Journal of Computer Science 2 (10): 758-764, 2006
ISSN 1549-3636

A Cooperative Intelligent Decision Support System for Contingency Management

Abdelkader ADLA
IRIT, University Paul Sabatier of Toulouse, France

Abstract: Traditional Decision Support Systems (DSS) give not enough possibilities of intervention to the user. These systems are reduced to an insular and very technical state in which the objective is not support decision but to dump data on the screen in the hope that the user will know what to do with. In complex situations, decision is not structured and it becomes primordial to design intelligent and cooperative systems allowing a joint resolution of problem based on dynamic sharing of the tasks between the user and the system and according to problems to be solved. In this perspective, we propose a cooperative architecture for intelligent decision support system. The framework embeds expert knowledge within the DSS to provide intelligent DSS using collaboration technologies by putting the decision maker effectively in the loop of the decision process. To this end, we used a structure based on domain and task conceptual modelling. Applicability and relevance of this model are illustrated through a case study where the system and the operator cooperate in decision problem which consists of identifying boiler defects, diagnosing and suggesting actions of cure.

Keywords: Decision process, cooperation, conceptual model, cooperative DSS

INTRODUCTION

Decision making is considered one of the most critical activities done in organizations[18]. To support this complex process for individuals, a variety of independent, standalone information systems called Decision Support Systems (DSSs) have been developed in the two last decades. They are defined as computer-based tools used to support complex decision-making and problem solving[17] . A complementary way of looking at DSSs is associated with the role and functions that DSSs have to fulfil[14], as seen from a user's perspective: they assist managers in their decision processes in semi-structured tasks; they support and enhance rather than replace managerial judgements; they improve the effectiveness of decision-making rather than its efficiency; they attempt to combine the use of models or analytical techniques with traditional data access and retrieval functions; they specifically focus on features that make them easy to use by non-computer people in an interactive mode; and they emphasize the flexibility and adaptability to accommodate changes in both the approach of the decision maker and the environment in which he acts.

At present, due to complex economic, social and political structures, the need for decision making techniques and support is greater than ever before. This is due to the complexity of business relationships, the greater number of decision makers and organizations that are involved in the decision process, online access to multiple external information sources, and the decreasing in the time allowed for decision making.

Due to that complex nature of "the problem", including lack of consistent and complete data and uncertainties, the process of finding optimized decisions cannot be limited to solving of mathematical optimization problems or performing complex simulations. Although many industrial systems have been realized, a feeling of doubt about their successfulness has emerged these last 15 years. This is due essentially to the fact: the user's ability is not taken positively into account. The relevant systems developed in the past, are mainly restricted to assist users using programs which limit users' access to computerized models and support systems. Indeed, the current decision support systems do not integrate the user in the whole decision making process. These systems predefine the roles of the agents giving the role of pure resolution to the system while the user is confined in data entry tasks, or even resolution of conflicts[1].

Moreover, one of the most important aspects is that, in decision-making processes, the man takes advantage on the machine, contrary to structured problems. To solve problems requires calling intuition and know-how of the decision-maker which becomes the preponderant element of the couple Man/machine. The system must be able to play collaborator's role with the decision-maker, that is, to know his intentions and the context of the decision problem, to be able to give an action coordinated with that of the decision-maker.

Corresponding Author: A. ADLA, IRIT University Paul SABATIER of Toulouse, France

This study aimed to create a Cooperative Intelligent Decision Support System (CIDSS). In particular, we envision leverage further traditional DSS by embedding expert knowledge with the DSS and implementing the DSS using collaboration technologies. Embedding expert knowledge with the DSS provides intelligent decision support, and implementing the intelligent DSS using collaboration technologies puts the decision maker effectively in the loop of such DSS.

The rest of the paper is organized as follows: First we take a literature survey of some related works in section 2. Then we propose a cooperative architecture for an intelligent decision support system in section 3. We also present an example application with some implementation issues to illustrate the feasibility of the idea in section 4. Finally, we conclude with a summary and future research direction in section 5.

LITERATURE SURVEY

Decision support systems (DSS) emerged in the 1970. It is defined as a computer-based system designed to actively interact with an individual decision maker in order to assist him to make better decisions based on information obtained[9,20]. The decision process is broadly defined as a bundle of correlated tasks that include: gathering, interpreting and exchanging information; creating and identifying scenarios, choosing among alternatives, and implementing and monitoring a choice[4,18]. Briefly, the decision process refers to some techniques or processing rules aiming at structuring the context, timing or content of communication.

DSS was designed to solve ill or non-structured decision problems[5]. Problems where priorities, judgements, intuitions and experience of the decision maker are essential, where the sequence of operations such as searching for a solution, formalization and structuring of problem is not beforehand known, when criteria for the decision making are numerous, in conflict or hard dependent on the perception of the user and where resolution must be acquired at restricted time.

A number of frameworks or typologies have been proposed for organizing our knowledge about decision support systems[16]. The two most widely implemented approaches for delivering decision-support are Data-Driven and Model-Driven DSSs. Data-Driven DSSs help managers organize, retrieve, and synthesize large volumes of relevant data using database queries, OLAP techniques, and data mining tools. Model-Driven DSSs use formal representations of decision models and provide analytical support using the tools of decision analysis, optimization, stochastic modelling, simulation, statistics, and logic modelling. Three other approaches have become more wide spread and sophisticated because of collaboration and web technologies: Communication-Driven DSSs rely on electronic communication technologies to link multiple decision makers who might be separated in space or time, or to link decision makers with relevant information and tools. Knowledge-Driven DSSs can suggest or recommend actions to managers. Finally, Document-Driven DSSs integrate a variety of storage and processing technologies to provide managers document retrieval and analysis. The basic structure of classic DSS comprises components for: (1) database management capabilities with access to internal and external data, information and knowledge; (2) powerful modelling function accessed by a model management system; and (3) user interface design that enable interactive queries, reporting and graphic functions.

A regular decision support system helps decision-makers to manipulate data and models. It does not play the role of an intelligent assistant to the decision maker. IDSS is needed and is economically feasible for generic problems that require repetitive decisions. Intelligent decision support systems (DSSs) are interactive computer-based systems that use data, expert knowledge and models for supporting DMs in organizations to solve semi structured problems by incorporating artificial intelligence techniques[21]. They draw on ideas from diverse disciplines such decision analysis, artificial intelligence, knowledge-based systems and systems engineering. There may be different ways to make a DSS more intelligent; the most frequently suggested method is to integrate a DSS with an ES[21]. Recent applications of intelligent DSS include IDSSFLEX for the design and evaluation of flexible manufacturing systems[22], Markex for product development decisions[10] and an IDSS for selecting IT applications that match company strategy[22]. A review of intelligent DSSs that combine mathematical modelling with knowledge-based systems can be found in[16].

However, several research communities have examined aspects of intelligent system behaviour, but often these contributions are merely parts of a solution. For example, many of the models, algorithms and knowledge-based reasoning capabilities that have been generated through artificial intelligence[2] research have led to important contributions to the intelligent systems approach advocated here. Other algorithmic work generated out of systems engineering[7], and optimization techniques[6] have also contributed greatly. However, even though these systems have failed in use due to their brittleness[2,5], complexity[15], and poor interfaces to other systems and to the people having to use them.

Therefore, we will view the individual DSS as a set of computer based tools integrating expert knowledge and using collaboration technologies that provide decision-maker with interactive capabilities to enhance his understanding and information base about options through the use of models and data processing, and collaborate with him. The machine interaction is not restricted to a simple message exchange or a strict dialog mode but especially is extended and regarded as an aid to carry out concrete tasks in real world. Putting the human expert user effectively in the loop of such decision support system represents the major guarantee of mastering efficiently the inherent complexity of the problems. It's quite accepted now to consider the user and machine as a whole instead of as separate entities.

COOPERATIVE INTELLIGENT DECISION SUPPORT SSYSTEM

One of the most important aspects is that, in decision-making processes, the man takes advantage on the machine contrary to structured problems. To solve problem requires calling intuition and know-how of the decision-maker which becomes the preponderant element of the couple Man / machine. The system must be able to play collaborator's role with the decision-maker, that is, to be able to give an action coordinated with that of the decision-maker..

Man-machine cooperation is issued from man-machine interaction (HIM) and computer supported collaborative work (CSCW)[22]. It is, in a way, about enriched IHM. For systems designers, the objective consists in endowing the machine of additional capacities to guide the user in its problem solving process. To an effective cooperation between a system and an end-user, adequacy between presentations and strategies used by both cognitive systems is critical. This adequacy between presentations gives rise to a cooperation defined as a mutual assistance to lead to an optimum decision[8]. Therefore, a man-machine cooperation situation is characterized by the implementation of a coordination mechanism where a group of activities are mutually supported with the intention of accomplishing a common objective while sharing skills, know-how and knowledge of the partners.

To cooperate, particularly means distributing tasks to be carried out among both agents (system and user). According to Millot [14] , collaboration man-machine can be envisaged in two ways: (1) vertical cooperation where the operator (the man) has the responsibility for the whole problem. It follows all stages of its decision-making step as he can solicit the system to possibly assist him in some tasks, (2) horizontal cooperation where the tasks are distributed dynamically among the operator and the system. The operator follows all stages of his decision-making step, but only on a set of sub-problems, the others being carried out by the system. Sharing tasks is a condition to implement cooperation between the two agents. The task that is the subject of cooperation must be decomposed in consistent subsets. The task distribution among the two agents is dynamically made, according to the performances of the couple man/machine and of the workload of the user. Competences of the user and the system are sometimes complementary, sometimes "redundant". In the latter case, user and system are often able to play the same role. The choice question of the appropriate agent which will have to play one role settles therefore. According to the context, different indications could be made to direct this choice. The set of indications on the manner to allocate different roles to the agents defines the cooperation modes.

For the implementation of man/system cooperation, we use a structure based principally on conceptual models of expertise: Domain Conceptual Model Task Conceptual Model (of the application for the system, but also of the users). The proposed architecture, see figure (1), for the design of a cooperative intelligent decision support system extends that of Soubie[19] developed for cooperative knowledge-based systems. Several models are suggested: Conceptual Model of the Application (ACM), Conceptual Models of the Users (UCM) and Control Model of resolution (CM). The ACM is a set of three bases: Database (DB), Model base (MB), and Knowledge base (KB). Therefore, the model of the application integrates a representation of the domain knowledge (Domain Conceptual Model) and a representation of the task expertise (Task Conceptual Model) based on the task-method paradigm. Thus we can express the Application Conceptual Model in terms of tasks, methods and domain knowledge.

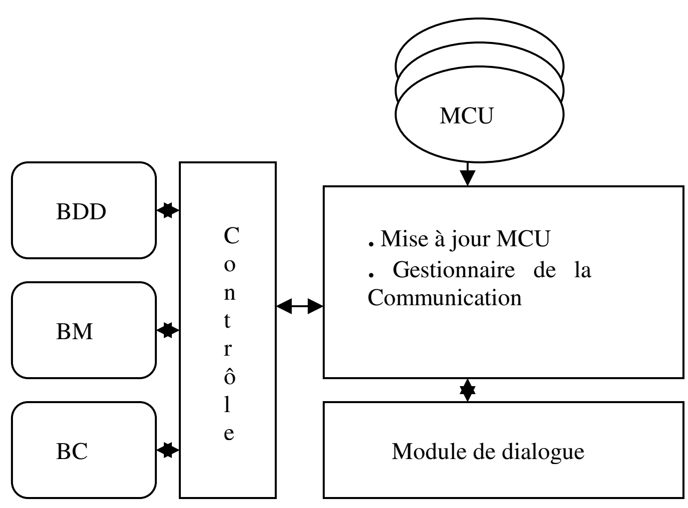


Fig. 1: Architecture of a Cooperative Intelligent Decision Support System (CIDSS)

The domain conceptual model: The domain knowledge contains concepts of the domain represented in an object-oriented diagram. Knowledge of the domain represents different concepts manipulated by the methods or used in the definition of the tasks.

The Task conceptual Model: Knowledge of reasoning represents concepts of problem tasks and problem solving methods that can be represented in a hierarchy of tasks and associated methods.

A task represents all problems and sub-problems to be solved senseless a priori of the manner of solving them. It is characterized by a purpose, a set of pre-conditions, and a set of associated methods. To every task are associated the methods which are a priori declared as the best adapted to achieve it.

A method represents all the resolution mechanisms or the know-how which it is possible to implement to achieve different tasks. Two categories of methods are available:

. An "action plan" type method which is a composition of sub-tasks;

. A "feasible" type method which consists of heuristics (knowledge base), a code or a procedure (model base), requests (database) or else ask user method. It is characterized by a purpose, a set of pre-conditions, a set of results, and the solving process.

Control and cooperative resolution: Different methods to achieve a task can be envisaged. Given a task, the system can then choose a method dynamically to achieve it. In order to do that, given the name of the task to be solve (wording of problem), the system constructs an action plan to be carried out (a sub-graph of tasks-methods hierarchy).

The problem solving mechanism is based on four levels: (a) task and method modelling primitives, (b) abstract notions (candidate methods, preferred methods, etc.), (c) high-level actions (identify candidate methods, identify preferred methods, etc.), and (d) control of high level actions, see figure (3). The inference engine based on this structure guides cooperatively and intelligently the decision maker in his decision problem solving.

APPLICATION

Presentation: The manufacturing process of the oil plant (GLZ) is split into two subdivisions: Utility subdivision and Process subdivision. The Utility subdivision is constituted of Pumps, Desalination Unit, boilers, Turbo-generators and Air Compressors while the Process subdivision concerns the tasks of manufacturing of liquefied Gas. This subdivision is composed of 6 strings where a string is group of equipments. Every string contains 10 sections which are going to be used to liquefy gas.

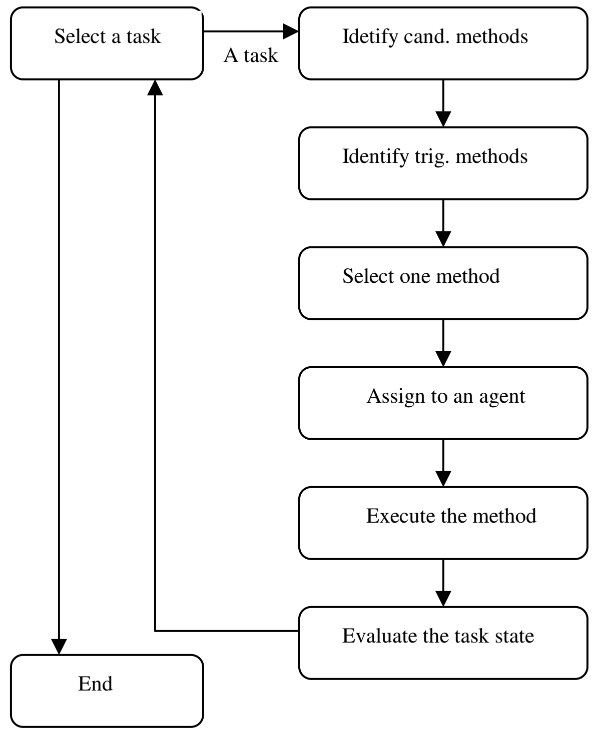


Fig. 3: Dynamic selection mechanism of tasks and methods

The management system of the boiler combustion is one of the most critical systems for the good functioning of the plant and has a high impact on the methods of cogitation and apprehension of various problems related to maintenance. The exploiting staff is often confronted with situations that impose a quick reaction of decision-making. This requires consequent human and material resources and adapted skills.

Different sensors are set up to detect anomalies at different stages of the process. Different sensors are set up to detect anomalies at different stages of the process. Breakdown can be automatically signposted by means of an alarm or intercepted by the exploiting engineer (case of defectiveness of the sensor where no alarm is triggered off but the boiler does not work). If there is a defect, an alarm will be triggered off. In case an alarm is signposted to the operator: the flag (the reference given to every alarm) is pointed out on the board (control room). It acquaints with an alarm and locates the defect. To solve this problem, diagnosis and actions of cure are generated by the system. Otherwise, a breakdown is directly raised by the operator (not triggered off alarm). This scenario intervenes when a sensor defect doesn't allow to automatically signpost the breakdown. In this case, the operator must explore a large research space of potential defects with a series of tests. In both cases, the operator tries first to solve the problem by using CIDSS.

The Domain knowledge Modelling: The domain knowledge relating to the problem is represented by the following diagram, see figure (4):

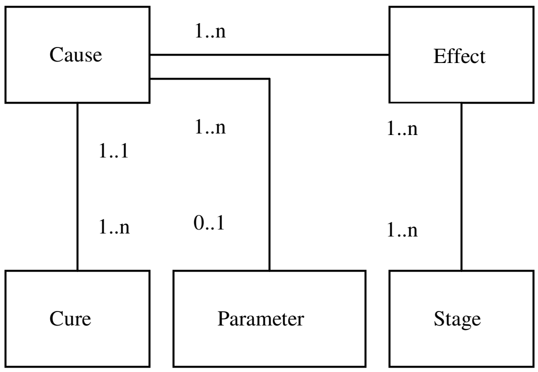


Fig. 4: conceptual model of domain

Stage: N ° stage, Designation. Effect: Encode effect, Description.Cause: Encode reason, Description. Parameter: Encode parameter, unit. Cure: Encode cure, Description.

The Task Modelling: The Task Conceptual Model is based on the task-method paradigm. A problem is represented by a hierarchy of tasks and their associated methods see figure (5). "Feasible" methods can be procedures, heuristics, codes, SQL requests or simply ask the operator to perform an action. All these methods are not exclusive to achieve a task and may be prioritized.

To implement these concepts, we defined a class named "model"; and Resolution is represented by "Resolution" function.

Example of session: We experiment our system on a case of boiler breakdown. It consists of discerning the defects while boiler functioning, diagnosing the defects, and proposing one or several appropriate actions of cure.

If a breakdown occurs, this can be:

. Automatically signposted to the operator by means of a triggered off alarm, the flag (the reference given to every alarm) is pointed out on the board (control room). It acquaints with a particular alarm and allows the operator to locate the defect from the database. Once the defect identified, diagnosis and actions of cure are automatically generated by the system and validated by the user, or

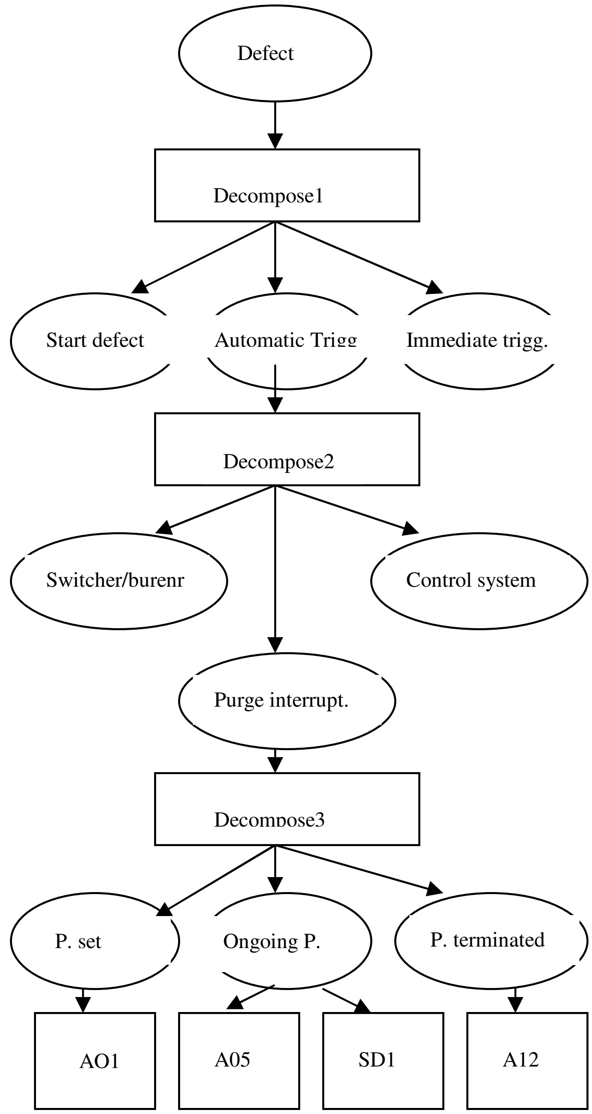


Fig. 5: Task hierarchy of boiler functioning

(Decompose1, 2, 3: Decomposition methods; A01, A05, A12, SD1: "feasible" methods).

. Intercepted by the operator (case of defectiveness of the sensor where no alarm is triggered off but the boiler does not work), the operator must explore a large research space of potential defects with a series of tests. Here also the system plays an important role to optimize the action plan to be carried out.

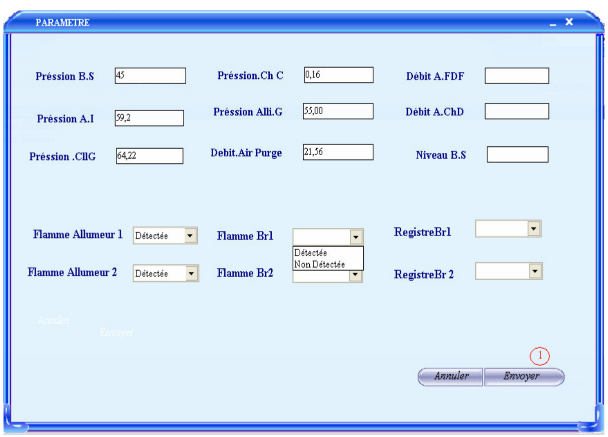
Before starting to solve this problem (diagnosis and actions of cure), the operator updates user model which meets his competences (a set of tasks and methods that his able to carry out), and initializes the cooperation mode (e.g.: decision and critic roles for the operator, aid and execution roles for the system).

After this system initialization, the cooperative resolution process may starts: The operator words to the system the task to be solved, then the system identifies the task in the task hierarchy, elaborate a plan of actions to carry out (a sub-graph of the task-method hierarchy). The action plan is elaborated according to parameter values (speed, temperature, oil level,) submitted by sensors or indications introduced by the operator. Two cases may occur:

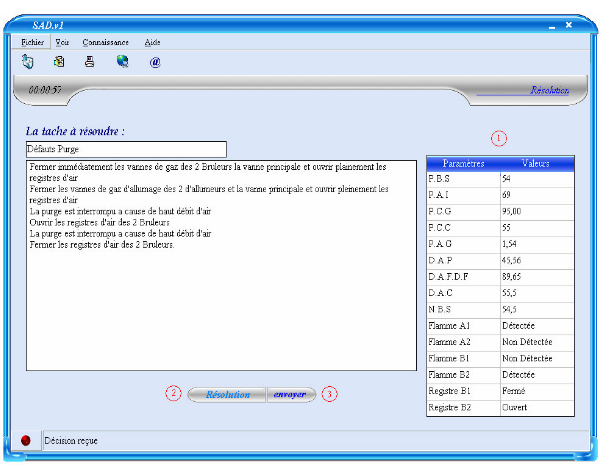
(i) “Final” method is « ask-user », task is therefore appointed to the user to solve it manually. An interaction settles between the system and the user for the introduction and the presentation of data and results.

(ii) “Final” method is "procedure", "SQL request" or "heuristic", the system checks from the user model, if the method is "redundant" (may be carried out by the system as well as by the user): According to the task and the cooperation mode, the task is appointed to the suggested agent (system or operator), otherwise the task is solved by the system.

Task and method definition



Parameter acquisition



Task resolution

CONCLUSION AND FUTURE WORK

We defined in this paper, a cooperative architecture for intelligent decision support system which allows to take into account competences of the user and so to integrate him into the problem solving process. Putting the human operator effectively in the loop of such decision support system represents the major guarantee of mastering efficiently the inherent complexity of the problems. The prototype developed is at a validation phase.

The structure used creates a problem solving environment based on models. The definition of new concepts and new strategies is possible, it is therefore progressive. Application to other case studies is enabled by modelling the domain and the task of the problem at hand and updating the Application Conceptual Model (ACM).

One perspective of this work is to integrate the proposed architecture and expand the research and consequently development of the prototype in a distributed field to allow dispersed group decision making. We consider here the paradigm of distributed decision-support systems, in which several decision-makers who deal with partial, uncertain, and possibly exclusive information must reach a common decision. The prospective development of the Distributed CIDSS is based on two types of cooperation: (1) Man-machine cooperation allowing every decision maker to solve problem and to generate an alternative, (2) Mediated man-man cooperation allowing the group of decision makers and the facilitator to make collective decision. This kind of cooperation uses a machine as an intermediate communication medium.

REFERENCES

1. ADLA, A., 2006. Systèmes d’aide à la décision coopératifs. In Proc on IHM, Montréal, Canada.
2. Bannon, L., 1997. Group Decision Support Systems: An analysis and critique. In Proc. Intl. Conf. on Information Systems, pp 526-539, volume 1, Cork.
3. Clarke, A. and Smith, M.G., 1993. A cooperative computer based on the principles of human cooperation. Intl J. of Man-Machine Studies, 1993.
4. DeSanctis, G and Gallup, B., 1987. A foundation for the study of group decision support systems. Management Science, Vol 13(12), pp 1589-1609.
5. Holtzman, S, 1989. Intelligent decision systems. Addison Wesley.

6. Huber, G., 1984. Issues in the design of group decision support systems. MIS Quarterly Vol (3), pp 195-204.
7. Jelassi, M.T. and Beauclair, R.A., 1987. An Integrated Framework for Group Decision Support Design. Information and Management, Vol 13(2), pp 143-153.
8. Karacapidilis, N.I. and Pappis, C, 1997. A framework for group decision support systems: combining AI tools and OR techniques. European Journal of Operational Research, Vol 103, pp 373-388.
9. Keen, P. and Scott-Morton, M., 1978. Decision Support Systems: an organizational perspective. Addison-Wesley Publishing.
10. Klein, M. and Methlie, L.B., 1990. Expert Systems: A Decision Support Approach with applications in management and finance. Addison Wesley Publishing Compagny.
11. Krivin, J.P. et al., 1994. Systèmes à base de connaissances coopératifs, rôles et modes de coopération. In Pavard (Ed) Systèmes coopératifs: de la modélisation à la conception. Octares, Toulouse.
12. Lévine, P. and Pomerol, J., 1989 Systèmes interactifs d'aide à la décision et systèmes experts. Editions Hermès.
13. Marakas, G.M., 2003. Decision support systems in the 21st century. Prentice Hall.
134.Millot, P., 1987. Task allocation between human and machine, design and evaluation, State Thesis, University of Valenciennes.
15. Nunamaker, J. et al., 1997. Lessons from a dozen years of group support systems research. Journal of MIS Vol 13(3), pp 163-207.
16. Power, D.J., 2000. Supporting Decision-Makers: An Expanded Framework. World Wide Web, http://dssresources.com/papers/supportinggdm/sld001.htm, version 1.0.
17. Shim J.P., et al., 2002. Past, present and future of decision support technology. Decision Support Systems Vol 33, pp111-126.
18. Simon, H.A., 1997. Administrative behaviour: a study of Decision-Making process in Administrative Organizations. New York, Free Press.
19. Soubie, J.L., 1998. Modelling in cooperative based systems, In Proc. of COOP'98, Cannes (France).
20. Sprague, R. and Carlson, D., 1982. Building Effecive Decision Support Systems. Prentice-Hall, Inc, Englewood Cliffs.
21. Turban, E and Aronson, J., 2001. Decision Support and Intelligent systems, Prentice-Hall International, Upper Saddle River, New Jersey.
22. P. Zarate. Des systèmes interactifs d'aide à la décision aux systèmes coopératifs d'aide à la décision: Contributions conceptuelles et fonctionnelles, Mémoire HDR, INP de Toulouse, 2005