JOINT ADVANCED STUDENT SCHOOL 2008, ST. PETERSBURG

KNOWLEDGE-BASED CONTROL AND ENGINEERING SYSTEMS

Final Report by

Natalia Danilova

born on 24.04.1987
address:
Grazhdanski pr. 28
Saint-Petersburg, Russia
Tel.: +79030996275

Department of
DISTRIBUTED INTELLIGENT SYSTEMS
Saint-Petersburg State Technical University

Supervisor: Gavrilova T.A.
Submitted: 01.03.2008

Contents

1 Introduction

Knowledge-based systems are the most mature and widely-used commercial field of artificial intelligence. They are most valuable to organizations that have a high-level of know-how experience and expertise that cannot be easily transferred to other members. They are designed to carry intelligence and information found in the intellect of experts and provide this knowledge to other members of the organization for problem-solving purposes. Knowledge-based systems achieve high levels of performance in task areas that, for human being, require years of special education and training. The applications find their way in many areas of knowledge work, such as science, agriculture, business, computer systems, education, manufacturing, process control, engineering (diagnostics, design, decision support), and so on.

Knowledge-based systems have several features:

- recognition of problem and its solution;
- explanation of the choice of the solution;
- selection of applicable solution;
- dealing with incomplete information;
- problems restructuring;
- reduction of the need of research.

2 Basic concepts

It is very important to clarify the difference between ‘data’ and ‘knowledge’. Data consists of facts about the object; it is used for describing its attributes and values. Knowledge shows the relationship and rules concatenating the data and obtained from experience.

A knowledge-based system, also known as an expert system, is a computer program that contains the knowledge and analytical skills of one or more human experts, related to a specific subject. In other words, it is a software that performs a task that would otherwise be performed by a human expert.

The main difference of expert systems from other applied programs is that expert systems do not model the physics of the domain. They show the way of problem solving by human expert or the way of thinking and knowledge processing in the domain. Expert systems make all conclusions using knowledge that is put in by expert and knowledge engineer; heuristic and fuzzy methods are used instead of mathematics.

Every expert system contains three main blocs (see Figure 2):

- Knowledge base – consists of sentences which define knowledge with the use of super-high-level languages, called knowledge representation languages. It is the kernel of expert system and should be clear and comprehensive.
- Inference engine – a program that simulates the process of expert reasoning or decision making.
- User interface – presents questions and information to the user and supply the user’s responses to the inference engine.

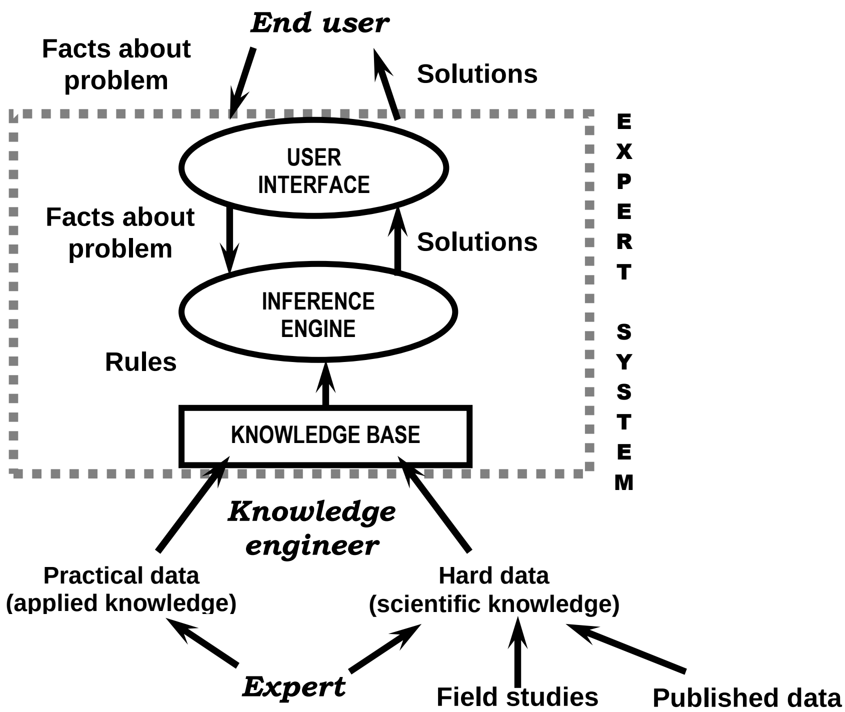

Figure 2 – Expert system structure

Expert system life cycle includes five main stages:
1) problem definition;
2) knowledge elicitation;
3) knowledge representation;
4) prototype system;
5) commercial system.

On the problem definition stage the appropriate domain is selected. The problem must require some type of specialized knowledge, but it must not be overly large. In business organizations it should be a problem that is handled often enough so that an investment is expected to have some payoff.

Knowledge acquisition is a structured way of developing knowledge-based systems.

Knowledge representation includes three major symbolic methods: rules, semantic objects and logic. In this stage knowledge about the domain must be fully coordinated, so that the future knowledge base will be consistent.

The prototype system evaluates three phases: demonstrative prototype, research prototype and executing prototype. Demonstrative prototype is a very small expert system which solves only a part of the problem. It should prove that it is possible to make an expert system. Usually after demonstration the prototype is abandoned. Research prototype is an expert system that solves the entire problem, but is not truly tested. Executing prototype is a complete expert system. It works only with the author and doesn’t have any documentation.

During the last stage the system if tested in full scope and the proper documentation is prepared for the users and knowledge engineers for latter knowledge-base correction, update and edition. It is ready for commercial installation in the market. Actually only 5% of all expert systems reached this stage.

There are several advantages and disadvantages of using expert systems. They provide consistent answers for repetitive decisions, processes and tasks, old and maintain significant levels of information. Expert systems encourage organizations to clarify the logic of their decision-making. Such a system never ‘forgets’ to ask a question, as a human might.

Unfortunately, under some unusual circumstances expert system cannot make some creative responses like human expert. Domain expert are not always able to explain their logic and reasoning. Expert system works only in the environment that knowledge base is adapted to. If the environment changes, the knowledge base should be changed as well. Expert system is a software, so if the error occurs, it may lead to the wrong decision.

2.1 History of Expert Systems

The growth of artificial intelligence could be classified into 3 time frames: the logical period, the knowledge-based period and the period of industrial artificial intelligence systems.

During the logical period (1960-1976) scientists formed a complex labyrinth to solve problems intelligently, but this failed due to the vast initial time and space requirements for generating these. Various heuristic approaches were used. But during the logical period people felt, that intelligence could be achieved by using mathematical logic. The axiomatic approach of solving problems was based on a set of axioms and a theorem (a problem) to be solved. But this approach failed as well, because most of the real-world problems were never axiomatic. In 1973 the first programming language PROLOG was invented, it was based on the above principle. In 1965 McCarthy invented LISP – the first language, that did symbol processing instead of numerical computation.

During the knowledge-based period (1976-1990) in the US there was a feeling that in order to achieve intelligence, conclusion should be made from vast amount of previous knowledge collected or generated. A human makes decisions based on knowledge obtained by years of experience. In this period the two first expert systems MYCIN and DENDRAL were developed in Stanford University.

Starting from 1990 until nowadays knowledge-based systems have been applied to a wide variety of areas.

3 Approaches

There are two main approaches in knowledge base theory: goal-driven reasoning and data-driven reasoning.

Goal-driven reasoning (backward chaining) is an efficient way to solve problems that can be modeled as ‘structured selection’ problems. The aim of the system is to make the best choice from many enumerated possibilities. For example, identification and diagnostic problems fit this model.

The main idea of the goal-driven approach is that the knowledge is structured in rules which describe how each of the possibilities might be selected. The rule breaks the problem into sub-problems. The system would try all of the rules which give information satisfying the goal. Each would trigger sub-goals. The lowest level sub-goals would be satisfied or denied by asking the user. The system effectively carries on a dialog with the user. The user sees the system asking questions and responding to answers as it attempts to find the rule which correctly corresponds to the right choice.

However, for many problems it is not possible to enumerate all of the possible answers in advance and have the system select the correct one. For instance, configuration problems fall in this category. Since the inputs vary and can be combined in an almost infinite number of ways, the goal-driven approach will not work.

The data-driven approach (forward chaining) uses rules similar to those used in backward chaining, but the inference process is different. The system keeps track of the current state of problem solution and looks for rules which will move that state closer to a final solution.

The important thing is that for a data-driven approach the system must be initially populated with data, in contrast to the goal-driven system which gathers as it is needed. Figure 3 illustrates the difference between forward and backward chaining systems for two simplified rules.

Backward chaining:

A=1 ← If A=1 & B=2 then C=3;   Iff C=3 then D=4 ← D=4
B=2

Forward chaining:

A=1 → If A=1 & B=2 then C=3;   If C=3 then D=4 → D=4
B=2

Figure 3 – Difference between forward and backward chainings

The forward chaining system starts with the data of *A=1* and *B=2* and uses the rules to derive *D=4*. The backward chaining system starts with the goal of finding a value for *D* and uses the two rules to reduce that to the problem of finding values for *A* and *B*.

4 Case-Studies

4.1 Case 1: Blast Furnace Control

In the beginning of 1990 NKK Steel Company's Fukuyama Works developed an expert system to predict abnormal conditions within its blast furnace. A blast furnace is a complex, distributed, non-linear process. Mathematical modeling techniques have never been able to predict future dynamic states of the furnace with enough accuracy to support automated control.

Because the blast furnace feeds all other processes in the steel mill, any instability in the operation of the furnace is compounded by the impact on other processes further down the production line. Avoiding unstable operation of the furnace requires characterizing the current state of the furnace and projecting the conditions which will occur over the next several hours while there is still time to make adjustments. Training a skilled blast furnace operator takes many years. Codifying the skill of experienced furnace operators reduces the training requirements.

Several factors contribute to the complexity of modeling a blast furnace. Material within it coexists in all three phases – solid, liquid and gas. The large size of the furnace leads to six to eight hours before a change in raw-material charging takes effect. There are no symmetries to simplify the geometric modeling. Moreover, the flow of material inside the furnace is itself a complex process. The thermal state of the furnace cannot be measured directly, but must be inferred from various sensor measurements. The challenge for the furnace controller is to minimize the uncertainty in the operating temperature. The smaller the uncertainty, the lower the overall temperature needed to produce the pig iron (see Figure 4.1).

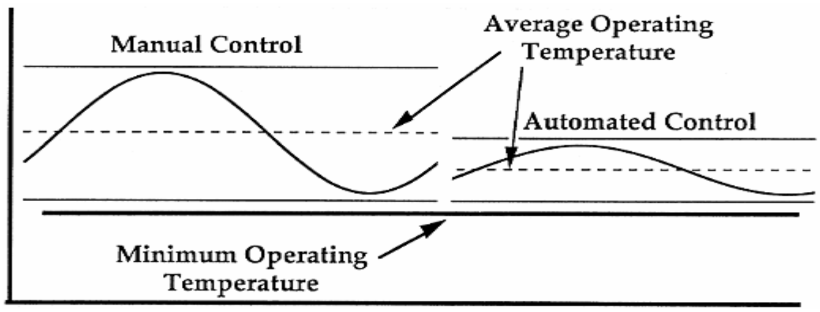

Figure 4.1 – Fuel Cost Savings

An expert system has been developed which successfully models the current state, predicts future trends with sufficient accuracy to make control decisions, and actually makes the control decisions. At any time, the operator can select either manual mode or automatic mode.

Figure 4.2 illustrates the blast furnace expert system structure.

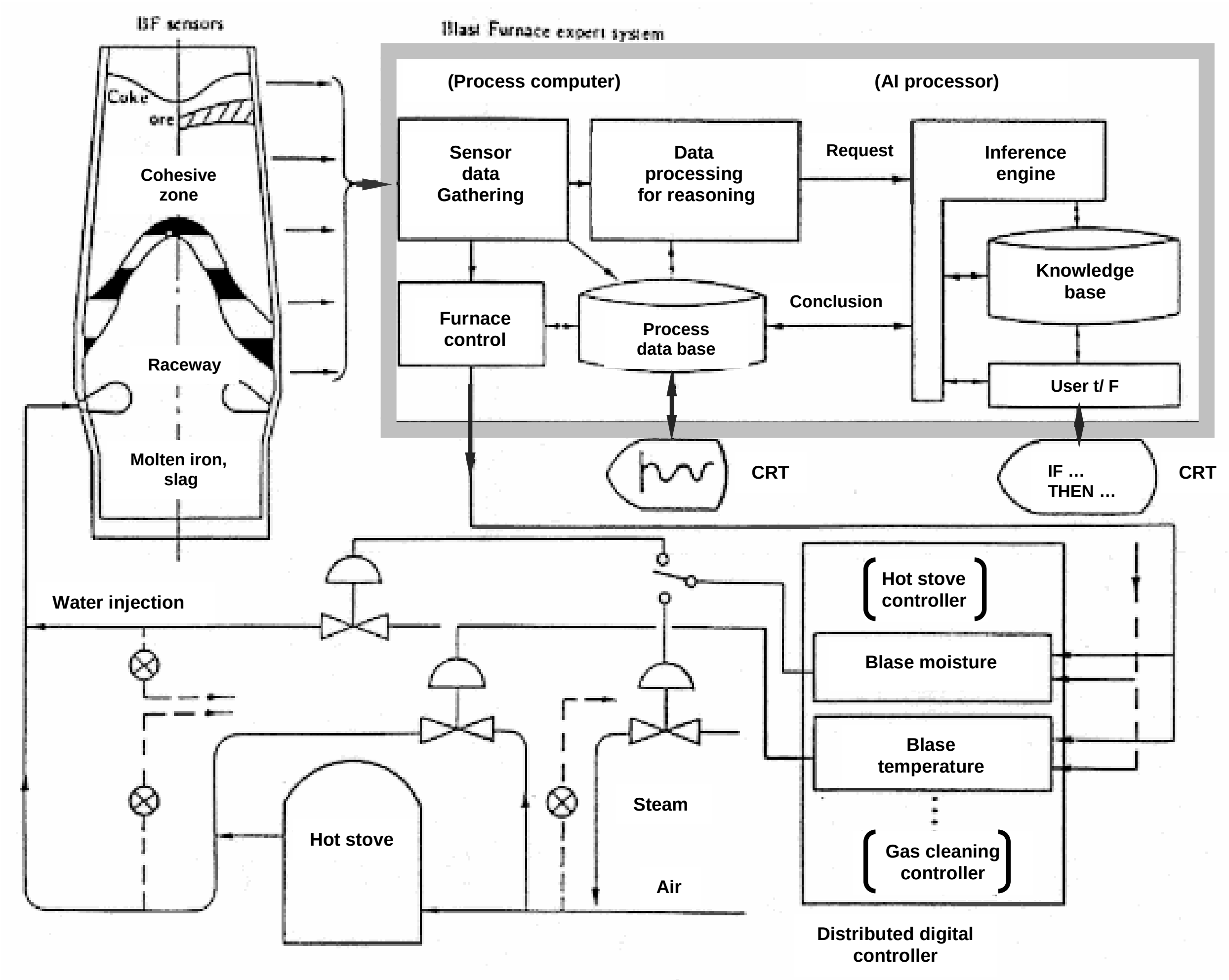

Figure 4.2 - Blast furnace expert system structure

The system consists of three main components:

- A process computer gathers input data from various sensors in the furnace, maintains a process database and generates furnace control information.
- An artificial intelligence processor provides the knowledge and reasoning for assessing and interpreting the sensor data, hypothesizing the internal state of the furnace, and determining appropriate control actions.
- A distributed digital controller uses the furnace control data from the process computer to control the actual blast furnace.

The system was implemented in LISP with FORTRAN used for data processing. The knowledge in artificial intelligence processor is contained in 400 rules, 350 frames, and 200 LISP procedures; fuzzy theory is employed in its inference engine. The system has a cycle time of 20 minutes instead of six to eight hours of the furnace time constant. Fuzzy set membership is used to relate the temperatures inferred from the instruments to actual temperatures. The membership functions are revised from time to time to tune the performance of the system.

The blast furnace control application is noteworthy for many reasons. The problem had not been solved previously by other techniques. The company reports an estimated annual savings of $6 million, a reduction in staff of four people, and an improvement in the quality of the furnace output because of reduced fluctuations in furnace temperature.

4.2 Case 2: MobES Expert System

Regarding to the usability and the important of the computers to the users, and the least number of the expertise in this field, a system called MobES (Motherboard Expert System) was developed to help the MSI’s motherboard users to recognize, manage and fix the small problems of their own personal computer. Knowledge of computer troubleshoot was collected from the experts by interviewing them, and also from the motherboard manual book and from the MSI’s website, so that MobES can identify and give accurate solutions to the users.

First of all, motherboard is the main component of a personal computer. From the motherboard users can recognize the problems happen with the computer. But when things happen, users are requested to access to the vendor’s website to list down their problems and wait for the reply from the staff.

Secondly, the users have to call to the service center and ask for the solution from the technician. When they are doing so, they are charged.

Lastly, the users have to refer to the motherboard manual which can be hard to understand.

MobES can easily provide the solution users need.

The objective was to develop an expert system for MSI’s motherboard users that would be capable to help them recognize, manage and fix their motherboard’s problems, and provide them with an appropriate solution base on the accurate diagnosis.

The development of MobES includes four stages of expert system life cycle.

In the stage of problem definition, the problems faced by the users are identified and assembled, so that to make sure that the data is suitable for the expert system. It is important to gather all the information, identify the requirements, the goals and the scope of the project.

During the knowledge acquisition stage the gathered knowledge is converted directly to the computer version. The collection of knowledge for the MobES was done by interviewing those who are experienced in the MSI motherboard structure, and by collecting data from the user manual books and the data from the MSI’s website. Analysis of gathered knowledge is important to make sure that the data is accurate, so that the system can work as the expert while giving solutions and the advice to the users.

In the stage of knowledge representation MobES is added with the rules, system programming part and the system interface. This phase is very important for the system because of the judging the successfulness of the whole project. The inference engine selects a rule for testing and then checks if the conditions for that rule are true. The conclusion will then be added to the knowledge base or may be displayed via the user interface for information.

During the stage of prototype system the system testing is crucial as to validate and verify the operations of the system at each stage of the system process. The test planning is concerned with setting out standard for testing process that serve as a guideline for the system developers when they carry out the system testing.

There are many possible input devices that make the communication between the user and the expert system effective. The most common are: mouse, keyboard, light pen, touch-sensitive screen, and voice input, but for MobES, mouse and keyboard are used. An expert system user interface takes the form of a set of questions, followed by some advice from the system. The expert system user interface will not only enable the user to answer questions, but allow the user to interrupt its operation and ask for explanations.

There are few stages of testing have done for the MobES.

Stage 1 - knowledge acquisition test. In this stage, testing on the accuracy of the knowledge has done to make sure all the knowledge and rules in the system are totally accurate with the knowledge that collected from the expertise.

Stage 2 - structure and design test. In this stage, the structure of the knowledge and the user interface are tested to ensure the MobES ability of solving the user’s problems. Besides that, the test of the system interface is also done in this stage to ensure that the users are comfortable with the design.

Stage 3 - full prototype test. In this stage, testing of the whole system is done to ensure it reaches the goals, whether it provides solution and advice to the users so that the users can handle their own computer problems.

This Motherboard Expert System is still under development, so it has limited knowledge and solutions about the MSI’s motherboard, such as sound card, monitor and display, mouse and keyboard, BIOS setting and other PC problems. MobES should collect more knowledge so that more rules can create and to have a better service for the users.

In future, MobES may be able to apply with other methods of Artificial Intelligence like artificial neural networks, fuzzy logic and genetic algorithms to improve the design of the system so that it can provide more accurate results to the users.

5 Conclusion

Knowledge base technology will clearly become more helpful in dealing with information overload. The current capability of machine intelligence is such that human knowledge will continue to be a valuable resource for the foreseeable future, and technology to help to leverage it will be increasingly valuable and capable. However, in many cases experts are being asked to surrender their knowledge and experience— the very traits that make them valuable as individuals.

This paper has provided a brief overview of knowledge-based systems. Knowledge-based systems appear to have a great deal of potential to assist the activities of managers in many organizations. They also have some dangers which must be appreciated and confronted.

Bibliography

1 Amzi inc. **Building Expert Systems in Prolog**: www.amzi.com/ExpertSystemsInProlog
2 Clarke R. **Knowledge-Based Expert Systems**: www.anu.edu.au/people/Roger.Clarke/SOS/KBT.html
3 Day J. **Knowledge-based Engineering – Automating for Profitability in Product Design**: www.designnews.com
4 Feigenbaum E., Friedland P. **Knowledge-Based Systems in Japan** - www.wtec.org/loyola/kb/c2_s3.htm
5 Gavrilova T. **Course of Lectures about Knowledge Engineering**, - Saint-Petersburg State Polytechnical University, 2007
6 Norman A. **Course of Lectures about Expert Systems**, - University of Texas at Austin: www.eco.utexas.edu/faculty/Norman/long.extra/Info.S98/Exp
7 Siew Wai. **MobES Expert System**: www.generation5.org/content/2005/MobES.asp
8 Taylor J., Stringer P. **An autosynthesizing non-linear control system using a rule-based expert system**: International Journal of Adaptive Control and Signal Processing, 5 Mar 2007, Volume 5, Issue 1 , Pages 21 – 39
9 **Wikipedia**: www.wikipedia.org