Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Design and Engineering 2 (2015) 137–147

www.elsevier.com/locate/jcde

Hybrid Type II fuzzy system & data mining approach for surface finish

Tzu-Liang (Bill) Tseng[a], Fuhua Jiang[b], Yongjin (James) Kwon[c,*]

[a]Department of Industrial, Manufacturing and Systems Engineering, The University of Texas at El Paso, El Paso, TX 79968, USA
[b]Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA
[c]Department of Industrial Engineering, Ajou University, Suwon 443-749, Republic of Korea

Received 28 November 2014; received in revised form 21 February 2015; accepted 22 February 2015
Available online 5 March 2015

Abstract

In this study, a new methodology in predicting a system output has been investigated by applying a data mining technique and a hybrid type II fuzzy system in CNC turning operations. The purpose was to generate a supplemental control function under the dynamic machining environment, where unforeseeable changes may occur frequently. Two different types of membership functions were developed for the fuzzy logic systems and also by combining the two types, a hybrid system was generated. Genetic algorithm was used for fuzzy adaptation in the control system. Fuzzy rules are automatically modified in the process of genetic algorithm training. The computational results showed that the hybrid system with a genetic adaptation generated a far better accuracy. The hybrid fuzzy system with genetic algorithm training demonstrated more effective prediction capability and a strong potential for the implementation into existing control functions.

Keywords: Data mining; Fuzzy set theory; Surface roughness; Metal cutting; Machining quality characteristics

1. Introduction

In this study, a data mining technique (i.e., a new heuristic algorithm) for reduct selection in the Rough Set Theory (RST) is applied to select significant factors (features). Literature review suggests that the RST has not been widely applied to metal cutting problems thus making this research novel [1,2]. In the RST, features characterize each object, and it discovers the dependencies between them. Compared to the usual statistical tools that use population-based approach, the RST uses an individual, object-model based approach that makes a very good tool for analyzing quality control problems [3]. The RST is also able to identify "defective" and "significant factors" simultaneously, which is unique and useful in solving quality control problems. After significant factors are identified, a Fuzzy Logic Theory (FLT) is used to construct the approach that adapts and predicts the surface finish because of the following reasons: (1) using a FLT enables fast and easy synthesis and modification of the control rule base; (2) if a rapid adaptation, using only a few data points, with good accuracy is obtained, the process can respond to the changes; and (3) adaptation is more suitable for today's machining environment because the adaptive approach can be integrated into the CNC controller to compensate for process variations. The applied Fuzzy Logic System (FLS) is a combined system of Type I and Type II. Different types of system express different strengths to handle heterogeneous factors as well as variables in the process. For example, Type I is effective to deal with "crisp" type of membership function, while Type II is adequate to handle "uncertain" type of membership. The Type II FLS has not been widely used to solve machining process problems thus making this study unique. Finally, the Genetic Algorithm (GA) is incorporated to the FLS for fuzzy adaptation. The combination of the unique strength in each domain is expected to provide a better solution space.

In current practice, setting machining parameters are usually conducted by the experience of skilled engineers. Once set, the parameters are usually unchanged during machining, unless prominent anomalies are present. The proposed scheme can be incorporated into the intelligent CNC controllers, and used as constant monitoring device as the machining

*Corresponding author. Tel.: +82 31 219 2418; fax: +82 31 219 1610.
E-mail address: yk73@ajou.ac.kr (Y. Kwon).

http://dx.doi.org/10.1016/j.jcde.2015.02.002

operations are carried out. Such practice can significantly improve the machining efficiency as well as the perceived quality of the machined parts. It is expected that the proposed approach will help compensate for the unforeseeable variations in the machining process, hence ultimately affects the overall quality of the CNC machining operations.

2. Literature survey

2.1. Fundamental of data mining and rough set theory

Data mining is the process of extracting and refining knowledge from large database [4–6]. The extracted information is used to predict, classify, model, and summarize the data being analyzed. The RST is a fundamental theory of data mining. This theory was originated by Pawlak [7] and was developed to classify imprecise, uncertain, or incomplete information or knowledge expressed in terms of data acquired from experience. Therefore, it complements the FST [8]. The rough set approach is suitable for processing qualitative information that is difficult to analyze by standard statistical techniques [9]. It integrates learning-from-example techniques, extracts rules from a data set of interest, and discovers data regularities [10]. The RST has been applied to address a variety of problems [11], including (1) representation of uncertain or imprecise knowledge; (2) empirical learning and knowledge acquisition from experience; (3) knowledge analysis; (4) analysis of conflicting; (5) evaluation of the quality of the available information with respect to its consistency and the presence or absence of repetitive data patterns; (6) identification and evaluation of data dependencies; and (7) approximate pattern classification. The RST is introduced as an extension of set theory for the study of intelligent systems characterized by using incomplete information to classify imprecise, uncertain, or incomplete information or knowledge expressed in terms of data. Indeed, the RST is an effective tool for multi-attribute classification problems. In RST, data is expressed in a decision table in which each row represents an object and each column represents an attribute. Formally, the decision table is represented by an information function [12]:

$$S = \langle U, Q, V, f \rangle \qquad (1)$$

where U is a finite set of objects, Q is a finite set of attributes, $V = \bigcup_{q \in Q} V_q$ QUOTE and V_q is a domain of the attribute q, and $f: U \times Q \to V$ is the total decision function such that $f(x,q) \in V_q$ for every $q \in Q$, $x \in U$ QUOTE . The main theme of RST is concerned with measuring what may be described as the "ambiguity" inherent in the data. The essential distinction is made between objects that may definitely be classified into a certain category, and those that may possibly be classified. Considering all decision classifications yields to what is referred to as the "quality of approximation" that measures the proportion of all objects from which definite classification may be achieved. A rough set can be described as a collection of objects that in general cannot be precisely characterized in terms of their values or sets of attributes, but can be characterized in the form of lower or upper approximations [13,14]. The upper approximation includes all objects that possibly belong to the concept, while the lower approximation contains all objects that definitely belong to the concept. As each object is characterized with attributes, discovering the dependencies between attributes and detecting the main attributes is of primary importance. Attribute reduction is one unique aspect of the rough set approach. A reduct is a minimal sufficient subset of attributes, which provides the same quality of discriminating concepts as the original set of attributes.

Let us consider the five objects in Table 1, each with four input features and an output feature (outcome). To derive the reduct, consider the first feature $F1$. The set of objects corresponding to the feature value F1 = 0 is {1, 2, 3, 5}. This set {1, 2, 3, 5} cannot be further classified solely using the relation F1 = 0. It is discernible over the constraint F1 = 0, which is expressed as $[x][\text{F1}=0]=\{1, 2, 3, 5\}$. For the objects in set {1, 5}, the output feature is $O=2$. For object 3, the output feature is $O=1$ and for object 2, the output feature is $O=0$. Therefore, additional features are needed to differentiate between $O=0$, 1, or 2. Applying this concept, the classification power of each feature can be evaluated. For instance, the feature value F1 = 1 is specific to $O=1$. This discernible relation can be extended to multiple features, e.g., $[x][\text{F1}=0] \wedge [\text{F2}=1]=\{1, 3\}$ and $[x][\text{F1}=0] \vee [\text{F2}=1]=\{1, 2, 3, 5\}$, where $\wedge$ and $\vee$ refers to "or" and "and", respectively.

2.1.1. Reduct generation

Most of the rough set based approaches may generate more than one reduct for an object. This paper adapts the reduct generation procedure proposed by Pawlak [12] and presents it in the form of the reduct generation procedure as illustrated in Fig. 1. The reduct generation procedure enumerates all

Table 1
Example data set.

Object no.	*F1*	*F2*	*F3*	*F4*	*O*
1	0	1	0	2	2
2	0	0	1	3	0
3	0	1	1	1	1
4	1	2	2	0	1

O: Not Applicable, 1: Low, 2: Medium, 3: High.

Step 0: Set object number i = 1;

Step 1: Select object i and find a set of reducts with one to m - 1 features;

Step 2: Set i = i +1. If all objects have been considered, go to Step 3; otherwise go to Step 1;

Step 3: Terminate the algorithm and output the result.

Fig. 1. Reduct generation procedure.

Table 2
Partial reducts for data in Table 1.

Object no.	*F1*	*F2*	*F3*	*F4*	*O*	Reduct no.	*Ui*	*F1*	*F2*	*F3*	*F4*	*O*
1	0	1	0	2	2	1	1	x	x	0	x	2
						2		x	x	x	2	2
						3	2	0	x	0	x	2
						4		0	x	x	2	2
						5		x	1	0	x	2
						6		x	1	x	2	2
						7		x	x	0	2	2
						8	3	0	1	0	x	2
						9		0	1	x	2	2
						10		x	1	0	2	2

possible reduct with one, two and three features that are presented in Table 2.

2.1.2. The rule extraction algorithm (REA)

Feature sets are used for predicting an object's outcome with algorithms. We adapt the reduct generation procedure proposed by Pawlak [12]. The data set is randomly divided into the training set and the testing set. The rule-extraction algorithm is developed to derive the rules from the training set. Then, we also propose the procedure to validate the derived rules on the basis of the testing set. The basic idea behind the heuristic algorithm developed by Kusiak and Tseng [15] is to consider a sequence of the rule sets $R_1, R_2, \ldots, R_q$. If one wants to construct the set $N=\{N_i\}$, follow this sequence: first $N_1 \in R_1$ is chosen, then $N_1 \in R_1, \ldots,$ and finally $N_q \in R_q$. Let $N_{(p)}=\{N_1, N_2, \ldots, N_p\}$ denote the selected elements at iteration p of the rule extraction algorithm.

This algorithm is initialized in Step 1. Rule sets $R_1, R_2, \ldots, R_q$, which correspond to each object generated. In Step 2, the final distance measure matrix is also generated. Selecting the best possible element $N_p \in R_p$ is performed in Step 4. In Step 5, the counter for iterations is incremented. Note that one can implement the same algorithm for solving a set of rules with two, three, or up to (n–1) features. After the desired rules have been derived through the REA (see Fig. 2), the next step is to compose those rules to elicit the significant features in the system.

Step 1. Group the rules by output feature and generate a set of rules with m feature;
Step 2. Generate the final distance measure matrix;
Step 3. $N^{(0)} = \varnothing$ *and* $p = 1$;
Step 4. Select the minimum value of distance matrix for all $i \in R_p$
If distance matrix is not unique
select N_p be the rule with the minimum number of the distance matrix
If the reduct with the minimum number of distance matrix is not unique
select N_p be the rule with the minimum value of total distances
Else go to Step 5
Else go to Step 5;
Step 5. $N^{(p)} = N^{(p-1)} + \{N_p\}$.
If p = q and go to step 6
Else set p = p + 1 and go to step 4;
Step 6. Stop and output the rules ($N^{(q)} = N$ is the final set of rules).

Fig. 2. REA procedure.

2.1.3. The rule-validation procedure

The validation of generated rules can be illustrated as follows: Fig. 3.

2.2. Fundamentals of Fuzzy Logic System

2.2.1. Type I and II FLS

A FLS is a control system, able to handle numerical data and linguistic knowledge simultaneously. In general, it is a nonlinear mapping of input data to a scalar output data. It includes fuzzifier, rules, membership functions, inference engine, and defuzzifier [16]. The key difference between the Type I and II FLS is the membership function. Basically, the membership function of Type-I FLS is completely "crisp," while the membership function of Type-II FLS remains "fuzzy (uncertain)." Therefore, Type-II FLS is more capable to handle the factor, which includes uncertain values by nature. Typical examples for the use of Type I and II FLS in this study include "cutting speed" that uses Type I because the content of the cutting speed is more certain in nature. "Tool wear" uses Type II since the tool wear changes over time and it is difficult to measure or define precisely.

Literature review reveals that there are four sources of uncertainties associated with the FLS [16]: (1) the linguistic meaning of the words that are used in the antecedents and consequents of rules can be uncertain. For different people, the same word may mean different things; (2) consequents may have a range of values and not just a single value; (3) measurements that activate the FLS may be uncertain; and (4) the data that are used to tune the parameters of the FLS may be noisy. Furthermore, Type II FLS is very suitable for solving problems under the following conditions [17]: (1) measurement noise is non-stationary; (2) a data-generating mechanism is time-varying; and (3) linguistic terms being used have a non-measurable domain. Consequently, Type II FLS is more capable of operating the uncertain cases because it provides greater flexibility. Furthermore, the output of Type II FLS is possibly a random value with a range.

The framework of Type II FLS is illustrated in Fig. 4. Generally, Type II FLS is able to incorporate uncertainties about measurements, fuzzy rules, consequent choices, and unreliable training data into its outputs without sacrificing relevance.

The membership function of Type-II FLS can be easily visualized beginning with a Type I FLS membership function. For example, assuming the membership function of Type I FLS is a triangle (see Fig. 5), one could blur the original "crisp" triangle to make a "fuzzy" triangle, which represents the membership function of Type II FLS. Note the contents of the input and output parameters in the membership function of Type II FLS can be set with a range.

Step 1. Compare each decision rule derived from the rule-composing algorithm with each new object from test set. Calculate how many objects are matched with the rule;
Step 2. Repeat comparison of the decision rules with objects from test set until no decision rule is left;
Step 3. Calculate the accuracy of each rule by using the total matched objects (for each rule) divided by summation of total correctly matched objects and total incorrectly matched objects. If accuracy of the rule is greater than a predefined threshold value (e.g., 60%) then go to Step 4; otherwise, remove the rules;
Step 4. Stop and output the results.

Fig. 3. Rule-validation procedure.

2.2.2. Membership functions and fuzzy rule derivation

Fuzzy membership functions of input and output variables and fuzzy rule derivation are illustrated in more detail in this section. Assuming “cutting speed” (CS) and “tool wear” (W) are the two input variables, which have been identified as the significant factors in CNC machining (in this case, turning opertions). The output variable is designated as “surface roughness” (SF). The cutting speed has seven membership functions. The row vector for cutting speed can be stated in the form:

$$CS^T = \{S, MS, M, MF, F\} \tag{2}$$

where S = slow, 600 ft/min; MS = medium slow, 650; M = medium, 700; MF = medium fast, 750; and F = fast, 800. An isosceles triangle is used to represent the membership function of cutting speed. Tool wear (measured along the tool flank) has three membership functions, such that

$$W^T = \{S, M, L\} \tag{3}$$

where S = small, 0.000 in.; M = medium, 0.025; L = large, 0.05. The maximum allowable wear is set 0.030 in. The membership functions for tool wear are defined as follows:

$$\zeta_s(W) = [0.015 - W]0.015^{-1}, \quad \text{where } 0 \le W \le 0.015 \tag{4}$$

$$\zeta_M(W) = [W - 0.000]0.025^{-1}, \quad \text{where } 0 \le W \le 0.025 \tag{5.1}$$

$$\zeta_M(W) = [0.040 - W]0.015^{-1}, \quad \text{where } 0.025 \le W \le 0.040 \tag{5.2}$$

$$\zeta_L(W) = [W - 0.025]0.025^{-1}, \quad \text{where } 0.025 \le W \le 0.050 \tag{6}$$

Surface roughness has five membership functions, such that

$$SF^T = \{F, MF, M, MR\ R\} \tag{7}$$

where F = fine, 50μ in.; MF = medium fine, 70; M = medium, 90; MR = medium rough, 110; R = rough, 130. A singleton fuzzy output represents the membership functions. Surface roughness values are selected from the experiment data, which represent the average values of surface roughness under three different levels of tool wear and cutting speed. Since there are five and three partitions in input variables, there would be 15 rules. Basically, fuzzy rules dictate the relationship between the input variables and the output variables, which allow the proper selection of control actions according to the characteristics of the fuzzy inputs. Fuzzy rule statements are summarized in Table 3. Each rule includes IF, THEN statement. For example, the first rule started from upper left corner states that IF $CS = S$ and $W = S$ THEN $SF = F$.

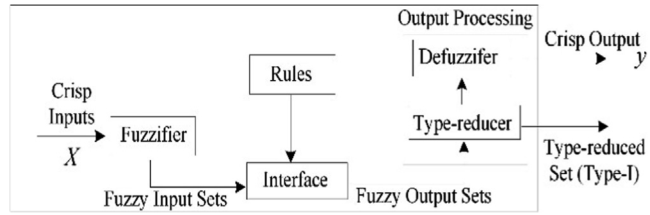

Fig. 4. Working of Type-II fuzzy logic system.

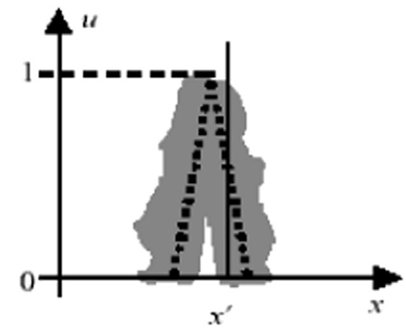

Fig. 5. Triangular membership functions of Type I and II FLS.

2.2.3. Hybrid fuzzy logic system

Here, a hybrid FLS is the combination of Type I and II FLS. Every FLS consists of at least four components such as fuzzifier, inference, defuzzifer and rules. In Type II FLS, one additional component called “type-reducer” is incorporated to deal with the interval type of output. Next, detailed description of the membership function of Type I and II FLS is introduced. Basically, the membership function of Type II FLS includes inner and outer triangles. The outer triangle is determined by the minimum, maximum, and most likely values called $x1$, m, and $x2$, while the inner triangle is determined by $x3$, m, and $x4$ (see Fig. 6). The membership function of Type I FLS included only one triangle as shown in Fig. 7.

The membership functions of Type I and II FLS are convertible. When $x1 = x3$ and $x2 = x4$, the output of the membership function of Type II FLS w^+ and w^- is overlapped as

$w^+ = w^- = w$. Therefore, the membership function of Type I FLS can be perceived as a special case of Type II FLS. As a result, the T-Norm and S-Norm of the membership function of Type I and Type II FLS can be verified in accordance with Mendel [16].

$$T - \text{Norm}\left(w^+, w^-\right) = \left(\min\left(w_1^+, w_2^+, \ldots, w_n^+\right), \min\left(w_1^-, w_2^-, \ldots, w_n^-\right)\right) \tag{8}$$

$$S - \text{Norm}\left(w^+, w^-\right) = \left(\max\left(w_1^+, w_2^+, \ldots, w_n^+\right), \max\left(w_1^-, w_2^-, \ldots, w_n^-\right)\right) \tag{9}$$

Table 3
Fuzzy rule table.

		Cutting speed				
		S	MS	M	MF	F
Tool Wear	Small	MR	MR	MR	MF	F
	Medium	R	MR	R	MR	MF
	Large	R	R	R	R	R

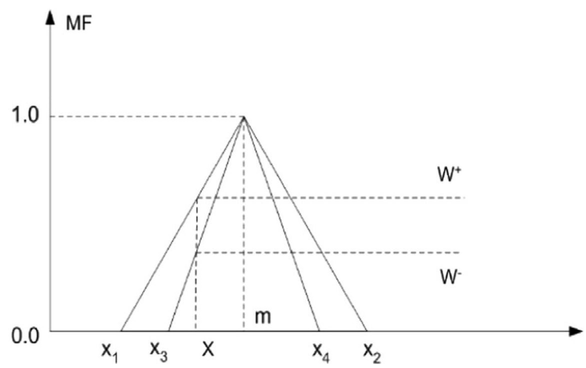

Fig. 6. The membership function of Type II FLS.

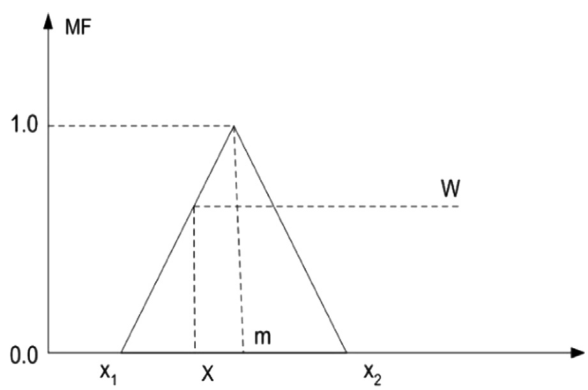

Fig. 7. The membership function of Type I FLS.

where n is the number of fuzzy variables. In the type-reducer component, the type reduced set is

$$f\left(w^{+}, w^{-}\right)=\alpha(1-S)+(1-\alpha)\left(w^{+}+w^{-}\right)/2 \tag{10}$$

where S is the segment area between w^{+} and w^{-}, α is a coefficient between 0 and 1. To simplify the computation of fuzzy output, one can take the average of w^{+} and w^{-} as the output. However, the average value might lead to overlooking some important information. For example, the average of the following two sets of data is identical: $w^{+}=0.8$, $w^{-}=0.2$ and $w^{+}=0.6$, $w^{-}=0.4$. Note that if the value of S increases (i.e., the segment area between w^{+} and w^{-} increases), then the range of the fuzzy output augments. In order to overcome pitfall of the "average" approach, a new segment area $(w^{+}–w^{-})$ to compensate the fuzzy output is introduced. To simplify the computation of area S, we assume the value of $(w^{+}–w^{-})$ is equivalent to S:

$$f\left(w^{+}, w^{-}\right)=\alpha\left(1-w^{+}+w^{-}\right)+(1-\alpha)\left(w^{+}+w^{-}\right)/2 \tag{11}$$

where $\alpha=0$, $w^{+}=w^{-}$, $f\ (w^{+},\ w^{-})=f(w)=w$. Therefore, this formula is also suitable for the membership function of Type I FLS.

2.3. Genetic Algorithm

The Genetic Algorithm (GA) operates on a population $P(k)$ of solutions rather than a single solution [18,19]. The operation of the standard GA is shown in Fig. 8.

In Fig. 8, the initialization and evaluation are the first steps of performing the standard GA and followed by the selection function, which intends to select a part of the initial population. The crossover function aims to achieve genetic diversity in the population by exchanging some genetic material in the relevant population members, while the mutation function aims to introduce an element of randomness. Despite its promise, a quite serious limitation of the GA is its primarily intention for unconstrained search. A number of techniques have been proposed for handling constraints in the GA. One such technique is to penalize infeasible solutions when evaluating each member of the population. An example of a well-known design system, which uses penalty functions, is Engineous [20]. Constraints in Engineous have a goal, actions, a weight, and conditions [21]. While penalty functions are useful when all variables are measured in the same unit, extreme care must be taken to avoid problems in scaling when applying penalty functions [22]. Perhaps, the primary difficulty with penalty functions is that slightly infeasible solutions, which otherwise would be of high quality, can contain a great deal of useful "genetic" materials. With high penalties, these solutions and their genetic materials are lost. With low penalties, the population may become filled with infeasible solutions.

This problem has led to a number of penalty functions, which increase with time [18,22]. Another basic technique for handling constraints is to use backtracking whenever constraints are violated. During initialization, crossover, or mutation, any attribute values that cause constraint violations are retracted and new values are assigned. The cycle repeats itself until a feasible solution is obtained.

2.4. Fuzzy adaptation using Genetic Algorithm

The prediction accuracy and learning speed of the GA are proved much better than back propagation algorithm in the artificial neural networks (ANNs) [18–20]. GA adaptation starts with approximate control rules derived from the empirical models and refines the control rules through a learning process when process variations occur. The fuzzy input remains the same, while the fuzzy output membership functions are adapted to minimize errors. In GA, the weights associated with the degree of input membership values are adapted because the inputs (CS and W) are assumed to affect the process output differently under process variations. Unlike

Begin

Set generation $k = 0$

Initialization: Initialize population, P(k)

Evaluation: Evaluate each population member of P(k) according to some measurement criterion

Do the following

Selection: Select part of P(k)

Crossover: Crossover (usually) each in turn two members of the selected part of P(k) to form a new population member

Set $k = k+1$

Mutation: Mutate at random the new population members P($k+1$)

Evaluation: Evaluate P($k+1$)

Until Termination Condition. This can be set for a variety of conditions.

End

Fig. 8. Operation of the Genetic Algorithm.

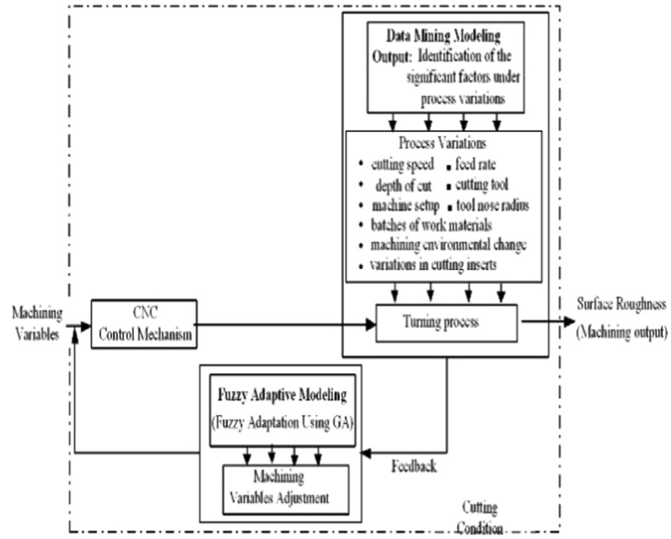

Fig. 9. Structure of the closed loop turning operation process.

the conventional ANNs, where a large number of training data is necessary, a single-step learning method can be selected. When the first part is made, the outcome of the surface roughness is measured and the weights go through a series of adaptation processes to minimize errors between the process output and the predicted values for successive parts. An error evaluation function is given as follows:

$$\text{Min } E(i) = 1/2\sum_{i=1}^{n}(y_i - d_i) \qquad (12)$$

where $E(i)$=error between the actual surface roughness and the fuzzy output, y_i=fuzzy output, d_i=process output, and n=a number of parts.

3. Methodology

The factors with significant impact on the quality were investigated in CNC turning. Specifically, conditions of the factors, which could meet the required surface finish, were identified. Recorded those specific conditions were used to develop a data mining and fuzzy logic based control system. This system can be used by industry to optimize the surface finish of machined metal (e.g., aluminum, steel) components (see Fig. 9). The study was conducted in two phases.

In data mining modeling phase (Phase I), all factors, which could influence surface roughness, were inquired (see Table 4). The significant factors, which have impact on the quality of surface, were determined through the rule extraction algorithm. The decision produced by the algorithm has become decision rules that are stored in the process expert system and is used to develop data mining and fuzzy logic based control system. In fuzzy adaptive modeling phase (Phase II), experimental data sets have been collected to facilitate and construct input membership functions. Then, a panel of experts was inquired to contribute their expert knowledge into the inference mechanism to develop the initial fuzzy rule base. Note that GA is used to train the fuzzy system.

Modifications of the rules were made coherent to the process variations. Third, based on the input membership functions and fuzzy rule base, the output membership functions were deduced. Finally, the output membership functions were used to cultivate the fuzzy adaptive predictor, extending the applicability of empirical models under variations as well as simulating the surface roughness curve. Adjustment of machining parameters based on the fuzzy modeling of surface roughness was performed along with implementation of fuzzy adaptive predictor into a CNC controller. The performance of the fuzzy adaptive predictor has been measured. One work piece material (6061-T6 aluminum) was investigated. The effects of cutting speed, depth of cut, feed rate, cutting tool, tool nose radius, and vibration on the performance of surface roughness estimation were studied. The turning experiments were performed on a Cincinnati Hawk CNC Turning Center.

4. Results and discussion

4.1. Phase I

The Department of Mechanical and Industrial Engineering at The University of Texas at El Paso has collected over 400 records (objects) of machining data and planned to investigate the machining factors to determine which factors have significant impact on the quality of surface finish. Specifically, conditions of the factors, which could meet the required surface roughness, were identified. The work piece is of 6061-T6 aluminum, cut by the Cincinnati Hawk CNC Turning Center. The effects of cutting speed, depth of cut, machine set up-modal stiffness, feed rate, cutting tool, tool nose radius and resultant cutting force on the performance of surface roughness estimation were studied. After roughing and semi-finishing operations, the surface roughness was measured by means of a Taylor Hobson surface profilometer. The contents of the outcome are recorded in binary format. “One” means surface

Table 4
Factor (feature) set of the turning operation process.

	Factor	Unit
*F*1	Resultant cutting force	N
*F*2	Cutting speed	sfm
*F*3	Depth of cut	in.
*F*4	Machine set up-modal stiffness	N/mm
*F*5	Feed rate	ipr
*F*6	Cutting tool	N.A.
*F*7	Tool nose radius	in.
*F*8	Tool wear (TW)	in.
Outcome 1	Surface roughness (Ra)	μ in.

roughness is acceptable, while “Zero” means unacceptable. The significant factors were determined through the rule extraction algorithm and the rule-validation procedure. The “Rough Set Based Decision Support System” software (see Fig. 10) was developed and implemented. It was installed using Apache 1.3 web server to enable the remote use. The system was developed with C++ language and the Common Gateway Interface (CGI) is used as a communication protocol between the server and client ends.

The data were split into two sets. One is for training, which was used to derive the decision rules; the other is for testing, which was used to verify the decision rules. Kusiak [3] suggested the split of the data set using the bootstrapping method according to the following ratio: 0.632 for training and 0.368 for testing. In this study, the training data set was collected for 267 parts and the testing data set was collected for 133 parts. Sixteen out of 267 parts in training set were unacceptable, while 7 out of 133 parts in the testing set were rejected. All decision rules derived by the rule extraction algorithm were expressed as IF–THEN rules as illustrated in Table 5. Number of support (see the 3rd column) was recorded from the training set, while the accuracy was verified by the rule-validation procedure from the testing set. One can observe that all preferred rules (e.g., high accuracy and more number of support) include Feature 5 (feed rate) and F8 (tool wear). Therefore, Features 5 and 8 have the significance in this rule induction.

4.2. Phase II

The decision rules were induced and the significant factors (features) were identified through the RST. The induced rules are beneficial to this study since they clearly indicate which factor under what condition can cause acceptable/unacceptable surface finish. However, those rules still cannot respond and/or model process variations. Many surface roughness control models for turning have been based on the experimental data. Empirical models are limited to a narrow domain and are very sensitive to the process variations so that even well-defined empirical models may become inaccurate under process variations. In time, a machining process output may drift and the surface roughness of turned parts becomes different than the values predicted by empirical equations. The empirical models have no means of incorporation changes in the process or responding to the process variations. Consequently, fuzzy adaptation using the GA is performed.

In Phase I, Feature 5 (feed rate) and Feature 8 (tool wear) are significant in the process. Therefore, both features have the most influence on the part quality. Basically, the feed rate and tool wear are different in nature. For example, the feed rate is easy to measure and it will not be affected by machining time, once it is set. On the other hand, the tool wear is relatively difficult to measure and it definitely is affected by machining time. Therefore, different types (e.g., Type I and II) of FLS are suitable to apply. The membership function of Type-I FLS is applied to “feed rate,” while the membership function of Type-II FLS is used for “tool wear.” Since two different types of the membership functions are incorporate in FLS, the combined FLSs are called a Hybrid FLS.

The “Data Mining and Fuzzy Logic Based Simulation System” software (see Fig. 11) was developed using Java Language, which incorporates multiple inputs and outputs. It is consisted of five modules: (1) user defined fuzzy input/output variables; (2) user defined fuzzy rules including all input variables (see Fig. 12); (3) GA for training historical data; (4) batch format for predicting the outcome; and (5) exchange (import/export) data with EXCEL®.

4.2.1. Membership function and fuzzy rules of the FLS

The triangle membership function is applied. Three points are incorporated in this membership function. They are left point $x1$, middle point m, and right point $x2$ on the x-axis. The membership function of the output (SF) is singleton, which is the special case of Type I FLS (i.e., $x1=m=x2$). The feed rate has seven membership functions (see Table 6). An isosceles triangle is used to represent the membership shape. Tool wear has three membership functions (see Table 7). A singleton fuzzy output represents the membership functions. Surface roughness values are selected from the experiment data, which represent the average surface roughness under three different levels of tool wear and feed rate (see Table 8). The partition of fuzzy input determines the number of rules. Since there are seven and three partitions for each input variable, there are 21 rules generated (see Table 9). Each row represents a level of tool wear, while each column represents a level of feed rate.

Fuzzy rule statements are as follows:

Rule 1: IF Feed Rate = VS AND W = S, THEN SF = VF;
Rule 2: IF Feed Rate = VS AND W = M, THEN SF = MF;
Rule 3: IF Feed Rate = VS AND W = L, THEN SF = R;
Rule 4: IF Feed Rate = S AND W = S, THEN SF = F;
Rule 5: IF Feed Rate = S AND W = M, THEN SF = MR;
Rule 6: IF Feed Rate = S AND W = L, THEN SF = VR;
Rule 7: IF Feed Rate = MS AND W = S, THEN SF = MF;
Rule 8: IF Feed Rate = MS AND W = M, THEN SF = MR;
Rule 9: IF Feed Rate = MS AND W = L, THEN SF = VR;

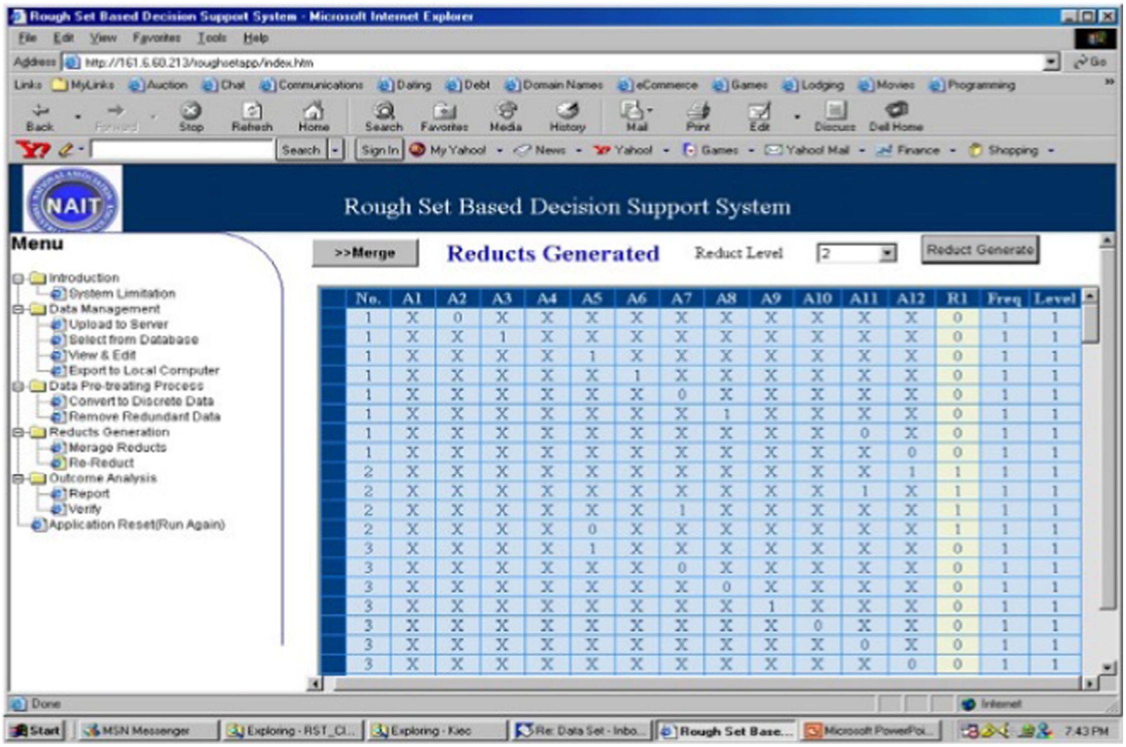

Fig. 10. Rough set application software.

Table 5
Results of decision rules.

Rule no.	Rule expression	No. of support	Accuracy (%)
1	IF ($F5=0.007$) AND ($F8=0.015$) THEN ($D=1$)	41	95
2	IF ($F5=0.007$) AND ($F8=0.015$) THEN ($D=1$)	36	100
3	IF ($F2=750$) AND ($F3=0.05$) AND ($F4=7$) THEN ($D=1$)	50	71
4	IF ($F3=0.05$) AND ($F5=0.007$) THEN ($D=1$)	30	89
5	IF ($F1=200$) AND ($F5=0.017$) AND ($F8=0.03$) THEN ($D=0$)	3	95
6	IF ($F5=0.017$) AND ($F8=0.03$) THEN ($D=0$)	3	100

Note: F1: resultant cutting force, F2: cutting speed, F3; depth of cut, F4: machine set up-modal stiffness, F5: feed rate, F8: tool wear.

Rule 10: IF Feed Rate = M AND W = S, THEN SF = MR;
Rule 11: IF Feed Rate = M AND W = M, THEN SF = R;
Rule 12: IF Feed Rate = M AND W = L, THEN SF = VR;
Rule 13: IF Feed Rate = MF AND W = S, THEN SF = MR;
Rule 14: IF Feed Rate = MF AND W = M, THEN SF = R;
Rule 15: IF Feed Rate = MF AND W = L, THEN SF = VR;
Rule 16: IF Feed Rate = F AND W = S, THEN SF = R;
Rule 17: IF Feed Rate = F AND W = M, THEN SF = R;
Rule 18: IF Feed Rate = F AND W = L, THEN SF = R;
Rule 19: IF Feed Rate = VF AND W = S, THEN SF = R;
Rule 20: IF Feed Rate = VF AND W = M, THEN SF = VR;
Rule 21: IF Feed Rate = VF AND W = L, THEN SF = VR.

4.2.2. Evaluation of Type I and Hybrid FLS

The performance of Type I and Hybrid FLS is compared. Again, the Hybrid FLS is combination of Type I corresponding to feed rate and Type II corresponding to tool wear. The key difference is that Type I is a special case of Type II. The Type I only needs three points to construct triangle membership function, while Type II needs five points to construct triangle membership function. Membership functions and input data of Type II are listed in Table 10. Experimental data (6061-T6 aluminum) are used in genetic algorithm training system. The data are shown in Table 11.

The objective of training is to minimize the errors. At the initial stage, the population size is set 100 and mutation rate is selected as 0.3. The ending condition is based on the generation number, and the maximum generation number is set 20. After several iterations of selecting different GA parameters, the user selects the best feasible solution to be the final solution. Since the GA is embedded in FLS, the surface roughness control system then automatically justifies/modifies the membership functions.

After performing training operations, the performance measurement of Type I and Hybrid FLSs is analyzed. The comparison of the end results between Type I (before/after training) and Hybrid (after training) FLS is listed in Table 12.

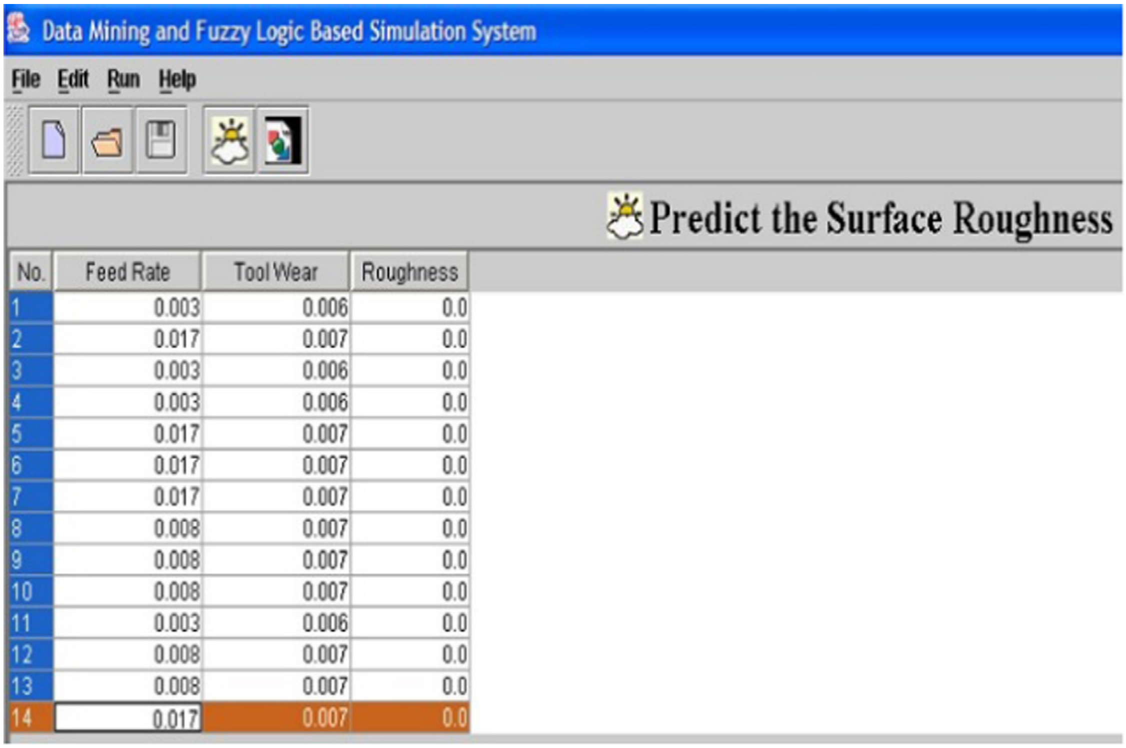

Fig. 11. Fuzzy set application software.

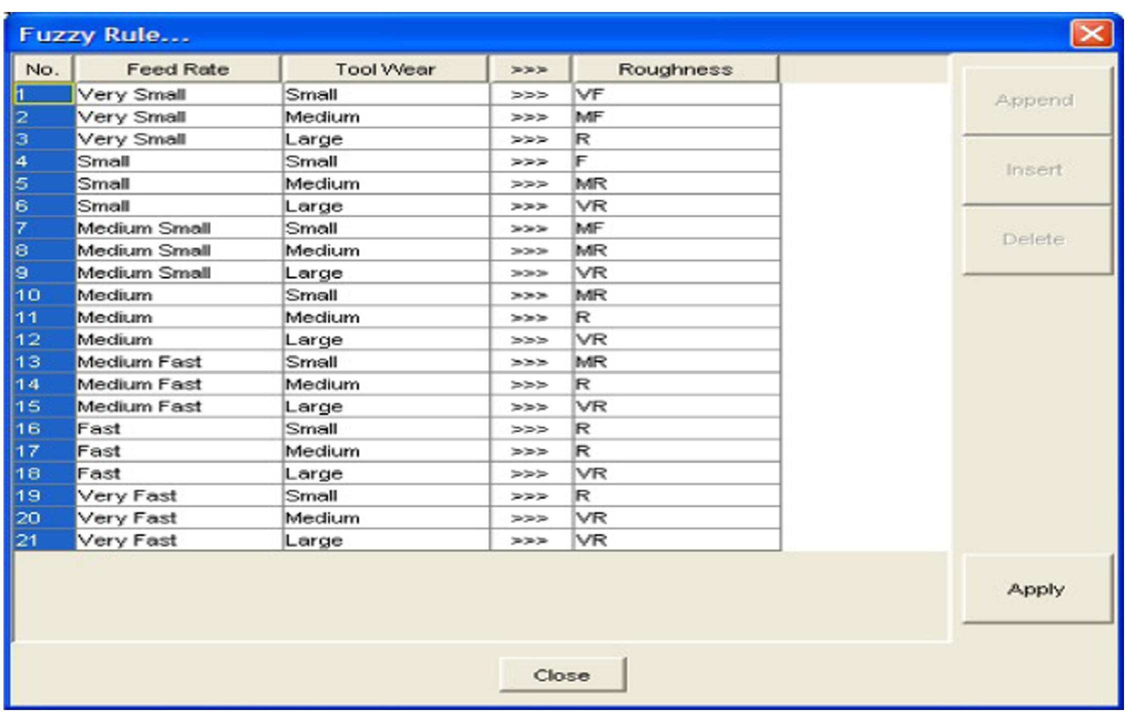

Fig. 12. User-defined fuzzy rules interface.

Table 6
Membership function and input data of tool wear.

Level	VS	S	MS	M	MF	F	VF
*x*1	0.003	0.003	0.005	0.007	0.010	0.013	0.015
m	0.003	0.005	0.007	0.010	0.013	0.015	0.020
*x*2	0.003	0.005	0.007	0.010	0.013	0.015	0.020

Table 7
Membership function and input data of tool wear.

Level	Small	Medium	Large
*x*1	0.000	0.000	0.015
m	0.000	0.015	0.030
*x*2	0.010	0.025	0.030

Table 8
Membership function for singleton output variable surface roughness.

Level	VF	F	MF	M	MR	R	VR
Value	30	50	70	90	110	130	150

Table 9
Fuzzy rule table.

	VS	S	MS	M	MF	F	VF
Small	VF	F	MF	MR	MR	R	R
Medium	MF	MR	MR	R	R	R	VR
Large	R	VR	VR	VR	VR	VR	VR

Table 10
Membership function and input data (W) of Type II FLS.

Level	Small	Medium	Large
*x*1	0.000	0.000	0.014
*x*3	0.000	0.001	0.016
m	0.000	0.015	0.025
*x*4	0.009	0.024	0.029
*x*2	0.011	0.026	0.031

Table 11
Experimental data used in the Genetic Algorithm training system.

Part no.	Feed rate	Tool wear	Ra
1	0.004	0.51779	18.23
2	0.015	0.58342	39.78
3	0.004	0.58211	24.41
4	0.004	0.53231	27.82
5	0.015	0.55256	32.31
6	0.015	0.56123	36.55
7	0.015	0.61167	39.36
8	0.010	0.62619	48.23
9	0.010	0.61614	48.28
10	0.010	0.60812	46.59
11	0.004	0.53451	25.51
12	0.010	0.60151	46.31
13	0.010	0.61125	47.53
14	0.015	0.55236	36.71
Level	Small	Medium	Large
*x*1	0.000	0.000	0.014
*x*3	0.000	0.001	0.016
m	0.000	0.015	0.025
*x*4	0.009	0.024	0.029
*x*2	0.011	0.026	0.031

Another testing data set (6061-T6 aluminum, 14 parts) is used for performance measurement of the aforementioned three different cases.

To be able to compare the Hybrid system, Type I FLS was applied as a baseline. Since the construction and rule derivation of fuzzy membership function depend on expert's knowledge and experience, a subjective judgment might lead into inaccuracy. Consequently, the GA-embedded FLS is used to refine the original fuzzy membership function with the empirical data. There are three different cases of comparison of prediction performance: (1) Type I FLS (before GA training), (2) Type I FLS (after GA training), and (3) Hybrid FLS (after GA training). The comparison is shown in Fig. 13. One can observe that Case II is better than Case I, since the yellow curve is more close to deep blue curve (the real surface finish). This indicates that the GA training is successful in modifying fuzzy membership functions in the Type I FLS. In general, Case III is prevailing over any other cases. This indicates that the Hybrid FLS after GA training results in the best feasible solutions. The results are anticipated since it is assumed that the Hybrid FLS is more effective in predicting surface finish than Type I FLS. The possible reason is that the Type II FLS handles uncertainty in the input variables more effectively than Type I FLS. In conclusion, the Hybrid FLS and the GA embedded FLS, which are different from conventional approaches, is promising in solving surface finish prediction problems in dynamic conditions.

Table 12
Comparison of prediction performance of three different cases: Type I before/after training and Hybrid FLS after training.

Part no.	Feed rate	Tool wear	Ra	Type I (before training)	Type I (after training)	Hybrid (after training) (after training)
1	0.003	0.60322	17.32	5.966	15.5724	15.5079
2	0.017	0.69897	38.87	49.8022	46.2889	41.6264
3	0.003	0.62237	23.34	11.711	19.2781	19.2426
4	0.003	0.64535	26.78	18.605	23.7248	23.7243
5	0.017	0.66067	31.23	45.7948	46.6331	37.7059
6	0.017	0.67982	35.65	47.7578	46.4645	39.6263
7	0.017	0.7028	38.13	50	46.2719	41.8199
8	0.008	0.72195	47.32	50	46.2719	41.8199
9	0.008	0.72961	47.22	50	46.2719	41.8199
10	0.008	0.71429	45.45	50	46.2719	41.8199
11	0.003	0.64152	24.45	17.456	22.9837	22.9773
12	0.008	0.71046	45.43	50	46.2719	41.8199
13	0.008	0.71429	46.65	50	46.2719	41.8199
14	0.017	0.66833	35.67	46.5382	46.5693	38.4332

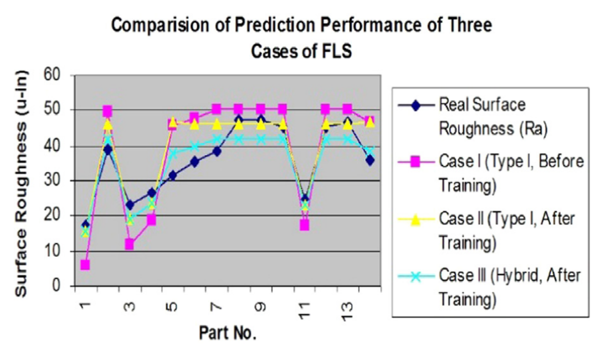

Fig. 13. Comparison of prediction performance of three cases of FLS.

5. Conclusion

In this study, the factors that impact surface finish in the machining process are identified. Numerous decision rules, which indicate under what condition which significant factor is able to be exploited to predict acceptable/unacceptable surface finish, were investigated. Rough set application software was used to derive the inductive rules. A surface roughness control system is developed in this study. This control system is a fuzzy logic based system to predict surface finish in CNC turning. Two different types of membership functions based on degree of uncertainty are applied in the fuzzy logic systems. The combination of those two different types of membership functions called a hybrid system was also developed. The computational results show the hybrid system prevails over others. Genetic algorithm was used for fuzzy adaptation in the control system. Fuzzy rules are automatically modified in the process of genetic algorithm training. The hybrid fuzzy system with genetic algorithm training demonstrated more effective prediction capability. The developed scheme may be incorporated into the CNC controller, where intelligent controls as well as unattended machining operations are expected.

Conflict of interest

The authors declare no conflict of interest associated with this manuscript.

Acknowledgments

This work was partially supported by the National Science Foundation (Award no. DUE-1246050) and the US Department of Education (Award nos. P031S120131 and P120A130061). This research was also supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (Grant no. NRF-2013R1A1A2006108. The authors wish to express sincere gratitude for their financial support received the duration of the research.

References

[1] Park K, Kim S. Artificial intelligence approaches to determination of CNC machining parameters in manufacturing: A review. *Artificial Intelligence in Engineering* 1998;**12**:127–34.

[2] Matsumura T, Obikawa T, Shirakashi T, Usui E. Autonomous turning operation planning with adaptive prediction of tool wear and surface roughness. *Journal of Manufacturing Systems* 1993;**12**(3)253–62.

[3] Kusiak A. Rough set theory: a data mining tool for semiconductor manufacturing. *IEEE Transactions on Electronics Packaging Manufacturing* 2001;**24**(1)44–50.

[4] Berry M, Linoff G. *Data Mining Techniques: For Marketing, Sales, and Customer Support*. New York: John Wiley & Sons; 1997.

[5] Dhar V, Stein R. *Seven Methods for Transforming Corporate Data into Business Intelligence*. Upper Saddle River, New Jersey: Prentice-Hall; 1997.

[6] Cheung DW, Ng VT, Fu AW, Fu Y. Efficient mining of association rules in distributed databases. *IEEE Transactions on Knowledge and Data Engineering* 1996;**8**(6)911–22.

[7] Pawlak Z. Rough sets. *International Journal of Computer and Information Sciences* 1982;**11**(5)341–56.

[8] Dubois D, Prade H. Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems* 1990;**17**:191–209.

[9] Heckerman D, Mannila H, Pregibon D, Uthurusamy R. *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Menlo Park*. CA: AAAI Press; 1997.

[10] Komorowski J, Zytkow J. *Principles of Data Mining and Knowledge Discovery*. New York: Springer-Verlag; 1997.

[11] Ziarko WP. *Rough Sets, Fuzzy Sets and Knowledge Discovery*. New York: Springer-Verlag; 1994.

[12] Pawlak Z. *Rough Sets: Theoretical Aspects of Reasoning About Data*. Boston: Kluwer Academic Publishers; 1991.

[13] Wojcik ZM. Rough sets for intelligent image filtering. In: Proceedings of the International Workshop on Rough Sets and Knowledge Discovery. Banff, Alberta, Canada; 1993; p. 399–410.

[14] Cabena P, Choi HH, Kim IS, Otsuka S, Reinschmidt J, Saarenvirta G. Intelligent Miner for Data Application Guide; 1999. Available from: ⟨http://www.readbooks.ibm.com⟩.

[15] Kusiak A, Tseng T-L. Modeling approach to data mining. In: Proceedings of the Industrial Engineering and Production Management Conference, Glasgow, Scotland; July 1999; p. 1–13.

[16] Mendel JM. Fuzzy logic systems for engineering: A tutorial. *IEEE Proceedings* 1995;**83**(2)345–77.

[17] Liang Qilian, Mendel Jerry M. Interval Type-2 fuzzy logic systems: Theory and design. *The IEEE Transactions on Fuzzy Systems* 2000;**8**:5.

[18] Michalewicz Z. *Genetic algorithms + aata Structures = Evolution Programs*. Berlin: Springer-Verlag; 1994.

[19] Goldberg D. *Genetic Algorithms in Search, Optimization, and Machine Learning*. MA: Addison Wesley, Reading; 1989.

[20] Powell D, Skolnick M. Using genetic algorithms in engineering design optimization with non-linear constraints. In: Forrest S, editor. *Proceedings of the ICGA-93*. San Mateo, CA: Morgan Kaufman; 1993. p. 424–31.

[21] Powell D, Tong SS, Skolnick M. EnGENEous do main independent, machine learning for design optimization. In: Schaefer D, editor. *Proceedings from the Third International Conference on Genetic Algorithms*. Morgan Kaufman; 1989. p. 151–9.

[22] Richardson J, Palmer M, Liepins G, Hilliard M. Some guidelines for genetic algorithms with penalty functions. In: Schaffer D, editor. *Proceedings of the ICGA-89*. San Mateo, CA: Morgan Kaufman; 1989. p. 191–7.