Decision support system extraction

Jelenk a SAVKOVIC-STEVA}NOVIC
Faculty of Technology and Met}allurgy, Belgrade University
Karnegijeva 4, 11000 Belgrade, Serbia
savkovic@tmf.bg.ac.rs

ABSTRACT
This paper presents knowledge processing and representation in decision support system. Knowledge achievement processes by semantic rules were examined. Product of the knowledge achievement processes is not only knowledge, than model of knowledge. Transfer and transformation facts and rules for problem solving were described. Method for decision mechanism extraction and representation was presented. Decision making mechanism was implemented.

Keywords: Knowledge processing, modeling, information, extraction, uncertainty.

1.0 INTRODUCTION
Knowledge base can be understand as a special case decision support systems and the other hand knowledge base present new stage in development evaluation step of information technologies. Knowledge base is the basic element of the expert system. Expression expert system is very often applied to program which use knowledge for behavior man-expert simulation, or whose function has some attributes man–expert behavior[1],[2]. It has power to learn from experience, general knowledge achievement, reconceptualization, analogy reasonable, transfer knowledge from one domain to the other, flexibility and changeable approach for problem solution. Behind these, expert selecting alternative solutions, explaining its diagnosis as well as to learn from previously experiences adding knowledge base new elements which achieving during the problem solution [3],[4].

Basic difference between expert systems and classical programs is that expert systems manipulate with knowledge and classical programs manipulate with data. Expert system has ability to solve complex problem which including uncertainty by information processing.

In this paper knowledge base for problem discovering and solving was developed.

2.0 KNOWLEDGEABLE SYSTEM
The two areas of model development and analysis are addressed through the discussion of generic simulation environment. The knowledge based simulation environment is an expression of some control law or cognitive theory. To the extent that the rule base is derived from set of assumptions about the environment and performance expectations, it is a belief system. However, in the existing form, the goals are not expressed and the underlying assumptions are not evident. Consequently, they are opaque to the analyst and cannot be directly applied to the learning process. When expressed in hierarchical form the relationship that exist between goals and subgoals provide a basis for relating overall goal based system performance to specific assumptions about the variability and contribution of the supporting subgoals. In this form, the belief system is a full expression of some control theory in that the system’s relationship with the environment, as expressed in a set of feasible state conditions, can be related either in overall system performance measures to be relationships and the subgoals that support them.

In recent years, many applications of expert systems to simulation have evolved as a computer aided knowledge engineering tools. There exist considerable success in developing
knowledge aided simulation systems. Intelligent simulation highlights to potential to meet with the demand. The technological advance in simulation has addressed the research interest of intelligent simulation [1].

The research and development in this disciplinary has continued for several years, and its effort has produced three types of intelligent simulation systems: single expert systems, coupling systems and integrated intelligent systems.

Single expert systems only process symbolic information, and provide assistance to system engineers in decision, making process for of-line simulation and modelling.

Coupling systems that couple numerical computation programs into expert system such that in can be used to solve engineering simulation problems.

Integrated intelligent systems are large intelligence integration environments, which can integrate different expert systems or numerical packages together to solve complex problems.

In the analysis and synthesis of engineering systems, simulation is a major technique. The traditional simulation techniques are algorithm based. They are often inflexible and provide limited means to the user. In fact, such techniques can not clearly simulate the dynamic behavior of the real processes. The segregation of the database, knowledge base and inference engine in the expert system allows us to organize the different models and domain expertise efficiently because each of these components can be designed and modified separately.

Presently expert systems are extensively developed in the research of intelligent simulation systems. Among the successful artificial intelligence applications, most of expert systems are production systems. Production systems facilitates the representation of heuristic reasoning such that expert systems can be built incrementally as the knowledge of expertise increases. The expertise knowledge for the problem is described by a set of production rules. The typical production rule is described as IF (condition )….THEN (action). Inference engine in executor. It must determine which rules are relevant to a given knowledge base and select one of them to apply.

3.0 A FRAMEWORK FOR KNOWLEDGE EXTRACTION

Knowledge based system must represents information abstractly so that it can be stored and manipulated effectively. Although experts have difficulty formulating their knowledge explicitly as rules and other abstractions (Fig.1). They find it easy to demonstrate their expertise in specific performance situations. Schemes for learning abstract representations, or concepts, from examples to interact directly with systems to transfer their knowledge.

---

Fig.1 Knowledge abstraction

Expert systems can be used to develop rules, based models and augment other types of models. Since can capture the experience, of experts, they can be used to forecast problems, advice, operators, validate data and ensure that the results from other are reasonable.

A functional approach to designing expert simulation systems was proposed many authors. They chose the differential games models is described using semantic networks. The model generation methodology is a blend of several problem solving paradigms, and the hierarchical dynamic goal system construction serve as the basis for model generation. Discrete event approach, based on the geometry of the games, can obtain the solution
generally in much shorter time. Cooperation between systems is achieved through a goal hierarchy.

Many expert systems have been introduced in such areas as medical diagnosis, chemical and biological reaction synthesis, pharmaceutical manufacturing, mineral and oil exploration, circuit analysis and equipment fault diagnosis. These expert systems have emphasized the development of the knowledge acquisition process, the knowledge base, the inference procedure or control structure and maintaining the independence of each of these functions. Many days, computers have been widely used in simulation, but the use has been limited almost exclusively to purely algorithmic solutions. Many engineering problems are partial structured problems that deal with the non-numeric information and non-algorithmic procedure, and suitable for the application of artificial intelligence techniques. Expert systems provide programming methodology for solving non-structured problems which are difficult to be handled by purely algorithmic methods. The experience from building expert systems has shown that their power is most apparent when the considered problem is sufficiently complex.

In simulation, both qualitative and quantitative analysis are often applied together. Usually, qualitative decision efficiently made with symbolic and graphic information, and quantitative analysis is more conveniently performed by numerical information. Both methods often complement each other. Any numerical solution is only an approximation to the true solution, which is always represented analytically. Analytical solutions can only be obtained by symbolic processing.

4.0 SYMBOLIC AND NUMERICAL PROCESSING

Most of the existing expert systems were developed for specific purposes. Usually, they were implemented with the symbolic language, and production rules were used to represent domain expertise. In light of application, such expert systems can only process symbolic information and make heuristic inference. Lack of numerical computation and uncoordinated single application make them very limited on the capability of solving the real engineering problems. Expert systems need data processing.

The coordination of symbolic reasoning and numerical computation is required heavily for simulation with expert systems. A few developers tried to develop expert systems with conventional languages. Other suggested to field expert systems in conventional languages, in order to achieve integration. Another disadvantage is that the procedural language environment cannot provide many good features that the symbolic language provides, such as easy debugging allowance for interruption by human experts.

Numerical languages often have a procedural flavor, in which the program control is command driven. They are very inefficient when dealing with processing strings. Symbolic languages are more declarative and data driven. However, it is very slow for symbolic languages to execute numeric computations. Complex problem can not be solved by purely symbolic or numerical techniques. Coupling of symbolic processing and numerical computing is desirable to use numeric and symbolic languages in different portion of software system. The coupled systems approach is often required when domain expertise is needed to provide the user suggestion or to direct the problem solving process. The most appealing approach is to achieve deep coupling of numerical and symbolic module representing the modules function, inputs, outputs, usage constraints. This allows the system to be applied to a wide range of problems, and makes it more robust system.

5.0 SUPERVISION AND CONTROL SYSTEM AS A CASE STUDY

As a case study the liquid flow system fault diagnosing shown in Fig.2 was used. The system consists of a two tanks, two mixers and pipes.

The study of fault detection and supervision control of the flow system is concerned with designing a system that can assist a human operator in detecting and diagnosing faults in order to prevent disturbance.
The system topology or component interconnections are defined by the process connections of the working process model (Fig.1). The level of aggregation is defined by the modular component interconnections which define propagation paths of attributes within the system. Initial research starting by phase of the development of a conceptual framework which facilitate the modular specification of models, and second phase the development of a logic framework which will permit object using attributes and simulation techniques to be linked into executable models.

The fault events of a system are in the first instance generally formulated in an IF-THEN form. This can be immediately reformulated using the operators AND, OR and NOT in Boolean form, if one can assume that the primary events have only two states existence and non-existence.

This system represent qualitative events model expressed by logic algebra, M, B, and L are independent logic variables representing the basic events malfunction, blockage and leakage, respectively.

Starting with the basic variables and their interrelations, the qualitative event model of the system can be formulated as shown in Fig.3.

To organize the logic of the rules, states variables must be defined within the system. Three types of state variables can be defined within a given component. The first are those variables whose values can be controlled by the system operator. These are controllable variables. The second set of state variables are those variables whose values are observable to the system operator. Last are those variables whose values are not immediately discernible by the system operator such as inner pressure and so on.

Scenarios are used to set initial states of the system state variables and attributes to predefined values prior to a model simulation run. This is necessary to evaluate “what if” scenarios concerning component malfunctions such as leakage due to worn or blockage due.

The system can diagnose causes of faults associated with state variables pressure and flow rate, supply. The qualitative variables are described in three discrete values (low, medium, high). The following faults are considered blockage-B, leakage-L, malfunction or missoperation-M.

A knowledge based decision support system building is consisting from the following steps:

2. Goals and subgoals definition.
4. Decision mechanism definition.
5. For diagnostic purposed need to monitoring system symptoms.

The symptom/scenario matrix displaying the final values of each symptoms which evolve from initial scenario states.

Various rules can be applied to this system based on the transitivity relationships of the qualitative variables of the fluid flow. For instance, fluid pressure implies fluid supply and fluid flow implies fluid pressure. Thus, fluid flow implies fluid supply. But the same can not be said for fluid supply implying fluid flow.

Each component provides attribution transformations depending upon its operating states.

Scenarios are used to set initial states of the system state variables and attributes to
predefined values prior to a model simulation run. This is necessary to evaluate “what if” scenarios concerning component malfunctions such as leakage due to worn or blockage due.

The system can diagnose causes of faults associated with state variables pressure, flow rate and temperature, supply. The qualitative variables are described in three discrete values (low, medium, high). The following faults are considered blockage-B, leakage-L, malfunction or missoperation-M and supply absent-SA.

6.0 DECISION MECHANISM

The best way to solve complicated problem by expert systems is to distribute knowledge and to separate domain expertise. In such case, several expert systems may be used together. Each of them should be developed for solving a sub domain problem. Here, it is faced the problem of knowledge integration and management. Many expert systems can only be used alone for a particular purpose inflexibility. There are lack of coordination of symbolic reasoning and numeric computation, lack of integration of different expert system, lack of efficient management of intelligent systems and capability of dealing with conflict facts and events among the various tasks, being difficulty in modifying knowledge bases by end users other than the original developers.

Many integrated intelligent systems are a large knowledge environment, which consists of several symbolic reasoning systems and numerical computation packages. They are under the control of a supervising intelligent system, namely, meta-system. The meta system manages the selection, operation and communication of these programs.

The key issue to construct the integrated intelligent system is to organize a meta – system, which can thus be referred to as a control mechanism of meta level knowledge. Meta system has its data base, rule base and inference engine, but it decomposes its activities into the separated, strictly ordered, phases of information gathering and processing.

Fig.3 The diagnostic system knowledge base

The main functions of meta-systems are coordination all symbolic reasoning systems and numerical computation routine in an integrated intelligent systems, distribution knowledge into separate expert systems and numeric routines, acquiring new knowledge, finding a near optimal solution for the conflict
solutions, and providing the possibility of parallel processing.

7.0 CONCLUSION

The obtained results show power in problem of knowledge acquisition and examining of own reasoning.

This paper describes supervision control diagnostic system building. A simple knowledge base was formed. In final phase the symptoms are ordered by relative information value for use within a decision tree.

The obtained results in this paper can be applied in the other domain.

Acknowledgment. The authors wish to express their gratitude to the Fund of Serbia for financial support.

8.0 REFERENCE


