MINE 432
Industrial Automation and Robotics in Mining

Lecture 10A

Expert Systems

An Expert is someone who has already made all of the possible mistakes in a very narrow field of study - Niels Bohr

Introduction

Expert Systems have been around since the early 1970's although the idea of symbolic processing (or soft-computing) was first suggested in the late 1950's. Spawned by the famous Dartmouth College Conference in 1958, the field of Artificial Intelligence has grown at a great pace over the past 40 years with Expert Systems techniques leading the way as commercial products for industrial use.

Numerous Expert System development tools entered the market-place in the 1980's together with the explosion of microcomputers and today there are many examples of successful applications. Processing plants worldwide, now use AI to solve real-world problems.[1-10] Early methodologies have been distilled into rule-based techniques that link incoming data to sub-goals and then to final conclusions. Frame-based approaches have evolved into object-oriented methods that group information into classes allowing construction of efficient rule-structures with both reduced numbers of rules and processing time.

But, as more and more technical people have become familiar with these methods, it is apparent there are limitations to the ease with which applications are developed. It takes considerable skill in the following areas to ensure success: psychology, computer programming, knowledge acquisition and representation, etc. Without training, it is difficult to build useful and productive systems - ones with big payback!

Expert Systems have promised to embody "human intelligence" into a computer program. This, they do well and a system can function with only a single rule in its knowledge base making development relatively painless. Such systems have limited ability and knowledge but not all. There are techniques today to incorporate expertise into a Hypertext Interface so Users control and manipulate the domain as they desire.[7]

Early systems were considered small or toy-like unless they contained hundreds or even, thousands of rules. Today, a system containing this number of rules is either an extremely large application with extensive domain-specific knowledge with perhaps, some global-knowledge capability, or it has been built using old techniques. Computationally-intelligent methods such as Fuzzy Logic, Artificial Neural Networks and Evolutionary Computing significantly enhance our ability to represent knowledge effectively and efficiently.

"Intelligent" Systems

Expert Systems possess the ability to embody "intelligence" into a computer program in much the way **human communication** takes place. Facts are stored symbolically with their truth or falsity dependent upon current circumstances when using the system. A Knowledge Engineer can build a system using the exact words of the Expert expressed as symbols and rules-of-thumb that link these symbols. For example, an Operator might express the following idea:

> "When froth conditions on the lead rougher bank are porridge-popping, I reduce the collector addition rate by about 100 cc/min unless it is already at minimum. In that case, I increase the water addition to the rougher feed provided the Pb feed grade is not high. If it is high, then I reduce the feed tonnage."

This simple paragraph can be formulated into the following code:

```
Rule Porridge-popping
IF     froth.conditions.porridge-popping
AND    collector.addition_rate.minimum is FALSE
THEN   collector.addition_rate_change.Negative_Big is TRUE
ELSE   MACRO ("Water_addition_increase")

Rule Water_addition_increase
IF     froth.conditions.porridge-popping
AND    collector.addition_rate.minimum
AND    feed.Pb_grade.high is FALSE
THEN   feed.water_addition.Increase is TRUE
ELSE   feed.tonnage_rate.Decrease is TRUE
```

Fuzzy Sets can be used with these rules to map discrete measurements into the linguistic terms in the rule statements. Explanations can be attached to each fact and rule descriptions to each rule as desired to provide Users with the ability to probe the knowledge base for further details should these rules fire successfully.

These and other rules are added to the system incrementally with static and dynamic testing of the knowledge base at points in time as the system grows. Other "experts" might be polled to ensure general agreement about the strategy to be implemented. If conflict arises, a method to arbitrate disagreement must be arranged, usually with support of suitable metallurgical staff. During a consultation with an Expert System, a User can obtain justification for the advice given or ask the system about the meaning of certain dialog: "Why are you asking me for this information?" or "What does that term mean?" or "How did you arrive at your conclusion?" In this way, such systems serve an important training function in addition to fulfilling their original goal: diagnosis, monitoring or control.

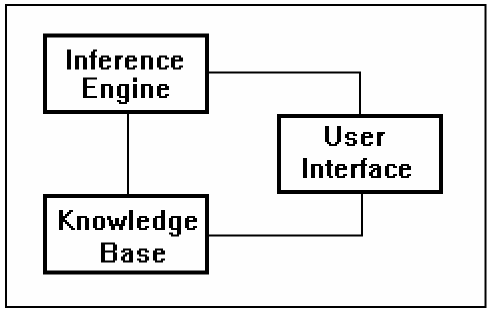
While this provides a dialog to mimic how a real "Expert" might communicate expertise to a novice (or lesser expert), there is more to human intelligence than providing advice and explanations. For example, these systems don't learn from experience. Neither can they derive new knowledge from an examination of their current knowledge. They simply perform what a human has created them to do. So based on today's techniques, conventional systems have reached a watershed particularly for real-time monitoring and control.

Review of Conventional Expert Systems

The real success of Expert System technology began with the decision to separate knowledge and data from how this information is processed. As a result, all Expert Systems consist of at least three separate computing modules:

- An Inference Engine - the thinking part of the system
- A Knowledge Base - the storage of memories symbolically
- A User Interface - the communication of the knowledge with others

The diagram below shows these basic modules and how they interact with one another.



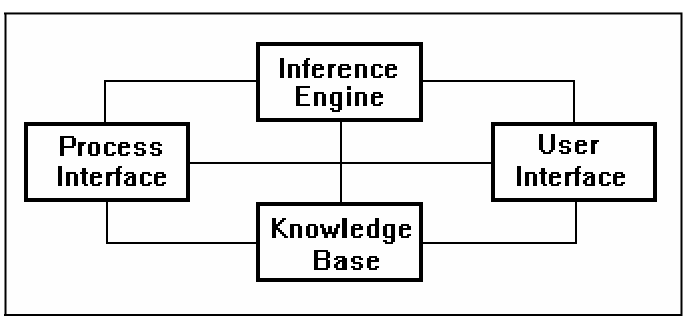
The Inference Engine contains the algorithms used to mimic the human-thought process. These methods are used as desired or required by the Knowledge Base and/or User Interface.

The Knowledge Base contains the domain knowledge relevant to the application in question. It is the centre of attention for the people responsible for building the system. A Knowledge Engineer builds the system in consultation with an Expert (or several). Eventually, the System Users are requested to comment on the system and its User Interface. These comments are used to modify the system after feedback to the Experts.

The User Interface is the gateway into the Knowledge Base. Users receive advice or can ask questions of the system such as: How?, Why?, and What does that mean? A variety of input-output screens are available in most system development tools:

- customized questions and explanations
- hypertext documents
- forms for input and output
- animations and pictures

Expert Systems for Process Control and Monitoring
When an Expert System is applied to monitor or control a process, a fourth module is required -- The Process Interface.



The Process Interface consists of a variety of modules depending on the process in question and the particular application. It may contain device drivers connected to process instrumentation such as DCS systems, PLCs, analog/digital converters, etc. The data derived from the process is stored in RAM-resident data files for access by the Inference Engine and Knowledge Base. These files represent process states as well as historical trends.

In some cases, The Process and User Interface Modules may be an existing Process Interface system such as Factory Link, Wonderware, The Fix, etc. By adding an Inference Engine and Knowledge Base, Intelligent Supervisory Control can be added to existing process software.

Knowledge Base Components

The elements that make up most Expert System knowledge bases consist of:

Facts	- classes and object-value combinations
Rules	- premise and conclusion statements
Procedures	- functions that operate on rules and facts
Meta-Knowledge	- text messages attached to rules and facts

Classes of objects allow a system to develop rules that operate on multiple units. For example: a process might consist of several parallel unit operations such as several ball-mills. Rules can be developed to deal with problems in ball-mills in general as a class: all ball-mills will be examined in sequence by these rules.

Rules are written in ordinary English allowing development and maintenance to be done by unskilled personnel. Rules form the backbone of the system: they represent the route by which changes in observed data are processed by the Inference Engine/Knowledge Base. In some systems, the cycle time through the Knowledge Base is fixed according to the rule structure and interactions. With a good real-time system however, processing through the Knowledge base is event-driven. This means that whenever certain facts change in the point data base, certain related rules are examined.

Procedures are essentially methods that are applied without regard to conditional statements such as IF or WHILE unless a specific statement contained within a procedure required examination of a rule or set of rules. Some of these procedures deal with temporal reasoning to examine the trend of variable changes. Others deal with I/O methodologies such as displaying data, activating features in a User Interface window, communicating with a database, spreadsheet or other application module external to the Expert System.

Meta-Knowledge refers to specific methods that are attached to facts, rules or procedures to provide answers to User questions such as:

- How did you derive that conclusion?
- What do you mean by that question?
- Why are you asking that question?

These questions are answered **on demand** in a real-time or consultative system. They do not pop up on the screen automatically untill a request is made to the system.

Search Strategies in Expert Systems

As the Inference Engine operates on the Knowledge Base with the current state of information, rules are either fired successfully or not. If a rule is successful, its conclusions may provide directions for the system to move to examine next. There are a number of strategies used to process data through an Expert System. These include:

- forward-chaining
- backward-chaining
- depth-first searching
- breadth-first searching

Forward-chaining is a search technique that is data driven. As facts become know (input or sub-goal facts), they are used by the system to select the next set of rules to fire.

Backward-chaining is a goal-driven search technique. A fact is selected as a goal and the system backs up to rules that conclude about this fact. If other unknown facts are needed in these rules then the system continues to back up to other rules that conclude about these new facts. This process continues until the system can return to the initial rules to "instantiate" the goal. ("Instantiate" refers to the assignment of belief in a fact on a truth scale that may vary from 0 to 100 or from -1 to +1.)

Depth-first searching is the most direct way to chain. Once a fact becomes known, the system continues on with its new task. Depth-first can be conducted in either a forward- or backward-chaining mode.

Breadth- first searching refers to the system continuing to examine all rules in question concerning a needed fact until all rules have been exhausted. As with depth-first searching, breadth-first searching can be conducted in either a forward- or backward-chaining mode. Breadth-first searching is the most accurate method to use while depth-first searching is usually the faster way to reach a final conclusion although it may be less accurate.

Dealing with Uncertainty

One of the most important features of Expert Systems is their ability to run even when data are unknown or uncertain. There are many techniques used in Expert Systems to handle uncertainty. These include:

- Probability theory
- Bayesian logic
- Certainty factors
- Confidence levels
- Fuzzy logic

When uncertainty is used in an expert system, rules can be fired with less than full certainty. These means that approximate answers can be given to a particular set of process variable states. By interpolating across such partial rule firings, it is possible to generate "close-enough" solutions to a problem.

The real power of expert systems shines through when uncertainty management tools such as fuzzy logic are used. In the future, by integrating Artificial Neural Network and Genetic Algorithm techniques into these systems, full adaptability and true system learning will result.

References

1. McDermott, K., Clyle, P., Hall, M. and Harris, C.A., 1992. "An Expert System for Control of No.4 Autogenous Mill Circuit at Wabush Mines", Proc. Canadian Mineral Processors, Paper 24, pp.20.

2. Freeman, N., Kemp, T. and Legg, J., 1990. "Development of Operator Guidance System for Pb-Blast Furnace Operations", Proc. Australia AI90 Conf., Perth, pp. 11.
3. Eggert, J., Folinsbee, J. and Benford, P., 1994. "SAG Mill Control at Dome Mines Using an Expert System", Proc. Canadian Mineral Processors, Ottawa, pp.16.
4. Benford, P. and Meech, J.A., 1992, "Advising Flotation Operators Using a Real-Time Expert System", Minerals Engineering, 5(10-12), 1325-1331.
5. Lacouture, B, Russell, C., Griffin, P. and Leung, K., 1991. "Copper Flotation Expert System at Mt. Isa Mines Limited", Copper91, Eds. Dobby, G.S., Argyropoulos S.A. and Rao, S.R., 429- 437.
6. Edwards, R. and Mular, A.L., 1992. "An Expert System Supervisor of a Flotation Circuit", CIM Bulletin, 84(959), 69-76.
7. Meech, J.A. and Kumar, S., 1994. "A HyperManual on Expert Systems - Version 3.0" CANMET Publication, Ottawa, 4723 electronic pages.
8. Harris, C.A. and Kosick, G.A. 1988. Expert System Technology at the Polaris Mine. Proc. Canadian Mineral Processors Conf., Ottawa, pp.25.
9. Poirier, P. and Meech, J.A., 1993. "Using Fuzzy Logic for On-Line Trend Analysis", Proc. 2nd IEEE Conference on Control Applications, Vancouver, Vol. 1, 83-86.
10. Poirier, P., Raabe, H. and Meech, J.A., 1993. Using Froth Identification in an Advisory Expert System for Copper Flotation Operations. Proc. Canadian Mineral Processors Conf., Ottawa, No. 36, pp.14.
11. Meech, J.A. and Harris, C.A., 1992. "Expert Systems for Gold Processing Plants"; Randol Gold Forum, Vancouver, B.C., 45-61.
12. Harris, C.A. and Meech, J.A., 1987. Fuzzy Logic: A Potential Control Technique for Mineral Processing. CIM Bulletin., 80 (905), 51-59.