行政院國家科學委員會專題研究計畫　成果報告

混合式柔性計算系統於營建知識發掘之研究(II)

計畫類別：個別型計畫
計畫編號：NSC93-2211-E-216-012-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：中華大學營建工程學系

計畫主持人：余文德

計畫參與人員：羅紹松、黃文郁

報告類型：精簡報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 94 年 10 月 6 日

行政院國家科學委員會補助專題研究計畫 ■ 成 果 報 告
□期中進度報告

混合式柔性計算系統於營建知識發掘之研究—第二年

計畫類別：■ 個別型計畫　　□ 整合型計畫
計畫編號：NSC 93－2211－E－216 －012－
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日

計畫主持人：余文德
共同主持人：
計畫參與人員： 羅紹松、黃文郁

成果報告類型(依經費核定清單規定繳交)：■精簡報告　□完整報告

本成果報告包括以下應繳交之附件：
■赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：中華大學營建工程學系

中　華　民　國　　94 年　10 月　05 日

（一） 計畫中、英文摘要

中文摘要

營建工程為一以經驗為基礎之專業，從過去執行專案中累積與發掘之營建工程知識為各營建業者(包括業主、設計者、施工單位及營建管理顧問)確保未來專案成功之基石。因此營建業也屬於一種特殊的「知識經濟」產業。然而專業知識之累積不易，因此如何從過去所完成的歷史案例資料中去發掘有用之資訊，並轉換為工程師與管理者能夠應用的實務知識，更是一件關鍵的任務。近年來由於知識發掘(KDD)與資料探勘(DM)技術的突飛猛進，使得其他產業在歷史資料加值處理與創造知識商品上得到極大進展。然而營建業在此一領域的應用卻極為有限，究其原因不外兩個主要理由：(1) 營建業對於知識發掘與資料探勘技術感到陌生而不知採用；(2) 現有的知識發掘與資料探勘技術對於營建業的特殊資料性質無法有效應用而被捨棄。然而我國營建業若欲擺脫目前之困境，朝向知識經濟轉型，則上述原障礙必須加以排除。本研究旨在探討營建業在知識發掘上所遭遇之問題，主要著眼點有二：(1) 發展出適用於處理營建特殊資料性質之資料探勘演算法；(2) 發掘出具有「可讀性」之營建工程知識，俾知識之傳承、再利用與專家驗證。為達到此一目標，本研究將廣泛地回顧現有之知識發掘與資料探勘方法，並分析營建業在應用此一技術時所遭遇之特殊問題及障礙。其次詳細回顧並分析目前資料探勘技術中最常用之各種柔性計算(Soft computing)方法(包括模糊理論、類神經網路、基因演算法、粗集合理論及案例式推理等)在營建業知識發掘應用之適用性，並針對營建業特性提出一資料探勘演算法，以處理營建資料不確定、不完整、不純及數量不足等問題。從現有的文獻得知，上述五種理論在資料探勘用各有其長處。其中模糊理論在表達專家知識以及處理不確定之資訊上已成為普遍接受的方法；類神經網路則在分佈型知識庫、平行處理及局部最佳化搜尋方面之應用極為成功；基因演算法在處理非結構性映至問題及全域搜索有強大之功能；粗集合理論在分析缺漏及不完整之資料，並用以萃取決策法則上之應用方面展現其卓越潛力；案例式推理方法在資料量極少的情況下即可提供相當有用之推理結果。本研究擬整合上述五種柔性計算理論以發展出一套可應用於營建資料庫之「工程知識發掘系統」的演算法，並依據所推導出之演算法撰寫一套電腦軟體程式，以驗證所建構之演算法的可行性與正確性。另外，將選擇營建工程與管理領域之典型資料探勘問題類型作為應用研究，探討上述工程資料探勘中常見之資料缺漏、資料不確定性、資料不純等問題，及所建構之系統在解決實際問題時之效益。本研究之成果除開發一套「混合式柔性計算系統」電腦軟體程式以對歷史資料做加值處理並創造知識商品外，亦可提供營建業界應用知識發掘與資料探勘技術之參考案例。

關鍵字：資料探勘、知識發掘、柔性計算、類神經模糊系統

Construction has been conceived as an experience-based discipline. Knowledge learned from previous projects plays important role in successful performance of future projects. This has made construction an ideal industry for the knowledge-based economy. However, modern KDD (knowledge discovery in databases) or DM (data mining) technologies are not yet widely exploited and adopted in the field of construction engineering and management. This is due to two main causes: (1) the construction industry is not familiar with KDD and DM technologies; (2) the existing KDD and DM technologies do not fit the special characteristics of data in the field of construction engineering and management. Should the construction industry be pursuing knowledge-based economy, obstacles caused by the above two reasons must be removed and the reusable domain knowledge must be generated from historical data. For this end, this research is proposed to tackle problems encountered in knowledge discovery in real world construction databases. The focuses are: (1) development of DM algorithms for the knowledge discovery of the unique construction data characteristics; (2) generation of human understandable knowledge, so that domain experts can visualize and verify it. At first, the existing KDD and DM methods are reviewed. Problems faced in applications of KDD and DM for construction engineering and management are broadly surveyed to identify the special characteristics of construction data, which hinder the implementation of KDD and DM in construction industry. The existing soft computing techniques, including fuzzy sets, artificial neural networks, genetic algorithms, rough sets, and case-base reasoning, are thoroughly reviewed to propose the most appropriate hybridization for handling unique domain data characteristics. The data mining algorithms are developed to discover knowledge from construction data, which are usually uncertain, incomplete, partially true, and scarce in their nature. A Hybrid Soft Computing System will be developed for implementation of data mining and knowledge discovery in construction industry. Various real world databases provided by the industrial partners are used for validation and verification of the proposed system. The anticipated results of the proposed research will provide not only an effective tool for KDD implementation in construction industry but also an important reference for future researchers, industrial practitioners, public work officials, and school educators.

Keywords: Data mining, knowledge discovery in databases, soft computing, neuro-fuzzy systems

（二） 報告內容

1. 前言

Construction has been conceived as an experience-based discipline (Ardery, 1991); therefore, knowledge acquired from previous works plays a key role for successful performance of the new projects. Not only the construction know-how's of the contractors, but also the design capabilities of the design firms and the management skills of CM consultants rely heavily on such knowledge. This has made construction an ideal industry for the knowledge-based economy (Ofori, 2003). In the past two decades, tremendous efforts have been contributed to the formation and application of construction knowledge provided by experienced engineers and managers to the new construction projects. However, modern KDD (knowledge discovery in databases) or DM (data mining) technologies were not yet widely exploited and adopted in the field of construction engineering and management to acquire valuable knowledge from historic databases, which results in leaking of knowledge from construction firms. This was due to two main causes: (1) the construction industry is not familiar with KDD and DM technologies (Yu and Yang, 2002; Yu and Skibniewski, 1999); (2) the existing KDD and DM technologies do not fit the special characteristics of data in the field of construction engineering and management (Yu and Yang, 2002).

2. 研究目的

As a result, this research is intended for two objectives: (1) development of DM algorithms for the knowledge discovery of the unique construction data characteristics; (2) generation of human understandable knowledge, so that domain experts can visualize and verify it. To achieve these goals, the existing KDD (Han and Kamber, 2000) and DM (Fayyad and Uthurusamy, 1996; Mitra et al., 2002) techniques are reviewed. Problems faced in applications of KDD and DM for construction engineering and management are analyzed to identify the special characteristics of construction data, which hinder the implementation of KDD and DM in construction industry. The existing soft computing techniques, including fuzzy sets (Zadeh, 1965), artificial neural networks (Tickle et al., 1998), genetic algorithms (Flockhart and Radcliffe, 1996), and case-based reasoning (Yang and Yau, 2000), are reviewed to propose the most appropriate hybridization for handling unique domain data characteristics. The data mining algorithms are developed to discover knowledge from construction data, which are usually uncertain, incomplete, and scarce in their nature. A Hybrid Soft Computing System (HSCS) is developed for implementation of data mining and knowledge discovery in construction. Case-studies applying the proposed HSCS for KDD from construction databases are selected to verify the proposed method with problems of data incompleteness, uncertainty, and scarcity.

3. 文獻探討

KDD Process

KDD was defined by Fayyad and Uthurusamy (1996) as the process to identify valid, novel, potentially useful, and ultimately understandable patterns in data. The general concept of KDD is to transfer raw data (usually of no direct use) into useful and valuable knowledge, which makes patterns understandable to humans. A general process of KDD consists of the following steps (Han and Kamber, 2000; Cabena et al., 1998):

(1) Understanding the domain problem—such as the relevant prior knowledge and goals of the application;

(2) Extracting the target data set—e.g., selecting a data set or focusing on a subset of variables;

(3) Data cleaning and pre-processing—e.g., noise removal and handling of missing data. Data from real-world sources are often erroneous, incomplete, and inconsistent, perhaps due to operation error or system implementation flaws. Such low quality data needs to be cleaned before data mining;

(4) Data integration—e.g., integrating multiple, heterogeneous data sources;

(5) Data reduction and projection—tasks such as finding useful features to represent the data (depending on the goal of the task) and using dimensionality reduction or transformation methods;

(6) Choosing the function of data mining—deciding the purpose of the model derived by the data-mining algorithm (e.g., summarization, classification, regression, clustering, web mining, image retrieval, discovering association rules and functional dependencies, rule extraction, or a combination of these);

(7) Choosing the data mining algorithm(s)—selecting method(s) to be used for searching patterns in data, such as deciding on which model and parameters may be appropriate;

(8) Data mining—searching for patterns or rules of interest in a particular representational form or a set of such representations;

(9) Interpretation—interpreting the discovered patterns, as well as the possible visualization of the extracted patterns. One can analyze the patterns automatically or semi-automatically to identify the truly interesting/useful patterns for the user;

(10) Using discovered knowledge—incorporating the discovered knowledge into the performance system, taking actions based on knowledge.

Data Mining (DM)

DM is the most critical step in the KDD process. KDD refers to the overall process of turning raw data into value-added knowledge, and DM is the core mechanism that extracts useful knowledge from databases. Fayyad et al. (1996) considered DM as an interdisciplinary field with a general goal of predicting outcomes and uncovering relationships in data. Mitra et al. (2002) defined DM as a process using automated tools, that employs sophisticated algorithms, to discover hidden patterns, associations, anomalies and/or structure from large amounts of data stored in data warehouses or other information repositories. Furnkranz et al. (1997) addressed that DM tasks can be descriptive (i.e., discovering interesting patterns describing the data) and predictive (i.e., predicting the behavior of the model based on available data). DM involves fitting models to or determining patterns from observed data. The fitted models play the role of inferred knowledge. Mitra et al. (2002) concludeds that, a typical DM algorithm constitutes some combination of the following components: (1) model—the function of the model (e.g., prediction, association) and its form (e.g., linear discriminates, ANN). A model contains parameters that are to be determined from the data; (2) preference criterion—a basis for judgment of better model, depending on the given data; (3) search algorithm—that is a specific algorithm for finding particular models and parameters, which significantly influences the mining results.

Characteristics of Construction Databases

Construction industry differentiates itself from other industries in the way of production, the environment of workspace, the format of products, and the constitution of organization. Such characteristics have contributed to the special problems confronting KDD in construction databases. The following point out three types of problems existing in construction databases.

Data Scarcity

Yu and Liu (2005) defined two types of data scarcity are found in construction databases: (1) scarcity in data volume—as the construction projects are unique in their nature and huge in their scales, it is very difficult to accumulate sufficient data required by available DM techniques; (2) sparsity in data coverage—the data are insufficiently and unevenly distributed in the range of

interested domain, i.e., the uneven distribution is due to the lack of certain types of projects that the firm has never performed before, so the associated information of that project type is missing. While encountering data scarcity problems, many DM techniques (such as ANN-related methods) may fail due to the under-determination of model variables. The sparsity in data may cause severe problems in generalization of predicative model and result in misleading inferences.

Data Incompleteness

Data incompleteness is common in traditional construction databases due to the harsh outdoor environment, the attitudes of workmen who collect the data, and merging of different databases. Two type of incompleteness were defined by Yu and Lin (2005) as follows: (1) missing data—incomplete coverage of data in some intervals of the universe of discourse; (2) missing values—incomplete information in some interesting attributes of a dataset. Han and Kamber (2000) describes six traditional approaches for processing incomplete data include: (1) ignoring the tuple; (2) filling in the missing value manually; (3) using a global constant to fill in the missing value; (4) using the attribute mean to fill in the missing value; (5) using attribute mean for all samples belonging to the same class as the given tuple; and (6) using the most probable value to fill in the missing value. Traditional approaches for processing incomplete data may be misleading or biased. It's better to perform DM directly on the raw data to retain the essential property of data.

Data Uncertainty

Major uncertainties in construction data may be due to the weather effects on outdoor construction operations. Moreover, the project-based delivery system of construction industry induces uncertainties from the various team members that are organized tentatively for a specific project. As the uncertainty is inevitable, DM algorithms should be able to tackle the uncertainty of data in construction.

Soft Computing Techniques

In traditional hard computing, achieving precision, certainty, and rigor in calculation are the primary goals. On the contrast, soft computing paradigm perceives that precision and certainty are costly so that computational schemes should exploit (wherever possible) the tolerance for imprecision, uncertainty, and approximate reasoning for obtaining low-cost solutions (Mitra and Hayashi, 2000). The above perspective is especially true for construction industry, where historical data are usually uncertain, incomplete, and scarce in their nature. Therefore, the flexible information processing capability of soft computing provides promising solution for DM and KDD in construction industry. That is, devise the DM algorithms that lead to an acceptable solution at low cost by seeking for an "approximate", instead of "accurate" or "exact", solution to an unstructured or ill-defined problem (Zadeh, 1965).

Due to its unique characteristics, the model and the associated search algorithms adopted for DM of construction databases should be specialized to tackle the abovementioned problems. Among the many existing data mining algorithms, soft computing techniques (involving fuzzy sets, neural networks, genetic algorithms, and rough sets) are most widely applied for a KDD process (Mitra et al., 2002). The fuzzy sets (FSs) provide a natural framework for the process in dealing with uncertainty (Pedrycz, 1998). Artificial Neural networks (ANNs) (Tickle et al., 1998) and rough sets (RSs) (Hirano, et al., 2002) are adopted for classification and rule generation. Genetic algorithms (GAs) are involved in various optimization and search processes, like query optimization and template selection. Other approaches like case-based reasoning (CBR) (Yang and Yau, 2000) and decision trees (Furnkranz et al., 1997) are also widely employed to solve data mining problems.

4. 研究方法

PROPOSED HYBRID SOFT COMPUTING SYSTEM (HSCS)

Framework

Basic Network

In this section, the knowledge representation model and the computational algorithms of the proposed hybrid soft computing system (HSCS) are proposed. In order to extract the readable knowledge, the linguistic Fuzzy IF-THEN rules are adopted for knowledge representation of the proposed system. For this end, a fuzzy inference system (FIS) is developed for HSCS to perform human-like reasoning process of the discovered knowledge. Previous researchers have shown that the self-organizing capabilities of ANNs can be adopted for automatically construction of the FIS (Lin and Lee, 1991; Jang, 1993; Wang and Mendel, 1992). Therefore the neuro-fuzzy system (NFS) became a appropriate choice for the proposed HSCS. Among the many existing NFSs, the FALCON model (Lin and Lee, 1991) adopts Mamdani general fuzzy decision rules and is selected for the proposed system. Figure 1 shows the structure of FALCON model.

A standard FALCON comprises five layers. Each layer consists of nodes with proper numbers of fan-in and fan-out connections represented by weights assigned to the nodes. The fan-in connections connect the nodes of the previous layer with the nodes of the current layer. The fan-out connections connect nodes of the current layer with nodes of the subsequent layer. In Layer 1, the values of input attributes are transmitted directly into FALCON. In Layer 2, the fuzzification on input attribute values from Layer 1 is performed, so that the input data are converted into fuzzy sets (or linguistic terms). The nodes at Layer 3 represent fuzzy rules. The connections between the second and the third layers represent the pre-conditions of fuzzy IF-THEN rules. Through these links, all pre-conditions of a fuzzy rule are linked to the associated rule node. Therefore, the operation performed at rule nodes is fuzzy AND. Usually, intersection is adopted for neuro-fuzzy decision systems. That is, the minimum of memberships from all pre-conditions is selected as the fired strength of this rule. The links between Layer 3 and Layer 4 represent the consequences of fuzzy rules. There should be no more than one consequence for each fuzzy rule node in a single output network. Thus, only one link connects one rule node at Layer 3 with one term node at Layer 4. The nodes and links of Layer 5 act as the defuzzifier. After defuzzification, the conclusion or decision is derived. The above structure is identical to an FIS. Not only the knowledge is represented in the format of human readable IF-TEHN rules in FALCON, but also the data processing flow in FALCON is also similar to human reasoning process. The FALCON model is tentatively considered as the basic structure of FIS for the proposed system.

Variable-Attribute Network Structure (VANS)

Similar to other neuro fuzzy systems, the traditional FALCON accepts only data with complete attribute values. Any missing attribute value will cause difficulty in performing Fuzzy AND operation in Layer 3 of FALCON. Further propagations can not proceed consequently, and thus the system output can not be derived from the network at Layer 5.

In order to improve this problem, a Variable Attribute Network Structure (VANS) was proposed by Yu and Lin (2005). The VANS adopts a flexible network structure that can adjust to the available attribute values in processing every single input dataset. Considered the FALCON in Figure 2, where input attribute *a* is missing. The rule nodes connecting to fuzzy terms of attribute *a* are prohibited from further propagation. The idea of VANS is to ignore the attributes with missing values. Thus the network of Figure 2 degrades to the one with only input attribute *b* and *c*, where the attribute *a* (with missing attribute value) and its associated fuzzy term nodes are deleted from the network. The signal propagation process of the degraded network follows the rules of original FALCON. The modified network is named Variable-attribute Fuzzy Adaptive Control Network (VaFALCON) (Yu and Lin, 2005). The HSCS based on VaFALCON is able to process the incomplete data with any combination of missing attributes.

Algorithms

The computational algorithms of the proposed HSCS consist of three categories: (1) Self-organization; (2) Supervised learning; (3) Global search. Following sub-sections describe these steps.

Self-organization

The first step for constructing a HSCS is to determine the primitive fuzzy partitions, the membership functions of the input and output fuzzy terms, and the primary structure of the rule base. The self-organized learning process is employed for this purpose. There are two learning stages in the self-organized learning process: (1) determining the centers and spreads of the

membership functions associated with the input and output nodes by Kohonen learning rule (Kohonen, 1988); (2) determining fuzzy logic rules by reinforcement competitive learning (Yu and Skibniewski, 1999).

The Kohonen learning rule consists of two stages:

Similarity matching stage: $$\left|x - \hat{w}_i^k\right| = \underset{1 \le j \le n}{Min}\left\{\left|x - \hat{w}_j^k\right|\right\} \quad (1)$$

In this stage, the most similar cluster for input data x is found to be the ith cluster by minimizing the difference between x and the center of the cluster ($\hat{w}_i^k$), where superscript k represents the kth iteration and $\hat{w}$ means a normalized value of the cluster center (w).

Updating stage: $$\hat{w}_i^{k+1} = \hat{w}_i^k + \eta^k\left(x - \hat{w}_i^k\right),$$
$$\hat{w}_j^k = \hat{w}_j^k, \text{ for } j = 1, 2,..., \text{n} \quad j \neq i. \quad (2)$$

In equation (2), η^k is a proper learning coefficient at the kth iteration. In the updating stage, the center of the ith cluster, the *winner* cluster, at the kth iteration ($\hat{w}_i^k$) is adjusted toward the incoming training data, x. The rest of the clusters are kept the same. Since only the winner cluster is adjusted, the rule is also called the winner-takes-all learning rule.

The reinforcement competitive learning rule consists of three stages:

Winner competition stage: $$\mu_j^k = e^{-\left(\frac{o^k - m_j^k}{\sigma_j^k}\right)^2}, \quad \mu_i^k = \underset{1 \le j \le m}{Max}\ \mu_j^k. \quad (3)$$

Reinforcement learning stage: $$w_{li}^{k+1} = w_{li}^k + \xi\mu_i^k\,\mu_l^k, \quad \text{for } i = 1,2,...,\text{m}; \ l = 1,2,...,\text{L}. \quad (4)$$

Rule selection stage: $$w_{li} = 1 \quad \text{if } w_{li} = \underset{1 \le i \le m}{Max}(w_{li}),$$
$$w_{lj} = 0 \quad \text{if } j \neq i, \quad \text{for } l = 1,2,...,\text{L}. \quad (5)$$

In the winner competition stage, the most strongly responding node of the output term layer is found. In the reinforcement learning stage, the connection between the fired rule nodes and the most strongly responding term nodes is reinforced by increasing the connection weight. The learning process can be performed for only one cycle (i.e., all training examples are fed into the network for only once) or many cycles to differentiate the correlative and non-correlative nodes between Layer 3 and Layer 4. Finally, the strongest connection between a rule node and one of the term nodes is set to be equal to one and the rest of the connections are disconnected.

Supervised learning

In the supervised learning, parameters of the membership functions for input and output linguistic terms are fine-tuned by back-propagation algorithms. The supervised parameter learning process consists of three basic steps: (1) forward data propagation; (2) backward error propagation; (3) parameter adjustments.

In the forward data propagation, the input data are first fuzzified in Layer 2 by input linguistic term nodes. Then, through pre-condition connections, the firing strength of each rule node is calculated based on *fuzzy AND* operation. The firing strengths are then propagated via the consequence links to the output linguistic term nodes and aggregated based on *fuzzy OR* operation. Finally, the crisp output is generated by defuzzification of output linguistic terms.

The backward error propagation begins with calculation of the difference between the actual output of the forward pass and the desired output provided by the training example. The error function E and error signal δ_5 at Layer 5 are defined as follows:

$$E = \frac{1}{2}(d(t) - o(t))^2, \quad (6)$$

$$\delta^5 = \text{calculated output - desired output} = d(t) - o(t), \quad (7)$$

where t indicates the time sequence of the learning iteration. Since only a single output is considered in the proposed methodology, there is no summation in Equation (6). In each layer, the error signal is calculated and used for back propagation. That is, the error signal of Layer 4 (δ^4) is a function of δ^5, and so forth. The parameter adjustment principle is based on the steepest gradient descent method, which can be described as follows:

$$\Delta w = \eta\frac{\partial E}{\partial w}, \qquad w(t+1) = w(t) + \Delta w, \quad (8)$$

where w can be any parameter to be adapted, η is a proper constant.

Global search

Previous research has found that traditional FACLON encountered severe local minimum problems while dealing with complex construction data (Yu and Skibniewski, 1999). In this paper, the messy genetic algorithm (mGA) is adopted to variably construct the fuzzy rule base in the NFS in light of global search. The traditional simple genetic algorithm (sGA) is a non-deterministic search algorithm based on the ideas of genetics. While applying to NFSs, the problem arises with the encoding of the NFS parameters. In an sGA, a coded chromosome is in fixed length that highly fit allele combinations are formed to obtain a convergence towards global optima. Unfortunately the required linkage format (or the structure of the NFS to be coded) is not exactly known and the chance of obtaining such a linkage in a random generation of coded string is poor. Although inversion and reordering methods can be used to adaptively search tight gene ordering, these are too slow to be considered useful. Some researchers had turned to mGAs for constructing NFSs (Chowdhury and Li, 1997). The primary difference between an mGA and a regular sGA is that the mGA uses varying string lengths; the coding scheme considers both the allele positions and values; the crossover operator is replaced by two new operators called *cut* and *splice*; and it works in two phases—*primordial* phase and *juxtapositional* phase. The selection mechanism is as in regular GA but is executed in *primordial* and *juxtapositional* phases. During the *primordial* phase, the population is first initialized to contain all possible building blocks of a particular length; thereafter only the selection operator is applied. This results in enriched population of building blocks whose combination will create optimal or near optimal strings. Also, during this phase, the population size is reduced by halving the number of individuals at specified intervals. The *juxtapositional* phase follows the primordial phase, and here the GA invokes the cut, splice and the other GA operators. With the flexible structure-learning scheme of mGA, the optimal NFS rule base can be searched globally.

Application of mGA in learning NFS structure is not of no objections. All currently available algorithms for NFS learning are limited to adapt the rule nodes and membership function types separately. However, the optimal parameters of membership functions in the NFS are not guaranteed with these algorithms. While coping with the proposed learning process depicted in Figure 3, the parameters of membership functions are preliminary determined by Kohonen's learning rules, and thus are not optimal. The proposed algorithms adopt the error back-propagation algorithm to fine-tune the parameters of membership functions after certain cycles of mGA adaptations. This setup equips the proposed mGA with powerful local search capability. Therefore the proposed NFS learning algorithms are both globally and locally optimal in their search process.

Integrated learning process

The integrated learning process of the proposed system is shown in Figure 3, where the learning process is divided into two phases: (1) Phase I—Preliminary Structuring Phase that constructs the primitive structure of the neuro-fuzzy knowledge-based system using Kohonen's learning rule (Kohonen, 1988) and a reinforcement competitive learning rule proposed by Yu and Skibniewski (1999); (2) Phase II—Parameters and Structure Optimization Phase that optimizes NFS structure with a messy GA (mGA) and fine-tunes the membership functions of the NFS with error back-propagation method.

Integrated Knowledge Discovery Process

The proposed HSCS is not only capable of processing uncertain, incomplete, and scarce data for construction databases, but also providing human understandable fuzzy IF-THEN rules for decision makers. This section describes the complete process of knowledge discovery process for the proposed hybrid soft computing approach.

Data preprocessing

The first step in knowledge discovery process is the pre-treatment of data, since without quality data there cannot be quality mining results. The data stored in construction databases are usually *dirty*, which means incomplete, noisy, and inconsistent. The incomplete data means lacking attribute values, lacking certain attributes of interest, or containing only aggregate data; the noisy data contain errors or outliers; the inconsistent data contain discrepancies in codes or names. Such dirtiness should be cleaned before knowledge discovery can proceed. The task of

pre-treatment is named data preprocessing.

There are several major tasks in data preprocessing including (Han and Kamber, 2000): (1) Data cleaning—smoothing noisy data, identifying or removing outliers, and resolving inconsistencies; (2) Data integration—integrating multiple databases, data cubes, or files; (3) Data transformation—performing normalization and aggregation of data; (4) Data reduction—obtaining reduced representation in volume but produces the same or similar analytical results; (5) Data discretization—partitioning data for optimal representation of qualitative or quantitative data.

Data mining

The next step of knowledge discovery is to perform data mining on the preprocessed data. The HSCS proposed in this paper provides hybrid soft computing algorithms that integrate FLDS, ANN, and mGA techniques for constructing VaFALCON. The learning process has been described in Figure 3 including two phases: preliminary structuring and parameters and structure optimization. In traditional FALCON, preliminary structuring was achieved by self-organized Kohonen Future Map (Kohonen, 1988) and competitive learning rule (Yu and Skibniewski, 1999), and the parameters optimization was performed by back-propagation (Lin and Lee, 1991). However, the structure optimization function was lacked in original FALCON. In the proposed HSCS, the structure of the NFS is optimized in Phase II by global search with mGA algorithm and convergence with BP, so that the erroneous rule structure organized Phase I can be improved to find out the optimal rule base for the FLDS.

Knowledge presentation

The proposed system provides desirable fuzzy IF-THEN rules as the result of data mining, so that human decision makers are able to realize and validate the patterns found. A fuzzy IF-THEN rule in the proposed FALCON network consists of two parts: (1) preconditions—a set of fuzzy linguistic variables characterized by a set of fuzzy terms defined by their associated membership functions; (2) consequence—a single fuzzy linguistic variable characterized by a fuzzy term defined by its associated membership function. In the preconditions, each fuzzy linguistic variable is associated with a parameter related to the system input. The consequence is the output that the decision maker is seeking. After data mining, a set of fuzzy IF-THEN rules are found to form a knowledge base.

5. 結果

System Testing Results

Data Scarcity

The R_s for each of the three cases are calculated using Equation (10) and (11). Basically, 80% of the collected data were used for training, and the other 20% used for testing. The results of testing were benchmarked with a hybrid soft computing approach, which integrating CBR with other numeric learning schemes, proposed by Yu and Liu (2005). In order to compare the testing results, the fuzzy partitions of the three cases were controlled the same as those in literature. The R_s‘s for the three cases are shown in Table 1. The testing results on data scarcity are shown in Table 2. It is found from Table 2 that the proposed HSCS performs similar to the hybrid CBR approach and much better than the other three traditional methods (CBR, ANN, and ANFIS). Even though the hybrid CBR approach slightly outperforms HSCS, it does not provide meaningful fuzzy IF-THEN rules for explicit knowledge presentation.

Data Incompleteness

The data incompleteness was tested with *POI* at five different levels: 0%, 5%, 10%, 15%, and 20%. The testing results on incompleteness are shown in Table 3. The information recovery ratio (*IRR*) defined by Yu and Lin (2005) is used here to show the capability of knowledge recovery under data incompleteness. It is found that the *IRR* is generally higher than the *POI*. That is, the available information is fully utilized for KDD.

Uncertainty

The uncertainty of data is modeled by Equation (12), where the original data were disturbed with random number for various uncertainty ranges, *p*. In the experiment, the *p* is controlled at 5%, 10%, 15%, and 20% for training sets. The testing sets were not disturbed. The testing results on uncertainty are shown in Table 4. The testing results show that the proposed HSCS is more

sensitive to uncertainty than the other two types of problems. However, the capability of knowledge discovery of HSCS is still robust as long as the disturbance of data is not very severe (e.g., higher than 15% of the data range).

Knowledge Presentation

The knowledge mined by the proposed HSCS is stored in the fuzzy rule base in form of matrix indicating the connection of rule nodes and output membership functions. The fuzzy rule base contains a set of fuzzy IF-THEN rules. Each fuzzy IF-THEN rule consists of a set of fuzzy linguistic terms for expressing the values of attributes in the precondition part; it also contains a set of fuzzy linguistic terms for the single output in the consequence part. Each fuzzy linguistic term is associated with a fuzzy membership function. The fuzzy partitions for the input parameters are subjectively determined by the decision maker and shown in the third column of Table 1. As a result, there are totally $\prod_{x=1}^{n_{in}} V_{fp}^{in}(x)$ fuzzy IF-THEN rules for each case; where V_{fp}^{in} is the vector of fuzzy partitions in the input layer as defined in Equation (11). Therefore, there are [3×2×2×3]=36 rules for Case I, [2×2×2×3]=24 rules for Case II, and [2×2×2×3]=24 rules for Case III.

The fuzzy membership functions of the linguistic terms for the input/output attributes of the three cases are shown in Figure 4 to 6, respectively. Each fuzzy decision rule is defined by multiple preconditions and single consequence. The relationships of adjacent layers indicating preconditions and consequence are stored in matrices. An example of the fuzzy IF-THEN rules for Case I is as follow:

"IF **type-of-earth-retaining-method** is *Simple* AND **No.-of-floors-above-ground** is *Small* AND **No.-of-floors-underground** is *Small* AND **total-floor-area** is *Small*,
THEN **construction-cost** is *Medium*."

The fuzzy IF-THEN rules can be visualized and evaluated manually by the domain experts. By investigating all fuzzy IF-THEN rules, the knowledge acquired from data mining process can be verified manually.

6. 結論與建議

This paper proposed a hybrid soft computing approach, namely HSCS, for mining of construction databases. The proposed HSCS integrates FLDS, ANN, and mGA to form a new paradigm for knowledge discovery of construction databases. The proposed approach combines several merits of the soft computing techniques, such as the human understandable fuzzy IF-THEN rules, the learning ability of ANN, and the global searching of mGA. Such hybridization offers desirable features for problems confronted in KDD of construction databases. Three cases of real world construction data repositories were tested with HSCS to verify its capability in discovering knowledge from scarce, incomplete, and uncertain databases. The testing results show that the proposed HSCS provides promising solution for KDD in construction.

The proposed HSCS is also able to mine fuzzy IF-THEN rules that can be visualized and verified by domain experts; however, the resulted knowledge base is too huge for human expert to verify manually. Some rule pruning or screening method should be developed to reduce the rule base mined by HSCS in order to make such system realistic for practical usage.

參考文獻

Ardery, E. R. (1991). "Constructability and constructability programs: White paper." *J. of Construction Engineering and Management*, ASCE, V. 117, N. 1, pp. 67-89.

Cabena, P., Hadjinian, P., Stadler, R., Verhees, J., and Zanasi, A. (1998). *Discovering Data Mining: from Concept to Implementation*, Prentice-Hall, NJ.

Chowdhury, M. and Li, Y. (1997). "Evolutionary reinforcement learning for neurofuzzy control." *Proc. Seventh International Fuzzy Systems Association World Congress (IFSA'97)*, volume II, pp. 434-439, Prague, Czech Republic.

Fayyad, U., Haussler, D., and Stolorz, P. (1996). "Mining scientific data." *Commun. ACM*, Vol. 39, pp. 51-57.

Fayyad, U. and Uthurusamy, R. (1996). "Data mining and knowledge discovery in databases." *Commun. ACM*, Vol. 39, .pp. 24-27.

Flockhart, I. W. and Radcliffe, N. J. (1996). “A genetic algorithm-based approach to data mining”, *Proc. 2nd Int. Conf. Knowledge Discovery Data Mining (KDD-96*). Portland, OR, USA, Aug. 2-4, 1996, p. 299.

Furnkranz, J., Petrak, J., and Trappl, R. (1997). “Knowledge discovery in international conflict databases.” *Applied Artificial Intelligence*, Vol. 11, pp. 91-118.

Han, J. and Kamber, K. (2000). *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Diego, U.S.A.

Hirano, S., Tsumoto, S., Okuzaki, T., Hata, Y., and Tsumoto, K. (2002). “Analysis of Biochemical Data Aided by a Rough Sets-Based Clustering Technique.” *International Journal of Fuzzy Systems*, Vol. 4, No. 3, pp. 759-765.

Jang, J. S. (1993). “ANFIS: Adaptive-network-based fuzzy inference system.” *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 23, No. 3, pp. 665-685.

Kohonen, T. (1988). *Self-organization and Associative Memory*, Springer-Verlag, Berlin, German.

Lin, C. T. and Lee, C. S. G. (1991). “Neural-network-based fuzzy logic control and decision system.” *IEEE Transactions on Computers*, Vol. 40, No. 12, pp. 1320-1336.

Mitra, S. and Hayashi, Y. (2000). “Neuro-fuzzy rule generation: Survey in soft computing framework.” *IEEE Trans. Neural Networks*, Vol. 11, pp. 748-768.

Mitra, S., Pal, S. K., and Mitra, P. (2002). “Data mining in soft computing framework: A survey”, *IEEE Transactions on Neural Networks*, Vol. 13, No. 1, pp 3-14.

Ofori, G. (2003). “Preparing Singapore's construction industry for the knowledge-based economy: practices, procedures and performance.” *Construction Management and Economics*, Vol. 21, No. 2, pp. 113-125.

Pedrycz, W. (1998). “Fuzzy set technology in knowledge discovery.” *Fuzzy Sets and Systems*, Vol. 98, pp. 279–290.

Tickle, A. B., Andrews, R., Golea, M., and Diederich, J. (1998). “The truth will come to light: Directions and challenges in extracting the knowledge embedded within trained artificial neural networks.” *IEEE Transactions on Neural Networks*, Vol. 9, pp. 1057-1068.

Wang, L. X., and Mendel, J. M. (1992). “Back-propagation fuzzy systems as nonlinear dynamic system identifiers.” *IEEE Trans. on Neural Networks*, Vol. 3, No. 5, pp. 807-814.

Yang, J. B. (1997) “An integrated knowledge acquisition and problem solving model for experience-oriented problems in construction management,” *Dissertation in Partial Fulfillment of Requirements for Degree of Ph.D.*, National Central University, Chungli, Taiwan.

Yang, J. B. and Yau, N. J. (2000). “Integrating case-based reasoning and expert system techniques for solving experience-oriented problems”, *Journal of the Chinese Institute of Engineers*, Vol. 23, No. 1, pp. 83-95.

Yau, N. J.; and Yang, J. B. (1998) “Applying case-based reasoning technique to retaining wall selection,” *Automation in Construction*, Vol. 7, No. 4, pp. 271-283.

Yu, J. S. (2001). “Developing building cost estimating system using case-based reasoning approach.” *Master Thesis*, Department of Civil Engineering, National Central University, Chungli, Taiwan. (in Chinese)

Yu, W. D., and Lin, H. W. (2005) “A VaFALCON neuro fuzzy system for mining of incomplete construction databases,” *Automation in Construction*, 13 pp. (accepted, in press)

Yu, W. D., and Liu, Y. C. (2005) “Hybridization of CBR and numeric soft computing techniques for mining of scarce construction databases.” *Automation in Construction*, 14 pp. (accepted, in press)

Yu, W. D. and Skibniewski, M. J. (1999). “A neuro-fuzzy computational approach to constructability knowledge acquisition for construction technology evaluation.” *Journal of Automation in Construction*, Vol. 8, No. 5, pp. 539-552.

Yu, W. D. and Yang, J. B. (2002). *Final Report on the Development of a Neuro-Fuzzy Knowledge-based System for Construction Conceptual Estimation*, CECI, Taipei, Taiwan. (in Chinese)

Zadeh, L. A. (1965). “Fuzzy sets.” *Information and Control*, Vol. 8, No.3, 338–353.

附表

Table 1 Scarcity ratio of the three cases

Case	Input		Output	No. of training sets	No. of testing sets	NV_{model}	R_s
	No. of inputs	Fuzzy partition	Fuzzy partition				
I	4	[3×2×2×3]	[3]	22	3	62	2.818
II	4	[2×2×2×3]	[3]	24	3	48	2.000
III	4	[2×2×2×3]	[3]	21	6	48	2.286

Table 2 Testing results on scarcity

Case	Accuracy %				
	CBR	ANN (BP)	ANFIS	Hybrid CBR	HSCS
I	85%	86.63%	67%	93.50%	90.97%
II	82.6%	81.11%	79.30%	95.37%	94.74%
III	68%	66.70%	66.70%	100%	100%

Table 3 Testing results on incompleteness

Case		*POI*				
		0%	5%	10%	15%	20%
I	*Acc.**	92.6%	90.7%	89.6%	86.5%	83.4%
	*IRR***	100%	98%	97%	93%	90%
II	*Acc.*	95.9%	89.5%	85.1%	86.2%	83.0%
	IRR	100%	93%	89%	88%	86%
III	*Acc.*	100%	100%	100%	100%	83.3%
	IRR	100%	100%	100%	100%	83.3%

**Acc.*—Accuracy defined in Equation (9)

** *IRR*—Information recovery ratio

Table 4 Testing results on uncertainty

Case	Uncertainty *p*				
	0%	5%	10%	15%	20%
I	96.6%	88.5%	85.4%	80.8%	73.3%
II	95.9%	90.4%	87.3%	84.6%	80.2%
III	100%	100%	100%	83.3%	83.3%

附圖

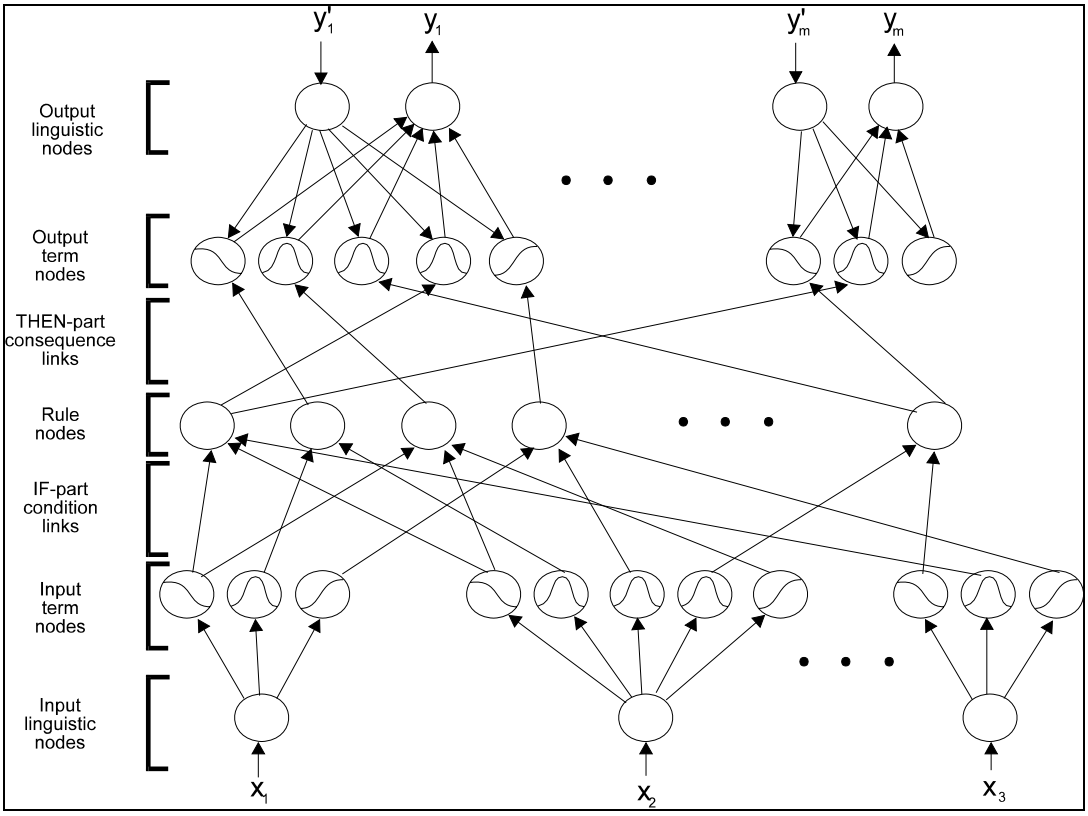

Figure 1 A generic FALCON model (Lin and Lee, 1991)

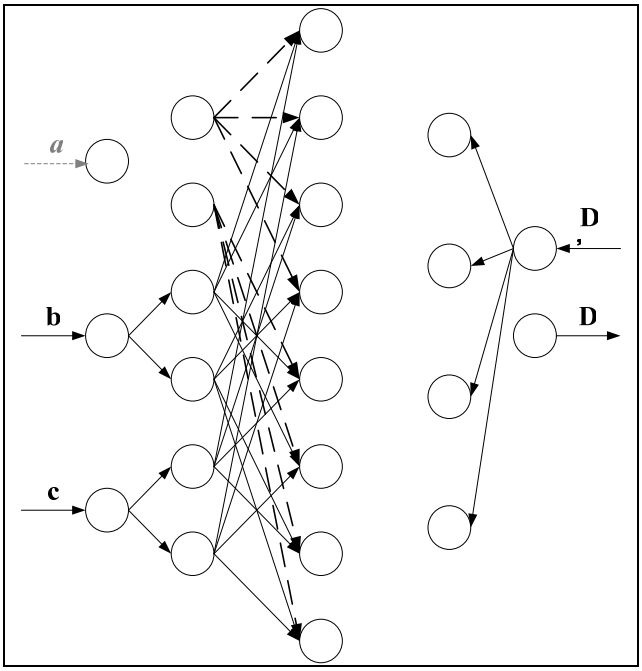

Figure 2 Connections of FALCON for incomplete attribute values (Yu and Lin, 2005)

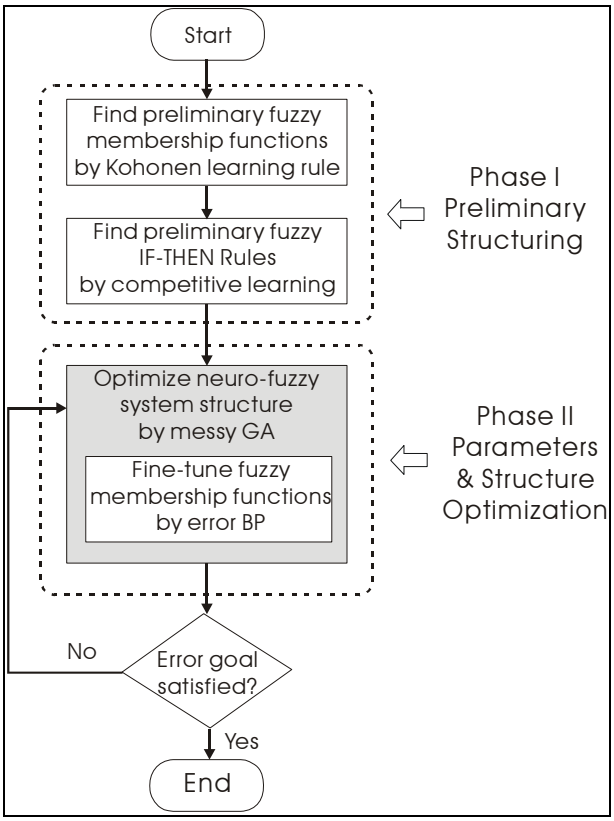

Figure 3 Integrated learning process of the proposed HSCS

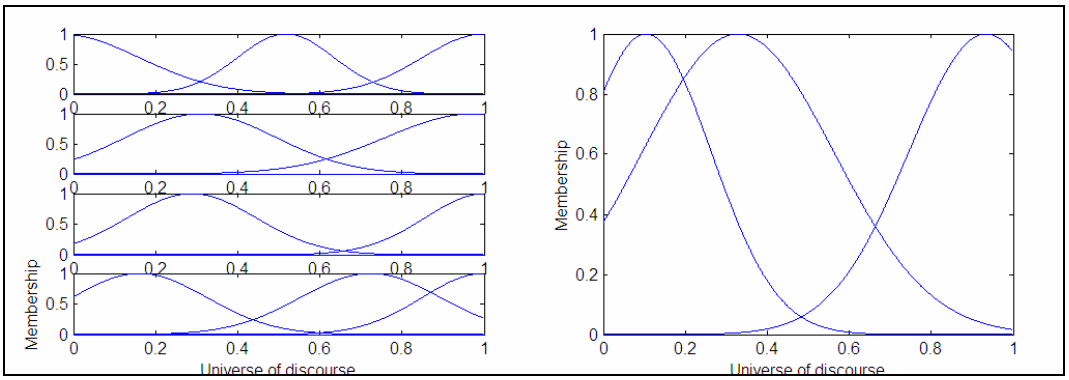

Figure 4 Memberships functions of Input/Output attributes of Case I

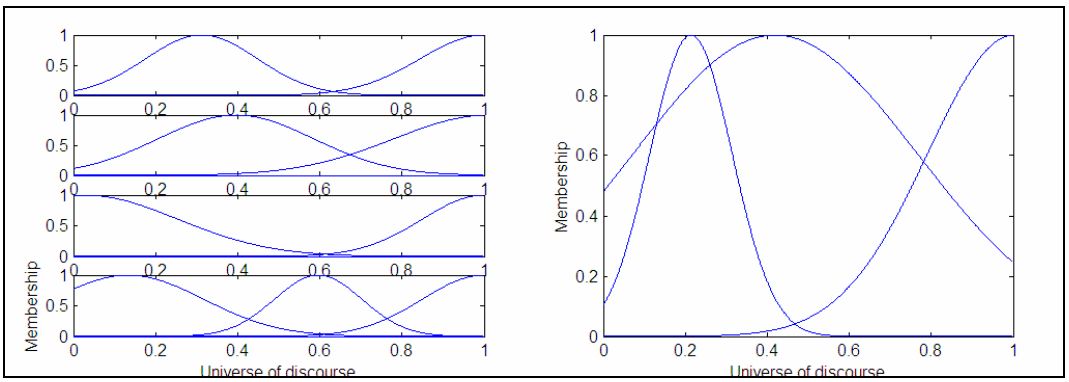

Figure 5 Memberships functions of Input/Output attributes of Case II

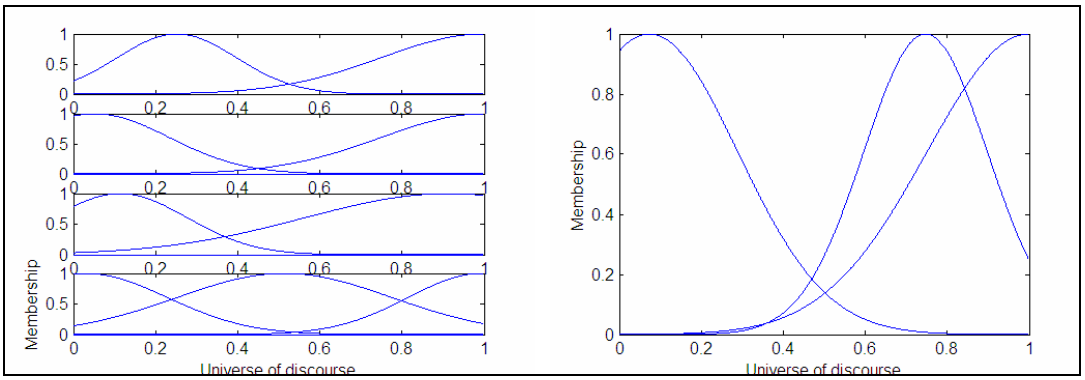

Figure 6 Memberships functions of Input/Output attributes of Case III

（三） 計畫成果自評

本計畫為一個三年度中程研究計畫之第二年，其主要研究工作在於針對營建業知識發掘(Knowledge Discovery)所遭的三大問題(不確定性、資料不足、資料缺漏)，發展出一套資料探勘之演算法，並撰寫 Matlab 程式，以測試所發展之方法的可行性。經過連續兩年之研究，已經完成相關演算法，並撰寫完成一套 Hybrid Soft Computing System (HSCS)，可做為營建業資料庫進行資料探勘之用。此一方法結合了柔性運算法中的模糊理論(FS)、類神經網路(ANN)及快速混元基因演算法(fmGA)，經測試發現確實可以解決營建資料庫資料不確定、資料不足及資料缺漏等三大問題。本年度之研究並進行三個實際案例之測試，團隊對於測試結果皆十分滿意。因此，本年度之研究與原計畫十分符合，研究結果亦符合預期目標。至於學術之實用價值必須透過第三年之驗證方能進行評估。

本研究第一年之成果已完成兩篇期刊論文及兩篇研討會論文之發表：

1. Yu, W. D., and Lin, H. W., “A VaFALCON neuro fuzzy system for mining of incomplete construction databases,” *Automation in Construction*, 15 pp., 2004.08. (accepted, 2005/1)
2. Yu, W. D., and Liu, Y. C., “Hybridization of CBR and numeric soft computing techniques for mining of scarce construction databases,” *Automation in Construction*, 14 pp., 2004.08. (accepted, 2005/1)
3. Yu, W. D., and Lee, Y. R., “Mining of Conceptual Cost Estimation Knowledge with a Neuro Fuzzy System,” *Proceedings of ISARC 2004*, Session SA 03-05, Sep. 21~25, Jeju, Korea, pp. 118-124, 2004.
4. Yu, W. D., and Lin, H. W., “A neuro fuzzy system for knowledge discovery of incomplete construction data,” *Proceedings of ISARC 2004*, Session SA 05-02, Sep. 21~25, Jeju, Korea, pp. 183-185, 2004.

本年度之成果亦已完成兩篇研討會論文之發表及一篇期刊論文之投稿：

1. Yu, W. D., and Fan, G. W., “Hybrid Soft Computing Approach for Knowledge Discovery in Construction Engineering,” *Proceedings of CC 2005*, Session II, Ag. 29~Sept. 2, Rome, Italy, 17 pp., 2005.
2. Yu, W. D., Lo, S. S., and Fu, J. W., “Real-Time Decision-Making with Partial Information for Construction Management,” *Proceedings of ISARC 2005*, Session MN&P-2-5, Sept. 11~14, 2005, Ferrara, Italy, 7 pp., 2005.
3. Yu, W. D., “Hybrid Soft Computing approach for knowledge discovery in construction,” *Journal of Computing in Civil Engineering*, ASCE, 30 pp., 2005. (under review by ASCE) (SCI, EI)

本計畫之主要發現與成果在於發展出一種可以同時處理資料不足與資料項缺漏之演算方法，且在結合模糊理論與類神經系統後，其對於不確定性資料之處理能力亦大幅提高。對於未來進行營建資料庫知識發掘之工作，具有極大之價值。未來對於不完全資訊下之緊急決策問題亦有潛在之價值。