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(III)

計畫類別：個別型計畫
計畫編號：NSC94-2211-E-216-024-
執行期間：94 年 08 月 01 日至 95 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

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 95 年 10 月 20 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—第三年

A Hybrid Soft Computing System for Knowledge Discovery in Construction Engineering and Management—Year Three

計畫類別：■ 個別型計畫　□ 整合型計畫
計畫編號：NSC　94-2211-E-216 -024
執行期間：　94 年　8 月　1 日至　95 年　7 月 31 日

計畫主持人：余文德
共同主持人：
計畫參與人員：　范綱緯

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　95 年　10 月　19 日

1. INTRODUCTION

Construction was conceived as an experience-based discipline (Ardery, 1991); knowledge acquired from previous works plays a key role for successful performance of new projects. Not only the construction know-how's of contractors, but also the design capabilities of design firms and the management skills of CM consultants rely heavily on such knowledge. This has made construction an ideal industry for knowledge-based economy. In the past decade, researchers from construction engineering and management area have engaged in developing methods and applications of knowledge discovery in databases (KDD) and data mining (DM). Soibelman and Kim (2002) applied KDD process to identify the cause(s) of construction activity delays from the data of RMS (Resident Management System) provided by the U. S. Corps of Engineers. Caldas, and Soibelman (2003) developed a construction information classification system (CICS) based on automatic hierarchical classification of construction project documents according to project components. In their work, pattern classification algorithms are used to create the classification models including naïve Bayes, k-nearest neighbors, Rocchio, and support vector machines (SVM). Such system can be applied to classify large project document databases. Another work by Soibelman et al (2003) applied similar concept to acquire corporate lesson-learned (CLL) from experienced personnel. They developed a Design Review Checking System (DrChecks) to provide a framework for a standardized review process.

In spite of some research works conducted by academic institutions, modern KDD or DM technologies were not yet widely applied by the industry in practical problems to acquire valuable knowledge from historic databases, which results in leaking of knowledge from construction firms. Previous works identified two causes for the lack of industrial applications: (1) the construction industry is not familiar with KDD and DM technologies (Yu and Skibniewski, 1999); (2) the existing KDD and DM technologies do not fit the special characteristics of complex construction databases.

2. RESEARCH OBJECTIVE

As a result, this research is intended for two objectives: (1) development of DM algorithms for mining of construction databases with complex characteristics such as data scarcity, data

incompleteness, and uncertainty; (2) generation of human understandable knowledge, so that domain experts can visualize and verify it. To achieve these goals, the existing KDD and DM techniques are reviewed. Problems faced in applications of DM to construction engineering and management are analyzed to identify the special characteristics of construction data. The existing soft computing techniques, including fuzzy sets, artificial neural networks, genetic algorithms, and case-based reasoning, are reviewed to propose appropriate hybridization in order to handle unique characteristics of construction data. A Hybrid Soft Computing System (HSCS) is developed for implementation of mining and knowledge discovery in construction databases. Case-studies applying the proposed HSCS for KDD from construction databases are selected to validate the proposed method.

3. LITERATURE REVIEW

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 domain problem—such as relevant prior knowledge and goals of the application; (2) Extracting 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; (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 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; (7) Choosing the data mining algorithm(s); (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; and (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. 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, which 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 are viewed as inferred knowledge. Mitra et al. (2002) concluded that, a typical DM algorithm constitutes some combination of the following components:

- 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;
- Preference criterion—a basis for judgment of better model, depending on the given data;
- Search algorithm—that is a specific algorithm for finding particular models and parameters, which significantly influences the mining results.

SOFT COMPUTING FOR MINING CONSTRUCTION DATABASES

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 unique problems confronting mining of construction data. The following point out three types of complexity existing in construction databases.

Data Scarcity

Yu and Liu (2006) defined two types of data scarcity are existing in construction databases: (1) scarcity in data volume—due to unique nature and huge scales of construction projects, 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 were never performed before, so the associated data 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 omnipresent 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 (2006) 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 (Yu and Lin, 2006).

Data Uncertainty

Major uncertainties in construction data may be due to the weather effects on outdoor construction operations (Halpin and Woodhead, 1998; Hendrickson, 1998). 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 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 the unique characteristics of construction databases, the model and associated search algorithms adopted 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 to a KDD process (Mitra et al., 2002). The fuzzy sets (FSs) provide a natural framework for the process to deal 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. METHODOLOGY—Proposed Hybrid Soft Computing System (HSCS)

Framework

Basic Network

In this section, the knowledge representation model and the computational algorithms of the

hybrid soft computing system (HSCS) are proposed. In order to extract 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 automatic construction of FIS (Lin and Lee, 1991; Jang, 1993). Therefore the neuro-fuzzy system (NFS) became an appropriate choice for the proposed HSCS. Among many existing NFSs, the FALCON model (Lin and Lee, 1991) that adopts Mamdani FIS scheme is selected for the proposed HSCS. Figure 1 shows the generic structure of FALCON model.

A standard FALCON comprises five layers: (1) Layer 1—the values of input attributes are transmitted directly into FALCON; (2) 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); (3) Layer 3—the connections between the second and the third layers represent the pre-conditions of fuzzy IF-THEN rules; (4) Links between Layer 3 and Layer 4—represent the consequences of fuzzy rules; (5) Layer 5—act as the defuzzifier to derive conclusion of fuzzy reasoning from FALCON. The above structure is identical to an FIS.

Variable-Attribute Network Structure (VANS)

Similar to other NFSs, the traditional FALCON tackles only data with complete attribute values. Any missing attribute value will cause difficulty in performing *fuzzy AND* operation in the 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, the Variable Attribute Network Structure (VANS) was proposed by Yu and Lin (2006). The VANS adopts a flexible network structure that can adjust to available attribute values in processing every single input dataset. Consider 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 underlying 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, 2006). The proposed HSCS adopts VaFALCON and 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 self-organized learning process is employed to determine the primitive fuzzy partitions, the membership functions of the input and output fuzzy terms, and the primary structure of the rule base for the HSCS. There are two learning stages in the self-organization: (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 *i*th cluster by minimizing the difference between *x* and the center of the cluster ($\hat{w}_i^k$), where superscript *k* represents the *k*th 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 *k*th iteration. In the updating stage, the center of the *i*th cluster, the *winner* cluster, at the *k*th 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: $\mu_j^k = e^{-\left(\frac{o^k - m_j^k}{\sigma_j^k}\right)^2}$, $\mu_i^k = \underset{1 \le j \le m}{Max}\ \mu_j^k$. (3)

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

Rule selection stage: $w_{li} = 1$ if $w_{li} = \underset{1 \le i \le m}{Max}(w_{li})$,

$w_{lj} = 0$ if $j \neq i$, for $l = 1,2,...,L$. (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, the parameters of the membership functions for input and output linguistic terms are fine-tuned by back-propagation algorithms (BP). 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:

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

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

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:

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

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

Global search

Previous research has found that traditional FACLON may encounter severe local minimum problems while dealing with complex construction data (Yu and Skibniewski, 1999). In this report, 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 sGA, a coded chromosome is in fixed length that highly fitted allele (feature value of gene) 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 sGA 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 finding 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 algorithm adopts error back-propagation algorithm (BP) to fine-tune the parameters of membership functions after certain cycles of mGA adaptations. This integrated algorithm equips the proposed HSCS 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 NFS using Kohonen's learning rule (Kohonen, 1988) and a reinforcement competitive learning rule proposed by Yu and Skibniewski (1999). The Kohonen's feature map constructs a primitive

NFS with roughly determined centers and spreads of the fuzzy membership functions in both input and output layers of the VaFALCON. In the mean while, the fuzzy IF-THEN rules are also roughly linked between Layer 3 and Layer 4 in the VaFALCON. After Phase I, a preliminary fuzzy IF-THEN rule-based system is constructed and the knowledge mined from construction databases is stored in terms of fuzzy IF-THEN rules.

(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. In Phase II, the parameters of the membership function in the output and input terms and the consequent connections of fuzzy IF-THEN rules are "globally" searched by a messy-genetic algorithm (mGA) and then "locally" fine-tuned by the error back-propagation (BP). Learning process in Phase II is repeated until the error is below the error goal or a pre-determined number of learning iterations has been performed. Should there exist a local minimum so that the network error is too high to be accepted, the mGA adaptation is performed again to escape from the local minimum. The learning process is repeated until the error goal is satisfied.

Integrated into Knowledge Discovery Process

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. 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. The data preprocessing tasks are performed before DM is preceded.

Data mining

The next step of knowledge discovery is to perform data mining on the preprocessed data. The HSCS proposed in this research 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 original VaFALCON, preliminary structuring was achieved by self-organized Kohonen Future Map and competitive learning rule, and the parameters optimization was performed by BP. However, the structure optimization function was lacked in original VaFALCON. The structure of NFS in the proposed HSCS is optimized in Phase II by global search with mGA algorithm and then fine-tuned with BP, so that the erroneous rule structure organized in Phase I can be improved to find optimal NFS.

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 discovered patterns. A fuzzy IF-THEN rule in the proposed VaFALCON 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 discovered to form the knowledge base.

SYSTEM VALIDATION

In this section, system validation methodology for the proposed HSCS is described. Three case-studies are conducted by applying the proposed HSCS. The objective of system validation is to validate the capability of HSCS in mining of scarce, incomplete, and uncertain construction databases. Therefore, measures of the three types of complexity are established: (1) scarcity ratio (R_s); (2) percentage of overall incompleteness (*POI*); and (3) degree of uncertainty (p%). The validation is based on system performance with respect to various degrees of complexity measures. Three construction data repositories were collected from published literature, including: (1) Building Construction Cost Estimation (Yu, 2001); (2) Estimation of Curtain wall Construction Duration (Yang, 1997); and (3) Retaining Wall Selection (Yau and Yang, 1998). It was found that the running-time of HSCS is case-sensitive. For a generic application such as the demonstrated examples, it takes about 30 min. to 2 hr. to converge.

Case Background

(1) Case I—Building Construction Cost Estimation

Yu (2001) applied CBR technique to the conceptual cost estimation of building construction projects. In his research, nearly 30 parameters are originally collected for knowledge representation of CBR. After reviewing with experienced engineers, four most important parameters were identified: (1) type of earth retaining method; (2) No. of floors above ground; (3) No. of floors under ground; and (4) total floor area. One single output, construction cost estimation, is provided by the CBR system. Totally 25 data are collected from historical building construction project through surveying the final project reports provided by public owners. 22 data sets are used for learning and the rest 3 data are used for testing.

(2) Estimation of Curtain wall Construction Duration

Yang (1997) developed a CBR system for duration estimation of curtain wall construction in his Ph.D. research. Totally 27 historical datasets were collected from major consultant firms of Taiwan. Among which, 24 are used for training and 3 are used for testing. The input attributes identified by Yang are: (1) excavation depth; (2) quantity of walls; (3) construction method; and (4) soil type.

(3) Case III—Retaining Wall Selection

Yau and Yang developed a retaining wall selection expert system named CASTLES (Yau and Yang, 1998). The training examples of CASTLES are generated by interviewing with domain experts from industry. Example projects were first selected and collected from real world and then presented to the domain experts who evaluated the project characteristics and selected the most appropriate method according to their knowledge. Totally 27 datasets are selected, among which 21 data are used for training and the rest 6 used for testing. The four input attributes identified are: (1) excavation depth; (2) sufficiency of working space; (3) level of water table; (4) soil type.

Design of Validation Experiments

Definition of the DM capability for knowledge discovery

The capability of knowledge discovery should represent the ability of the DM algorithm to figure out interesting patterns and rules behind the data. In this research a simple index of DM capability, *accuracy*, is adopted for measure of the DM capability for knowledge discovery of the proposed HSCS. Such approach has been adopted by Leu et al. (2001) in their similar research, too. The *accuracy* of system output is defined in the following equation.

$$Acc.(\%) = \left\{1 - \left|1 - \frac{Estimated}{Actual}\right|\right\} \times 100\%, \tag{9}$$

where *Estimated* is the output generated by the system, *Actual* is the actual result observed from real world, and *Acc.* is the *percentage accuracy* of the estimation. Absolute value is taken within the parenthesis to avoid minus values.

Measure of Data Scarcity

The scarcity of data can be measured with the ratio of the number of variables in the model over the number of available training sets. Following equation define scarcity ratio (R_s).

$$R_s = \frac{NV_{model}}{N_{ts}}, \tag{10}$$

where R_s is the scarcity ratio that measures the severity of data scarcity; N_{ts} is the number of

available training sets; and NV_{model} is the number of variables in the HSCS model to be determined by training examples. The NV_{model} can be roughly estimated by summing up all attributes in the network including the centers and spreads of the membership functions in input layer, the consequence connections of rule nodes, and the centers and spreads of the membership functions in output layer.

$$NV_{\mathrm{model}} = 2 \times \sum_{x=1}^{n_{in}} V_{fp}^{\ in}(x) + \prod_{x=1}^{n_{in}} V_{fp}^{\ in}(x) + 2n_{out}, \tag{11}$$

where $V_{fp}^{\ in}$ is the vector of fuzzy partitions in the input layer; n_{in} is the number of input attributes; the mathematic symbol Π represents the *product* calculation of elements in the vector; and n_{out} is the number of output attributes. It is noted that the first and third terms in the right hand side are multiplied by a coefficient, 2, which counts for the number of the undetermined variables, centers and spreads of the membership functions.

The higher the value of R_s means the severer the data scarcity problem. The experiment is then designed for testing the data of the three cases with various degrees of R_s to see its impact on the system accuracy defined in Equation (9).

Measure of Data Incompleteness

Yu and Lin (2006) defined a measure of data incompleteness as "percentage of overall incompleteness (*POI*)", which measures the ratio of the total number of incomplete attributes over the number of all attribute values of all datasets. The *POI* is defined in the following equation.

$$POI = \frac{\sum_{x=1}^{N_{ts}} N_{ma}^{\ i}}{n_{in} \times N_{ts}}, \tag{12}$$

where $N_{ma}^{\ i}$ is the number of missing attributes in the i^{th} training set; n_{in} and n_{out} are as defined previously.

Similar to problem of data scarcity, the higher value of *POI* means the severer data incompleteness problem. The experiment is then designed for testing the data of the three cases with various degrees of *POI* to view its impact on the system accuracy.

(1) Uncertainty in Data

Testing of uncertainty is relatively simple, as the uncertainty existing in the construction events is usually modeled by a probability density function (PDF) of normal distribution. There are two parameters in the normal PDF, the mean (μ) and square-root of variance (σ). It is σ that relates to the uncertainty of a random event. The uncertainty is then modeled by applying the following equation.

$$x_i^{p\%} = x_i \times Rand(\cdot) \times p\%, \quad (13)$$

where $x_i^{p\%}$ is the generated dataset from original dataset x_i with σ of $p\%$; *Rand(.)* is a set of random numbers generated by the computer system.

The higher value of *p%* implies the higher uncertainty existing in data. The experiment is designed for testing the data of the three cases with various degrees of *p%* to view its impact on the system accuracy.

Validation Results

Data Scarcity

The R_s for each of the three cases are calculated using Equation (10) and (11). 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 (2006). 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 and Figure 4. It is found that the proposed HSCS performs similar to the hybrid CBR approach and much better than the other three traditional methods (CBR, ANN, and ANFIS) under severe data scarcity conditions (with scarcity ratio $R_s \geq 2$). Even though the *hybrid CBR* approach slightly outperforms HSCS, it does not provide meaningful fuzzy IF-THEN rules for explicit knowledge presentation. It is concluded that HSCS is more suitable for mining of scarce construction databases while compared with most of the other soft computing methods.

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, Figure 5, and Figure 6. The information recovery ratio (*IRR*) defined by Yu and Lin (2006) measures the ratio of the accuracy with incomplete data over the accuracy with complete data. It is found from the testing results that both accuracy and *IRR* are relatively high (> 83%) when the *POI* is less than 20%. It is validated that HSCS can mine incomplete data as data incompleteness is not severe (e.g., *POI* < 20%).

Uncertainty

The uncertainty of data is modeled by Equation (13), where the original data were disturbed with random number of various uncertainty ranges, *p*. In the experiment, the *p* was 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 Figure 7. The testing results show that the proposed HSCS is more sensitive to uncertainty than the other two attributes (scarcity and incompleteness). However, the capability of HSCS in mining of uncertain data is validated as long as the disturbance of data is not very severe (e.g., *p*% < 15%).

Knowledge Presentation

The knowledge mined by the proposed HSCS is stored in the fuzzy rule base that 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. 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 8 to 10, respectively. Each fuzzy decision rule is defined by

multiple preconditions and single consequence. The relationships of the preconditions and consequence are stored in matrices. An example of the fuzzy IF-THEN rules for Case I is 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 domain experts. By investigating all fuzzy IF-THEN rules, the knowledge acquired from data mining process can be verified manually.

5. DISCUSSION

Broad and Narrow Definitions of KDD

Previous researchers have defined KDD broadly as "the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" (Fayyad, Piatetsky-Shapiro, Smyth, and Uthurusamy, 1996; Cios, Pedrycz, and Swiniarski, 1998). Any system provides functions of the above can be viewed as a system for KDD. A narrower definition described some textbook such as "Data Mining: Concepts and Techniques" (Han and Kamber, 2000), where KDD has a strict process of target dataset creation, Data cleaning and pre-processing, Data reduction and transformation, Data mining, Pattern evaluation and knowledge presentation. The definition of KDD adopted in this research is closer to the former.

DM Functionalities

The functionalities of a generic DM system include (Han and Kamber, 2000): (1) Concept description—characterization and discrimination; (2) Association--correlation and causality); (3) Classification and Prediction; (4) Cluster analysis; (5) Outlier analysis; (6) Trend and evolution analysis; (7) Other pattern-directed or statistical analyses. The current version of HSCS provides several DM functionalities such as: association (can also be described in IF-THEN rules); predictions; classification. Characterization and discrimination are just simplified form of

classification. Clustering and outliers analysis is not provided in the current version. However, with proper design of pre-processing and post-processing functions, the abovementioned DM functionalities can also be implemented.

Validation Methodology

The validation methodology adopted in this research is based a simple index, *accuracy*, of classification function of the proposed HSCS. It should be noted that classification serves the fundamental functions for all other DM functionalities. Should the function of classification be validated, further extension of functionalities can be developed. Validation of classification accuracy also validates other functionalities such as association, concept description, etc.

6. CONCLUSION AND RECOMMENDATION

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

Three major contributions are concluded for the research including: (1) proposing a new data mining system that can tackle complex characteristics of construction databases such as data scarcity, data incompleteness, and uncertainty; (2) establishing measures for softness of DM method such as scarcity ratio (*Rs*), percentage of overall incompleteness (*POI*), and degree of uncertainty (*p*%); (3) the proposed HSCS does not only provide prediction function of traditional linear and nonlinear mapping schemes (such as linear regression model or ANN), but also

generate human understandable fuzzy IF-THEN rules.

Although HSCS is able to mine fuzzy IF-THEN rules that can be visualized and verified by domain experts; the resulted fuzzy rule base is too huge for human expert to verify manually. Some rule pruning or screening method need to be developed to reduce the rule base in order to make such system realistic for practical usage. Moreover, other soft computing techniques, such as C4.5 and C5.0, can be considered in comparison with the proposed HSCS in future works.

REFERENCES

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.

Caldas, C. H. and Soibelman, L. (2003) "Automating hierarchical document classification for construction management information systems," *Automation in Construction*, Vol.12, 395–406.

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.

Cios, K. J., Pedrycz, W., and Swiniarski, R. (1998) *Data Mining Methods for Knowledge Discovery*, Dordrecht, The Netherlands.

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.

Halpin, D., and Woodhead, R. W. (1998) *Construction Management*, 2nd edition, John Wiley &Sons, New York, USA.

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

Hendrickson,C., *Project Management for Construction--Fundamental Concepts for Owners, Engineers, Architects and Builders*, Prentice Hall, New York, USA, 1998.

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.

Leu, S.-S.; Chen, C.-N.; and Chang, S.-L., "Data mining for tunnel support stability: neural network approach," *Automation in Construction*, Vol. 10, No. 4, pp. 429-441, 2001.

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.

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

Soibelman, S. and Kim, H. (2002) "Data Preparation Process for Construction Knowledge Generation through Knowledge Discovery in Databases," *Journal of Computing in Civil Engineering*, Vol. 16, No. 1, pp. 39-48.

Soibelman, L., Liu, L. Y., Kirby, J. G., East, E. W., Caldas, C. H., and Lin, K. Y. (2003) "Design Review Checking System with Corporate Lessons Learned," *Journal of Construction Engineering and Management*, Vol. 129, No. 5, pp. 474-484.

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.

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. (2006) “A VaFALCON neuro fuzzy system for mining of incomplete construction databases,” *Automation in Construction*, Vol. 15, No. 1, pp. 33-46.

Yu, W. D., and Liu, Y. C. (2006) “Hybridization of CBR and numeric soft computing techniques for mining of scarce construction databases.” *Automation in Construction*, Vol. 15, No. 1, pp. 20-32.

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.

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

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%

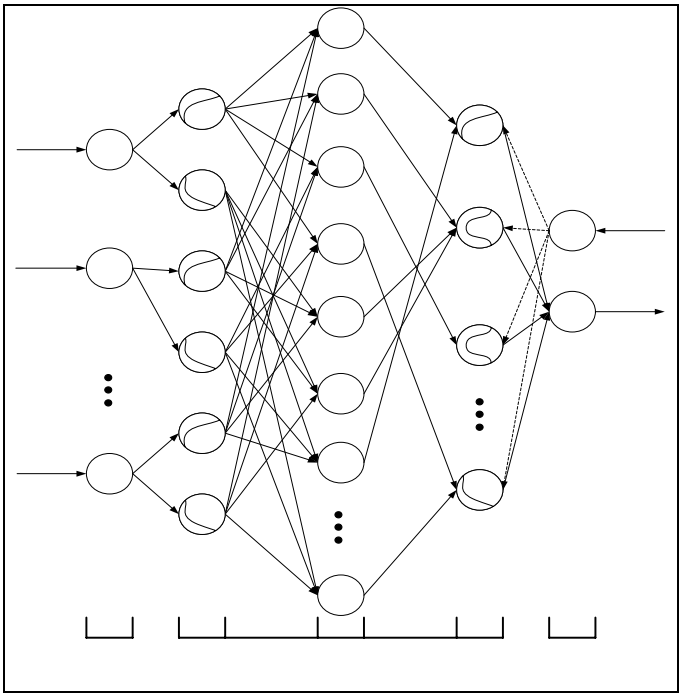

Figure 1 Generic FALCON model (Modified from Lin and Lee, 1991)

X_1

Y'

X_2

Y

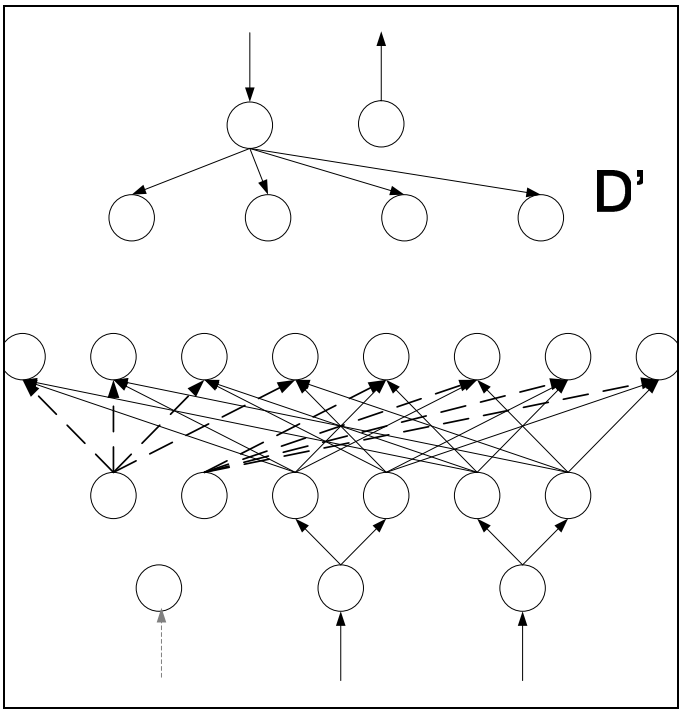

Figure 2 Connections of FALCON for incomplete attribute values (Modified from Lin and Lee, 1991)

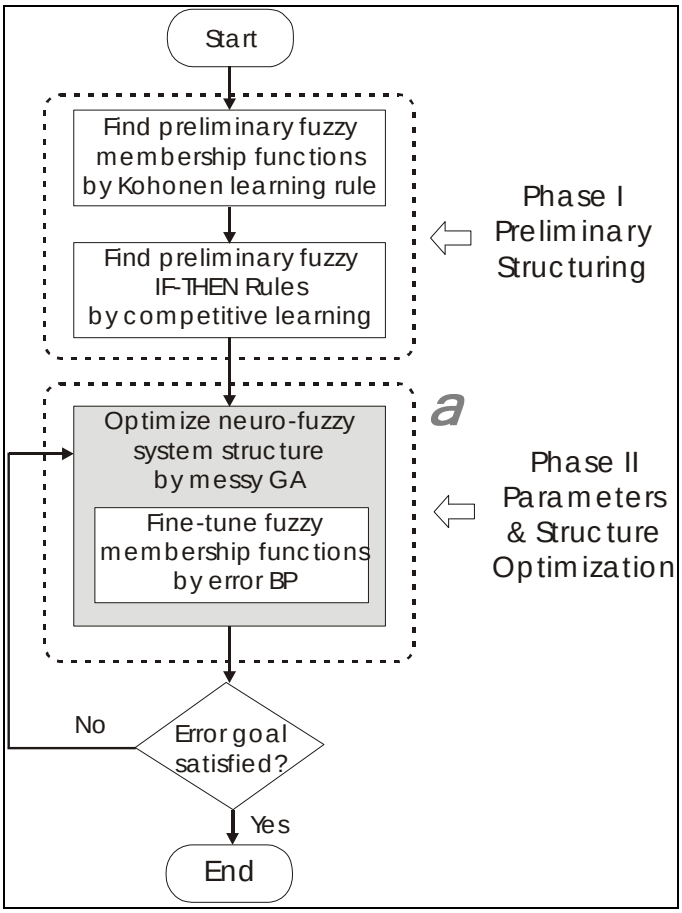

Figure 3 Integrated learning process of the proposed HSCS

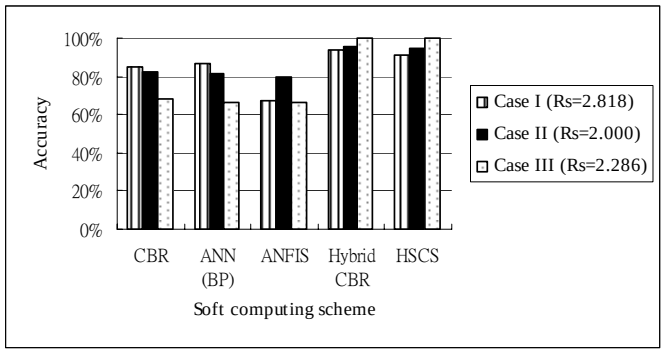

Figure 4 Testing results on accuracy vs. scarcity for the five soft computing schemes

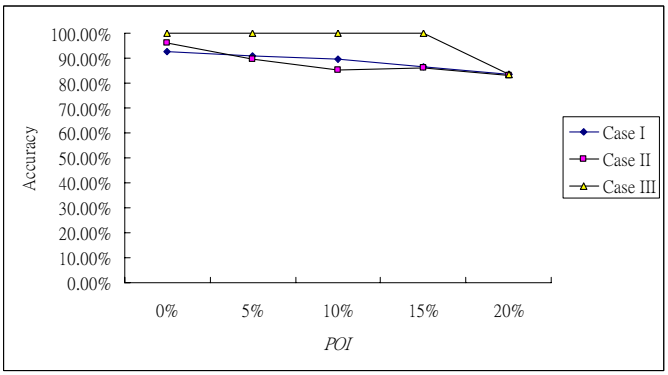

Figure 5 Testing results on accuracy vs. *POI*

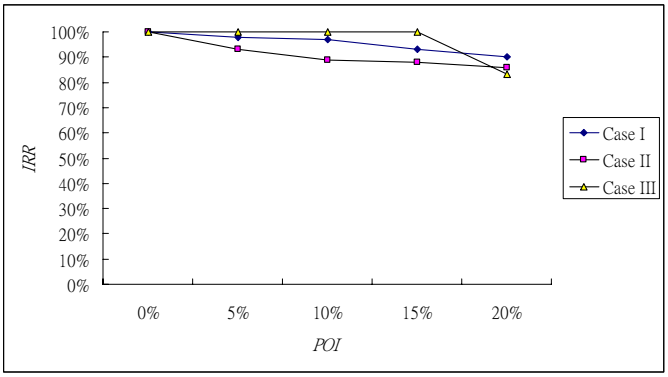

Figure 6 Testing results on information recovery ratio (*IRR*) vs. *POI*

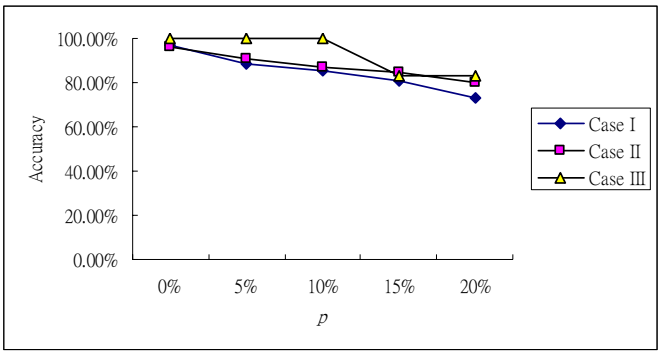

Figure 7 Testing results on accuracy vs. uncertainty ratio *p*

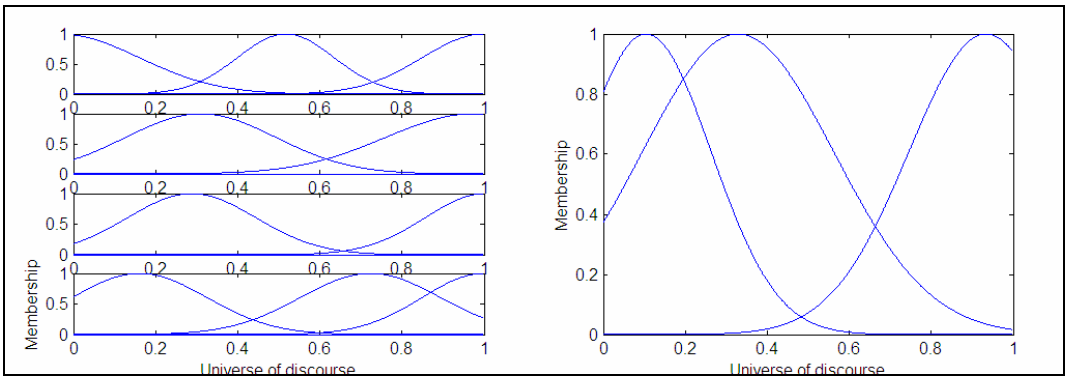

Figure 8 Memberships functions of Input/Output attributes of Case I

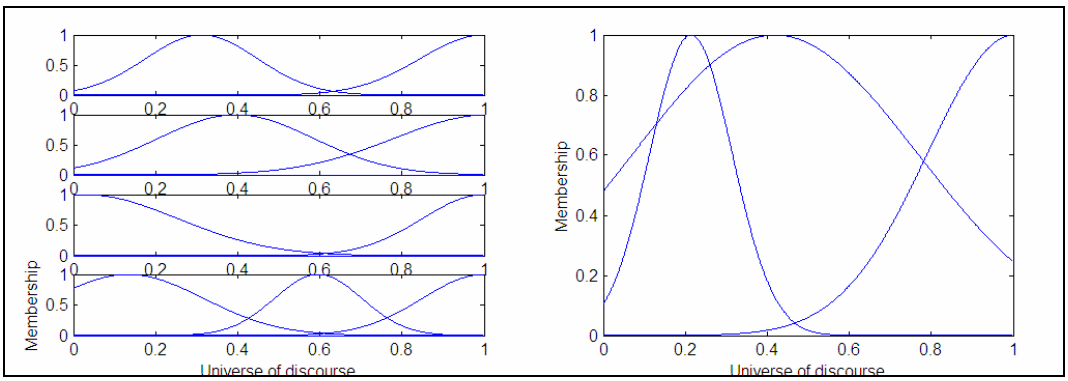

Figure 9 Memberships functions of Input/Output attributes of Case II

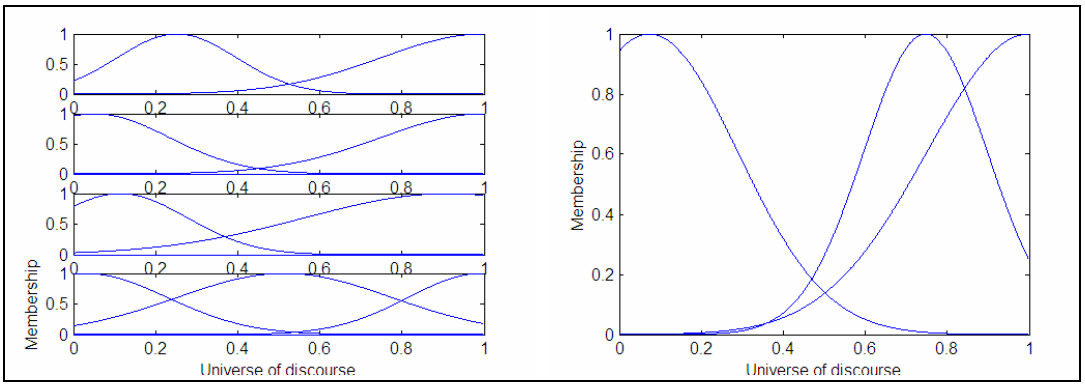

Figure 10 Memberships functions of Input/Output attributes of Case III

可供推廣之研發成果資料表

□ 可申請專利　■ 可技術移轉　日期：95 年 10 月 19 日

國科會補助計畫	計畫名稱：混合式柔性計算系統於營建知識發掘之研究—第三年 計畫主持人：余文德 計畫編號：NSC 94-2211-E-216 -024　學門領域：土木(營建)
技術/創作名稱	混合式柔性計算系統
發明人/創作人	余文德
技術說明	中文： 本研究發展出一套混合式柔性運算系統(HSCS)作為複雜型營建資料庫知識探勘之用。所提出之 HSCS 混合了模糊邏輯、類神經網路、混原基因演算法等柔性運算技術，構成了一個探勘能力極強之工具。在三項柔性運算測試(資料不足、資料缺漏及不確定性)後發現，HSCS 能夠有效地發掘複雜資料之隱含知識。本 HSCS 對於發展營建企業智能等相關應用，應具有潛在之價值。
	英文： The research develops a hybrid soft computing system for mining of complex construction databases. The proposed approach hybridizes soft computing techniques, such as fuzzy logic, artificial neural network (ANN), and messy genetic algorithms (mGA), to form a novel computational method for mining of human understandable knowledge from historical databases. The hybridization combines merits of explicit knowledge representation of fuzzy logic decision-making system (FLDS), learning abilities of ANN, and global search of mGA. A Hybrid Soft Computing System (HSCS) is developed for mining complex databases in construction with three characteristics: scarcity, incompleteness, and uncertainty. Real world construction data repositories are selected to test the capabilities of the proposed HSCS for data-mining under the abovementioned complex conditions. The testing results show promising potential of the proposed HSCS for mining of complex databases in construction.
可利用之產業及可開發之產品	營建及其他複雜資料庫之探勘工具
技術特點	對於資料不足、資料缺漏及不確定性等三種特性交互影響之資料庫具有強大之資料探勘能力。

推廣及運用的價值	可結合現有資料庫管理系統，成為資料探勘工具。

※ 1.每項研發成果請填寫一式二份，一份隨成果報告送繳本會，一份送　貴單位研發成果推廣單位（如技術移轉中心）。

※ **2.本項研發成果若尚未申請專利，請勿揭露可申請專利之主要內容。**

※ 3.本表若不敷使用，請自行影印使用。