A Case Study of a Compex System with the Decimas Framework

Marina V. Sokolova[1,2];
Antonio Fernández-Caballero[1]; Francisco J.Gómez[1]

Abstract— This paper presents the DeciMaS framework , which supports the vital stages of information system creation. The case study for the DeciMaS framework is also presented. For this reason an agent-based decision support system (ADSS) is created and described in detail. We discuss the structure and the data mining methods of the designed ADSS. The intelligent ADSS described here provides a platform for integration of related knowledge coming from external heterogeneous sources, and supports its transformation into an understandable set of models and analytical dependencies, with the global aim of assisting a manager with a set of decision support tools.

Keywords— multi agent systems, data mining, decision making.

1 Universidad de Castilla - La Mancha, Departamento de Sistemas Informáticos & Instituto de Investigación en Informática de Albacete, Campus Universitario s/n, 02071-Albacete, Spain
marina@dsi.uclm.es, caballer@dsi.uclm.es
2 Kursk State Technical University, ul.50 let Oktiabrya, 94, Kursk, 305040, Russia

Introduction

The principal objective of complex systems (CS) study and analysis is to give a possibility not only to understand and estimate it, but fundamentally to be able to forecast, control and manage it. It is clear that in case of CS we can not intend to embody ideas of rigid "command - execution" style management. On the contrary, we can only rely on flexible preemptive/anticipated correction, which would be harmonized with the nature and dynamics of the respective CS. The degree of control over different components of a CS vary. Indeed, we can control the technical or managerial subsystems, but the independent components of the CS generate their own uncontrolled decisions. Here, the classical decision making process converts in shared and constant collaboration between specialists, and in a supporting tool.

In the last years, some proposals for intelligent and agent-based decision support systems (e.g. Liu, Qian & Song, 2006; Ossowski et al., 2004; Petrov & Stoyen, 2000; Urbani & Delhom, 2005) have been described. New approaches of researching intelligent decision support systems (IDSS) appear following the rapid progress of agent systems and network technologies. Thus, a large range of works dedicated to environment and human health have been implemented as multi-agent systems (MAS), which are in the center of active research for more than ten years and resulted in many successful applications. On the other hand, the use of data mining techniques for environmental monitoring, medicine, and social issues is also a rather common hot topic.

Moreover, using intelligent agents in IDSS enables creating distributed and decentralized systems and localizing control and decision making, as agents by their proper nature themselves continuously make decisions. In an IDSS, control and decision making can be viewed simultaneously, as the internal process, when the system (considered as a community of intelligent entities) solves problems and takes responsibilities for the chosen actions, and, at the same time, as an instrument, which prepares the necessary recommendation information for the human decision maker. Such diligence of responsibilities is essential for the IDSS, dedicated to work with complex systems, such as social, economical, environmental ones.

In this article, the framework that facilitates creation of decision support systems for complex domains is offered. We are going to present its main stages and describe them in more detail.

The Decimas Framework

The framework for decision support system creation denominated "Agent Based Framework for DECIsion MAking in Complex Systems" (DeciMaS), supports the vital stages of information system creation.

The purpose of the DeciMaS framework is to provide and to facilitate complex systems analysis, simulation, and, hence, their understanding and managing.

The overall approach used in the DeciMaS framework is straightforward. The system is decomposed into subsystems, and we use intelligent agents to study them. Then we pool together the obtained fragments of knowledge and we model general patterns of the system behavioral tendencies.

The framework consists of the following three principal phases:

Preliminary domain and system analysis: This is the initial and preparatory phase when an analyst, in collaboration with experts, studies the domain of interest, extracts entities and discovers its properties and relations. Then, he/she states the main and additional goals of the research, and the possible scenarios and functions of the system. During this exploration analysis, the analyst searches answers to the next questions: what has the system 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, and by the Prometheus Design Kit, which is used to design the multi-agent system and to generate the skeleton

code for posterior implementation of the agent-based decision support system (ADSS).

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 a support at this phase, the Jack Intelligent Agents and JACK Development Environment software tools are used. Once the coding has finished and the system has been is tested, the second phase of the DeciMaS concludes.

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, constructing solutions and policies, and estimating consequences of possible actions on the basis of simulation models.

The Case Study for the Decimas Framework

To exemplify the usage of the DeciMaS framework and have designed an ADSS and have applied it to environmental issues in the sense that the system calculates the impacts imposed by the pollutants on the morbidity, creates models and makes forecasts, permitting to try different variants of situation change.

The DeciMaS framework consists from three phases, that is reflected in the architecture of the ADSS. The proposed system is logically and functionally divided into three layers; the first is dedicated to meta-data creation (information fusion), the second is aimed to knowledge discovery (data mining), and the third layer provides real-time generation of alternative scenarios for decision making. Within the logical levels act agents.

The levels do not have strongly fixed boundaries, because the agents construct a community, in which agent spheres of competence can overlap, and the boundaries smooth.

The goals of the ADSS repeat the main points of a traditional decision making process, which includes the following steps: (1) problem definition, (2) information gathering, (3) alternative actions identification, (4) alternatives evaluation, (5) best alternative selection, and, (6) alternative implementation.

The first and the second stages are performed during the initial step, when the expert information and initial retrospective data is gathered, the stages 3, 4 and 5 are solved by means of the MAS, and the 6th stage is supposed to be realized by the decision maker.

Though the goals of the ADSS are determined and keep being constant for various domains, the goals of the case study are not visible and clear at the first sight, that is why the domain of interest has been studied with creation of a goal tree. Prometheus Design Tool (PDK) offers graphical tools to construct it.

The every level of the proposed system is oriented to solve a global goal. The first layer is dedicated to data retrieval, fusion and pre-processing, the second one discovers knowledge from the data and the third deals with making decisions and generating the output information. Let us observe in more details the tasks solved at each level.

In first place, data search, fusion and pre-processing is being delivered by two agents, which perform a number of tasks, following the next workflow:

Information Search -> Data Source Classification -> Data Fusion -> Data Pre-processing ->Believes Creation.

The second logical level is completely based on autonomous agents, which decide how to analyze data and use their abilities to do it. The principal tasks to be solved at this stage are:

- to state the environmental pollutants that impact on every age and gender group and determine if they are associated with previously examined diseases groups;
- to create the models that explain dependencies between diseases, pollutants and groups of pollutants.

Thus, the aim is to discover the knowledge in form of models, dependencies and associations from the pre-processed information, which comes from the previous logical layer. The workflow of this level includes the following tasks:

State Input and Output Information Flows -> Create Models ->Assess Impact -> Evaluate Models -> Select Models -> Display the Results

The third level of the system is dedicated to decision generation. So, both the decision making mechanisms and the human-computer interaction are important here. The system works in a cooperative mode, and it allows decision maker to modify, refine or complete the decision suggestions, providing them to the system and validating them.

This process of decision improvement is repeated indefinitely until the consolidated solution is generated. The workflow is represented below:

State Factors for Simulation -> State the Values of Factors ->Simulate -> Evaluate Results -> Check Possible Risk -> Display the Results -> Receive Decision Maker Response -> Simulate ->Evaluate Results -> Check Possible Risk -> Display the Results

Agents communicate to each other and are triggered by events and sent messages, and share common data. A preliminary system specification was realized by means of the Prometheus Development Kit (PDT), which was chosen due to its possibilities to determine the system structure, the functionalities, the agents' communications and their internals. The other advantage is that PDT incorporates a graphical interface and the possibility to generate the primary code for the JACK Intelligent Agents^{TM} software agent tool. We used Jack to code and test the ADSS.

Data Mining Tools within the ADSS

Agents for Data Search, Fusion and Pre-processing

Our system is an intelligent agent-based decision support system, and as such it provides a platform for integration of related knowledge from external heterogeneous sources, it supports their transformation into an understandable set of models and analytical dependencies, assisting a manager with a set of decision support tools. The ADSS has an open agent-based architecture, which would allow an easy incorporation of additional modules and tools, increasing the number of functions of the system.

Information Search obliges agents to search for data storages that might contain the necessary information, and then classify the found sources in accordance with their type, the presence of ontology concepts and the file structure organization.

After these tasks have been solved, the next work to do is to search the necessary values and their characteristics in agreement with the domain ontology. The crucial task here is to provide the semantic and syntactic identity of the retrieved values, saying they have to be pre-processed before being placed into the ontology and the agent's beliefs set. The properties for the "pollutant" concept include scale, period of measurement, region, value, and pollutant class, whereas "disease" properties include age, gender, scale, measurement period, region, value, and disease class.

Thus, the **Data Aggregation** agent (DAA) firstly searches for information sources and reviews them trying to find if there was a key ontological concept there. If the file contains the concept, the **Data Aggregation** agent sends an internal event to start data retrieval, and passes the identifier of the concept. The plan responsible for execution with the identified concept starts reading the information file and searching for terms of interest.

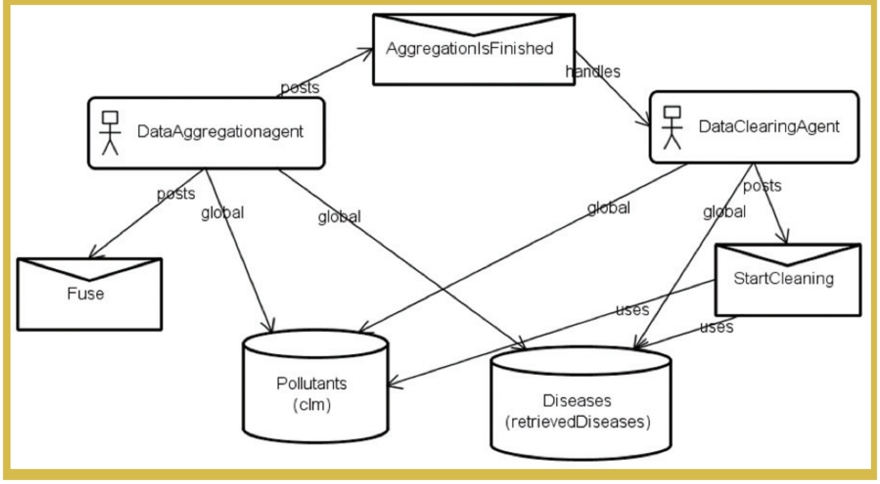


Figure 1. Interaction between the ***Data Aggregation*** *and* ***Data Clearing*** *agents.*

After having checked the information sources presented, and having called plans to recover data, the DAA forms two belief types: "pollutants" and "diseases". Then, the **Data Aggregation** agent sends a message about fusion termination to the **Data Clearing** agent (DCA). The **Data Clearing** agent searches for gaps and outliers. The DCA uses event StartCleaning, capability Cleaning and plans Smooth, FillGaps and Outliers, which respectively do outliers identification and elimination, gaps filling and smoothing, and the believes the agent possess.

There are two types of believes for the DCA: "Pollutants" and "Diseases". The "Pollutants" type currently stores information about pollutants in Castilla-La Mancha (a Spanish region), and contains the following key fields: identity number, region, pollutant name, and value fields, which store yearly records for pollutants. The "Diseases" type determines the beliefs structure for diseases, and includes the same fields as the "Pollutants" type plus the key fields: age and gender. Figure 1 offers a view on the DAA and the DCA and their interactions.

There are two global named data believes created, which can be later used by all the other agents. There is a global believe "Diseases", used in internal plans (and later for data visualization), and the private belief PollutantsN, which belongs to "Pollutants" type and is used in some plans of the DCA for internal calculations.

Also, double data are filtered during data fusion. Before pasting a value into its place in the MAS believes, the **Data Aggregation** agent checks if a record with the same properties has already been pasted. This procedure appeared to be very effective, as the sources of information for sequential years contain data about previous years, and, while searching for the values, on the first stage, DAA copies them all. Every record has its identification, which codes its properties. So, the DAA analyzes identifications of retrieved values and eliminates the similar ones.

If the recovered values satisfy all the requirements imposed or have been adjusted properly, they are placed in the ontology. DCA then is triggered by the AggregationIsFinished message and starts executing plans to pre-process the newly created data sets (they are checked for anomalies, double and missing values, then normalized and smoothed) and creates a global belief, prepared for further calculations.

Agents for Data Mining

Our system is an intelligent agent-based decision support system, and as such it provides a platform for integration of related knowledge from external heterogeneous sources, it supports their transformation into an understandable set of models and analytical dependencies, assisting a manager with a set of decision support tools. The ADSS has an open agent-based architecture, which would allow an easy incorporation of additional modules and tools, increasing the number of functions of the system.

Data fusing and further cleaning compose the preparation phase for data mining. We check the consistency of the obtained data series, and, first of all, outliers have to be detected.

The most well known method of outliers identification is the Z-score standardization, which sets a value as an outlier if it is out of [-3σ, 3 σ] intervals of the standard deviation. The only disadvantage of this method, which makes it not suited to apply here, is that it is too sensitive to the presence of the outliers in our input data. That is why we decided to try more robust statistical methods of outlier detection, based on using the interquartile range.

It states, that a data value is an outlier if:

- it is lower than (Q1-1.5(IQR)),
- it is lower than (Q3+1.5(IQR)),

where Q1 is a 25th percentile, Q3 is a 75th percentile, and IQR=Q3-Q1.

Data normalization is required in order to proceed with further modeling, for example for neural networks creation. DCA can execute Z-score standardization as follows:

$$X^* = \frac{X - mean(X)}{SD(X)} \qquad (1)$$

Where mean(X) is the mean and SD(X) is the standard deviation, or the Min-Max deviation:

$$X^{*} = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (2)$$

These types of normalization are used in different plans by **Function Approximation** (FAA) and **Impact Assessment** agents (IAA). There is a number of ways to replace values for missing data. For instance, we replace values with the mean of the k neighboring values, and the number of values depends on the position of the gap, whether in the middle of the time series or in the edge. The fields with missing values cannot be omitted, as we analyze time series, and as they are usually short, every value in the series is valuable. DCA uses the exponential smoothing, where recent observations are given relatively more weight in forecasting than older observations.

Before starting the modeling itself, we state the inputs (the pollutants) and the outputs (the diseases) for every model. The principal errors to be avoided here are to include input variables which are highly correlated to each other and to include the variables which correlate with the dependent output variables in the model. In this case, we would not receive independent components and the model would not be adequate. These difficulties are anticipated and warned by correlation analysis and factor dimension decomposition, which is based on a neural-network approach.

The **Impact Assessment** agent establishes the groups of factors that can be used to model the dependent variable using the non-parametrical correlation analysis. More precisely, the Mann-Whitney test is used. Those variables, which demonstrated correlation with a given pollutant, are excluded from the set of factors for that concrete pollutant.

To select the most influencing pollutants for every disease, we create neural networks with pollutants as inputs and the variable of interest as output. After training, we make sensitivity analysis for the network and mark the variables that have greater weights as the most influencing ones for that variable (or pollutant).

To be able to make decisions in the system, we need to have adequate functional models of the type Y=f(X). This way, it is possible to simulate disease tendencies and to calculate their values depending on the studied factors, on the one hand. Besides, we require autoregressive models to calculate factor dynamics caused by them. So, the **Function Approximation** agent (FAA) has to be able to execute many data mining strategies. It executes a set of plans, which create statistical regression models (linear and non-linear), the models based on feed-forward neural networks (FFNN), GMDH-models and their hybrids, represented in form of committee machines.

Committee machines provide universal approximation, as the responses of several predictors (experts) are combined by means of a mechanism that does not involve the input signal, and the ensemble average value is received. As predictors we use regression and neural network based models.

The set of created models is wide and contains linear and non-linear regression, neural-networks based models, inductive models based on the group method of data handling approach (GMDH) and their hybrids (Madala and Ivakhnenko, 1994).

After their creation, the models are validated. The selection of the best models for every disease is realized by statistical estimators, which validate the approximation abilities of the models. **Function Approximation** agent uses a set of data mining techniques, including regression linear and non-linear models, autoregression models, and neural networks based on multilayer perceptron.

As we deal with short data sets, we create data models, using GMDH, which is based in sorting-out of gradually complicated models and selecting the best solution by the minimum of external criterion characteristic. Also it is supposed that the object can be modeled by a certain subset of components of the base function. The main advantage derived from such a procedure is that the identified model has an optimal complexity adequate to the level of noise in the input data (noise resistant modeling).

Agents for Simulation

Simulation, together with the previous information about impact assessment and modeling, forms a foundation knowledge, which facilitates the process of making a decision to the user. The final model for every process is a hybrid committee machine of cascading type, which includes the best models received during the data mining procedure.

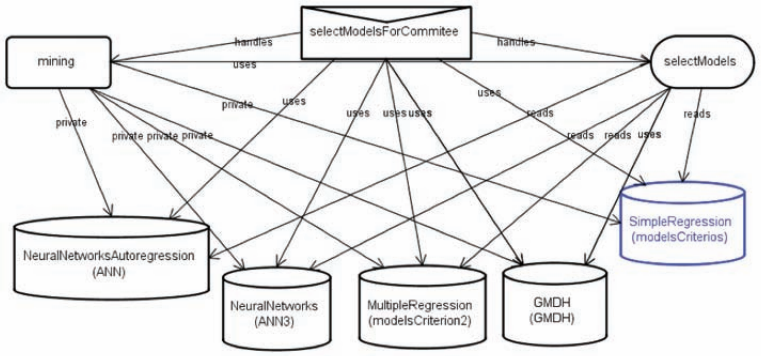


Figure 2. *JACK diagram of committee machine creation.*

The committee machine (see Figure 2) incorporates the FFNNs of autoregression models of factors, the best of the created regression, neural networks and GMDH models of dependent variables, and the block, which calculates the weighted final value. This way of combining models enables increasing the quality of the prediction by incorporating different models of the process and proposing their weights.

FFNN are trained by the backpropagation algorithm, with momentum term (Haykin, 1999). The training process stops when the error reaches the minimal value and then stays at this level. Experiments have shown that the error function curve for the studies processed has the "classical" view. In other words, the error value decreases quickly during the first epochs of training, and then continues decreasing more slowly.

Results

The ADSS has an open agent-based architecture, which would allow us 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. In Figure 3 a part of the code is shown. There, the **Data Aggregation** agent is constructed with a constructor:

```
"DataAggregationAgent DAA1 = new
DataAggregationAgent ("DAA")",
```

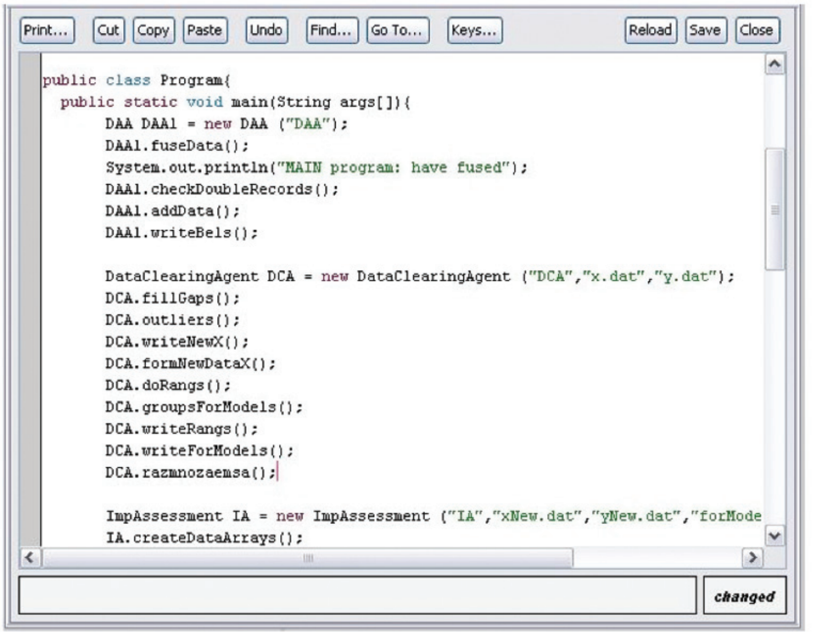
And then some of its methods are called, for example, DAA1.fuseData(). The DataClearingAgent is constructed as

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

Where "x.dat" and "y.dat" are agents believes of "global" type. This means that they are open and can be used by the other agents within the system. Finally, the ViewAgent, which displays the outputs of the system functionality and realize interaction with the system user, is called.

As the system is autonomous and all the calculations are executed by it, the user has only access to the result outputs and the simulation window. He/she can review the results of impact assessment, modeling and forecasting and try to simulate tendencies by changing the values of the pollutants.

To evaluate the impact of environmental parameters upon human health in Castilla-La Mancha, in general, and in the city of Albacete in particular, we have collected retrospective data since year 1989, using open information resources offered by the Spanish Institute of Statistics and by the Institute of Statistics of Castilla-La Mancha. As indicators of human health and the influencing factors of environment, which can cause negative effect upon the noted above indicators of human health, the factors described in Table 1 were taken.



```
public class Program{
  public static void main(String args[]){
       DAA DAA1 = new DAA ("DAA");
       DAA1.fuseData();
       System.out.println("MAIN program: have fused");
       DAA1.checkDoubleRecords();
       DAA1.addData();
       DAA1.writeBels();

       DataClearingAgent DCA = new DataClearingAgent ("DCA","x.dat","y.dat");
       DCA.fillGaps();
       DCA.outliers();
       DCA.writeNewX();
       DCA.formNewDataX();
       DCA.doRangs();
       DCA.groupsForModels();
       DCA.writeRangs();
       DCA.writeForModels();
       DCA.razmnozaemsa();

       ImpAssessment IA = new ImpAssessment ("IA","xNew.dat","yNew.dat","forMode
       IA.createDataArrays();
```

Figure 3. The main program window in JACK.

The ADSS has recovered data from plain files, which contained the information about the factors of interest and pollutants, and fused in agreement with the ontology of the problem area. It has supposed some necessary changes of data properties (scalability, etc.) and their pre-processing. After these procedures, the number of pollutants valid for further processing has decreased from 65 to 52. This significant change was caused by many blanks related to several time series, as some factors have started to be registered recently. After considering this as an important drawback, it was not possible to include them into the analysis. The human health indicators, being more homogeneous, have been fused and cleared successfully.

The impact assessment has shown the dependencies between water characteristics and neoplasm, complications of pregnancy, childbirth and congenital malformations, deformations and chromosomal abnormalities. Part of Table 2 shows that within the most important factors apart from water pollutants, there are indicators of petroleum usage, mines outcome products and some types of wastes.

The ADSS has a wide range of methods and tools for modeling, including regression, neural networks, GMDH, and hybrid models. The function approximation agent selected the best models, which were: simple regression – 4381 models; multiple regression – 24 models; neural networks – 1329 models; GMDH – 2435 models. The selected models were included into the committee machines.

We have forecasted diseases and pollutants values for the period of four years, with a six month step, and visualized their tendencies, which, in common, and in agreement with the created models, are going to overcome the critical levels. Control under the "significant" factors, which cause impact upon health indicators, could lead to decrease of some types of diseases.

On Figure 4 there is a picture of the "View Results" window, where the user can choose the region, disease and age category, and get information about impact assessment and forecast for the factor of interest.

Conclusions

The agent-based decision making problem is a complicated one, especially for a general issue as environmental impact upon human health. We should note some essential advantages we have reached, and some directions for future research.

First, the ADSS supports decision makers in choosing the behavior line (set of actions) in such a general case, which is potentially difficult to analyze and foresee. As for any complex system, ADSS allows pattern predictions, and the human choice is to be decisive.

Type of Disease/Pollutant	Disease class
Endogenous diseases:	Certain conditions originating in the perinatal period; Congenital malformations, deformations and chromosomal abnormalities.
Exogenous diseases:	Certain infectious and parasitic diseases; Neoplasm; Diseases of the blood and blood- forming organs and certain disorders involving the immune mechanism; Endocrine, nutritional and metabolic diseases; Mental and behavioral disorders; Diseases of the nervous system; Diseases of the eye and adnexa; Diseases of the ear and mastoid process; Diseases of the circulatory system; Diseases of the respiratory system; Diseases of the digestive system; Diseases of the skin and subcutaneous tissue; Diseases of the musculoskeletal system and connective tissue; Diseases of the genitourinary system; Pregnancy, childbirth and the puerperium; Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified; External causes of morbidity and mortality.
Transport:	Number of Lorries, Buses, Autos, Tractors, Motorcycles, Others.
Water pollution:	Total volume of spilled water, Total volume of reutilized water, Dissolved oxygen (DOB, DBO5), Solids in suspension, Metals, Nitrites.
Industry:	Petroleum liquid gases, Car gasoline, Gasoline, Kerosene, Gasohol, Fuel-oil, Asphalts, Main mining products.
Dangerous wastes:	Urban waste products (categorized), Industrial waste products (categorized).

Table 1. Diseases and pollutants studied in research.

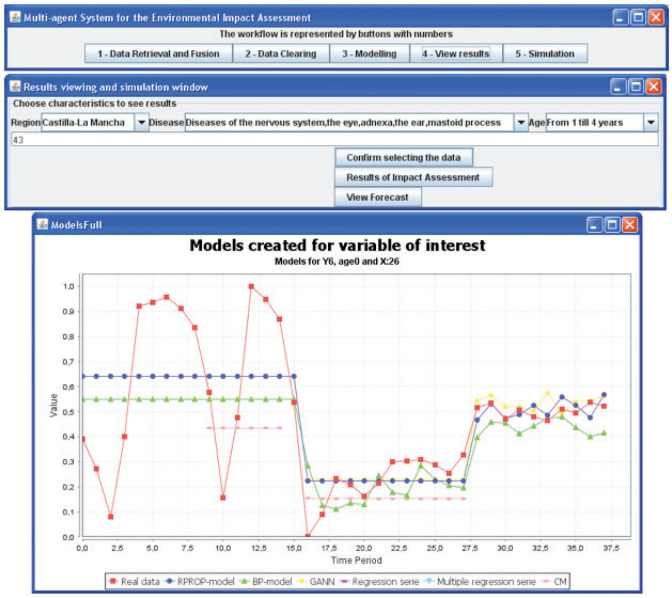
Second, as our work is very time consuming during the modeling, we are looking forward to both revise and improve the system and deepen our research. Third, we consider making more experiments varying the overall data structure and trying to apply the system to other but similar application fields.

Region	Castilla La Mancha
Neoplasm	Nitrites in water; Miner products; DBO5; Dangerous chemical wastes; Fuel-oil; Petroleum liquid gases; Water: solids in suspension; Asphalts; Non-dangerous chemical wastes.
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
Pregnancy, childbirth and the puerperium	Kerosene; Petroleum; Petroleum autos; Petroleum liquid gases; Gasohol; Fuel-oil; Asphalts; Water: DQO; DBO5; Solids in suspension; Nitrites.
Certain conditions originating in the prenatal period	Non-dangerous wastes: general wastes; mineral, constriction, textile, organic, metal. Dangerous oil wastes.
Congenital malformations, deformations and chromosomal abnormalities	Gasohol; Fuel-oil; DQO in water; Producing asphalts; Petroleum; Petroleum autos; Kerosene; Petroleum liquid gases; DBO5 in water; Solids in suspension and Nitrites.

Table 2. Part of the Table with the outputs of impact assessment.

The ADSS provides all the necessary steps for standard decision making procedure by using intelligent agents. The levels of the system architecture, logically and functionally connected, have been presented. Real-time interaction with the user provides a range of possibilities in choosing one course of action from among several alternatives, which are generated by the system through guided data mining and computer simulation. The system is aimed to regular usage for adequate and effective management by responsible municipal and state government authorities.

We used as well traditional data mining techniques, as other hybrid and specific methods, with respect to data nature (incomplete data, short data sets, etc.). Combination of different tools enabled us to gain in quality and precision of the reached models, and, hence, in recommendations, which are based on these models. Received dependencies of interconnections and associations between the factors and dependent variables helps to correct recommendations and avoid errors.


Multi-agent System for the Environmental Impact Assessment

The workflow is represented by buttons with numbers

1 - Data Retrieval and Fusion
2 - Data Clearing
3 - Modelling
4 - View results
5 - Simulation

Results viewing and simulation window

Choose characteristics to see results

Region: Castilla-La Mancha

Disease: Diseases of the nervous system,the eye,adnexa,the ear,mastoid process

Age: From 1 till 4 years

43

Confirm selecting the data
Results of Impact Assessment
View Forecast

ModelsFull

Models created for variable of interest

Models for Y6, age0 and X:26

Figure 4. *Result viewing window of the ADSS.*

Bibliography

• Haykin, S., 1999. Neural Networks: A Comprehensive Foundation. Prentice-Hall.

• Jack™ Intelligent Agents home page.
http://www.agent-software.com/shared/home/.

• Liu, L., Qian, L. & Song, H., 2006. Intelligent group decision support system for cooperative works based on multi-agent system. In Proceedings of the 10th International Conference on CSCW in Design, CSCWD 2006, pp. 574–578.

• Madala H.R. & Ivakhnenko A.G. , 1994. Inductive Learning Algorithms for Complex System Modeling, CRC Press, ISBN: 0-8493-4438-7.

• Ossowski, S., Fernandez, A., Serrano, J.M., Perez-de-la-Cruz, J.L., Belmonte, M.V., Hernandez, J.Z., Garcia-Serrano, A. & Maseda, J.M., 2004. Designing multiagent decision support system: The case of transportation management. In 3rd International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2004, pp. 1470–1471.

• Padgham, L. & Winikoff, M., 2004. Developing Intelligent Agent Systems: A Practical Guide. John Wiley and Sons.

• Padgham, L. & Winikoff, M., 2002. Prometheus: A pragmatic methodology for engineering intelligent agents. In Proceedings of the Workshop on Agent Oriented Methodologies (Object-Oriented Programming, Systems, Languages, and Applications), pp. 97-108.

• Petrov, P.V. & Stoyen, A.D., 2000. An intelligent-agent based decision support system for a complex command and control application. In Sixth IEEE International Conference on Complex Computer Systems, ICECCS'00, pp. 94–104.

• Prometheus Design Tool home page.
http://www.cs.rmit.edu.au/agents/pdt/.

• Sokolova, M.V. & Fernández-Caballero, A., 2007. A multi-agent architecture for environmental impact assessment: Information fusion, data mining and decision making. In 9th International Conference on Enterprise Information Systems, ICEIS 2007, vol. 2, pp. 219-224.

• Urbani, D. & Delhom, M., 2005. Water management policy selection using a decision support system based on a multi-agent system. In Lecture Notes in Computer Science, 3673, pp. 466–469.

• Weiss, G., 1999. Multi-agent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press.

Author. Marina V. Sokolova

Born in Russia, Kursk,. In 2000 graduated from the Engineering Department and in 2004 obtained her Ph.D. in "Administration in complex systems" from the Kursk State Technical University. Now she is a associate professor of the Safety of Vital Activity Department of the Kursk State Technical University and is a visiting professor in the University of Castilla-La Mancha, Albacete, Spain. Marina V. Sokolova is interested in complex systems, systems for decision support and multi-agent systems.