A cooperative intelligent decision support system for boilers combustion management based on a distributed architecture

Abdelkader Adla*, — Jean-Luc Soubie*— Pascale Zarate*,*****

** IRIT, Paul Sabatier University, Toulouse, France*

*** Computer Science Departement, Oran University, Algeria*

**** GI, INPT, ENSIACET, Toulouse, France*

ABSTRACT. *The management system of the boilers combustion is one of the most critical systems for the good functioning of GLZ-I oil plant. It has a high impact on the methods of cogitation and apprehension of various problems related to maintenance. It is a multi-participant process with high level of interactivity. In this work we consider the paradigm of distributed decision-support systems where several decision-makers must reach a common decision. As a solution we propose an integrated framework based on a distributed architecture where each decision-maker uses a specific cooperative intelligent decision support system. The support system is viewed as a set of computer-based tools integrating expert knowledge and using collaboration technologies that help decision-makers and provide them with interactive capabilities to enhance their understanding and information base about options through use of models and data processing.*

KEYWORDS: *DSS, GDSS, Intelligent DSS, Cooperative DSS, Distributed DSS.*

RESUME: *Le système de gestion de combustion des chaudières est un des systèmes les plus critiques pour le bon fonctionnement de l'usine pétrolière GLZ-I et a un grand impact sur la maintenance. C'est un processus multi participant et hautement interactif. Nous considérons, dans ce papier, le paradigme des systèmes d'aide à la décision distribuée, dans lequel plusieurs décideurs doivent atteindre une décision commune. Pour cela, nous proposons un cadre intégré basé sur une architecture distribuée où chaque décideur utilise un système coopératif intelligent d'aide à la décision. Ce dernier consiste en un ensemble d'outils intégrant des connaissances expertes et des technologies de coopération qui coopèrent avec le décideur et lui fournissent un contexte interactif lui permettant d'enrichir son savoir.*

MOTS-CLES: *SIAD, SIAD de Groupe, SIAD Intelligent, SIAD Coopératif, SIAD Distribué.*

1. Introduction

In many organizational or social settings, a decision does not appear as an outcome given by a "single" decision-maker, but as a compromise between various divergent interests and points of view. In the early 1990s, a shift from a mainframe based DSS to a network DSS occurred. As decision-making moves from being an individual activity toward a group activity, many organizations are forming virtual teams of geographically distributed knowledge workers to collaborate on a variety of tasks.

Teams of experts performing decision tasks together face many issues and traps characteristic of team work only. Geographic dispersion might limit access to and free movement of information, working with others could lead to fear of expressing ideas and so on. The complexity of business relationships as well as the number of decision-makers and organizations that are involved in the decision process has increased, while the time allowed for decision making has typically shortened. The need for Group Decision Making (GDM) techniques and support that may facilitate team work and could help overcome above issues is greater than ever before.

Research studies of group decision support systems reported in the literature mainly use face-to-face facilitated GDSS (Fjermestad et al., 1998; Warkentin et al., 1997). Some of its results may not apply to distributed teams (Chen et al., 2005). It is difficult for distributed teams to arrange face-to-face meetings. Moreover, although most presented GDSS environments try to solve problems in the real world, the lack of an integrated procedure (from decision identification, basic information acquiring, to final decision proposed), makes the systems only partially supportive or even needful of outside assistance. Still, despite the existence as well as the extensive use of numerous general-purpose commercial systems, it is our belief that these systems do not readily fulfil the needs and operational requirements of specialists and experts working in different organizations to render their expertise in GDM processes.

Our aim was to create a Distributed Cooperative Intelligent Decision Support System (DCI-DSS) for a specific task, contingency management. The system is applied on a case of boiler breakdown to detect a functioning defect of the boiler, to diagnose the defect and to suggest one or several appropriate actions of repair.

The rest of the paper is organized as follows: First we describe the application area. Then in section 3, we present a literature survey of some related works on decision support systems. In section 4, we propose a distributed architecture for an intelligent decision support system. Section 5 illustrates the feasibility of our DSS architecture and how it is used. Finally, we conclude with a summary and future research directions in section 6.

2. The boilers combustion management system

Gas Liquefying Zone I (GLZ-I) is a plant specialised in liquefying gas. It is one of several plants (a dozen on a national scale) which compose a parent oil company. In GLZ-I the manufacturing process is split into two subdivisions: Utility subdivision and Process subdivision. The Utility subdivision is composed of Pumps, Desalination Unit, boilers, Turbo-generators and Air Compressors while the Process subdivision concerns the tasks of manufacturing of liquefied Gas.

The management system of the boiler combustion is one of the most critical systems for the good functioning of the plant. It has a high impact on the methods of cogitation and apprehension of various problems related to maintenance. To supervise the good functioning of the boilers, different sensors are set up to point out anomalies at different stages of the process. The exploiting staff is often confronted with situations that impose a quick reaction of decision-making. This requires consequent material resources and highly trained and experienced operators in oil process management.

Currently, in a contingency situation (i.e. breakdown of a boiler) it is the duty of the process administrator at a local site to identify and diagnose the breakdown. The handling of a boiler breakdown consists of three main steps: discerning defects while the boiler is functioning, diagnosing defects, and proposing one or several appropriate actions of repair. There are two types of breakdown that may occur:

1) Automatically signposted to the operator by means of a triggered-off alarm, the flag (the reference given to each alarm) is pointed out on the board (control room). It acquaints with a particular alarm.

2) 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.

For the operator, there is two ways to solve the problem. In the former, the operator refers manually to several textbooks provided by the manufacturer to seek for the causes and determine the actions of repair. In the latter, he performs a set of tests on the site. He may also recall from his knowledge a previously encountered and successfully resolved case with similar characteristics and signs.

If no solution can be found locally, the operator informs the process administrator who makes contact with other process administrators and/or operators of the parent company and even calls on the technical services of the boilers manufacturer located abroad via traditionally communication media (phone call, fax, etc.). A lot of communication messages are necessary between the process administrator and the other actors to solve the problem knowing that the communication is set up between two persons at the same time, which takes a lot of time. This type of situation results in the plant working with degraded functions and it might even be necessary to stop the process (causing a shutdown alarm) waiting for the problem to be solved.

A lot of overhead might occur when the various actors try to exchange their knowledge and ideas which could undermine efficiency – when it is needed the most. The current set up of team work keeps the actors too distant from the actual site as often it is not possible to meet. Therefore, it is the process administrator who plays the role of a mediator and attempts to interact with all the actors. Consequently, this situation is mainly characterized by highly decentralised data sets coming from various sources. There is constant need to access this decentralised information at any time, from anywhere, under tight time constraints. We will show how a cooperative intelligent DSS based on a distributed architecture may provide answers to this challenge.

3. Decision making support

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, implementing and monitoring a choice (Mintzberg, 1979; McGrath, 1984). Briefly, the decision process refers to some techniques or processing rules aiming at structuring the context, timing or content of communication.

3.1. *Decision support systems*

Decision support systems (DSS) are computer-based systems designed to support and enhance managerial decision making. They are designed to actively interact with individual decision-makers in order to assist them to make better decisions based on information obtained and to solve ill or non-structured decision problems (Keen, 1978; Sprague et al., 1982).

A number of frameworks or typologies have been proposed for organizing our knowledge about decision support systems (Power, 2000). The two most widely implemented approaches for delivering decision support are Data-Driven and Model-Driven DSS. Data-Driven DSS help managers organize, retrieve, and synthesize large volumes of relevant data using database queries, OLAP techniques, and data mining tools. Model-Driven DSS 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: Document-Driven DSS integrate a variety of storage and processing technologies to provide managers document retrieval and analysis. Knowledge-Driven DSS or Intelligent DSS can suggest or recommend actions to managers. Finally, Communication-Driven DSS or Group Decision Support Systems 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.

3.2. *Intelligent decision support systems*

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. Recently, many improvements have been noticed in the DSS field, with the inclusion of artificial intelligence techniques and methods, as for example: knowledge bases, fuzzy logic, multi-agent systems, natural language, genetic algorithms, neural networks and so forth. The new common denomination is: Intelligent Decision Support Systems – IDSS (Ribeiro, 2006).

Intelligent decision support systems are interactive computer-based systems that use data, expert knowledge and models for supporting decision-makers in organizations to solve complex, imprecise and ill-structured problems by incorporating artificial intelligence techniques (Ribeiro, 2006).

The inclusion of Artificial Intelligence (AI) technologies in DSS is an effort to develop computer based systems that mimic human qualities, such as approximation, reasoning, intuition, and just plain common sense. The use of IDSS is intended to improve the ability of operators and decision-makers to better perform their duties and work together.

DSS research has evolved to include several additional concepts and views. Executive Information Systems (EIS) have extended the scope of DSS from personal or small group use to the corporate level. Model management systems and knowledge-based decision support systems provided smarter support for the decision-maker. The latter began evolving into the concept of organizational knowledge management about a decade ago, and is now evolving into a broader notion of DSS, Group Decision Support Systems (GDSS), serving as knowledge sources or connecting decision-makers with diverse sources (Courtney, 2001).

3.3. *Group Decision Support Systems*

A Group Decision Support System is developed to provide decision aid to groups or organizations by supporting collaborative and interactive works. DeSanctis and Gallup (1987) present one of the first formal definitions of a GDSS, evolved from the research of: “A GDSS combines communication, computing, and decision support technologies to facilitate formulation and solution of unstructured problems by a group of people". Such systems aim at facilitating the communication between individuals and supporting the expression of their opinions and beliefs in a way that is commonly accepted and understood, while providing the necessary technical infrastructure (Dennis et al., 2001; Jessup and Valacich, 1992).

DeSanctis and Gallup (1987) identify three environmental contingencies as critical to GDSS design: group size, member proximity, and group task. Based on differences in group size and dispersion of group members, four environmental settings are placed allowing the GDSS design and other technologies to be compatible. A space/time grid (including same-time and same-place or different-time and different-place scenarios) is used to classify various collaboration technologies.

Tools that support distributed teams which have been empirically tested are mainly synchronous computer conferencing systems (i.e. discussion forum software). These systems do not have explicit support for decision-making processes and often do not provide tools for alternative evaluation, therefore, they are also called Groups Support Systems. GSS products, such as GroupSystems (Chen et al., 2005), are LAN-based client-server applications often supporting same-time and same-place groups working in face-to-face settings. An audio/video conference system is a major example of a collaboration technology that supports groups working at the same time but at different places. This category of tools focuses on enhancing the virtual presence of meeting participants. The support for group processes and decision making are mostly missing from products in this category. Asynchronous technologies, such as e-mail and discussion forums, are commonly used as in the business world by distributed teams (Chen, 2005). Asynchronous technologies tend to focus on supporting group information exchange and sharing (Zaraté, 2005). However, these tools do not have strong support for group decision-making processes comparing to traditional GDSS.

Despite their impressive functionalities, DSS of all of types must focus on supporting, not replacing, a human decision maker for important decision tasks, as many of the problem situations faced by managers are unstructured in nature and require the use of reasoning and human judgement. Therefore, as articulated by Lévine and Pomerol (1995): "The DSS and the decision maker form a united problem solver". In decision-making processes, contrary to structured problems, the man takes advantage on the machine. Indeed, solving problems requires calling intuition and know-how of the decision-maker which becomes the preponderant element of the coupled 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 one of the decision-maker.

3.4. *Cooperative and distributed decision making*

Making a simple machine act intelligently may be much less useful or important than being able to cooperate in an environment with other intelligent agents, whether they are humans or machines. Beyond being able to communicate with others, detect and correct mistakes, and take advantage of others' abilities, so that overall intelligence or effectiveness may be an emergent property of all the smart agents working on the problem in a relatively coordinated fashion. It is found that "lack of attention to the human and organizational aspects of IT is a major explanatory factor and is manifest in a failure to involve users appropriately" (Clegg, 1997). As Keen (1981) stated, "Decision support systems do not automate the decision process nor impose a sequence of analysis on the user". Therefore, judgement and decision-making must occur throughout the entire problem solving process, that is, during the user's physical interaction with the system, and as the final human decision is being made. Because of this, the user's decision processes must be taken into account into the design process of successful cooperative DSS.

Moreover, according to Keen (Keen, 1981), decisions may be individual, group, or organizational tasks. When it comes to group or organizational tasks and when its users and/or components cannot be at the same place at the same time, a distributed DSS should be considered as a convenient alternative not as a replacement.

Automation is then more important for distributed group decision making in which group members are dispersed geographically, where coordination and control through traditional means of facilitation are very difficult, or even impossible.

We believe that successful distributed cooperative intelligent decision support systems and their subsystems must act intelligently and cooperatively in a complex domain with potentially high data rates and make judgements that model the very best human decision makers. It is also crucial that human decision makers maintain control over the final judgments, either by focusing the system on particular reasoning goals, or by modifying the basic knowledge on which the systems judgements rely.

To this end, the benefits of the traditional DSS can be further leveraged by both embedding expert knowledge 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. In this way, several distributed entities (humans and machines), possibly mobile along networks, cooperate to reach an acceptable decision (distributed decision). Distributed decision making must be possible at any moment. It might be necessary to interrupt a decision process and to provide another, more viable decision.

We propose, in the next section, a distributed architecture for cooperative intelligent decision support system in which the group decision making process is facilitated (a facilitator manages the group decision activity using the tools provided by the system) and each decision-maker uses a cooperative intelligent support system enabling him or her to generate locally and communicate alternative solutions of the problem at hand.

4. A Distributed Cooperative Intelligent Decision Support System

Decision making is mostly used as a multi-participant process with high level of interactivity. In this work, we consider the paradigm of distributed decision-support systems. The advantages of a distributed DSS running on a distributed architecture are twofold: on the one hand, it is difficult for distributed teams to arrange face-to-face meeting, the system enhances communication and information exchange from various expertise domains. This increased knowledge reusability is essential in a DSS. A decision problem is rarely completely new and often reuse parts of the knowledge used to solve past problems. On the other hand, a DSS is only useful if it is able to quickly generate high-quality alternatives. The flexibility offered by a distributed architecture does allow a DSS to retrieve and combine data, models, and other types of knowledge under tight constraints, amongst the currently available knowledge sources, to generate the required decision alternatives.

Based on this architecture, several decision-makers, who deal with partial, uncertain, and possibly exclusive information, must reach a common decision. To this end, the use of a distributed cooperative system makes possible the collaboration of distant users dispersed over a network and important volumes of data. The cooperative work so initiated is synchronous owing to the problem nature (contingency) but may be asynchronous in other situations. A small group or a whole organization can be supported. The application can be performed from several sites sharing a common information base (group memory, corporate memory and so on). If allowed, the decision making participants may know each other and work together, their influence on the decision making process may vary according to their individual levels of responsibilities at work, or they may have different abilities, arguments and points of view. The main benefits of the system in the our case are:

- Each decision maker is supported by a specific cooperative intelligent decision system which integrates the local process expertise and the experience gained in a plant. Beyond the traditional electronic brainstorming, this is helpful for novices and inexperienced operators who are really supported in alternative solutions generation;

- Provide means to share experiences and knowledge of several decision-makers. All cases resolved are stored in the group memory and can be retrieved continually;

- Using the system during a contingency, the process administrator should be able to gather information about the incident, access databases related to the incident, activate predictive models, communicate plans to operators and monitor the decision execution and the progress of the situation;

• The decision-maker is dynamically guided through the problem-solving process. First, the resolution is jointly performed by the two cooperating agents (decision-maker and system) based on dynamic tasks allocation between the agents. If no solution can be found locally, the system suggests submitting the case to the group. The process administrator is then persuaded to organise a meeting.

4.1. Architecture

We propose to develop a Distributed Cooperative Intelligent Decision Support System (DCI-DSS), see figure 1. The system should be, in essence, decentralized in terms of databases, model bases and knowledge engines. It should make the best use of the available knowledge to offer optimal cooperation capabilities for the users. The main features of the intelligent cooperative DSS considered here are: they operate on knowledge-bases, they are based on decentralized distributed architecture, and they are designed to facilitate collaboration and communication among decision making group members. Due to their distributed nature, they urge for solutions that go beyond centralized client/server DSS designs. The networked decision-makers (DM) work together to solve a particular problem although they might neither be present at the same time in the same place nor constitute a permanent organization.

The consequent development of the DCI-DSS based on above distributed approach is built upon the following assumptions: 1) Prospective users of DSS are involved in the decision-making process and should be supported locally, particularly during the generation stage; 2) the users may be experts or non-experienced in decision-making, so embedding expert knowledge would help both to generate alternatives; 3) the tools should allow for the generation and aggregation of alternatives and for the selection of a solution, and should present consequences of the selected decision in a meaningful way; and 4) time intervals between formulation of a query and receipt of the response should be minimal to allow for an effective on-line experience while using the DSS.

In this perspective, two co-operation types are implemented:

1) Man-machine cooperation allowing every decision-maker to locally solve problems and to generate alternatives. Here, each man-machine association is considered as a whole and indivisible entity within the cooperation network. There must be at least one communication protocol between the two components of the cooperative entity, as well as a machine, and formal data and information are directly available for computing.

2) Mediated man-man cooperation allowing the group of decision makers and the facilitator to make a collective decision. This kind of cooperation uses a machine as an intermediate communication medium. This technology can bring more

individual knowledge to the group discussion by supporting knowledge sharing, representation and visualisation tools, levels of participation are enhanced with contributions from members located in different sites of the organization, and organizational memory can be augmented with more relevant data: issues, comments, votes, and decisions.

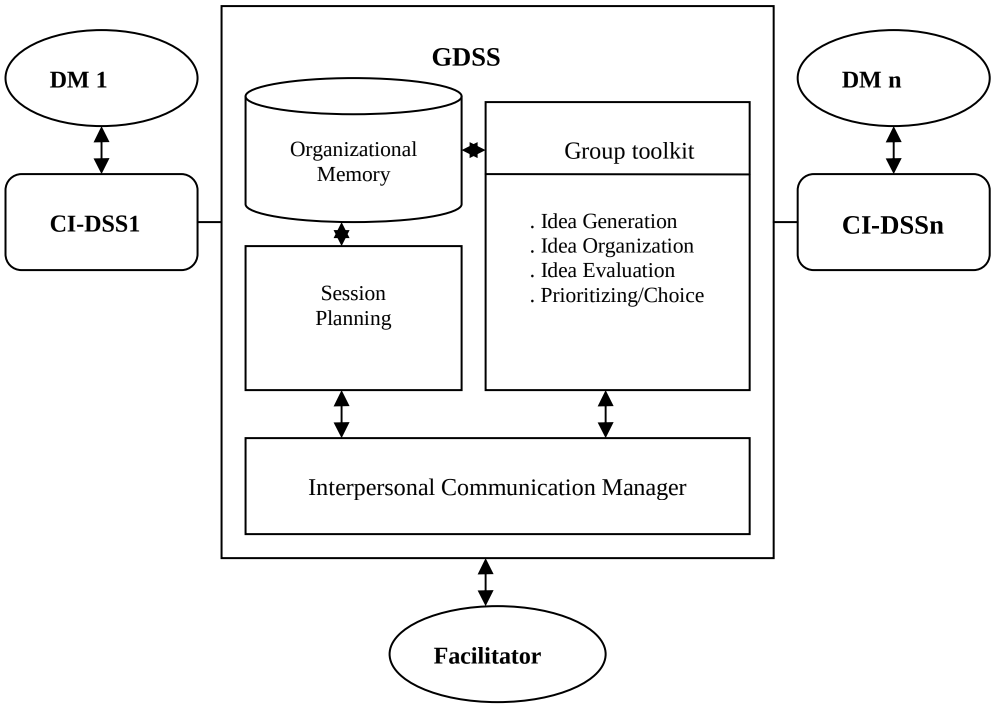

Figure 1. *DCI-DSS Architecture*

The system architecture consists of the following modules:

Interpersonal Communication Manager: In decision-making processes, it is obvious to give a tool of communication to the decision-makers. This type of tool is useful in idea aggregation, evaluation and prioritizing. The interpersonal communication manager allows the facilitator and the participants to interact across a network which may be web-based to allow distant decision makers to participate.

Organizational Memory: A meeting repository is used to implement an organizational memory storing all the meeting related information including meeting setup information, data relating to different users (e.g. @IP), the trace of previous sessions, and intermediate results saved. It is essential to be able to capitalize knowledge of the decision-makers implicated in the distributed decision processes so that each can refer to it if necessary. Moreover, the decision-makers implicated in the distributed decision process will be supported by this tool through reusing existing solutions of already established solutions.

Session Planning: A session planner helps the facilitator to set up and manage the meeting. Meeting setup involves the following tasks: create a meeting agenda which consists of agenda items, invite existing users or create new users to participate in the meeting.

Group Toolkit: There is a set of tools to support group activities which may be classified into three major categories: (1) Idea generation: each decision maker tries to generate alternatives using a specific cooperative intelligent decision support system that integrates local expert knowledge; (2) Idea organization: the facilitator uses tools to organize the ideas transmitted by participants (e.g. remove redundant alternatives); (3) Idea evaluation tools: a set of tools are put at the group's disposal to rate, rank, select and evaluate the alternatives.

Each group tool has two versions: (a) participation version as private space: it is used by a meeting participant engaged in a meeting activity; (b) Facilitation version as public space: it is used by the meeting facilitator to set up parameters or alternatives associated with a meeting activity.

In the proposed system each networked decision maker is supported by a specific cooperative intelligent DSS.

4.2 The Cooperative Intelligent Decision Support System (CI-DSS)

We have to take into account: (a) A decision maker, facing a singular decision situation, who is not familiar with the decision task, and (b) an expert decision maker, who is familiar with the decision problem, has strategies already compiled in his long term memory. Embedded expert knowledge and software tools are, therefore, used to learn the expert decision maker strategies in order to assist him (possibly continuously), to help him improving his performance and to capitalize his know-how.

The architecture proposed for the design of a specific cooperative intelligent decision support system – see figure 2 – extends the one of Soubie (Soubie, 1998) and developed for cooperative knowledge-based systems.

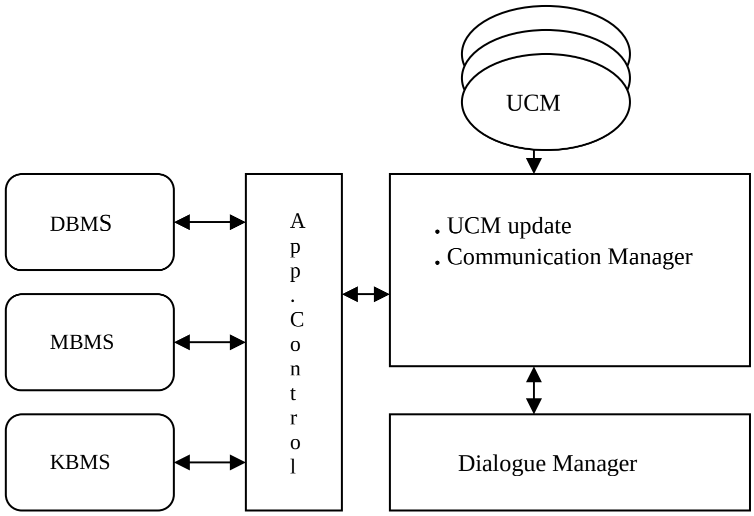

Figure 2. *CI-DSS structure*

The system architecture consists of the following components:

(1) **The Data Base Management System (DBMS)** mainly contains and manages a relational database that provides fast data retrieval, updating, and appending.

(2) **The Model Base Management System (MBMS)** includes many statistical, management scientific models, or other quantitative models that provide system's analytical or forecasting capability to solve future outcomes. Optimization models, such as linear programming and dynamic programming, are often adopted to determine the optimal resource allocation to maximize or minimize an objective function.

(3) **The Knowledge Base Management System (KBMS)** can support any of the other subsystems or play an independent role. It suggests alternatives or actions to decision makers. Additionally, it can be inter-connected with the data base.

(4) **The Dialogue Manager** allows the decision maker to interact with the system to achieve the task.

(5) **The Control and cooperation Management**: Sharing tasks is a precondition to implement cooperation between the two agents. The task that is the subject of cooperation must be decomposed into consistent subsets. The task distribution among the two agents is dynamically made according to the performances of the man/machine couple 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.

It is particularly useful to manage the task allocation in a dynamic way and, according to the control of the different tasks, to change planning initially implemented. This role is devolved into a planning tool. The interest of such a tool is to be able to provide an interactive and dynamic support during the collective problem solving; this type of support – interactive and dynamic – is a precondition for effective decision support. This tool constitutes essential provision in cooperation management. The man-machine cooperation is possible only if this task management tool allows a quick re-planning as well as a re-counting of allocations if there is a context modification or an evolution of the problem. This approach insures the dynamic character of the tool.

The cooperative architecture is based on three kinds of models:

- Conceptual Model of the Application (ACM): represents the functional capabilities of the system in the domain of problem solving;

- Conceptual Models of the Users (UCM): can be viewed as representation of users;

- Control Model of Cooperation (CCM): represents the cooperative behaviour of the system with respect to the context and the user.

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.

All the models used to implement the man-machine cooperation are based on the tasks-methods paradigm. The paradigm is based on the decomposition of objectives in sub-tasks allowing their performance. For each sub-task at least one method is associated in order to perform it. The problem to be solved is then modelled as hierarchical tree. Terminal nodes represent the last sub-tasks to perform.

Therefore, the reasoning of decision-makers is represented by a Task/Method paradigm. A Task represents all problems and sub-problems to be solved without a priori information about the manner of solving them. A Task is defined by the following components:

Name: Task name

Par: Parameter list, it represents the set of world objects (described in the domain model) handled by the task

Objective: Task’s Goal described as a state of the world after task performance

Methods: List of Methods achieving the task

To each task are associated the methods which are a priori declared as the best adapted to achieve it. A method represents a resolution mechanism or the know-how which it is possible to implement in order to achieve a task. A method is characterized by the following fields:

Name: Method name

Heading: Task achieved by the method

App-cond: Applicability conditions needed for the instantiation of the method

Prec: Preconditions that must be satisfied to apply the method

Effects: Effects generated by the successful application of the method

Control: Achievement order of the sub-tasks

Sub-tasks: Sub-tasks set

Two types of tasks are available: an "action plan" task which is a composition of sub-tasks; and "feasible" or "terminal" task which is self performable. Its execution does not require any decomposition. A terminal task is a task having only one method to achieve it. The field sub-tasks is empty and the control can point at an executable program.

Some tasks represented in the proposed Task/Method paradigm can be performed by an execution engine that can be roughly described in the following algorithm:

Start (T: task)

1. If the task T is a terminal then execute it else
2. Select an applicable method
3. For all subtasks ST of T, Start (ST) according to the control order

5. Modelling the boilers combustion management

We present here a simplified and partial Conceptual Model of Application (ACM) related to the boilers combustion management.

5.1. *The domain conceptual model*

The domain conceptual model is represented in the relational model. It allows the representation of all the knowledge relating to the application domain independently of the actions which manipulate them. We present a partial relational data model of the application in figure 3:

Stage: (#Stage-Number, Designation ….)

Effect: (#Effect-Code, Description …)

Cause: (#Cause-Code, Description …)

Parameter: (#Parameter-Code, unit, Normal value, Alarm Values (min, max) …)

Cure: (#Cure-Code, Description ….)

Figure 3. *Partial relational data model of the application*

5.2. *The task conceptual model*

The analysis of the problem related to a boiler defect results in the hierarchy of tasks, sub-tasks and associated methods partially presented in figure 4.

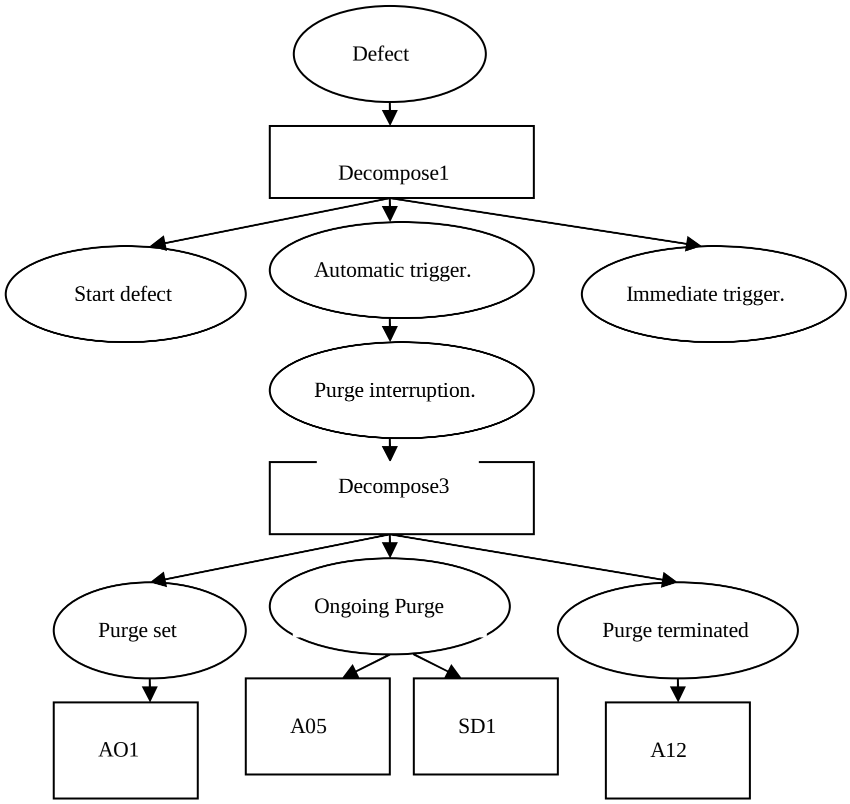

Figure 4. *A partial hierarchy of tasks and methods of the application*

Instead of support based actually on papers, a data base, a model base and a knowledge base are created upon these models within the DSS. These bases enable the decision-maker to interactively access information and execute predictive models. The decision-maker can query the system about certain parameters (speed, oil level, etc.) which are at the origin of the defect, the causes and the actions of repairs already stored in the database, and may also activate the predictive models to analyze other causes related to potential defects.

5.3. *Resolution*

When a boiler breakdown occurs, the process administrator (or the operator) first tries to solve the problem locally by using the specific Cooperative DSS (C-DSS). According to the parameter values accessed from the database, a reasoning process is launched to identify and diagnose the defect. If the defect is identified and diagnosed, actions of repair are automatically generated by the CI-DSS system and validated by the user. If no solution is reached, the system suggests the problem to be submitted to the group (process administrators or operators of parent company and technicians of the boiler manufacturer). We consider the latter case.

An on-line “meeting” is set up by the process administrator to represent a group solving process for the specific problem at hand where all other actors are invited to participate. The model for the DCI-DSS assumes that decision-makers are located in different places. A computer network is presumed that connects these different locations of participants. The decision-making process is controlled by a process administrator as a facilitator. The facilitator initiates and prepares the phases of the decision making process. S/he defines the issue(s) for decision and organizes the human group of decision makers for the decision-making process. It is also his responsibility to distribute the results among the participants. During the process the mediator has a main responsibility to allow for the convergence of the decision making process. He is responsible for the complete process and its deliverable, namely the decision.

The decision making process consists of three phases: meeting preparation, during the meeting phase, and post meeting phase.

5.3.1. *Meeting preparation*

The process administrator plans the decision making process and defines an agenda of the tasks and activities to be undertaken**,** defines the ground rules for the process and time limits, and select the decision-makers to participate. A list of potential participants (operators, process administrators, and constructor technicians) is put at his disposal. Participants have unique names within the environment. This name is a composite of the natural name and network address (@IP) of a participant. The model requires the uniqueness of names of participants.

5.3.2. *During meeting*

The problem solving is achieved at two levels: the decision maker level (alternative generation) and the group level (alternative organization and evaluation, and solution choice).

5.3.2.1. Alternative generation

This activity can be set up by the meeting facilitator in three modes of anonymity: (1) complete anonymity: the participant’s user name is not associated with the alternatives that was created by him, (2) Semi-anonymous: only a meeting facilitator can view the name of the participant who created the alternatives, (3) No anonymity: all participants can see the user name of the creator of the alternatives.

To create the solution alternatives to the problem at hand, each participant (decision-maker) uses his proper specific cooperative intelligent decision support system. 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 performed (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 a set of cycles until the entire problem is solved. Each cycle consists of the following steps: (1) identifying candidate methods; (2) identifying triggered methods; (3) selecting a method; (4) assigning the method to an agent (system or user); (5) executing the method; and (6) evaluating the task state.

Before launching the problem solving process (diagnosis and actions of repair), the operator updates the user model which meets his competences (a set of tasks and methods that he is 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 start: The operator assigns the task to be solved to the system. Then the system identifies the task in the task hierarchy and elaborates a plan of actions to carry out (a sub-graph of the task-method hierarchy). The action plan is elaborated according to the given parameter values (speed, temperature, oil level,) submitted by sensors or indications introduced by the operator. Two cases may occur:

(a) “Final” method is « ask-user »: thus the task is assigned to the user to solve it manually. An interaction is created between the system and the user for the introduction and the presentation of data and results.

(b) “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 assigned to the suggested agent (system or operator), otherwise the task is performed by the system.

The resulting alternatives are then placed into the private space. Each participant can select some of his private alternatives to be shown to the group. Participants have a delay for private creation of alternatives. Afterwards, the process administrator makes the alternative proposals public to the group through the shared space.

5.3.2.2. Alternative organization

Using the specific CI-DSS the group of decision makers may generate several alternatives in a short period of time. These alternatives may be similar or duplicated and may need to be merged. The facilitator may retrieve alternatives containing the same keywords, review them and then merge or delete them. Organization ideas in a distributed environment is mainly the responsibility of the facilitator. It can be a very challenging task.

5.3.2.3. Alternative evaluation

Participants can submit their evaluations and may also vote. They can view group results that include group averages and standard deviations. A large standard deviation may indicate a lack of consensus on an alternative or issue. The facilitator can bring issues with large standard deviations to participant's attention for further discussion. Participants may recast their votes to see whether the team can come to a consensus. Four evaluation tools can be developed: rating, ranking, selection, and multi-criteria evaluation tools.

1) Rating tool: this tool allows participants to evaluate a set of alternatives based on a 1 to X scale, with the default value for X being 10. The rating tool can be used to evaluate a long list of alternatives quickly and cut them down to a smaller set. Group evaluation results contain group average and standard deviation that can be viewed by participants.

2) Ranking tool: it allows participants to evaluate a set of alternatives by ordering them. Participants can select an alternative and move it up or down in a list of alternatives to change its ranking.

3) Voting tool: this allows participants to evaluate a set of alternatives by voting yes or no on each. A higher number of yes votes on an alternative means that the alternative is more favourable.

4) Multi-criteria evaluation tool: this tool allows participants to use a set of weighted criteria to evaluate a set of alternatives.

5.3.2.4. Decision choice

At this stage one alternative is chosen according to the evaluation tool used. This decision constitutes the collective decision made by the group.

This four-step resolution strategy used is illustrated in figure 5.

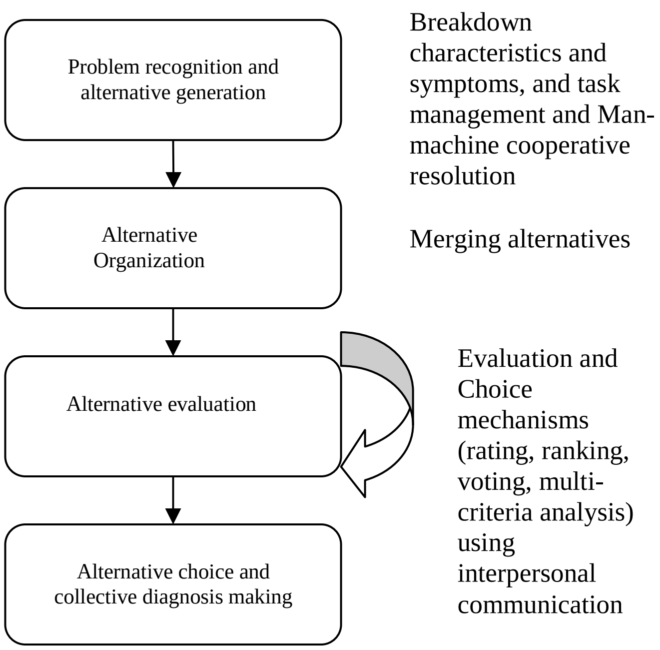

Figure 5. *Group diagnosis strategy*

5.3.3. *Post- meeting*

The process administrator provides a summary of the meeting to the participants, disseminates the results immediately to reinforce the agreements made, and generates post-meeting reports.

6. Conclusions

In this work we proposed a distributed cooperative intelligent decision support system to support operators during contingencies by giving them detailed and real time information, and enabling them to monitor the execution of the decision and the progress of the situation.

We considered 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 development of the Distributed Cooperative Intelligent DSS is based on two cooperation modes: (1) Man-machine cooperation allowing every decision-maker to solve problem and to generate alternative solutions, (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.

To support each decision maker, we used an intelligent decision support system based on a cooperative architecture that takes into account the capabilities of users and integrates them during the problem solving process, especially while generating alternatives. Placing the human operator effectively in the loop of such adecision support system represents a main guarantee for the efficient mastering of the inherent complexity of certain problems.

The structure used creates a model-based problem solving environment. 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).

On the other hand, during the past decade there have been numerous studies in the literature reporting about distributed facilitation (Antunes et al., 2001). Facilitators are most likely to either adapt a generic process or select one from a toolkit. Both the generic and the toolkit approach require prior experience with a large range of problems and thus are not applicable in the context of an experienced facilitator. Another question would consider how inexperienced facilitators are enabled to successfully facilitate a group decision making process.

One perspective of this work is to integrate an expert system approach capable to develop facilitation skills. The facilitation expert system considers the support to expert and inexperienced facilitators by incorporating a decision making process model in the support system. Therefore, the facilitator is supported in managing the group activity along the three phases of the decision process, i.e. in pre-meeting, during meeting, and post meeting phases.

8. References

Adla A., "Système coopératif d'aide à la décision", in *Proceedings of IHM'06,* Montréal, Canada, 2006.

Adla A., "A distributed architecture for cooperative DSS", *Workshop of the EWG-DSS,* London, United Kingdom, 2006.

Alter S.L., *Decision Support Systems: Current Practice and Continuing Challenges*, Addition-Wesley, Reading, MA, 1980.

Antunes P., Ho T., "The Design of a GDSS Meeting Preparation Tool", *Group Decision and Negotiation*, 10, 2001, p. 5-25.

Arnott D., Pervan G., "A critical analysis of decision support systems research", *Journal of Information Technology*, 20(2), 2005, p. 67-87.

Bannon L., “Group Decision Support Systems: An analysis and critique”, in *Proceedings of International Conference on Information Systems*, Cork, Vol. 1, 1997, p. 526-539.

Bostrom R., “Group facilitation and group support systems”, in *Group Support Systems: New Perspectives*, Mcmillan, New York, 1993.

Chen M., Liou Y., Wang C.W, Fan Y.W., Chi Y.P.J., TeamSpirit: Design, implementation, and evaluation of Web-based group decision support system, Elsevier B.V., Decision Support System, 2005.

Cheung W., “An Intelligent decision support system for service network planning”, *Decision Support Systems*, Volume 39, 2005, p. 415-428.

Clarke M., Smith M.G., “A cooperative computer-based on the principles of human cooperation”, *International Journal of Man-Machine Studies*, 1993.

Clegg C., “Information technology: a study of performance and the role of human and organizational factors”, *Ergonomics*, 40(9), 1997, p. 851-871.

Coppin G., Skrzyniarz A., “Human-Centred Processes: Individual and Distributed Decision Support”, *IEEE Computer Society*, July/August, 2003.

Courtney J.F., “Decision making and knowledge management in inquiring organizations: toward a new decision making paradigm for DSS”, *Decision Support Systems*, 31/1, 2001, p. 17-38.

Dennis A.R., Wixom B.H., Vandenberg R.J., “Understanding fit and appropriation effects in group support systems via meta analysis”, *Journal of Management Information Systems Quarterly*, 25 (2), 2001, p. 167-193.

DeSanctis G., Gallup B., “A foundation for the study of group decision support systems”, *Management Science*, Vol. 13, No 12, 1987, p. 1589-1609.

Dodson G., “The client and User in Decision Support Systems: Review and Research Agenda”, in *Proceedings of International Conference on Creativity and Innovation in Decision Making and Decision Support*, London, England, 2006, p. 466-487.

Fjermestad F., Hiltz S.T., “An assessment of group support systems experimental research: methodology and results”, *Journal of Management Information Systems* 15 (3), 1998, p. 7-148.

Forgionne G., “I-DMSS: A Conceptual Architecture for the Next Generation of decision Making Support Systems in the Internet Age”, in *Proceedings of IFIP TC8/WG8.3*, Cork, Ireland, 2002, p. 154-165.

Gachet A., *A Software Framework for Developing Distributed Cooperative Decision Support Systems, Inaugural Dissertation*, University of Fribourg, Switzerland, 2003.

Guerlain G., “Interactive advisory systems”, in *Proceedings of HPSAA Conference*, Savannah, GA., 2000.

Holsapple C.W., Whinston A.B., *Decision Support Systems: A Knowledge-Based Approach*, West Pub., St. Paul, MN, 1996.

Holtzman S., *Intelligent decision systems*, Addison Wesley, 1989.

Huber G., “Issues in the design of group decision support systems”, *MIS Quarterly* Vol. 8, No 3, 1984, p. 195-204.

Janssen R., *Multi-objective decision support for environmental management*, Kluwer academic publishers, 1992.

Jelassi M.T., Beauclair R.A., “An Integrated Framework for Group Decision Support Design”, *Information and Management*, Vol. 13, No 2, 1987, p. 143-153.

Jessup L., Valacich J., “Future directions and challenges in the evolution of group support systems”, *Group support systems, new perspectives*, L. Jessup and J. Valacich (eds), 1992.

Karacapidilis N.I., Pappis C., “A framework for group decision support systems: combining AI tools and OR techniques”, *European Journal of Operational Research*, Vol. 103, 1997, p. 373-388.

Keen P., Scott-Morton M., *Decision Support Systems: an organizational perspective*, Addison-Wesley Publishing, 1978.

Keen P.G.W., “Value analysis: Justifying Decision Support Systems”, *MIS Quaterly*, 5(1), 1981, p. 15-25.

Klein M., Methlie L.B., *Expert Systems: A Decision Support Approach with applications in management and finance*, Addison Wesley Publishing Company, 1990.

Krivine J.P., “Systèmes à base de connaissances coopératifs, rôles et modes de coopération”, in *Systèmes coopératifs: de la modélisation à la conception*, Pavard (Ed)., Octares, Toulouse, 1994.

Lévine P., Pomerol J., *Systèmes interactifs d’aide à la décision et systèmes experts*, Editions Hermès, 1989.

Lévine P., Pomerol J-C., “The role of decision maker in DSSs and representation levels”, in *Proceedings of Hawaii International Conference on System Sciences*, 1995, p. 42-51.

McGrath J.E., *Groups – Interaction and Performance*, Prentice Hall, Englewood Cliffs, NJ, 1984.

Marakas G., *Decision support systems in the 21st century*, Prentice Hall, 2003.

Millot P., *Task allocation between human and machine, design and evaluation*, State Thesis, University of Valenciennes, 1987.

Mintzberg H., *The structuring of Organization*, Prentice Hall, 1979.

Nonaka I., Takeuchi H., *The Knowledge-Creation Company*, Oxford Press, 1997.

Nunamaker J., “Lessons from a dozen years of group support systems research”, *Journal of MIS*, Vol. 13, No 3, 1997, p. 163-207.

Parker B.J., Al-Utaibi G.A., “Decision support systems: the reality that seems hard to accept”, *OMEGA International Journal of Management Sciences*, 14 (2), 1986, p. 135-143.

Power D. J., *Supporting Decision-Makers: An Expanded Framework*, http://dssresources.com, version 1.0, December 15, 2000.

Ribeiro R., "Intelligent Decision Support Tool for Prioritizing Equipment Repairs in Critical/Disaster Situations", in *Proceedings of the Euro Working Group On Decision Support Systems Workshop,* Rapport IRIT n°14—2006, Zaraté P. (ed), 2006.

Shim J.P., Warkentin M., Courtney J.F., Power D., Sharda R., Carlsson C., "Past, present and future of decision support technology", *Decision Support Systems*, 33, 2002, p. 111-115.

Simon H.A., *Administrative behaviour: a study of Decision-Making process in Administrative Organizations*, New York, Free Press, 1997.

Soubie J.L., "Modelling in cooperative based systems", in *Proceedings of COOP'98,* Cannes, France, May, 1998.

Sprague R., Carlson D., *Building Effective Decision Support Systems*, Prentice-Hall International, Englewood Cliffs, 1982.

Turban E., Aronson J., *Decision Support and Intelligent systems,* Prentice-Hall International, Upper Saddle River, New Jersey, 1998.

Turban E., Aronson J., *Decision support systems and intelligent systems,* Prentice-Hall International, Upper Saddle River, New Jersey, 2001.

Warkentin M.E., Sayeed L., Hightwer R., "Virtual teams versus face-to-face teams: an exploratory study of a web-based conference system", *Decision Sciences*, 28(4), 1997, p. 975-996.

Wong Z., Aiken M., *Automated facilitation of electronic meetings*, Elsevier B.V., Information Management, 2003.

Zhao J.L., "Intelligent Workflow techniques for Distributed Group Facilitation", in *Proceedings of the 35th Hawaii International Conference on Systems Sciences*, 2002.

Zarate P., *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.

Zarate P., Adla A., "Group DSS: A proposal of architecture", *Euro XXI*, Reykjavík, Island, July 2-5, 2006.