12th International Research/Expert Conference
"Trends in the Development of Machinery and Associated Technology"
TMT 2008, Istanbul, Turkey, 26-30 August, 2008

ANALYTICAL STUDY OF DIFFERENT APPROACHES TO DETERMINE OPTIMAL CUTTING FORCE MODEL

Zeljko Krizek
Wienerzeitung Gmbh
Wiedner Gürtel 10, Vienna
Austria

Zoran Jurkovic
Faculty of Engineering, University of Rijeka
Vukovarska 58, Rijeka
Croatia

Miran Brezocnik
Faculty of Mechanical Engineering, University of Maribor
Smetanova ulica 17, Maribor
Slovenia

ABSTRACT
Determination of optimal machining parameters is an engineering task with aim to reduce the production cost and achieve desired product quality. Such exercise can be tackled on many different ways. The goal of this work is to present some of the possible approaches and to benchmark them among each other. These principles are analyzed: response surface methodology (RSM), evolutionary algorithms (GA & GP), support vector regression (SVR) and artificial neural networks (ANN). All methods implement completely different data handling philosophies with the same goal, to build the model which is able to predict cutting force in satisfying manner. Those aspects are chosen to be evaluated and compared: average percentage deviation of all data, ability to find generalized model and minimize the risk of over fitting and at least the runtime of each single model determination. Average percentage deviation is one of the best indicators of the quality of model. The ability to find generalized model is good indicator of the flexibility of model, and shows how model deals with unknown data. The runtime is important in a real time environment or in scenarios where conditions change frequently. Cutting force data used in this benchmark comes from experimental research of longitudinal turning process.
Keywords: Cutting Force; Modelling; Response Surface Methodology; Genetic algorithms; Genetic programming; Support Vector Regression; Artificial Neural Networks

1. INTRODUCTION
Determination of optimal cutting force model based on measured (experimental) data can be handled on many different ways. Some of possible approaches are going to be presented and to be benchmarked here among each other. The main goal is the model building of cutting force as value depending on independent values (spindle speed, feed rate and cutting depth). The benchmark regarding models compares the efficiency of approaches and measures the quality of chosen strategies trough the produced models. The strategies can be put in two categories. To the first category belong paradigms that have clear polynomial structure of models. Those are: response surface methodology and evolutionary techiques. And to the second category belong paradigms that are based on statistical learning theory and in this case the models don't have functional form but rather consists of parameter list (SVR and ANN). The second aspect watched here, is time needed to become a model from data
This point could be interesting in complex environments with conditions that change often and probably the machining parameters have to be adjusted a few times a day. Experimental data applied for this purposes consist of one training set of 15 data and one validation set with 26 data. The main performance indicator measured and presented in this work is the percentage deviation between experimental data and data won after executing models. Indirectly trough these percentage deviation one can observe the capability of concepts to generalize or to build solutions that tend to over fitting.

2. CONCEPTS

For each of the presented concepts the role of the fitness function is defined as the percentage deviation between sample data, and data produced (predicted) by model.

$$\Delta = \frac{\sum_i^n \Delta i}{n}, \quad \Delta i = \frac{| Ei - Gi |}{Ei} \, 100\,\% \tag{1}$$

n is size of the data, Δi is percentage deviation of single sample data, Ei is measured cutting force and Gi cutting force predicted by model.

2.1. Response surface methodology

The response surface methodology (RSM) technique for design of experiments (DoE) seems to be the most wide-spread methodology for prediction problem. In the RSM, the factors that are considered as most important are used to build a polynomial model. A functional relationship between main cutting force and the independent variables under finish turning process investigation is postulated by:

$$F_c = C_1 + b_1 v + b_2 f + b_3 a + b_{12} vf + b_{13} va + b_{23} fa + b_{123} vfa + b_{11} v^2 + b_{22} f^2 + b_{33} a^2 \tag{2}$$

Both the high adjusted R^2 value and the ANOVA show that model has a satisfactory goodness of fit.

2.2. Genetic Algorithm

Genetic algorithm (GA) strategy utilized by model generation presented here is based on the work described in [1]. In this approach GA is implemented as simulation in which a population of candidate solutions (individuals) evolves toward better solutions regarding fitness function. All of the common GA operators and concepts were applied (crossover, mutation, tournament selection, etc.). The individuals are represented as vectors of double numbers. Each single member of vector represents one of the constant factors in polynomial shown below (C_1, b_1, b_2, b_3, b_{12}, b_{13}, b_{23}, b_{123}, b_{11}, b_{22}, b_{33})
The searched polynomial has the same following structure as model (2):

2.3. Genetic Programming

Genetic programming (GP) [2, 3] is a specialization of genetic algorithms where each individual is a computer program. It is a machine learning technique used to find computer program that meets conditions defined trough fitness function. GP-modelling was used to determine polynomial based on following operators (+, -, *, /) that is able to meets conditions of the fitness function. GP evolves polynomial represented in a memory as Trees consisting of operators as nodes and operands as leafs. The main weakness of the GP approach is it computationally intensity, and these guides this approach to problematically runtime behaviour. The structure of the model is polynomial but the end structure couldn't be specified until the model is finished.

2.4. Support Vector Regression

Support Vector Machines [4, 5] are supervised learning systems based on statistical learning theory developed by Vapnik and Chervonenkis. These systems use a hypothesis space of linear functions in a high dimensional feature space. Such a learning system can be used for classification and regression. The basic idea in case of regression is to map the experimental data into a high-dimensional feature space F via a nonlinear mapping, and to do linear regression in this space. For the purposes presented here the best result was received with the so called (RBF) kernel (radial basis function) $K(xi, xj) = \exp(-\gamma * | xi - xj |\wedge 2)$ as kernel and υ-Regression. RBF kernel maps non-linearly samples into a high dimensional space. This kernel can handle cases when the relation between the data is nonlinear. The presence of outliners or noise in a data could significantly decrease affect of generalisation of SVM in this case SVR. This problem can be handled with the concept of soft margins. Normally in each SVM there is margin of training set, and in real world application there are also noise points that are outside of this margin. To handle this noise data soft margin concept was introduced. This concept allows the margin to make few mistakes. A kind of soft margin handling is υ-Regression. υ is an upper bound on the fraction of margin errors and is the lower bound on the fraction of support vectors. Important parameters are γ (allowed margin) and C (cost of constraints violation).

2.5. Artificial Neural Networks

An Artificial Neural Network (ANN) [6] is an information processing system that is composed of a number of interconnected processing elements (neurones) working to solve specific problems. ANN can be seeing also as an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. The architecture of such a system is made of an input layer, a hidden layer and an output layer consisting of neurons. A neuron is an information process unit implemented as mathematical function. The artificial neuron receives one or more inputs (representing the one or more dendrites) and sums them to produce an output (synapse). ANN presented here, is a two-layer feed-forward neural network trained with Levenberg-Marquardt back propagation.

3. EXPERIMENTS AND RESULTS

Different experiments executed on provided data offer results and indicators that are going to be presented here. Directly comparison of gotten models it is not so easy task. The concepts are so different that there is no way to compare single steps, configurations, strategies or settings in algorithms. The only way to check performance is to observe percent deviation and runtime (how long was calculating time) of final models. Another interesting point could be the readability of obtained models for human being and later usage in some calculations as part of the bigger systems, but such a property could be heavily quantified.

- Model obtained by response surface methodology approach has following form:

$$F_{cf} = 130{,}075 - 1{,}1076 v_c + 2115{,}06 f + 603{,}9025 a_p - 3{,}4836 v_c f - 0{,}94445 v_c a_p - 2302{,}8 f a_p + 8{,}322 v_c f a_p + \\ + 0{,}001612 v_c^2 - 1223{,}6 f^2 - 43{,}01875 a_p^2. \tag{3}$$

- Model obtained by GA:

$$F_{cf} = 354{,}5579 - 2{,}2926 v_c + 2435{,}2718 f + 672{,}9255 a_p - 3{,}77751 v_c f - 1{,}0907 v_c a_p - \\ - 2595{,}4597 f a_p + 9{,}0536 v_c f a_p + 0{,}0030701 v_c^2 - 2015{,}2932 f^2 - 48{,}7725 a_p^2 \tag{4}$$

- Model obtained by GP:

$$F_{cf} = (((-1.165+f)-((-5.910-2*a_p))+((((v_c-((((((((((-4.540+(((f*(a_p*(-1.165+f)))-7.654)-7.654))-7.654)-7.654)-7.654)-7.654)-7.654)+(((((((-4.540+(((f*(-5.910-a_p))-7.654)-7.654))-7.654)-7.654)-7.654)-7.654)-7.654)-7.654))-7.654)-7.654)-7.654))*(((f+v_c)-v_c)-7.654))+((((-1.165+f)+a_p)*v_c)+((v_c+f)+((v_c+(f+v_c))+((((f+v_c)+((2*v_c)*f))+((f-5.910)-7.654))+((2*v_c)*f))))))*((a_p*(-1.165+f))*f))) \tag{5}$$

- Models produced by SVR and ANN cannot be presented in a polynomial form.

SVR model parameters are: SVM-Type: nu-regression, SVM-Kernel: radial basis, cost: 969, gamma: 0 .015, nu: 0.5, Number of Support Vectors: 8.

ANN model parameters are: training epochs: 300, number of hidden neurons: 80, training function: *"trainlm"* (Levenberg-Marquardt back propagation), adaptation function: *"trains"* (trains a network with weight and bias learning rules with sequential updates), gradient function: *"calcjx"* (calculates the Jacobian of a network's errors with respect to its vector of weight and bias values X)

Table 1. Performance data of obtained models

Model	Percent deviation of training samples %	Percent deviation of test samples %	Algorithm runtime to determine model
RSM	1,72	3,88	1 second
GA	0,93	4,18	3 hours
GP	4,73	5,38	6 hours
SVR	0,91	3,70	1 second
ANN	4,33	10,83	1 second

The satisfying results reaches SVR model with the best deviation for both data sets. Runtime behaviour is also one of the best. The problematical point is the bad readability of determined model.

It is easier for a human being to read and understand polynomial form of model. ANN has the worst performance regarding percent deviation. It tends to over fitting. ANN has a model that is not so bed by describing of training data but it has problems to handle unseen data. RSM has the best ratio (smallest difference between percentage deviation of training and test data sets). GA has good performance, but it has bad runtime (it takes long time to get model). The worst one regarding runtime is GP. GP has the largest space to search trough for the best model, but ratio between training and test data is one of the best. Figure 1 show graphically the deviation between measured data from validation set and data won after models execution. This figure confirms also the results from Table 1. It easy to see that ANN has the biggest deviation and that SVR and RSM competitors, because they have almost the same deviations from measured data.

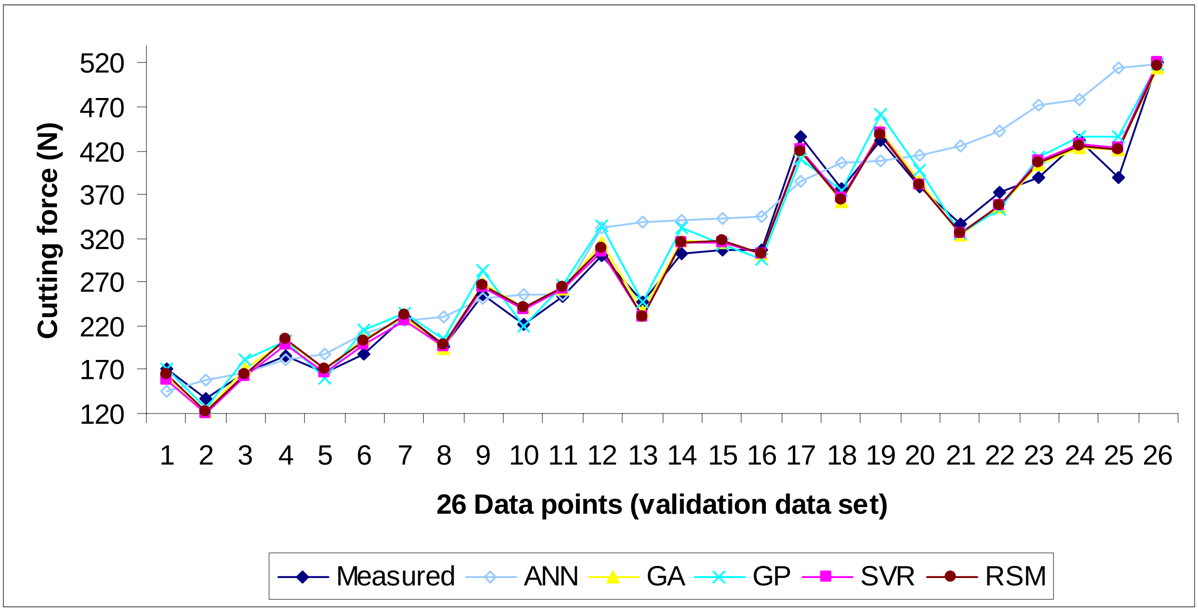

Figure 1. Performance of models presented for each single data in a validation set

4. CONCLUSION

Different strategies of model building from experimental data were analysed and presented. As mentioned earlier this work handles response surface methodology (RSM), evolutionary algorithms, support vector regression (SVR) and artificial neural networks (ANN). Percent deviation and runtime of algorithm was chosen to quantify the performance of the won models. The interpretation of these results allows following conclusion. The best percent deviation was reached by SVR model (training data 0,913% and test data 3,70%). The runtime of this SVR comparable to the others was ok. The only problem of this method is the bed readability for human being of the determined model. The main problem of the evolutionary methods is not so the generalisation ability but the runtime (time to get the model). The problem by the statistical learning theory is the strength preparation of the data for model building.

5. REFERENCES

[1] Holland John H.: Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, 1975.
[2] Koza J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, 1992.
[3] Brezocnik M., Kovacic M.: Integrated genetic programming and genetic algorithm approach to predict surface roughness, Materials and Manufacturing Processes, 475-491, 2003.
[4] Cristianini N., Shawe-Taylor J.: An Introduction to Support Vector Machines, Cambridge University Press, 2000.
[5] Smolay A. J., Schölkopf B.: A Tutorial on Support Vector Regression, 2003.
[6] Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, 2006.