2011 International Conference on Information and Intelligent Computing
IPCSIT vol.18 (2011) © (2011) IACSIT Press, Singapore

A Hybrid Genetic Programming – Artificial Neural Network Approach For Modeling of Vibratory Finishing Process

A. Garg and K. Tai

School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore
e-mail: AKHIL1@e.ntu.edu.sg, e-mail: mktai@ntu.edu.sg

Abstract. Finishing processes are gaining importance over the last decades as manufacturers seek to improve process efficiency while meeting increasingly severe cost and product requirements. A number of researchers have employed conventional modeling techniques such as response surface methodology and linear programming but very few or none has paid attention to the non-conventional modeling approaches such as Artificial Neural Network (ANN), Genetic Programming (GP) and Fuzzy logic (FL) for studying the vibratory finishing process. Unlike conventional approaches, the independency of non- conventional modeling techniques on statistical assumptions ensures trustworthiness on the prediction ability of a model. The present study proposes a hybridized Genetic Programming-Artificial Neural Network (GP-ANN) approach for the formulation of mathematical models for the finishing process. The approach is based on error compensation which is achieved using an Artificial Neural Network (ANN) model in parallel with a Genetic Programming model. It is found that the hybridized GP-ANN models perform better than Genetic Programming models in terms of accuracy. The characteristics of the hybrid technique used are compared with response surface methodology. The results and analysis concludes that the GP-ANN approach is vital in those circumstances where data samples are fewer and can avoid the excessive GP runs for generating an optimal model. The ANN error model can also provide trustworthiness on the generalization ability of GP model whenever a new data sample is to be evaluated.

Keywords: Genetic Programming; Artificial Neural Network; vibratory finishing; hybrid; Artificial Intelligence

1. Introduction

In the vibratory finishing process, extensive knowledge based on industrial experience and empirical information has been accumulated but fundamentals of this process are developed mostly though trial and error procedure on the shop floor [1]. A review of the literature shows that very little research has been carried out with respect to vibratory finishing process and only a few mathematical models exist. A number of empirical studies on this process have been published [2, 3, 4, 5]. These studies are of limited usefulness as they have limited predictive capability since they do not consider the relationship between the key parameters in the process. Hashimoto [6] in his work established the fundamentals of vibratory finishing process by carrying out experiments, and based on these fundamentals he proposed mathematical models for the prediction of surface roughness, stock removal rate and optimum polishing time. Although the models describe the transient and steady rate behavior of finishing, the only parameters included in the model are time and initial surface roughness. Sofronas and Taraman [7] developed a mathematical model for vibratory finishing process by adopting the response surface methodology. He considered five input factors (Brinell hardness, projection width, processing time, media size and frequency of bowl) and three output factors (surface roughness reduction, projection height reduction, edge radiusing). Adequacies of models are checked by using analysis of variance. A conventional optimization technique Linear Programming was used to determine the optimum levels (processing time, media size, and frequency) of the developed models. Pande and Patel [8] used response surface methodology for studying vibratory burnishing process where four input factors (burnishing speed, feed, ball force, frequency and amplitude of vibration) and two output factors (surface roughness and micro hardness) are considered . The literature review on vibratory finishing process reveals that the process has been studied using conventional modeling approaches such as response surface methodology, linear programming, and analytical modeling. The dependency of conventional modeling approaches on the statistical assumptions induces a sort of suspicion about the reliability and trustworthiness of the prediction ability of the model. Nonconventional modeling approaches such as Genetic Programming (GP), Artificial Neural Network (ANN) and Fuzzy Logic (FL) are powerful Artificial Intelligence (AI) modeling tools being void of statistical assumptions. Because of this reason, these tools have been used comprehensively to study machining processes such as drilling, grinding, milling, turning, electro discharge machining , etc. [9].

In the present work, a hybridized Artificial Intelligence (AI) approach of Genetic Programming-Artificial Neural Network (GP-ANN) is proposed for the formulation of mathematical models for vibratory finishing process. Twenty-two data points are referred from an earlier study conducted on model development and optimization of vibratory finishing process [7]. The process input parameters considered are Brinell hardness (x_1), projection width (x_2), processing time (x_3), media size (x_4) and vibratory frequencies (x_5). The output parameters are projection height reduction (H), edge radius (E) and surface finish reduction (S). The hybrid approach is based on error compensation parallel approach using Artificial Neural Network (ANN) model in parallel with Genetic Programming model (GP). The objective of the present study is to compare the performance of proposed approach with standardized Genetic Programming (GP). The comparison of characteristics of response surface methodology and current hybrid approach is also summarized. The typical features of the hybrid GP-ANN approach are highlighted from the current study. The remainder of the paper is organized as follows. The method of hybrid approach is discussed in detail in Section II, the results and discussions are presented in section III and some concluding remarks are given in Section IV.

2. Genetic programming-Artificial neural network (gp-ann)

The hybridized GP-ANN approach is proposed for the formulation of mathematical models. For understanding the conception of GP-ANN, the Genetic Programming (GP) and Artificial Neural Network (ANN) are discussed below.

Genetic Algorithms (GA) attempt to find the best solution to a problem by mimicking the process of evolution in nature using Darwin's theory of 'survival of the fittest'. The individual potential solutions are selected based on the fitness value and recombined to produce better solutions. The generalization of Genetic Algorithms (GA) is Genetic Programming (GP) which was first studied at length by Koza [10]. The goal of Genetic Programming is to find a variable length program that solves the original problem when executed. In GP, programs or models are represented in tree structures and a relatively large population of tree based individuals with high diversity level is generated. Each member or tree or model comprised of functions and terminals, and the functions can be chosen from basic mathematical operators (+,-,*,/) and also Boolean algebraic operators (eg., AND and OR) or any other user defined expressions. The terminals may involve numerical constants and external inputs from the program [11]. Multigene genetic programming have also been developed by Hinchliffe et al. [12] and Hiden [13]. In this multigene Genetic Programming, several trees make a model. All of the genes have specific optimal weights and summation of weighted genes plus a bias term would form the final formula as the best obtained numerical model. Multigene GP can be written as

$$Y = a_o + a_1 \times \text{gene}_1 + a_2 \times \text{gene}_2 + \dots\dots.. + a_n \times \text{gene}_n \quad (1)$$

where a_o is a bias term and a_i is the weight of the ith gene.

Artificial Neural Network (ANN) is considered to be a powerful computational technique that takes into account nonlinear relationship between the variables. It is used when the relationship between the variables is unknown. The architecture of ANN consists of three layers, input layer, hidden layer and output layer. In the input layers, data is input and weighted sum of input is evaluated. In the hidden layers, data is processed and in the output layer, the output is produced. Each layer consists of a node or neuron. A neuron computes the weighted sum of input by acceptance of input and weights. The weights connect input layer to the hidden layer and are determined by training of the system. The activation or transfer function (generally sigmoid) is used to produce the output by receiving the weighted sum from the hidden layer. The transfer function is a relationship between the internal activation level of the neurons and the outputs. The sigmoid function varies between -1 and +1 and is used to model nonlinear relationships. During training of the ANN network, the weights are adjusted in such a way that the difference between the actual and the network output, i.e. the error, becomes minimum. Back Propagation (BP) is a training algorithm used frequently in ANNs. This algorithm redistributes the error associated with the output through the model, and weights are thereby adjusted accordingly. Several iterations are required to achieve the minimum error. A gradient descent optimization algorithm can be used to determine optimal weights but it results in slower convergence. Hence, the Levenberg-Marquardt Algorithm (LMA) that works on the principle of second derivatives is used for faster convergence [14].

In the present study, the hybridized Genetic Programming-Artificial Neural Network (GP-ANN) approach which comprises of genetic programming model and artificial neural network error model in parallel is presented as shown in Fig.1. Error is defined as the difference between the experimental and predicted values of the genetic programming model. The Error model is formulated using an Artificial Neural Network (ANN) based on evaluating the errors of the predictions of the genetic programming model. During ANN training, the five process parameters are inputs and the error is the output.

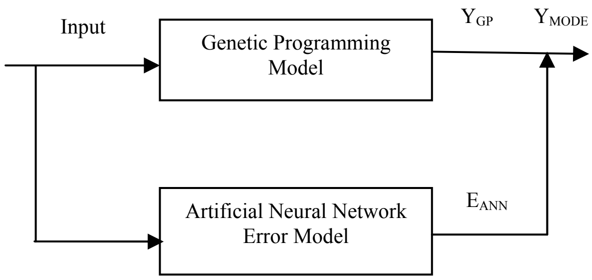

Figure 1. GP-ANN Model

The mathematical representation of the adopted approach is given by Equation (2) as follows:

$$Y_{MODEL} = Y_{GP} + E_{ANN} \tag{2}$$

A software tool GPTIPS is used in the current study to perform multi gene genetic programming for prediction of projection height reduction, surface finish reduction and edge radius. This software is a new "Genetic Programming and Symbolic Regression "code written based on Multigene GP for use with MATLAB [15]. This software tool has the capability of avoiding the problem of bloat in genetic programming by setting limitations on initial parameters such as the maximum number of genes, maximum depth of trees and genes, maximum number of nodes per tree, etc. For effective training of data set, the Kennard and Stone algorithm is used to select appropriate training and testing sets such that the whole data is distributed uniformly throughout the domain [16].The parameters adjusted for Genetic Programming are shown in Table I.

TABLE I. PARAMETERS FOR GENETIC PROGRAMMING

Parameters	**Values assigned**
Population size	100
Generations	150
Tournament selection and size	Lexicographic, 4
Termination criteria	0.25
Max depth of tree	4
Max genes	4
Functional set	(multiply, plus, minus, square)

Terminal set	(x_1, x_2, x_3, x_4, x_5, real numbers)

Three-layer feed forward neural network with back propagation learning is used for prediction of errors of the Genetic Programming (GP) model. The optimal architecture of ANN is determined based on minimum value of root mean square error of training and testing set. A trial and error approach is adapted to select number of neurons in the hidden layers. The lowest root mean square error (RMSE) is used as a criteria for selecting the optimal number of neurons. The parameters adjusted for Artificial Neural Network (ANN) are shown in Table II.

TABLE II. PARAMETERS FOR ARTIFICIAL NEURAL NETWORK (ANN)

Parameters	**Values assigned**
Training set	17
Testing	5
Number of hidden layers	1
Number of neurons in hidden layer	22
Activation function	Sigmoid
Number of epochs	1000
Learning rate	0.001
Architecture selection	Random
Target goal MSE	10^{-5}
Minimum performance gradient	10^{-5}
Optimization Algorithm	LMA

The percentage deviation is chosen as the criteria for evaluating the performance of the Genetic Programming and Genetic Programming-Artificial Neural Network (GP-ANN) models. Absolute percentage deviation is defined by the following expression

$$\%deviation = \sum \left| \frac{Predicted - Actual}{Actual} \right| \tag{3}$$

3. Results and Discussion

Genetic Programming and Artificial Neural Network (ANN) is carried out using the GPTIPS tool box [15] and MATLAB 2009a respectively. The models evolved during Genetic Programming (GP) for projection height reduction (*H)*, edge radius (*E*) and surface finishing reduction (*S*) respectively are as follows:

$$H = 0.0000051x_3x_4x_2^2 + 0.0000001x_3x_4x_2 + 2x_5^2 + x_3x_4 - 0.00005x_3x_4x_2x_5 + x_5 + x_3x_4 + 0.000000000313x_3x_3x_1 - x_5(x_3 - x_5 + 4.4662) - 0.00000000002619 \tag{4}$$

$$E = (0.000054x_3 + 1)(x_4 - 1)x_1^2 + (0.168 - 0.0070x_3 - x_4)(x_4 - 1) - 0.168x_4^2x_1 + 28.61x_4 - 18.56 \tag{5}$$

$$S = 0.00382(2x_3 - x_4 + 7.555)(x_2 - x_3 + x_4x_5) - 0.000177(1.2171x_2 + 1.21x_3)(x_1 - x_2 + x_5) - 0.00698(x_3 + 0.217x_4^2) - x_3 + x_4x_5 - 0.568x_3x_4^3 - 0.568x_2 - 7.59 \tag{6}$$

The error calculated from the Genetic Programming models is trained in the Artificial Neural Network (ANN) as outputs. The error predicted using the ANN network is finally compensated in the Genetic Programming model for the formulation of the GP-ANN model. The absolute percentage deviation of GP and GP-ANN models are calculated and shown in Table III.

TABLE III. PERFORMANCE OF GP AND GP-ANN MODELS

Error	**H**		**E**		**S**	
	GP	***GP-ANN***	***GP***	***GP-ANN***	***GP***	***GP-ANN***

Training	7.37	2.24	6.04	2.43	2.03	0.63
Testing	3.77	0.77	1.23	0.36	1.78	0.51
Overall	11.09	3.01	7.28	2.80	3.82	1.14

The models given by Equation (4), (5) and (6) are nonlinear and complex. The addition of error term predicted from neural network adds only a constant term to the Genetic Programming (GP) model to form the GP-ANN model. This results in the increase in accuracy of the Genetic Programming (GP) models maintaining its complexity. It is apparent from Table III that GP-ANN model outperforms GP model in terms of accuracy. The percentage reduction in error for training data set for projection height reduction (*H*), edge radius (*E*) and surface finish reduction (*S*) is 69.60%, 59.70% and 68.96% respectively and for testing data set, it is 79.50%, 70.73% and 71.34% respectively. The highest reduction in absolute percentage deviation is achieved in projection height reduction (*H*). In other words, maximum error compensation is provided by ANN model for prediction of projection height reduction. The reason could be that Genetic Programming (GP) performs poorly and the ANN model is able to be trained with the data satisfactorily. The characteristics of the currently used hybrid approach are compared with response surface methodology (RSM) [7]. The comparison is summarized in Table IV.

TABLE IV. COMPARISON OF GP-ANN AND RESPONSE SURFACE METHODOLOGY

Technique	RSM	GP-ANN
Interpretation	Easy	Complex
Model	Linear grey box	Nonlinear white box
Statistical Assumption	Yes	No
Generalization ability	No	Yes
Reproducibility of a Model	Yes	No

From the comparison in Table IV, it is obvious that for most of points such as type of model, assumptions, and generalization ability, Genetic Programming (GP) is better than response surface methodology. The disadvantage of using Genetic Programming (GP) is that the model is not reproducible and is not easily interpretable. The poor reproducibility means that the probability of getting the same model again using the same GP parameter settings is very low. This is because Genetic Programming (GP) works on randomization and initialization of population of models. The complex models evolved by GP as given by Equation (4), (5) and (6) also make interpretation difficult.

4. Conclusion

The results show that the GP-ANN approach provides accurate generalized models which perform better than the GP models. The nonlinear GP-ANN models for surface finish reduction (*S*), projection height reduction (*H*) and edge radius (*R*) are formulated by compensating Genetic Programming models with the ANN error models. Such models can be further optimized using Genetic Algorithm or can be coded into a system for online monitoring and prediction. The current study interprets that GP-ANN can be vital in the circumstances where data samples are few and there is necessity of rigorous tuning of GP parameter settings for determining the optimal model. In such situations, the time involved for evaluations i.e. number of runs using Genetic Programming can drastically increase the overall cost of data analysis. Hence to avoid high cost analysis, ANN can be used in parallel to provide accuracy or compensation to the Genetic Programming model. Moreover ANN error model can also provide the sense of trustworthiness in the prediction of Genetic Programming model whenever a new data sample is to be evaluated. In this way, we made use of the explicit form of Genetic Programming model pooled with the accuracy of an ANN model to produce an accurate hybrid model. In the future, experimental processes such as fused deposition modeling (FDM) where samples are collected based on design of experiment (DOE) and few in number could be appropriately studied using GP-ANN approach.

5. References

[1] J. Domblesky, R. Evans, and V. Cariapa, "Material removal model for vibratory finishing," International Journal of Production Research, vol. 42, pp. 1029-1041, 2004.

[2] L. Gillespie, “A quantitative approach to vibratory deburring effectiveness”, Society of Manufacturing Engineers, Technical paper MR75-11 (1975).

[3] P. C. Brust, Surface improvement by vibratory cascade finishing process. Dearborn, Mich.: Society of Manufacturing Engineers, 1997.

[4] D. A. Davidson, "MECHANICAL SURFACE PREPARATION - Mass Finishing Processes," Metal finishing., vol. 1, p. 75, 2006.

[5] S. Wang, R. S. Timsit, and J. K. Spelt, "Experimental investigation of vibratory finishing of aluminum," Wear, vol. 243, pp. 147-156, 2000

[6] F. Hashimoto "Modelling and optimization of vibratory finishing process," CIRP annals ... manufacturing technology., vol. 45, p. 303-306, 1996

[7] A. Sofronas and S. Taraman, "MODEL DEVELOPMENT AND OPTIMIZATION OF VIBRATORY FINISHING PROCESS," Int J Prod Res, vol. 17, pp. 23-31, 1979.

[8] S. S. Pande and S. M. Patel, "Investigations on vibratory burnishing process," International Journal of Machine Tool Design and Research, vol. 24, pp. 195-206, 1984

[9] M. Chandrasekaran, M. Muralidhar, C. M. Krishna, and U. S. Dixit, "Application of soft computing techniques in machining performance prediction and optimization: A literature review," International Journal of Advanced Manufacturing Technology, vol. 46, pp. 445-464, 2010

[10] J. R. Koza, Genetic programming : on the programming of computers by means of natural selection. Cambridge, Mass.: MIT Press, 1992

[11] M. H. Baziar, Y. Jafarian, H. Shahnazari, V. Movahed, and M. Amin Tutunchian, "Prediction of strain energy-based liquefaction resistance of sand-silt mixtures: An evolutionary approach," Computers and Geosciences, 2011

[12] Hinchliffe,M.P., Willis,M.J., Hiden,H.G., Tham,M.T., McKay,B., Barton,G.Modelling Chemical Process Systems Using a Multi-Gene Genetic Programming Algorithm Late Breaking Paper. GP'96, Stanford: USA.1996

[13] Hiden, H.G. Data-Based Modelling using Genetic Programming. PhD Thesis, Dept. Chemical and Process Engineering, University of Newcastle. UK: 1998

[14] Q.H.Wang , Journal of Qinghai University 22 (2004) 82.

[15] Searson, 2009a Searson, D.P., 2009a. GPTIPS: Genetic Programming & Symbolic Regression for MATLAB. http://gptips.sourceforge.net

[16] Daszykowski, M., Serneels, S., Kaczmarek, K., Van Espen, P., Croux, C. & Walczak, B.TOMCAT: A MATLAB toolbox for multivariate calibration techniques. Chemometrics and intelligent laboratory systems// an international journal sponsored by the Chemometrics Society.2007.85, 26