Multi-agent-based System Technologies in Environmental Issues

Marina V. Sokolova[1] and Antonio Fernández-Caballero[2]

[1] Kursk State Technical University, Kursk, ul.50 Let Oktyabrya, 305040, Russia

[2] Universidad de Castilla-La Mancha, Departamento de Sistemas Informáticos, Campus Universitario s/n, 02071-Albacete, Spain

Abstract

Environmental impact assessment (EIA) is one of the principal indicators for human health evaluation as well as for further issues in occupational medicine, public health and planning. Certain difficulties to evaluate EIA appear due to, on the one hand, the nature of contaminants and health characteristics, and, on the other, the difficulties found in data processing, such as heterogeneity of the information sources, lack of data quality and need to use numerous data mining methods. The application of multi-agent systems (MAS) to EIA helps operating with distributed decentralized information and generating general decisions and solutions. In order to facilitate the workflow of MAS planning and implementation in decision support systems (DSS) applied to EIA, a generalized multi-agent system technology (MAST) is proposed. This workflow is general enough to enable the related MAS to be used in any application area requiring decision making. We demonstrate that by only changing the domain ontology the MAS is easily oriented towards another problem area. The functional organization and the roles executed in the MAS conform to the logical sequence of the data transformation flow, which include information retrieval and fusion, data pre-processing, data mining and modeling, and simulation and decision making.

I.N. Athanasiadis et al., *Information Technologies in Environmental Engineering*, Environmental Science and Engineering, DOI 10.1007/978-3-540-88351-7_41,

Keywords: multi-agent systems, environmental impact assessment, decision support systems

1 Introduction

Recent advances and continuous changes in computer science, cybernetics and data management have penetrated into industries and business, and nowadays have become essential for many applications. Environmentalists have accustomed to use information technologies for environmental management, engineering and simulation. There are some explanations to this fact, caused by the complexity of the environmental issues, which require interdisciplinary approaches and their fusion, usage of distributed methods and techniques for their study. As shown by recent research reports, environmental problem solving benefits from the application of such methods and frameworks, and intelligent agents is a striking example of such a paradigm (Cortés and Poch 2009). The multi-agent approach seems to be an excellent technique that can help to reduce the complexity of a system by creating modular components, which solve private subtasks that constitute together the whole goal. Every agent utilizes the most effective technique for solving the subtask and does not apply the general approach, acceptable for the system in the whole, but not optimal for a concrete subtask.

In case of EIA, all the advantages of intelligent agents become crucial. EIA is an indicator, which enables to evaluate the negative impact upon human health caused by environmental pollution. Environmental pollution, as one of the factors with dominant and obvious influence, causes direct and latent harm, which has to be evaluated and simulated in order to create a set of preventive health-preserving solutions. That is why linking all the named components in one system and its study leads to the analysis of potential and present health problems, retrieval of the new ones, and to working out the in-depth view of situation development and strategies and activities oriented to situation management and control. Large amounts of raw data information describe the "environment-human health" system, but not all the information is of use though. It transforms from the initial "raw" state to the "information" state, which suggests organized data sets, models and dependencies, and finally, to the "new information" which has a form of recommendations, risk assessment values and forecasts.

After studying some related previous works (section 2), the proposal of a multi-agent system (MAS) for environmental impact assessment support system (section 3) and description of the agent teams are provided. Section 4 shows a significant example of the implementation of the obtained model using JACK Intelligent Agents (Winikoff 2005).

2 Related works

In Athanasiadis et al. (2005) it is reported about applying the software agents' paradigm to environmental monitoring informational system created by a three-layered MAS, which is aimed to provide measurements of meteorological information and air pollution data, to analyze them and to generate alarm signals. The authors present a multi-agent system for environmental monitoring, which was created by means of an intelligent platform Agent Academy. Another article related to environmental monitoring by intelligent agents may be found (Athanasiadis and Mitkas 2004).

In another article by Athanasiadis and Mitkas (2005), the authors report about the application of the agent paradigm towards the evaluation of socially-oriented advertising campaigns, aimed to affect consumers' behavior. Situation assessment has also been realized by an agent-based system dedicated to the creation of a general approach to situation assessment, which can be applied for different problem areas (Gorodetsky et al. 2005). The authors accept the JDL model (Ly et al. 2003) as a basis for situation awareness; the new approach to situation assessment learning is described and the structure of the MAS is presented as well. In another paper (Urbani and Delhom 2005), the authors present the framework of a decision support system for water management in the Mediterranean islands coupling a multi-agents system with a geographic information system. Recently, the development and experimental evaluation of an Internet-enabled multi-agent prototype called AgentStra (Li 2007) for developing marketing strategies, competitive strategies and associated e-commerce strategies has been introduced.

On the other hand, the specialists working with environmental sciences and public health store huge volumes of relevant monitoring factual information about pollutants and human health that are thought to be analyzed. Continuous processing and maintenance of the information requires essential efforts from the practitioners and professionals not only while handling and storing data, but also when interpreting them. Actually, it seems very hard to handle all the data without using data mining (DM) methods, which can autonomously dig out all the valuable knowledge that is embedded in a database without human supervision, providing a full life-cycle support of data analysis, using such techniques as clustering, classification, logical and association rule-based reasoning, and other methods, which are highly demanded for a comprehensive environmental data analysis. For instance, DM techniques for knowledge discovering and early diagnostics were utilized for early intervention in developmentally-delayed children (Chang 2006).

In Chen and Bell (2002) it is reported about the MAS named "Instrumented City Data Base Analyst", which is aimed to reveal correlations between human health and environmental stress factors (traffic activity, meteorological data, noise monitoring information and health statistics) by using wide range of DM methods, including regression analysis, neural networks, ANOVA and others. In Foster et al. (2006) the authors suggest using intelligent agents within an agent-based intelligent decision support system (IDSS) in the area of clinical management and research. The IDSS is aimed to improve the decision quality and be used in urgent cases.

A group of researchers presented a number of works dedicated to decision making support in wastewater treatment plant (WWTP) (Riaño et al. 2001). In order to provide an intelligent modeling and control for WWTP, the authors used eight kinds of autonomous agents, which support the decision-making process. The agents, which interact with WWTR and with users, are: the Monitoring Agent (MA) and the Actuation Agent (AA), the first captures some data from WWTP through the system of sensors, supplies retrospective information, displays it and supplies a basic alarm system. Another approach (Ceccaroni et al. 2004) continues and develops the idea and realization of the DSS noted above and offers environmental decision support system related with domain ontology. Authors aimed to check if incorporation of ontology and DSS, based on rule-based reasoning (RBR) and case-based reasoning (CBR) can improve decisions it generates. The article describes the OntoWEDSS architecture, which is derived from the model of the DSS for WWTP, but having an ontology embedded and a reasoning module that serves to facilitate knowledge sharing and reuse.

As all these works have demonstrated novel and convincing practical and theoretical outcomes, it seems to be important to create an agent-based decision support system (MAS) for knowledge discovery and assessment environmental tension upon the population by detail analysis of endogenous and exogenous diseases cases. Thus, the main practical aim of the paper is to introduce a fully developed situation assessment agent-based system, monitoring the environment pollution and following the correspondent changes in human health, generating a set of alternatives for successful and sustainable situation management.

3 The general approach for multi-agent system creation

3.1 Information change

Large amounts of raw data information describe the “environment-human health” system, but not all the information is of use though. For the situation modeling we orient to factual and context information, presented in data sets and we use intelligent agents to extract it. So, the information transforms from the initial “raw” state to the “information” state, which suggests organized data sets, models and dependencies, and finally, to the "new information" which has a form of recommendations, risk assessment values and forecasts. The way the information changes, is given on Fig. 1:

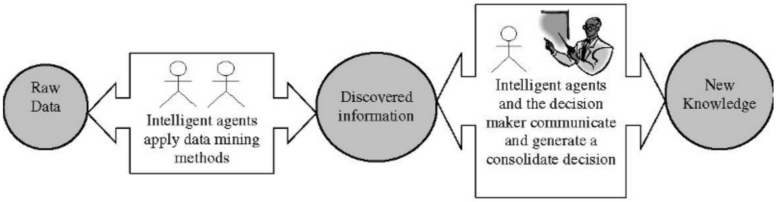


Fig. 1. The information transformation, which changes from weakly organized and heterogeneous view into knowledge

The hidden information is discovered by agents, but for new information construction not only intelligent agents, but the knowledge of a decision maker or expert is involved. The agent-based decision support system, which Sokolova and Fernández-Caballero (2009, 2008, 2007) are creating and updating, provides these information changes.

The process of information change, shown on Fig. 1, corresponds to the MAS lifecycle flow, which in case of MAS counts the following steps:

1. **Domain Analysis** - is related to the analysis of the project idea, problem definition, extraction of aims, creation of goal trees, sequencing of tasks and subtasks to be solved. This stage also implies the domain ontology creation, which covers the problem area, the set of relations between the concepts and the rules to incorporate new knowledge. The experience of domain area experts is required on this stage.
2. **Software Elements Analysis** - this stage also deals with private ontologies creation; but now ontologies are created for the system and its elements. The sets of goals and tasks are related to the sets of system functions (roles), required resources (commonly in form of informational files), interactions, and so on.
3. **Specification** - is the written description of the previous stages, which results in system meta-ontology creation.
4. **Software Architecture** - implies the abstract representation of the system to meet the requirements. The software architecture includes interfaces for human-computer communication.
5. **Implementation** - is the iterative process of program creation.
6. **Testing** - program testing under normal and/or critical conditions.
7. **Deployment and Maintenance** - program application and support until the software is put into use. Sometimes some training classes on the software product are made.
8. **End of Maintenance** - is the final stage of the software life cycle.

The workflow of tasks, which has to be solved for information integration (see Fig. 1) contains four sequential states of data transformation: (1) initial heterogeneous data sources, (2) storages of extracted data, (3) mapped (fused) meta-data, (4) shared global ontology of the problem area (domain ontology) and three flows/processes, which provide and organize the transformations: (1) data retrieval and extraction, (2) data mapping (fusion), (3) filling in the ontology of the problem area (domain ontology).

3.2 Multi-agent system organization and architecture

We have implemented an agent-oriented software system, dedicated to environmental impact assessment. The system receives retrospective statistical information in form of direct indicator values -water pollution, solar radiation -and in form of indirect indicator values -types and number of vehicles used, energy used annually and energy conserved, types and quantity of used fuel, etc. The indirect indicators are utilized in accordance with ISO 14031 "Environmental Performance Evaluation" standard in order to estimate air and soil pollution (ISO/IEC). The population exposure is registered as number of morbidity cases, with respect to International Statistical Classification of Diseases and Related Health Problems, 10th review (ICD-10).

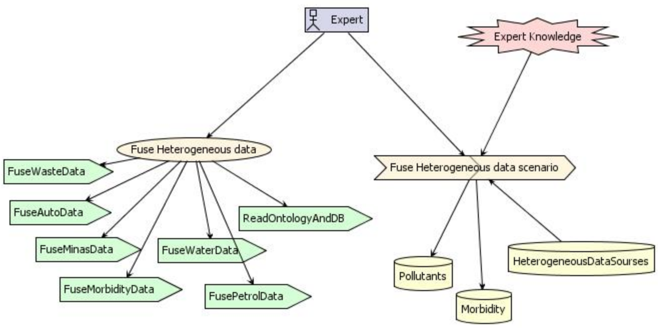
In order to provide the system design we use the Prometheus Development Tool (PDT), which opens a wide range of possibilities for MAS planning and implementation. These are the system architecture, the system entities, their internals and communications within the system and with outer entities. The most important advantages of PDT are an easy understandable visual interface and the possibility to generate code for JACK™ Intelligent Agents, which is used for MAS implementation, verification and maintenance. The process of multi-system creation, in accordance with the Prometheus methodology, consists of three phases, which are:

1. System specification, aimed to the identification of multi-agent system entities, such as actors, system goals, scenarios, actions, percepts and roles.
2. High-level (architectural) design, which is centered in the description of agent role coupling, general system structure and interaction protocols.
3. Detailed design, in which each agent is described in detail in terms of capabilities, events, plans and data.

The further analysis of the system has resulted in obtaining and describing the system roles and protocols. There, 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.

The goals drawn in Fig. 1 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.

Being implemented by means of the Prometheus Design Tool, the Analysis Overview Diagram of the MAS enables seeing the high-level view composed of external actors, key scenarios and actions (see Fig. 2). The proposed MAS presupposes communication with two actors. One actor is named as “Expert” and it embodies the external entity which possesses the information about the problem area - in more detail, it includes the knowledge of the domain of interest represented as an ontology - and delivers it through protocol ReturnEI to the MAS. The data source, named “The CS Results” stores the results of the simulation and forms a knowledge base (KB). Through the *Simulate Models* scenario user interacts with the KB, and gets recommendations if they have been previously simulated and stored before, or creates and simulated the new ones. As a result of the interaction within the *FuseHeterogeneousData* scenario, the raw information is being read, and it is shown as “Heterogeneous Data Sources” data storage, and there are “Pollutants” and “Morbidity” data sources are created.



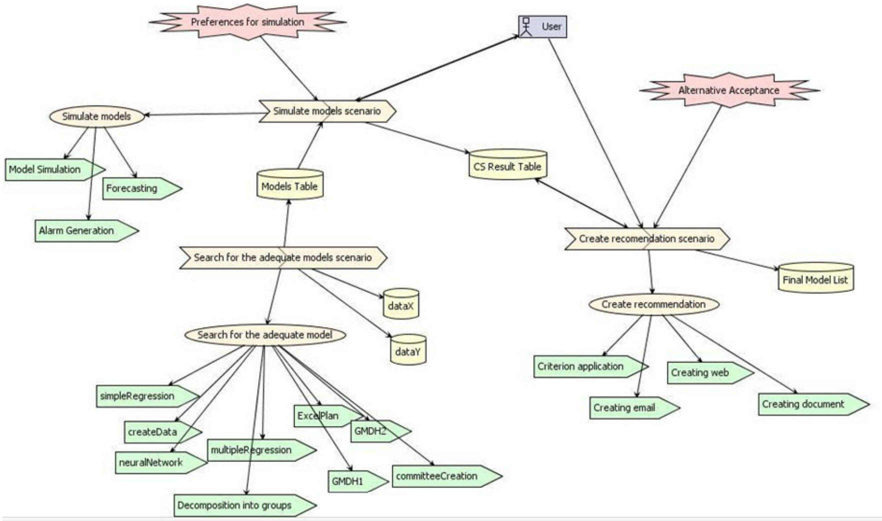


Fig. 2. The Prometheus Diagram of MAS interaction with actors

The second actor, named "Decision Maker", is involved in an interactive process of decision making and choosing the optimal alternative. This actor communicates with agents by message passing through protocol ReturnSUI, stating the model, simulation values, prediction periods, levels of variable change, etc. It accepts the best alternative in accordance with its beliefs and the MAS. The flow of works, which are essential for decision making, include three scenarios: the Simulate models scenario, the Create recommendation scenario and the Search for the adequate model scenario; and three goals, which related to every scenario and have similar names. Each goal has a number of activities, and within each scenario are used, modified or created informational resources in form of data sources.

In addition to the accepted MAS architecture and in order to gain time of the recommendation generation process and optimize interactions between agents, we used local agent teams, which coordinate and supervise task execution and resource usage. Agent teams have permitted to synchronize the work of the system, plans execution in a concurrent mode and strengthen the internal management by local decision making. As we use four agent teams within the system: two within the first level, and one team on the second and third level, and each "main" agent plays several roles.

During the system work cycle, agents manipulate with diverse input and output information flows: data transmission protocols, messages, input and output data, etc. These information sources differ by the "life time": it can be permanent and temporary, by the assessment levels – some can be used modified or deleted by agents, the decisions about others have to be taken by a system user. For example, the DPA administrates agents of its team, the agents have to execute sequentially and carry out data clearing and preprocessing methods. So, in this case the DPA has to operate as a planning agent, on the one hand, and has to pool the results of the subordinate agents' execution, on the other hand.

4 Description of MAS agents

4.1 The Data Aggregation agent

DAA has a number of subordinate agents under its control; these are the Domain Ontology agent (DOA) and the *fusion agents*: the Water Data Fusion agent (WFA), the Petroleum Data Fusion agent (PFA), the Mining Data Fusion agent (MFA), the Traffic Pollution Fusion agent (TFA), the Waste Data Fusion agent (WDFA) and the Morbidity Data Fusion agent (MFA).

First, the DAA sends the message *ReadOntology* to the DOA, which reads the OWL-file, which contains information about the ontology of domain, and make it available to the DAA. The DOA terminates its execution, sending the message *OntologyIsBeingRead* to the DAA. Next, the DAA sends the message *Start Fusion* to the fusion agents, which initiate to execute. When start to execute, each fusion agent searches for the files, which may contain information about the concept of its interest. Each fusion agent works with one or few concepts of the domain ontology: WFA searches for the information about water contaminants and their properties, PFA – about the use of petroleum and related concepts, MDF retrieves data about the contamination related to mining industry activity, the WDFA retrieves data about wastes and its components, the TFA – data about transport vehicles activity, and the MFA – data about morbidity and their properties. When it finds the information file, the agent retrieves the information about the concept and its values, and changes their properties (in order to get rid of heterogeneity and to homogenize information) and sends it to the DAA, which pools retrieved information together. Finally, DAA fills the domain ontology with data, and put data into a standard format. After that, the data files are ready to be preprocessed, and the DAA through the protocol ReturnDF says to the DPA, that data is fused and preprocessing can be started.

4.2 The Data Preprocessing agent

DPA provides data preprocessing and has a number of subordinate agents which specialize in different data clearing techniques: Normalization agent (NA), Correlation agent (CA), Data Smoothing agent (DSA), Gaps and Artifacts Check agent (GAA). They perform all data preprocessing procedures, including outliers and anomalies detection, dealing with missing values, smoothing, normalization, etc.

DPA starts to execute as soon as receives a triggering message from DAA. The main function of the DPA is to coordinate the subordinate agents and decide when they execute and in which order. Starting its execution, DPA sends the *StartDataConsistenceCheck* message, which triggers the GAA, which eliminates artifacts, searches for the double values and fills the gaps. Having finished execution, GAA sends to DPA a message. Then, DPA through the message *StartSmoothing* calls for DSA, which can execute exponential and weighted-average smoothing and terminates sending *SmoothingIsFinished* message to DPA. Then, NA and CA are called for in their turn. The outputs of the DPA work are: data, ready for further processing and modeling, and additional data sources with correlation and normalization results.

4.3 The Function Approximation agent

FAA has a hierarchical team of subordinate agents, which serve to support the roles: "Impact Assessment", "Decomposition" and "Function Approximation". FAA has under its control a number of *data mining agents*: the Regression agent (RA), the ANN agent (AA), and the GMDH agent (GMDHA), which work in a concurrent mode, reading input information and creating models. Then, if any agent from this group finishes modeling, it calls for the Evaluation agent (EA), which evaluates received models, and return the list of the accepted ones, the others are banned and deleted. The FAA pools the output of the agents work, creates the list with the accepted models and then, once RA, AA and GMDHA finished their execution, calls for the Committee Machine agent (CMA), which creates the final models in form of committees for each of the dependent variables, and saves them.

The FAA working cycle is the following. FAA sends StartDecomposition message and waits until DA finishes its execution. Then, having received the StartDataMining message, the data mining agents start execution in a concurrence mode. Each of them has plans with particular tool, and in case of AA, it has neuralNetwork and evaluateImpactAssessment plans, where the first plan is oriented to artificial neural network (ANN) creation and training, and the second plan aims to evaluate the environmental impact by the means of ANN with determined structure and characteristics. EA is called by each of the data mining agents to evaluate created models, and to check the adequacy of the model to the experimental data. EA is triggered by the StartEvaluation message from a data mining agent, and, wherever it is not busy, starts to execute. Having terminated the execution, it is ready to receive tasks and handle them. CM is the last to be called by FAA, as CM creates final hybrid models for every dependent variable. Each hybrid model is based on the previously created and evaluated models from the data mining agents, and uses the data sources created by them: Models Table, IAResults.

4.4 The Computer Simulation agent

The CSA interacts with user and performs a set of task within Computer Simulation, Decision Making and Data Distribution roles. It has the agent team, which includes Forecasting agent (FA), Alarm agent (AmA) and ViewAgent (VA).

The CSA execution cycle starts with asking for user preference, to be more precise, for the information of the disease and pollutants of interest, period of the forecast, and the ranges of their values change. Once the information from user is received, CSA sends a message *SimulateAlternative* to FA, which reasons and executes one of the plans, which are Forecasting, ModelSimulation, and CriterionApplication. When the alternative is created, CSA sends the *StartAlarmCheck* message to AmA. The AmA compares the simulation and forecast data from the FA with the permitted and alarm levels for the correspondent indicators. If they exceed the levels, AmA generates alarm alerts.

5 Results and conclusions

The MAS 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. The ViewAgent displays the outputs of the system functionality and realize interaction with the system user. 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.

At present, there is a full software running version of our proposal. And, 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.

The MAS 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.

The MAS 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. All this information is now being transferred to the regional experts who are going to evaluate the performance of the system.

As a result, the MAS 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. To conclude with, it is necessary to reason about our future plans regarding the work. As the work appeared to be 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.

Acknowledgements

This work is supported in part by the Spanish CICYT TIN2007-67586-C02-02 grant and the Junta de Comunidades de Castilla-La Mancha PBI06-0099 grant.

References

Athanasiadis IN, Mentes AK, Mitkas PA, Mylopoulos YA (2005) A hybrid agent-based model for estimating residential water demand. Simulation 81(3):175–187

Athanasiadis IN, Mitkas PA (2005) Social influence and water conservation: An agent-Based approach. Computing in Science and Engineering 7(1):65-70

Athanasiadis IN, Mitkas PA (2004) An agent-based intelligent environmental monitoring system. Management of Environmental Quality 15(3):238-249

Ceccaroni L, Cortés U, Sànchez-Marrè M (2004) OntoWEDSS: Augmenting environmental decision-support systems with ontologies. Environmental Modelling & Software 19(9):785–797

Chang CL (2006) A study of applying data mining to early intervention for developmentally-delayed children. Expert Systems with Applications 33(2):407–412

Chen H, Bell M (2002) Instrumented city database analysts using multiagents. Transportation Research Part C 10:419–432

Cortés U, Poch M (2009) Advanced Agent-Based Environmental Management Systems. A Birkhäuser book

Foster D, McGregor C, El-Masri S (2006) A survey of agent-based intelligent decision support systems to support clinical management and research. First International Workshop on Multi-Agent Systems for Medicine, Computational Biology, and Bioinformatics

Gorodetsky V, Karsaeyv O, Samoilov V (2005) Multi-agent and data mining technologies for situation assessment in security-related applications. Advances in Soft Computing, pp. 411–422

ISO/IEC 12207 home page. http://www.iso.org/iso/

Li S. (2007) AgentStra: An Internet-based multi-agent intelligent system for strategic decision-making. Expert Systems with Applications 33(3):565–571.

Ly TC, Greenhill S, Venkatesh S, Pearce A (2003) Multiple hypotheses situation assessment. Proceedings of the Sixth International Conference of Information Fusion, vol. 2, pp. 972–978

Riaño D, Sànchez-Marrè M, R.-Roda I (2007) Autonomous agents architecture to supervise and control a wastewater treatment plant. Lecture Notes in Computer Science 2070:804–811.

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

Sokolova MV, Fernández-Caballero A (2008) Agent-based decision making through intelligent knowledge discovery. Lecture Notes in Computer Science 5179:709–715

Sokolova MV, Fernández-Caballero A (2007) An agent-based decision support system for ecological-medical situation analysis. Lecture Notes in Computer Science 4528:511–520

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

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