Expert Systems for Engineering Applications

By Tan Chee Nian,
Ir. Dr Tan Chee Fai,
Dr Low Cheng Yee,
Ir. Dr Leong Wai Yie

From the first industrial revolution till the latest fourth industrial revolution, from the use of steam and water to generate force in the olden days to today's cyber-physical systems, it has been proven that digitalisation has become more prominent in our daily life. The fourth industrial revolution, which includes developments in artificial intelligence and machine learning, rapid prototyping, and genetics and biotechnology will require dramatic changes of skills sets. According to the World Economic Forum 2016, two million jobs will be created by digital industrial and service sectors with a loss of seven million jobs in traditional industrial and service sectors. Industry 4.0 also focuses on digital technologies like sensors or connectivity devices, and smart applications for manufacturing execution systems. Artificial intelligence (AI) which is closely related to cyber-physical systems is playing a paramount role in the multistage development of Industry 4.0. There are many types of AI in the current market, for instance, expert systems, artificial neural networks and fuzzy systems.

Expert systems, defined as computer systems that emulate the decision-making ability of a human expert, were among the first truly successful forms of artificial intelligence software. An expert system is defined as a programme that tries to mimic human expertise by applying the inference method to a particular body of knowledge (*Mansyur et al., 2013*). This system is able to computerise expert knowledge in a specific subject domain in a short time and provide users with easily accessible solutions in a useful and practical way.

Expert knowledge is a combination of the theoretical understanding of a problem and a collection of heuristic problem-solving rules that experience has shown to be effective in the domain. Constructing expert systems involves

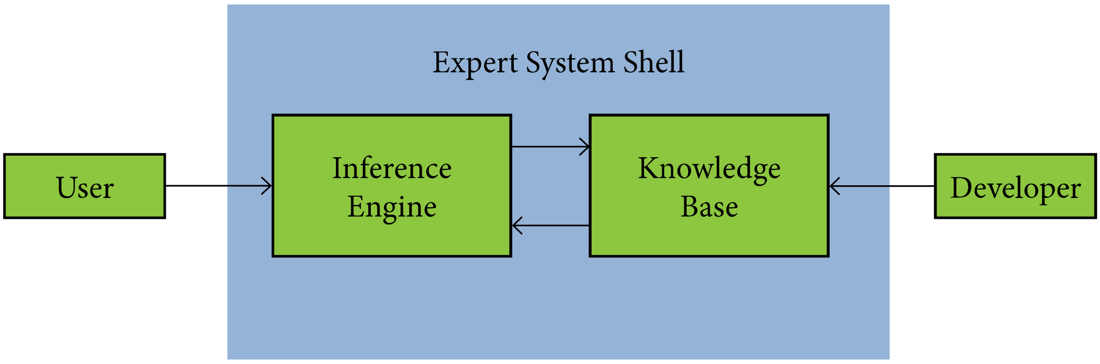

Figure 1: The structure of an expert system (P. Jackson, 1998)

obtaining knowledge from a human expert and coding it into a form that a computer may apply to similar problems (*Negnevitsky, 2005*).

Software Architecture

There are many expert system shells available in the market such as Exsys Developer, KEE, VP-Expert, KnowledgePro, Kappa-PC and others. Once the database of the system has been developed, it is followed by system development. Generally, the structure of the knowledge-based system consists of users, user interface, process information, database, knowledge base, inference engine and output.

The inference engine is an important part of an expert system which is based on rules. The inference engine scans and checks the condition given in the rules depending on the chaining process defined in the reasoning. The reasoning process is classified into two parts namely; backward or forward chaining. Rules are scanned until one is found where constraint values match the user input. The scanning resumes and the results are deduced. The final results are reported to the user. The process continues until the final selection is made. Figure 1 shows the fundamental structure of an expert system.

When a user enters the required information through the user interface, the inference engine is invoked. Then, the inference engine will try to search for the information from the knowledge base. In the knowledge base system, the knowledge base and database work together. The inference engine will process the data from the user interface based on the reasoning process defined in the reasoning database. Then, the inference engine will try to match the information with the data from the knowledge base. If the condition part of a rule in the knowledge base matches with the information given by the user, the conclusion of that rule is set as a result.

According to Figure 2, backward chaining is a type of reasoning by asking the inference engine whether a certain fact can be established. The objective is to find a rule whose conclusion matches this question, or goal. Backward chaining is often called goal-driven reasoning. Establishing the truth of a premise can also sometimes call for further rule application (*Rupnawar et al., 2016*). In backward chaining, or goal-driven reasoning, the inference engine tries to verify a fact (reach a goal) by finding rules that can prove the fact and then attempting to verify their premises. The premises in turn become new facts to be verified by other rules, and so on. Several points should be made about backward chaining in general:

- The backward chainer requires a predefined goal.
- Goals are written in the Goal Editor.
- The backward chaining process is continually trying to satisfy the goal.
- Multiple pattern rules cannot be used with the backward chainer.

There are three possible phases in the backward chaining process:

- *Expanding* - Expanding is when the backward chainer tries to further evaluate parts of the rules or the values of objects or slot pairs in order to try and satisfy the goal.

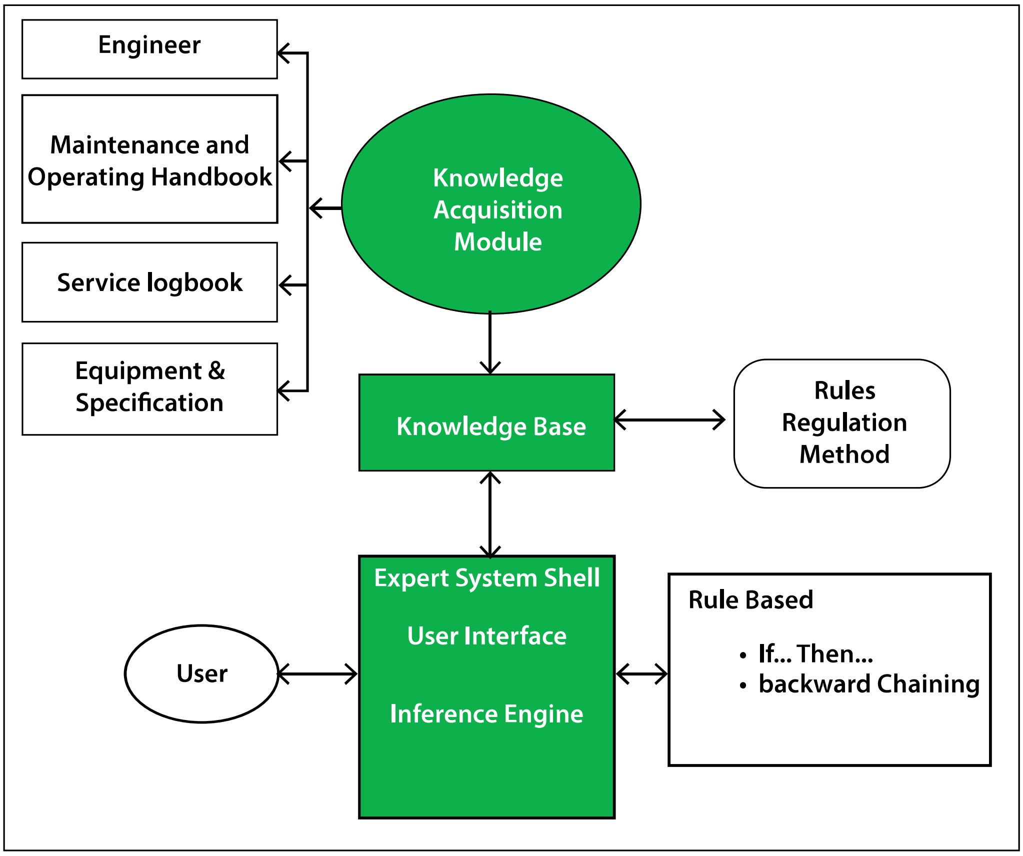

Figure 2: The designed framework for expert system shell

- *Collapsing* - Collapsing the goal is the process whereby the backward chainer tries to determine if the goal has been satisfied. The goal may have been satisfied by the recent addition of new facts caused by the Expansion process.
- *Asking* - The third phase of the backward chainer is where the backward chainer queries the user for the value of a slot once it has determined it cannot find that value on its own.

Figure 2 shows a framework design for fault diagnosis of a commercial bus design and manufacturing process.

Application

Human experts are able to perform at a high level because they know their areas of expertise. An expert system, as these programmes are often called, uses domain specific knowledge to provide "expert quality" performance in a problem domain. Generally, expert system developers acquire this knowledge with the help of human domain experts, and the system will emulate their methodology and performance. An expert system is a system with both theoretical and practical knowledge focusing on a narrow set of problems. However, an expert system is not human and thus is unable to learn from its own experience. Knowledge in an expert system must be extracted from humans and encoded in a formal language. This is the major task faced by designers of knowledge-intensive problem solvers *(Negnevitsky, 2005)*. Expert systems neither copy the structure of the human mind, nor are they mechanisms for general intelligence. They are practical programmes that use heuristic strategies developed by humans to solve specific classes of problems. Because of the heuristic, knowledge-intensive nature of expert-

Table 1: Differences between Conventional Systems and Expert Systems (Negnevitsky, 2005)

Conventional Systems	Expert Systems
Information and processing combined in a single sequential programme	The knowledge base is separated from the inference engine
The programme is never wrong	The programme could make a mistake.
Complete input data is required.	Incomplete input data is acceptable.
Changes of programme are time consuming.	Changes of programme can be made with ease.
Quantitative data	Qualitative data
Numerical data representation	Symbolic data representation.

level problem-solving, expert systems generally: (*Negnevitsky, 2005*).

1. Support inspection of their reasoning processes, both in presenting intermediate steps and in answering questions about the solution process.
2. Simplify and ease modification which enables adding and deleting skills from the knowledge base.
3. Reason heuristically, using knowledge to obtain useful problem solutions.

The reasoning of an expert system is open to inspection to provide useful information about the state of art and to explain the choices and decisions made by the programme. Explanations are important in expert systems for many reasons. Doctors or engineers who seek recommendations from an expert system must be satisfied with the solution. The explanations help people relate the advice with their existing understanding of the domain and apply it in a more confident and flexible manner. The major application area of an expert system includes (but is not limited to) the following: interpretation, prediction, diagnosis, planning design control and instruction. (*Shu, 2005*). Table 1 shows the differences between conventional systems and expert systems.

Expert systems are being applied to a variety of problem domains to assist in the decision making process (*Chen et al., 2012*). It is able to provide solutions for different problems in the industry ranging from planning to marketing strategies (*Li, 2005*) to re-engineering consultation (*Hvam et al., 2004; Ruiz et al., 2011*). Nowadays, expert systems are applied in various fields with different applications, such as:

- Constructing a nutrition diagnosis expert system (*Chen et al., 2012*)
- An expert system for diagnosing problems in boiler operations (*Artificial et al., 1992*)
- An expert system hybrid architecture to support experiment management (*Fiannaca et al., 2014*)
- A knowledge-based expert system for selection of appropriate structural systems for large spans (*Golabchi, 2008*)
- Expert system for making durable concrete for aggressive carbon-dioxide exposure (*Islam & Miah, 2011*)
- Expert system for Sri Lankan solid waste composting (*Jayawardhana et al., 2003*)
- A prototype rule-based expert system for travel demand management (*Mansyur et al., 2013*)
- Development of an expert system for the design of airborne equipment *(Kumar et al., 2004*)
- An expert system development tool for non-AI experts (*Ruiz et al., 2011*)

Figure 3: The main menu of Kappa-PC software.

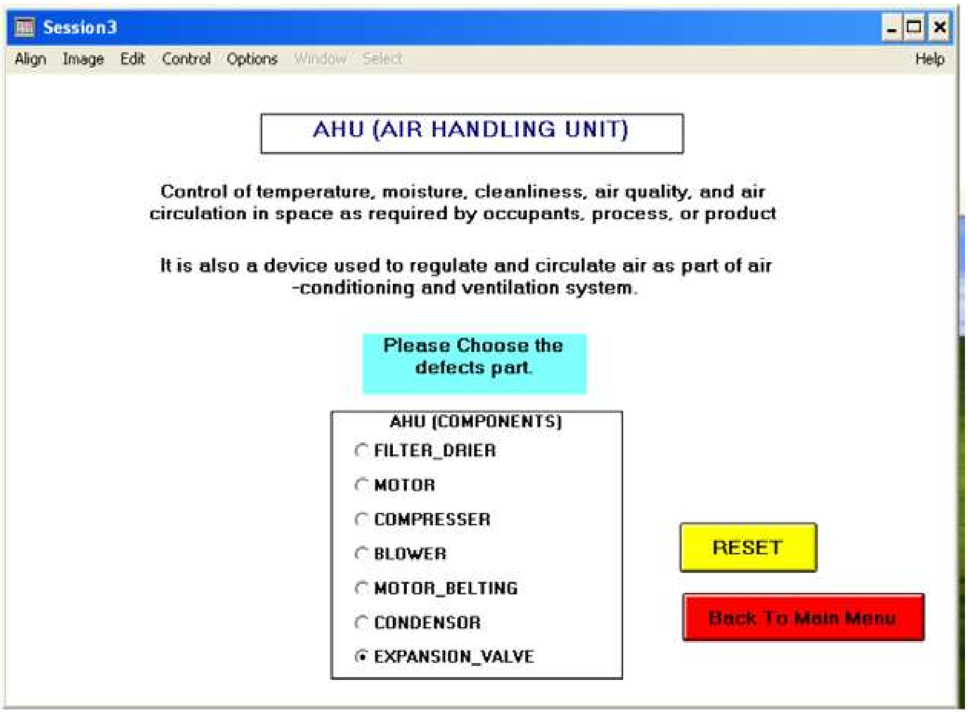

Figure 4: A fault diagnosis expert system for Air Handling Unit (AHU).

Figure 5: A Water Cooled Packed Unit (WCPU) in AHU room.

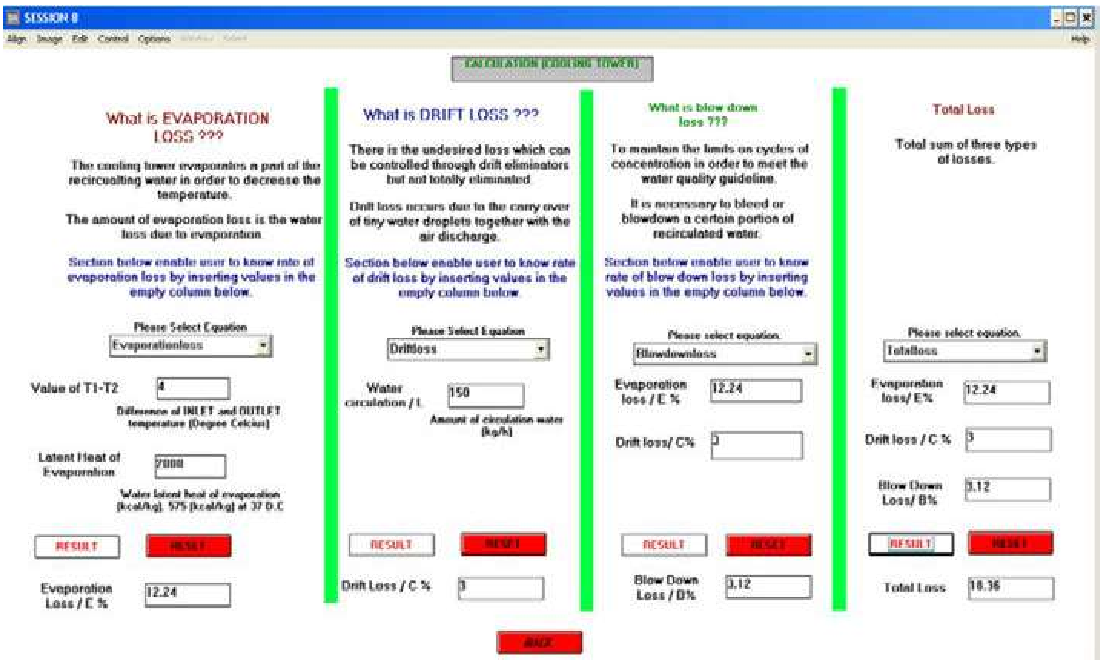

Figure 6: The calculation of cooling tower session window (value inserted)

- An expert system for fault diagnosis for gear boxes in wind turbines (*Ling et al., 2012*)

An expert system called Kappa-PC Software has been used to provide a standardised methodological approach for solving important and complex problems normally done by human experts (*Terry, 1995*). Kappa-PC is a knowledge based expert system shell that helps in developing an expert system (*Mansyur et al., 2013*). The main menu of the developed Kappa-PC system is shown in Figure 3. The main purpose of the developed expert system is to give a solution and recommendation to the person in charge when experts are not around. The system will generate friendly prompts according to the user data. Friendly prompts will be shown as advice to the user on how to tackle problems in the current situation. The recommendation prompts can give optimal advice on what to do in the next following steps or propose to send back to factory for service in a worst case scenario. Figure 4 shows a fault diagnosis expert system for an Air Handling Unit (AHU). Figure 5 shows a Water Cooled Packed Unit (WCPU) in an AHU room.

Figure 6 shows a developed expert system to calculate a cooling tower's total loss which consists of evaporation loss, drift loss, blow down loss and total loss. By identifying the values needed to obtain the total loss of a cooling tower in a building, the system acts as a platform to identify the working condition of the cooling tower as well as the alarm alert system. The algorithm input in Figure 6 can identify whether the cooling tower system is running ideally or over-designed by comparing the total loss with the design specifications.

Summary

In conclusion, an expert system is an artificial intelligence programme that is able to carry out diagnoses, interpretation, design, prediction and others useful activities to assist human experts. An expert system for air handling units has been developed by using Kappa-PC software. This can help technicians or engineers with a lack of knowledge and experience by providing them with the correct information and solution.

A well developed expert system provides useful suggestions and recommendations for the user. With such a system, time needed for diagnosing problems can be shortened when human expertise is not available. Expert systems are flexible and modular, and can be upgraded to enhance effectiveness in a more comprehensive way. ■

REFERENCE

Artificial, C, D, I., Monterrey, N, L. and Hall, H., 1992. An Expert System for Diagnosing Problems in Boiler Operations. *Expert System with Applications*, 5, pp. 323-336.

Chen, C., Lee, C. and Chen, C., 2010. To Enhance the Energy Efficiency of Chiller Plants with System Optimization Theory. *Energy & Environment*, 21(5), pp. 409-424.

Fiannaca, A., Rosa, M. L., Rizzo, R., Urso, A., and Gaglio, S., 2014. An expert system hybrid architecture to support experiment management. *Expert Systems with Applications,* 41(4), pp. 1609-1621.

Golabchi, M., 2008. A KNOWLEDGE BASED EXPERT SYSTEM FOR SELECTION OF APPRORIATE STRUCTURAL SYSTEM FOR LARGE SPANS. (*Asian Journal Of Civil Engineering (Building And Housing)*, 9(2), pp. 179-191.

Hvam, L., Malis, M., Hansen, B. and Riis, J., 2004. Reengineering of the quotation process: application of knowledge based systems. *Business Process Management Journal*, 10(2), pp. 200-213.

Islam, M. and Miah, M., 2011. Expert system for making durable concrete for aggressive carbon-dioxide exposure. *4th Annual Paper Meet and 1st Civil Engineering Congress*, pp. 978-984.

Jayawardhana, L., Manipura, A., Alwis, A. D., Ranasinghe, M., Pilapitiya, S., and Abeygunawardena, I., 2003. BESTCOMP: expert system for Sri Lankan solid waste composting. *Expert Systems with Applications,* 24(3), pp. 281-286.

Kumar, B., Shanmugam, J., Janarthanan, S., and Santhiseela, R., 2004. Development of expert system for the design of airborne equipment. *The 23rd Digital Avionics Systems Conference (IEEE),* pp. 4.1- 4.10.

Li, S., 2005. A Web-enable hybrid approach to strategic marketing planning: Group Delphi plus a Web-based expert system. *Expert System With Applications*, 29, pp. 393-400.

Mansyur, R., Rahmat. R.A.O.K. and Ismail, A., 2013. A Prototype Rule-Based Expert System for Travel Demand Management. *UNIMAS E-Journal of Civil Engineering,* 4(4), pp. 34–39.

Negnevitsky, M., 2005. *Artificial intelligence: a guide to intelligent systems*, Harlow, England: Addison-Wesley.

P. Jackson. 1998. Introduction to Expert System. Addison Wesley.

Ruiz-Mezcua, B., Garcia-Crespo, A., Lopez-Cuadrado, J. and Gonzalez-Carrasco, I., 2011. An expert system development tool for non AI experts. *Expert Systems with Applications*, 38(1), pp. 597-609.

Rupnawar, A., Jagdale, A. and Navsupe, S., 2016. Study on Forward Chaining and Reverse Chaining in Expert System. *International Journal of Advanced Engineering Research and Science*, 3(12), pp. 60-62.

Terry Anthony Byrd. (1995). Expert systems implementation: interviews with knowledge engineers, 95 (10), 3-7.

World Economic Forum, 2016. Available at [https://www.weforum.org/agenda/2016/05/industry-4-0]