Create recommendation`, which suggests collaboration with the external actor USER/DECISION MAKER, and contains activities for computer simulation and decision generation.

### 3.3.3. Roles of the proposed MAS

Roles represent agent's functions, responsibilities and expectations. A role enables pooling together the goals of the system in accordance with different types of behavior that an agent assumes when archiving a goal or a series of goals. The distribution of roles for agents determines the agent's specialization and knowledge.

One of the intentions for the system design was to assign one role to each agent or agent team. That requirement was met for the roles *Data Fusion* and *Data Clearing* where the teams of Data Fusion agent and of the Data Preprocessing agent carry out these roles. Moreover, the Function Approximation agent manages three data mining roles: *Impact Assessment*, *Decomposition* and *Function Approximation*, and the Computer Simulation agent takes on *Computer Simulation*, *Decision Making* and *Data Distribution* roles. The correspondence between agents and roles is demonstrated in Fig. 6.

### 3.3.4. Description of the agents

Once the multi-agent system notions have been defined and its logical architecture has been determined, with the set of goals, scenarios, interaction models and data usage, a global view of the system's layers and description of agent teams may be provided. With regard to the proposed multi-agent architecture and in order to

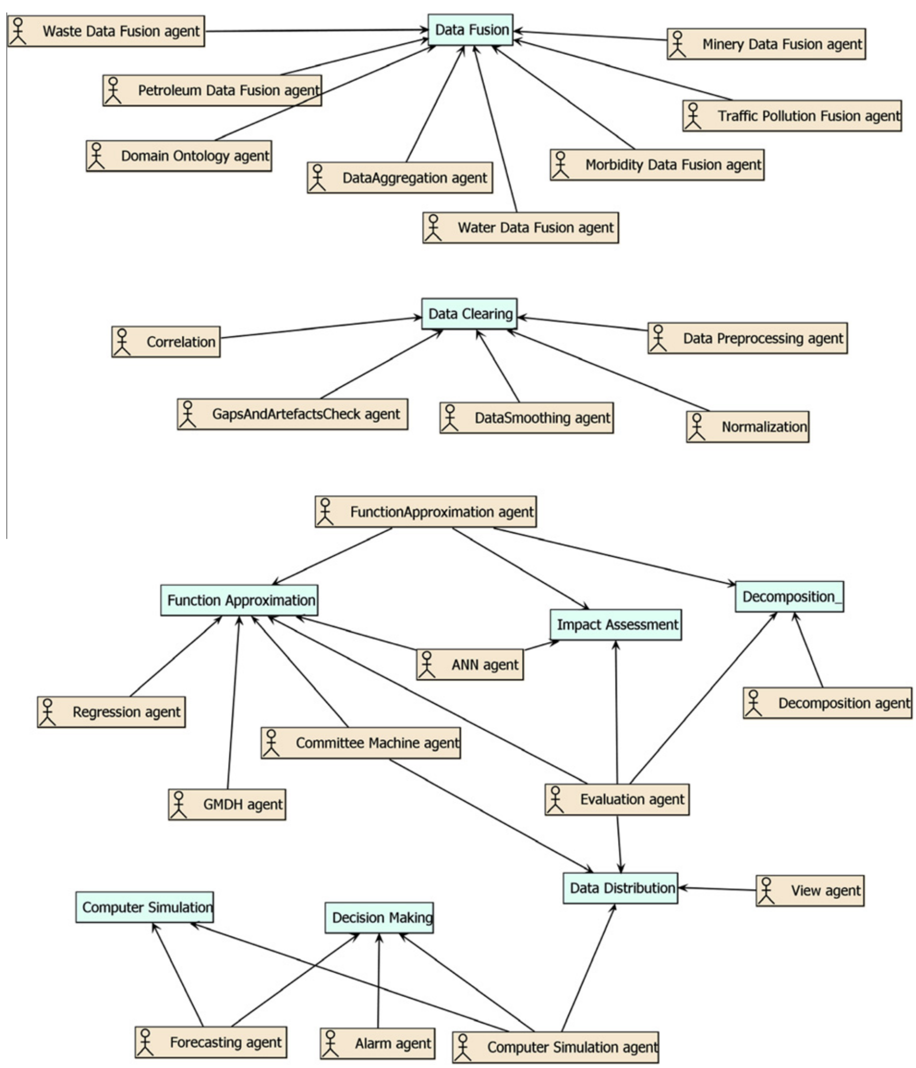


**Fig. 6.** The Agent–Role coupling diagram created in Prometheus.

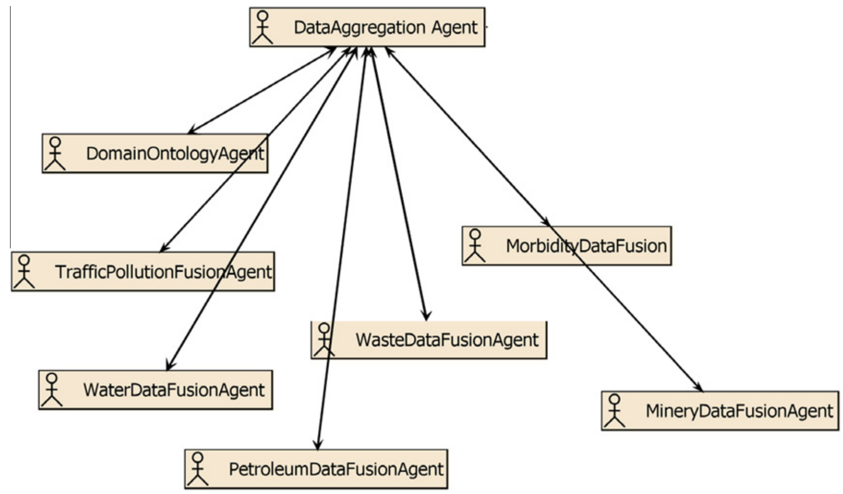


**Fig. 7.** The Data Aggregation agent and its team.

gain time of the recommendation generation process and optimize interactions between agents, local agent teams were used. The teams coordinate and supervise task execution and utilization of resources. Agent teams synchronize the work of the system, execute plans in a concurrent mode, and strengthen the internal management by local decision making. There are four agent teams defined within the system: two within the first level, and one team on the second and third level. Each "main" agent plays several roles.

The Data Aggregation agent is the principal agent, which acts within the role *Data Fusion* at the first logical layer. One of the agents is oriented to read the domain ontology, and the others have to retrieve information from the identified data sources. There are a number of subordinate agents under its control. These are:

1. The Domain Ontology agent.
2. The fusion agents:
   - Water Data Fusion agent.
   - Petroleum Data Fusion agent.
   - Mining Data Fusion agent.
   - Traffic Pollution Fusion agent.
   - Waste Data Fusion agent.
   - Morbidity Data Fusion agent.

The structure of the Data Aggregation agent and its team is shown in Fig. 7.

The process of information fusion requires working with multiple data sources. Some of them can vary significantly in their format and internal data structure. These are the reasons why the Data Aggregation agent receives several subordinate agents in its disposition. They facilitate data retrieval because each of them is specified in a particular type of pollutant.

In the current case study, information is weakly organized, and is presented in the form of plain text files, tables or consolidated forms. In this case, it is necessary to analyze the file structure, and localize the principal concepts and their properties, which can be found as intersections of the concepts or the concepts and their properties. Thereby, data extraction turns into file content analysis.

The Data Aggregation agent must achieve the following goals:

1. Obtain information from the ontology of the domain.
2. Search for information sources, which may contain information of interest stored in the ontology of the domain.
3. Retrieve information from the found sources.
4. Transform the retrieved information in order to avoid heterogeneity.
5. Fuse information.

The Data Aggregation agent interacts with Expert actor and receives information from it.

The Data Preprocessing agent aims to prepare the initial data for further modeling and acts within the *Data Clearing* role. It manages a number of subordinate agents, which make up its team. Each subordinate agent specializes in a different data clearing technique:

- Gaps and Artifacts Check agent clears fused raw information from missing and inconsistent values and fill the gaps.
- Data Smoothing agent carries out exponential and moving average smoothing procedures.
- Normalization agent normalize data sets.
- Correlation agent calculates correlation matrices.

Fig. 8 is a part of Prometheus agent acquaintance diagram, and it provides a view of the Data Preprocessing agent and its team.

The Function Approximation agent has a hierarchical team of subordinate agents, which serve to carry out the roles: *Impact Assessment*, *Decomposition* and *Function Approximation*. The Function Approximation agent has under its control a number of subordinate agents:

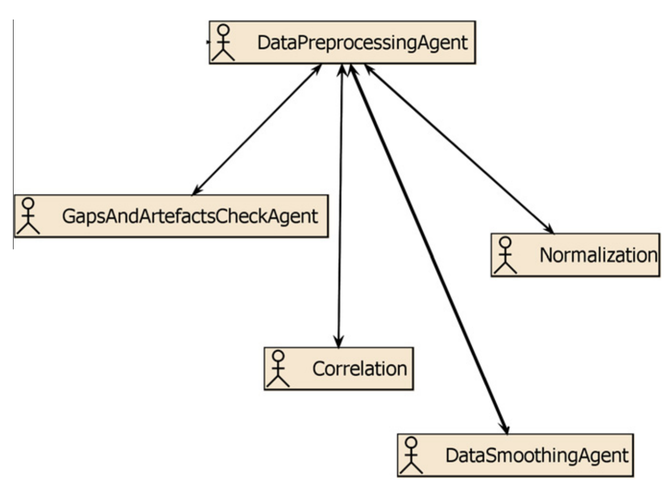


**Fig. 8.** The Data Preprocessing agent and its team.

- Data mining agents, which work in a concurrent mode and create models of the following types:
  - The Regression agent,which creates regression models.
  - The ANN agent, which creates neural network models.
  - The GMDH agent, which creates polynomial models with the group method of data handling.
- The Evaluation agent, that calculates evaluation criteria for models.
- The Committee Machine agent that creates hybrid models.
- The Decomposition agent that carries out the decomposition procedure.

Fig. 9 shows the Prometheus diagram of the Function Approximation agent and its team.

The Computer Simulation agent interacts with the user and performs a set of tasks within the *Computer Simulation*, *Decision Making* and *Data Distribution* roles. It's subordinate agents are the following: (Fig. 10).

- The Forecasting agent, which is used to create forecasts and predictions of dependent and independent variables.
- The Alarm agent, which is used to identify the values, which exceed permissible level.
- The View agent, which is used to organize the computer–user interaction and create textual, graphical, and other types of documents.

The Computer Simulation agent asks for the user's preferences, and, to be more precise, for the information of the disease and pollutants of interest, the period of the forecast, and the ranges of their value changes. Once the information from the user is received, The Computer Simulation agent sends the *SimulateAlternative* message to the Forecasting agent, which reasons and executes one of the relevant plans. When the alternative is created, the Alarm agent compares the simulation and forecast data from the Forecasting agent with the permitted and alarm levels for the correspondent indicators. If they exceed the levels, the Alarm agent generates alarm alerts.

### *3.3.5. Implementation in JACK*

The MAS has an open agent-based architecture, which allows for an easy incorporation of additional modules and tools, enlarging a number of functions of the system. The system belongs to the organizational type, where every agent obtains a class of tools and knows how and when to use them. Actually, such types of systems have a planning agent, which plans the orders of the agents' executions. In our case, the main module of the JACK™ program carries out these functions. The View agent displays the outputs of the system functionality and organizes the interaction with the system user. As the system is autonomous and all the calculations are executed by it, the user only has access to the resulting outputs and the simulation window.

The Data Aggregation agent is constructed with a constructor:

`"DataAggregationAgent DAA = new DataAggregationAgent ("DAA")",`

Its method can be called as *DAA.fuseData( )*. The DataPreprocessingAgent is constructed as follows:

`DataPreprocessingAgent DCA = new DataPreprocessingAgent ("DPA", "x.dat", "y.dat")`

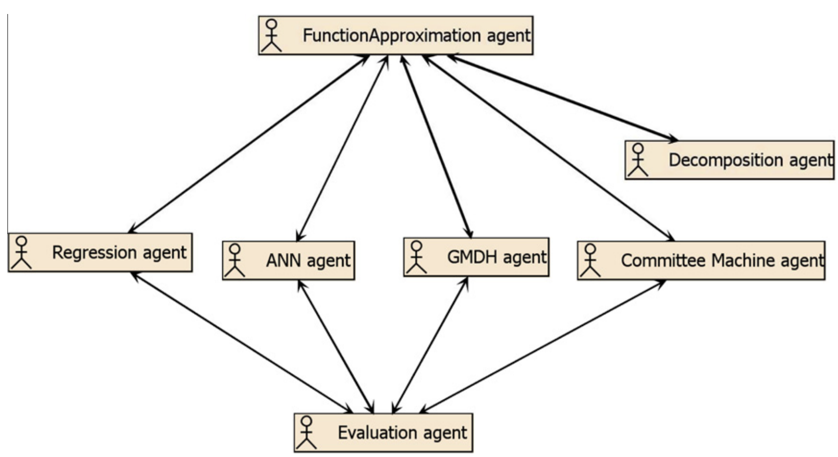


**Fig. 9.** Function Approximation agent and its team.

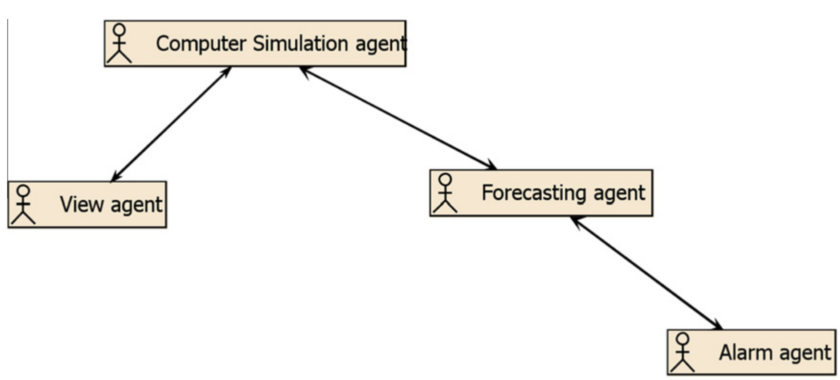


**Fig. 10.** Computer Simulation agent and its team.

where `x.dat` and `y.dat` are agent beliefs containing data about pollutants and morbidity.

In order to choose the appropriate plan, each agent must apply meta-reasoning. With this aim, it determines which plans are handled by the message event reading the `#handles event` statements and then inspects the content of the `relevant()` method of each plan.

```
static boolean relevant (StartReading eV)
   {
       return (eV.startID.equal(word));
   }
```

The Data Aggregation agent is launched when the program starts the execution.

The Data Preprocessing agent and its internal architecture, as presented in the Navigator window, are shown in Fig. 11. The Data Preprocessing agent is a subclass of the JACK's Agent class. The "Java" specifications for the agent include the package name, interfaces it implements, and imported libraries and packages.

The Function Approximation agent is responsible for data mining. It is launched by the triggering event message from the Data Preprocessing agent, and is initialized by the command:

`FunctionApproximationAgent FAA = new FunctionApproximationAgent (''FAA", ''x.dat", ''y.dat")"`

The Computer Simulation agent interacts with the user and receives its preferences for forecasting and simulation. It also manipulates the Forecasting agent, the Alarm agent and the View agent. The Computer Simulation agent is created with the command:

```
"ComputerSimulationAgent DCA = new
  ComputerSimulationAgent ("CSA",
"x.dat", "y.dat", "selectedModels.dat",
  "criticalValues.dat")
```

where `CSA` is the name of the agent, `dataX.dat` and `dataY.dat` are the files that contain initial data, `selectedModels.dat` contain information about selected models and `criticalValues.dat` contains information about critical levels for factors. These text files are used in the Computer Simulation agent beliefs creation during its initialization.

The View agent is implemented by the command:

```
ViewAgent VA = new ViewAgent ("VA", "regions.dat",
  "years.dat",
"pollutants.dat", "diseasesEn.dat",
  "forModels.dat",
"ages.dat", "xNew.dat", "yNew.dat", "yPred.dat",
  "rangs.dat");
```

where `VA` is a name of the agent; `regions.dat` is the textual file, which contains names of regions; `years.dat` is the textual file, that contains numbers of years; `pollutants.dat` is the textual file, which contains names of pollutants; `diseasesEn.dat` is the textual file, which contains names of diseases; `forModels.dat` contains links to the best models that was created, `ages.dat` is the textual file, which contains information about ages; `xNew.dat` contains `X` data sets with pollutants; `yNew.dat` contains `Y` data sets with diseases; `yPred.dat` contains approximated `Y` datasets; `rangs.dat` contains the results of the correlation matrix for `X` and `Y`. All this knowledge from the View agent's constructor is transformed into its beliefs.

The View agent contains both numerical and textual data with the aim to organize the human–computer interface and provide the user with all the necessary information. Fig. 12 gives a view of the principal window of the View agent.

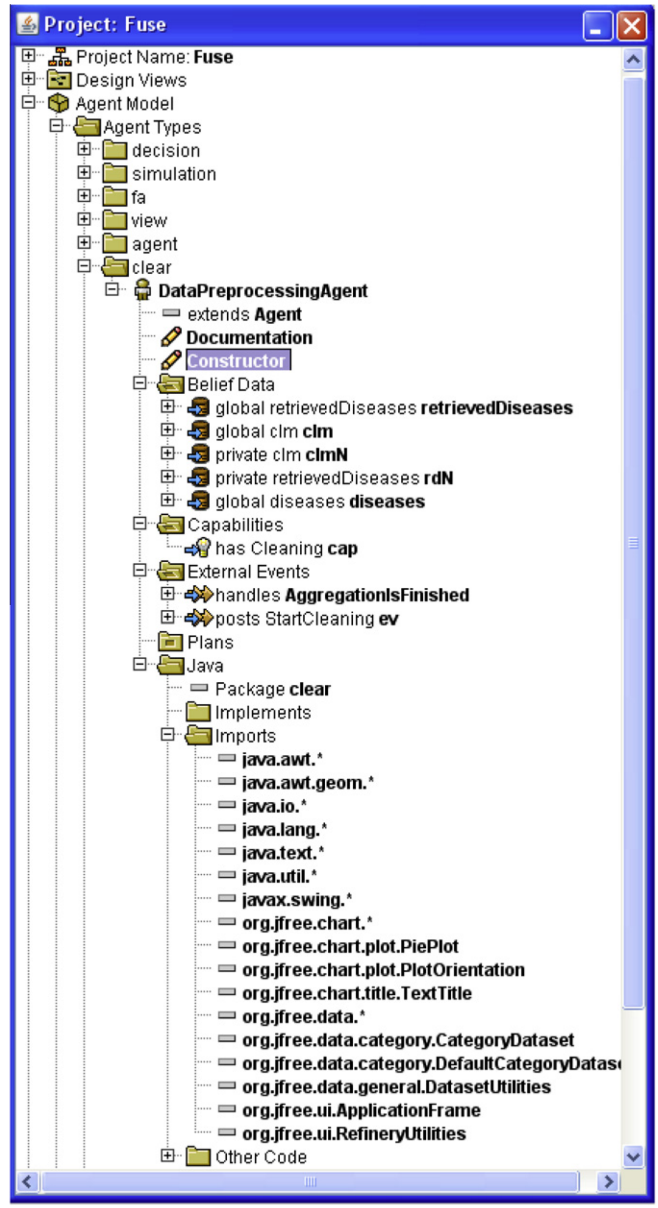


**Fig. 11.** The Data Preprocessing agent and the view of its internal structure. JACK Navigator View.

The window marked "1" is a principal window, which demonstrates a workflow carried out by agents. These works correspond to the logical levels of the agent-based decision support system and include:

- Data retrieval and fusion.
- Data clearing.
- Modeling.
- View results.
- Simulation.

The second window gives a view of the simulation window and in it marked with a "2". There is the option to choose a region, a disease type, and an age range, and shows results of impact assessment and forecasting. The view given below shows the committee machine model for selected disease "Diseases of the nervous system, eye, adnexa, the ear, mastoid process", for the selected region "Castilla-La Mancha", and for the specific age group

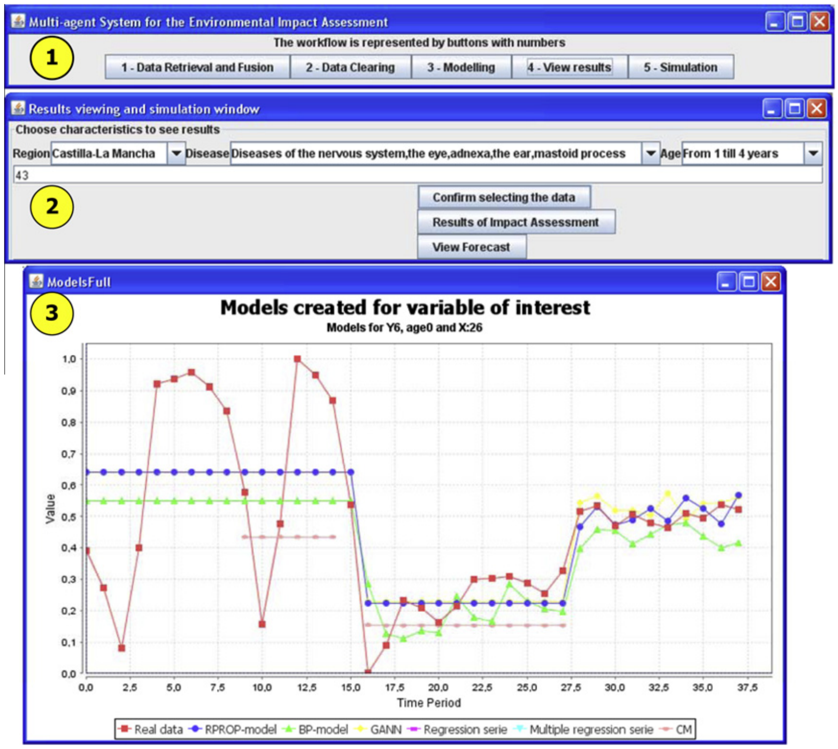


**Fig. 12.** The cascade of windows realized by the View agent.

"from 1 to 4 years". The chart shows models which have been included in the committee, and are defined in the legend field.

## 4. Data and results

### *4.1. Data for the experiment*

The experiment designed to evaluate the possible harm caused by environmental contamination upon public health was conducted for the region of Castilla-La Mancha. Retrospective data dated from 1989 until 2007 was used in order to evaluate the impact of environmental pollution upon human health in Castilla-La Mancha. Resources offered by the Instituto Nacional de Estadística (http://www.ine.es) and by the Instituto de Estadística de Castilla-La Mancha were used for the research (http://www.jccm.es/estadistica/). The factors, which was used in the experiment, are presented in Table 1.

Morbidity, classified by sex and age, was accepted as an indicator to evaluate human health. Table 2 gives a list of diseases examined in this case study. The diseases included in the research were chosen in accordance with the International Statistical Classification of Diseases and Related Health Problems (http://www.who.int/classifications/icd/en/). The sex groups included the following ones: "males", "females" and "total".

Information was retrieved from CSV, DOC and XLS-files and fused together. Data is checked for the presence of missing values and outliers. These can be caused by registration errors or misprints. As a result, the number of pollutants valid for further processing has decreased from 65 to 52. Inconsistent datasets are excluded from the analysis. As artifacts are eliminated in the previous step, they are marked as missing values or gaps. The presence of missing values skews the data and may lead to incorrect or unreliable conclusions. In the current study, some datasets suffer from the presence of gaps. The bar chart given in Fig. 13 visualize the filling gaps procedure for given datasets before (in red) and after (in blue).

The smoothing is applied next in order to homogenize data after the treatment of missing values. The exponential smoothing with the coefficient $\alpha$ equal to 0.15 is used. Next, data is normalized using two normalization methods: "Z-score standardization" and the "Min Max" normalization. Decomposition of the studied complex system, "environmental pollution–human health", is carried out by the means of correlation analysis. As correlation between variables can impede the correct execution of data mining procedures

**Table 1**
Pollutants studied in research.

| | Type of Disease/ pollutant | Disease class |
|---|---|---|
| 1 | Transport | Number of lorries, buses, autos, tractors, motorcycles, others |
| 2 | Usage of petroleum products | Petroleum liquid gases; petroleum autos<br>Petroleum; kerosene; gasohol; fuel–oil |
| 3 | Water characteristics | DQO; DBO5; solids in suspension; nitrites |
| 4 | Wastes | Non-dangerous chemical wastes; other non-dangerous chemical wastes<br>Non-dangerous metal wastes, wastes from used equipment of paper industry, dangerous wastes of glass, dangerous wastes of rubber, dangerous solid wastes, dangerous vitrified wastes, wastes from used equipment, metallic and phosphorus wastes |
| 5 | Principal miner products | Hulla/hull; mercury; kaolin; salt; thenardite; diatomite; gypsum; rock; others |

**Table 2**
Diseases studied in research.

| | Type of Disease/pollutant | Disease class |
|---|---|---|
| 1 | Endogenous diseases | Certain conditions originating in the perinatal period<br>Congenital malformations, deformations and chromosomal abnormalities |
| 2 | Exogenous diseases | Certain infectious and parasitic diseases<br>Neoplasm, diseases of the blood and blood-forming<br>organs and certain disorders involving the immune mechanism<br>Endocrine, nutritional and metabolic diseases<br>Mental and behavioral disorders, diseases of the nervous system<br>Diseases of the eye and adnexa, diseases of the ear and mastoid process<br>Diseases of the circulatory system, diseases of the respiratory system<br>Diseases of the digestive system, diseases of the skin and subcutaneous tissue<br>Diseases of the musculoskeletal system and connective tissue<br>Diseases of the genitourinary system, pregnancy, childbirth and the puerperium<br>Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified<br>External causes of morbidity and mortality |

and lead to false results, a set of non-correlated independent variables **X** for each dependent variable **Y** are created. The independent variables (pollutants) that demonstrated insignificant correlation with the dependent variable (disease), are also included into the set. The mutual correlation between the variables of a model are examined. The variables that have a correlation coefficient greater than 0.7 are marked for exclusion from the model. This procedure is applied for regressions, artificial neural networks, and so on.

### 4.2. Modeling

The ADSS has a wide range of methods and tools for modeling, including regression, neural networks, models, received with the group method of data handling (GMDH), and hybrid models. The function approximation agent selected the best models. This models included: 43 simple regression models, 24 multiple regression models, 4098 neural networks models; 1409 GMDH-models. The selected models are included into the committee machines. Next, the values for diseases and pollutants are extrapolated for the period of ten years with a six month step. This extrapolation allows visualize dynamics of the factors and detect if their values overcome the critical levels. Control under the "significant" factors, which impact health indicators, could decrease some types of diseases. Also, both traditional data mining techniques and other hybrid and specific methods, with respect to data nature (incomplete data, short datasets, etc.) are used. The combination of different tools enabled us to gain quality and precision of the reached models, and, hence, in making recommendations, which are based on these models. Received dependencies of interconnections and associations between the factors and dependent variables help to correct recommendations and avoid errors.

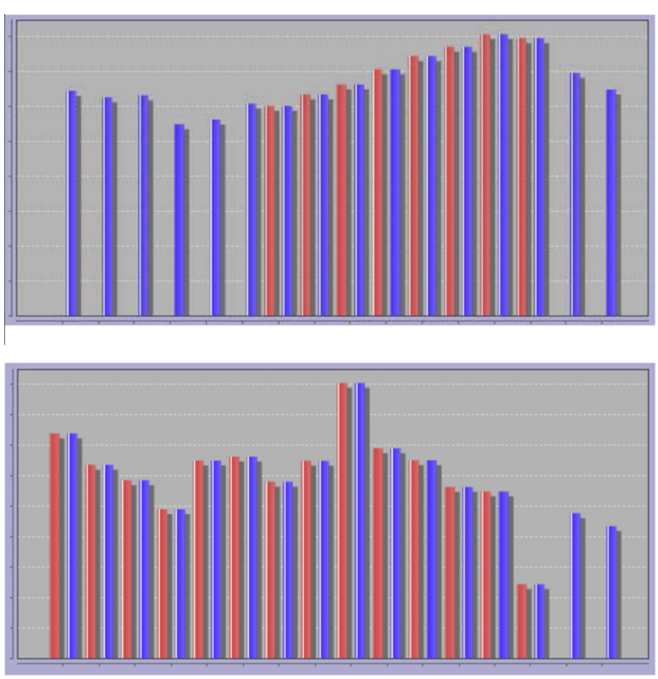

**Fig. 13.** The bar chart that exemplifies the filling gaps procedure: the data before (in red), and the data after filling the gaps (in blue). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

For every class of diseases, plotting morbidity value against pollutant or several pollutants, simple and multiple regressions were performed, both linear and non-linear. As a result, regression models were created of least-squared, power, exponential and hyperbolic types. Each model is evaluated with Fisher *F*-value. Generally, the number of accepted regression models is low, the predictability of the best performing univariate regression models ranges from 0.48 to 0.82 for the discrimination coefficient.

Neural network-based models, calculated for the experimental datasets, have demonstrated high performance results. Networks trained with resilient propagation and with backpropagation algorithms have similar architectures, and the training and testing procedures are equivalent. The best results are received from the networks with a limited number of hidden layers and neurons, as short training sets (which were used for the experiment) require networks with a simple structure. Feedforward networks trained with the backpropagation algorithm, the values of learning rate and momentum vary within the interval $[0, 0.99]$. Better results are obtained with the values of the learning rate within the interval $[0.85, 0.99]$ and the values of momentum within the range $[0.3, 0.4]$. Feedforward neural networks trained with the resilient propagation training algorithm have demonstrated high performance results with the zero tolerance equal to $10^{15}$, the initial update value within the range $[0.05, 0.15]$, and the maximum step equal to 50. Neural network are trained with genetic algorithms with the following parameters of training:

- the size of population, used for training is 1000,
- the percent of the population, to which the mutation operator will be applied is 30%,
- the part of the population, to which the crossover operator will be applied is 10%.

Models, based on Group Method of Data Handling (GMDH) have demonstrated high performance results and efficiency when working with short datasets (Farlow, 1984). The models were received

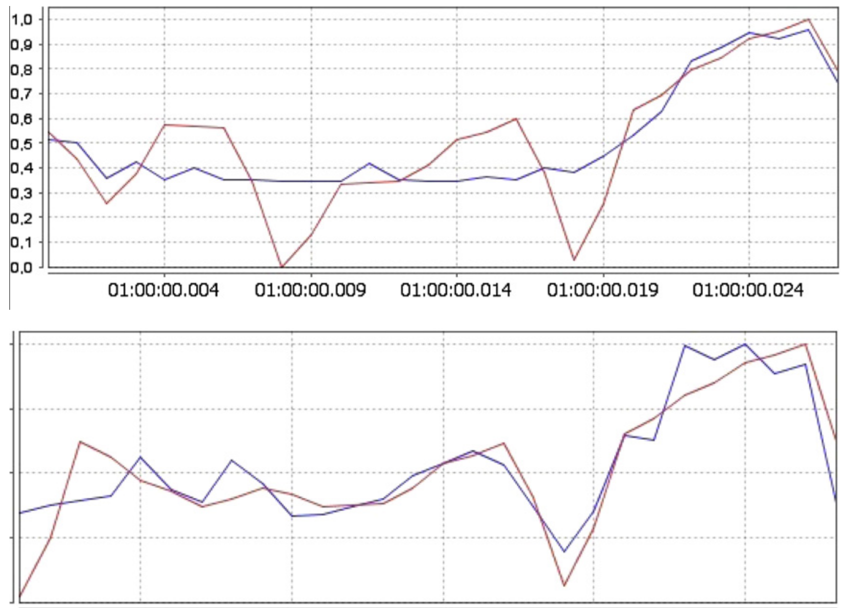


**Fig. 14.** Accepted models for the variable $Y_{35}$ "External causes of death", age group "under 1 year". Approximation of real data by BP-trained (above) and RPROP-trained (below) neural networks.

with combinatorial algorithm, where the combination of the following polynomials were used: $X,\ X^2,\ X_1X_2,\ X_1X_2^2,\ X_1^2X_2,\ X_1^2X_2^2,\ 1/X,\ 1/(X_1X_2)$. The selection of the model is stopped when the regulation criterion started to reduce.

A final model for every variable is a committee machine. As an example of a committee machine, the outcomes of modeling for the variable of interest $Y_{35}$ "External causes of death", age group "all ages" are discussed. First, after the decomposition of the number of variables (pollutants) that could be included into models for interest $Y_{35}$ is reduced and included the following factors: $X_8$, $X_9$, $X_{12}$, $X_{60}$, $X_{61}$, $X_{62}$, $X_{63}$, $X_{64}$. Several models that included these factors are created and evaluated for the variable, $Y_{35}$, and then the best are selected.

The best models that are received:

1. Multiple regression model $Y_{35} = f_1(X_9,X_{61})$.
2. Neural network trained with backpropagation algorithm $Y_{35} = f_2(X_8,X_{63},X_9)$.
3. Neural network trained with RPROP algorithm $Y_{35} = f_3(X_{60},X_{62},X_{12})$.
4. Neural network trained with genetic algorithm $Y_{35} = f_4(X_{64},X_{12})$.

The final model generated by the committee machine is:

$$Y_{35} = \frac{f_1(X_9,X_{61})D_{f_1} + f_2(X_8,X_{63},X_9)D_{f_2} + f_3(X_{60},X_{62},X_{12})D_{f_3} + f_4(X_{64},X_{12})D_{f_4}}{D_{f_1} + D_{f_2} + D_{f_3} + D_{f_4}}$$

where $f_i$ is a model, included into the committee machine, and $D_{f_i}$ is the determination coefficient for the $i$-th model, $i \in [0,\ldots,n]$, being $n$ the number of models.

Fig. 14 gives a graphical representation of the models. The factual information covers 28 years, which are given with a six months step. It starts at "0" and finishes at "27.5". The forecast is made for 10 years, and includes the marks starting from "28" and finishing with "37.5". To realize the forecast, the autoregressive neural networks models for all the factors, which are included

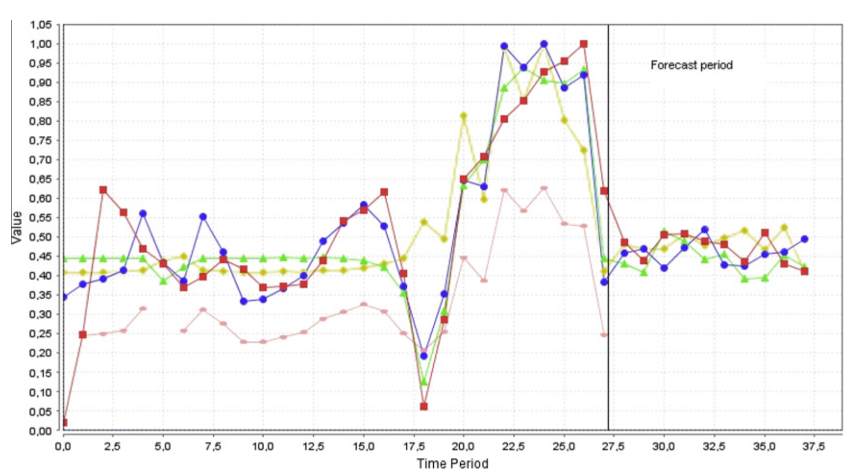


**Fig. 15.** Models for the variable $Y_{35}$ and prognosis for the determined period. Dependent variables are $X_8$, $X_9$, $X_{12}$, $X_{60}$, $X_{61}$, $X_{62}$, $X_{63}$ and $X_{64}$. The data received by the committee machine is in red, the data received by the neural network trained with RPROP algorithm is in blue, the data received by the neural network trained with backpropagation algorithm is in green, the data received by the neural network trained with genetic algorithms in yellow, and the data received by the multiple regression model is in magenta. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

**Table 3**
Table with the outcomes of impact assessment for selected diseases.

| No. | Disease class | Pollutant, which influence upon the disease |
|---|---|---|
| 1 | Neoplasm | Nitrites in water; miner products; DBO5; asphalts; dangerous chemical wastes; fuel–oil; petroleum liquid gases; water: solids in suspension; non-dangerous chemical wastes |
| 2 | Diseases of the blood and blood-forming organs, the immune mechanism | DBO5; miner products; fuel–oil; nitrites in water; dangerous wastes of paper industry; water: solids in suspension; dangerous metallic wastes |
| 3 | Pregnancy, childbirth and the puerperium | Kerosene; petroleum; petroleum autos; petroleum liquid gases; gasohol; fuel–oil; asphalts; water: DQO; DBO5; solids in suspension; water: nitrites |
| 4 | Certain conditions originating in the prenatal period | Non-dangerous wastes: general wastes; mineral, constriction, textile, organic, metal wastes, dangerous oil wastes |
| 5 | Congenital malformations, deformations and chromosomal abnormalities | Gasohol; fuel–oil; DQO water; producing asphalts; petroleum; petroleum autos; kerosene; petroleum liquid gases; water: DBO5, nitrites, solids in suspension |

**Table 4**
Simulation for the variable $Y_{35}$ "External causes of death", age group "less than 1 year" and the dependent variable $X_{62}$.

| Step | Predicted value | Changes of dependent variable, in % | | | |
|---|---|---|---|---|---|
| | | +50 | −50 | −10 | +10 |
| *R = 0.904, D = 0.818, MAE = 0.099, MSE = 4.0E−4, F = 7.354* | | | | | |
| 1 | 0.537 | 0.583 | 0.401 | 0.474 | 0.510 |
| 2 | 0.520 | 0.549 | 0.435 | 0.48 | 0.503 |
| 3 | 0.520 | 0.55 | 0.434 | 0.48 | 0.504 |
| 4 | 0.598 | 0.704 | 0.28 | 0.449 | 0.534 |
| 5 | 0.614 | 0.737 | 0.247 | 0.443 | 0.541 |
| 6 | 0.498 | 0.504 | 0.48 | 0.489 | 0.494 |
| 7 | 0.605 | 0.72 | 0.264 | 0.446 | 0.537 |
| 8 | 0.602 | 0.713 | 0.271 | 0.448 | 0.536 |
| 9 | 0.515 | 0.54 | 0.444 | 0.482 | 0.501 |
| 10 | 0.609 | 0.727 | 0.257 | 0.445 | 0.539 |

in the committee machine were calculated. The autoregressive model is a prediction formula that predicts an output $y(n)$ of a system based on the previous outputs, $y(n-1), y(n-2)\ldots$ and inputs, $x(n), x(n-1), x(n-2)\ldots$. For the current case, each autoregressive model is calculated as $x(t) = f(x(t-1), x(t-2), \ldots, x(t-4))$, where $t$ represents time, and has values $(1, 2, \ldots, n)$, $n$ is the length of the dataset, and $x(t)$ is the value of the factor at the step, $t$. Furthermore, each autoregressive neural network model belongs to the feedforward type, and was trained with RPROP algorithm. It's structure includes an input layer with five input neurons, a hidden layer with three or four neurons, and an output layer with one neuron. When the predictions for the factors from the formula of the committee machine are received, they are used to calculate the forecast for $Y_{35}$. The models demonstrate similar results, which do not vary much. In accordance with the forecast, the morbidity from external causes $Y_{35}$ has a tendency to decline (Fig. 15).

For the period of prediction, all the models give similar forecasts, which are not strongly dispersed. That similarity in predictions by different models proves the tendency of the situation development. The outcomes received by the committee machine are marked with red and the response of the committee is a composite from the best models.

### 4.3. Environmental impact assessment results

The impact assessment has shown the dependencies between water characteristics and neoplasm, complications of pregnancy, childbirth and congenital malformations, and deformations and chromosomal abnormalities. Table 3 shows the outcomes of impact assessment for several variables of interest (classes of diseases), which proves that within the most important factors apart from water pollutants, there are indicators of petroleum usage, mine output products and some types of wastes.

### 4.4. Decision creation and simulation

For this case study, a decision is to made by the specialist, however, information that could help him/her to ground it, is offered by the system. First, models in the form of committee machines and predictions are created, and hidden patterns and possible tendencies are discovered. Second, the results of impact assessment explain the qualitative and quantitative dependencies between pollutants and diseases. Finally, the possibility of simulation is supported by the ADSS.

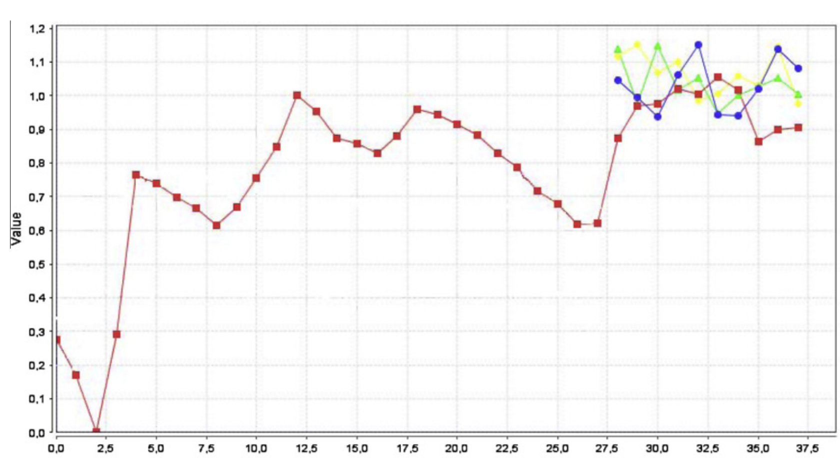


**Fig. 16.** Simulation and forecasting for the variable $Y_{35}$.

The variable $Y_{35}$ "External causes of death" and the age group "all the ages" are chosen in order to exemplify how simulation can be organized. The committee machine model for the variable of interest, $Y_{35}$, is used. Suppose, there is a need to make a sensitivity analysis changing the value of a pollutant and observing how a specific morbidity class would response to this change. Suppose that the pollutant is $X_{62}$. There are four models, which compose a committee machine for the variable $Y_{35}$, and the "RPROP-model" includes $X_{62}$ as an input variable. Table 4 shows outcomes of sensitivity analysis. The first column contains values predicted by a model, the others contain values of $Y_{35}$ calculated under the hypothesis that the variable $X_{62}$ is going to vary. With this aim the values of the variable $X_{62}$ are increased to 50% and 10% (see second and forth columns of Table 4) or decreased to 50% and 10% (see third and fifth columns of Table 4). The model is characterized with correlation coefficient $R = 0.904$ and $F = 7.354$ ($F > F_{table}$). The determination coefficient, $D$, shows that the variables $X_{62}$ and $X_1$ explain approximately 81.8% of the variable $Y_{35}$. The values of the variable $Y_{35}$ are given in a normalized scale, and represent relative growth/decrease of the process.

Calculations, similar to the example presented in the Table 4, were made for each variable of interest. Recommendations given in Table 4 show possible changes in case variable $X_{62}$ decreases or increases to 10% or 50%.

Fig. 16 shows the charts for the simulation of the variable, $Y_{35}$ "External causes of death", in the case when the predictions are calculated with different models and the variable $X_{12}$ is changed to +10%. Fig. 16 shows the outcomes of the simulation for the same variable, $Y_{35}$, in the case when the independent variable, $X_{12}$, is increased to +30%, +20% and +10%.

## 5. Conclusions

This article is dedicated to the creation of decision support systems, bringing together existing methods, and providing an interdisciplinary flexible methodology for complex, systemic domains and policies. The review of the related works presents an overview of comprehensive approaches to complex systems study with a particular emphasis on the problem of decision making for such systems. It is outlined that the problem lies in the existence of many overlapping methodologies which intend to manage complex systems, but nevertheless, do not comply with the requirements for integrated support in decision making.

The DeciMaS framework is a systematic sequence of methods that can be applied in order to study a complex system with respect to it's systemic properties: emergency, possibility to be divided into subsystems, existence of various types of internal and external relations, etc. The DeciMaS framework appears to be a consolidated set of interdisciplinary methods and techniques which can be applied to any complex domain. It is achieved just by changing the domain ontology of the meta-ontology used in the framework.

The case study is performed in order to apply the DeciMaS framework for the identification and evaluation of environmental impact upon human health and generation of alternative decisions sets. The computational experiment was carried out by the means of an agent-based decision support system, which sequentially executed and completed each stages of DeciMaS. The study resulted in several constitutive outcomes and observations, regarding both subjects and methods of the study.

## Acknowledgements

This work was partially supported by Spanish Ministerio de Ciencia e Innovación TIN2010-20845-C03-01 Grant, and by Junta de Comunidades de Castilla-La Mancha PII2I09-0069-0994 and PEII09-0054-9581 Grants.

## References

Annicchiarico, R., Cortés, U., & Urdiales, C. (2008). *Agent technology and e-health. Whitestein series in software agent technologies and autonomic computing*. Basel: Birkhäuser.

Athanasiadis, I. N., & Mitkas, P. A. (2004). An agent-based intelligent environmental monitoring system. *Management of Environmental Quality: An International Journal, 15*(3), 238–249.

Athanasiadis, I. N., & Mitkas, P. A. (2005). Social influence and water conservation: An agent-based approach. *Computing in Science and Engineering, 7*(1), 65–70.

Basson, L., & Petrie, J. G. (2007). An integrated approach for the consideration of uncertainty in decision making supported by life cycle assessment. *Environmental Modelling and Software, 22*(2), 167–176.

Briggs, D. (2008). A framework for integrated environmental health impact assessment of systemic risks. *Environmental Health, 7*(1), 61–72.

Campbell-Lendrum, D., & Bertollini, R. (2009). *Protecting health from climate change: Global research priorities. (2010)*. WHO (World Health Organization) Press.

Cornelius, T. L. (1997). *Fuzzy logic and expert systems applications*. New York: Academic Press.

Farlow, S. J. (1984). *Self-organizing methods in modeling: GMDH-type algorithms*. Marcel Dekker, Inc..

Garrido, J., & Requena, I. (2011). Proposal of ontology for environmental impact assessment: An application with knowledge mobilization. *Expert Systems with Applications, 38*(3), 2462–2472.

Gohlke, J. M., Hrynkow, S. H., & Portier, C. J. (2008). Health, economy, and environment: Sustainable energy choices for a nation. *Environmental Health Perspectives, 6*(116), 236–237.

ISO 14031:1999. (1999). *Environmental management–environmental performance – Guidelines*.

Jakeman, A. J., & Letcher, R. A. (2003). Integrated assessment and modelling: Features, principles and examples for catchment management. *Environmental Modelling and Software, 18*(6), 491–501.

Letcher, R. A., Croke, B. F. W., & Jakeman, A. J. (2007). Integrated assessment modelling for water resource allocation and management: A generalized conceptual framework. *Environmental Modelling and Software, 22*(5), 733–742.

Liu, K. F., & Lai, J.-H. (2009). Decision-support for environmental impact assessment: A hybrid approach using fuzzy logic and fuzzy analytic network process. *Expert Systems with Applications, 36*(3), 5119–5136.

Lussier, Y. A., Williams, R., Li, J., Jalan, S., Borlawsky, T., Stern, E., et al. (2007). Partitioning knowledge bases between advanced notification and clinical decision support systems. *Decision Support Systems, 43*(4), 1274–1286.

Lux, T., & Matthews, W. A. (2007). Advanced technology for environmental modelling. *Environmental Modelling and Software, 22*(3), 279–280.

Miller, S. N., Semmens, D. J., Goodrich, D. C., Hernandez, M., Miller, R. C., Kepner, W. G., et al. (2007). The automated geospatial watershed assessment tool. *Environmental Modelling and Software, 22*(3), 365–377.

Nastar, M., & Wallman, P. (2009). An interdisciplinary approach to resolving conflict in the water domain. In *Proceedings of the 4th international ICSC symposium on information technologies in environmental engineering* (pp. 411–424).

Purvis, M., Cranefield, S., Ward, R., Nowostawski, M., Carter, D., & Bush, G. (2003). A multi-agent system for the integration of distributed environmental information. *Environmental Modelling and Software, 18*(6), 565–572.

Rotmans, J. (2006). Tools for integrated sustainability assessment: A two-track approach. *Integrated Assessment, 6*(4), 35–57.

Sokolova, M. V., & Fernández-Caballero, A. (2007). A meta-ontological framework for multi-agent systems design. In *Proceedings of second international work-conference on the interplay between natural and artificial computation. Lecture notes in computer science* (Vol. 4528, pp. 521–530).

Sokolova, M. V., & Fernández-Caballero, A. (2007) A multi-agent architecture for environmental impact assessment: Information fusion, data mining and decision making. In *Proceedings of the ninth international conference on enterprise information systems* (Vol. DISI, pp. 219–224).

Sokolova, M. V., & Fernández-Caballero, A. (2008). The Protégé – Prometheus approach to support multi-agent systems creation. In *Proceedings of the tenth international conference on enterprise information systems* (Vol. AIDSS, pp. 442–445).

Sokolova, M. V., & Fernández-Caballero, A. (2009). Data mining driven decision making. In *International conference on agents and artificial intelligence. Lecture notes in computer science* (Vol. 5179, pp. 220–225).

Sokolova, M. V., Fernández-Caballero, A., & Gómez, F. J. (2010). Agent-based interdisciplinary framework for decision making in complex systems. In *Proceedings of the international conference on agents and artificial intelligence* (Vol. 2, pp. 96–103).

Sokolova, M. V., & Fernández-Caballero, A. (2008). Facilitating MAS complete life cycle through the Protégé–Prometheus approach. *Agents and Multi-Agent Systems: Technologies and Applications, Lecture Notes in Computer Science, 4953*, 63–72.

Sokolova, M. V., & Fernández-Caballero, A. (2009). Multi-agent-based system technologies in environmental issues. *Information Technologies in Environmental Engineering II*, 549–562.

Sokolova, M. V., & Fernández-Caballero, A. (2009). Modeling and implementing an agent-based environmental health impact decision support system. *Expert Systems with Applications, 36*(2), 2603–2614.

Weiss, G. (2000). *Multiagent systems: A modern approach to distributed artificial intelligence*. The MIT Press.

Winikoff, M. (2005). JACK™ intelligent agents: An industrial strength platform. *Multi-Agent Programming: Languages, Platforms and Applications. Multiagent Systems, Artificial Societies, and Simulated Organizations, 15*, 175–193.