Application of Genetic Algorithms to Optimize Neural Networks for Selected Tribological Tests

Tomasz Trzepieciński\(^1\) and Hirpa G. Lemu\(^2\)

1. Department of Materials Forming and Processing, Rzeszow University of Technology, Rzeszów 35-959, Poland
2. Dept. of Mechanical and Structural Engineering and Material Technology, University of Stavanger, Stavanger N-4036, Norway

Received: January 16, 2012 / Accepted: February 03, 2012 / Published: February 25, 2012.

Abstract: This paper presents a method of determining the friction coefficient in metal forming using multilayer artificial neural networks based on experimental data obtained from strip drawing test. The number of input variables of the artificial neural network has been optimized using genetic algorithm. This process is based on surface parameters of the sheet and dies, sheet material parameters and clamping force as input parameters to train the neural network. In addition to demonstrating the fact that regression statistics model using genetic selection and intelligent problem solver are better than models without preprocessing of input data, the sensitivity analysis of the input variables has been conducted. This avoids the time-consuming testing of neurons in finding the best network architecture. The obtained results from this study have also pointed out that genetic algorithm can successfully be applied to optimize the training set and the outputs agree with experimental results. This allows reduction or elimination of expensive experimental tests to determine friction coefficient value.

Key words: Friction, friction coefficient, genetic algorithm, artificial neural networks, intelligent problem solver.

1. Introduction

Friction regimes encountered during deep drawing of thin metal sheets are very complex and depend on several parameters such as the contact pressure, sliding velocity, sheet metal and tool surface roughness, kinematics of tool motion, tool and blank material, lubrication and temperature [1-3]. One of the main factors influencing frictional resistance is surface topography of deformed sheet. As friction between the sheet and tools is one of the important factors affecting the quality of drawpiece, clarifying the frictional condition for modeling and analysis of sheet metal forming processes is very essential. The workpiece surface topography and asperity contact are also important factors that control the mechanisms of lubrication in metal forming process.

Amontons-Coulomb simple friction model (constant coefficient of friction) is often used in many research works intended to describe the real frictional properties of deep drawing. This model is considered to be correct when the frictional resistance does not depend on nominal area of the contact. Furthermore, it is applicable within limited ranges of contact pressure because friction, to a significant degree, depends on sliding velocity and varied contact geometry. Measuring the frictional properties of a material always requires replicating the conditions under which the friction occurs, including the material sliding against test material, the geometry of contact, the surface conditions, and the relative speed of the sliding bodies.

A few regions of deep drawing process such as the wall, bottom and flange of the cup can have different stress and strain states, thus the sliding speed and friction conditions can vary. In certain applications such as microforming, friction size effects play significant role [4] and the values are determined...
based on measurement of friction force and normal force.

The aim of the experimental research reported in this article is to determine the coefficient of friction for different grades of sheets metals. The friction tests were carried out for wide range of contact pressures between sheet-tool interfaces. Considering significant amount of factors influencing frictional resistance during sheet metal forming, analytical determination of the friction coefficient is practically impossible. Due to this reason, a multilayer artificial neural network (ANN) is utilized. An important factor that influences the correct operation of ANN is selection of suitable input variables. A method that is often used to reduce the number of experiments is experiment optimization [5]. The experimental data set used in this research for ANN training is thus optimized by using genetic algorithm (GA). There has been growing interest in applying ANN to engineering fields for solving complicated problems including control, optimization, regression and prediction. Based on abductive modeling techniques, the neural networks represent sophisticated and uncertain relationship between input and output variables. The utilization of ANN enables the behavior of complicated system to be modeled and predicted based on known experimental data. Accordingly, the network can solve complex nonlinear problems using current and historical data [6]. The purpose of this study is to further examine the ability of neural networks to predict accurately the friction coefficient in sheet metal forming.

The rest of the article is organized as follows: Section 2 presents the experimental procedure used in the test and the measured results. After a short overview of ANN basics in Section 3, the formulation of the optimization model in GA is presented in Section 4. Finally, Section 5 presents the concluding remarks.

2. Experimental Procedure

Friction tests were conducted by strip drawing method placed between two fixed cylindrical rolls with equal radii (Fig. 1). The test was carried out in such a way that a strip of the sheet was clamped with specified force between two cylindrical rolls of equal radii 20 mm. Values of both forces, the clamping force $F_c$ and the pulling force $F_p$, were constantly recorded using electric resistance strain gauge technique, 8-channel universal amplifier of HBM’s QuantumX data acquisition system and computer PC. The specimens for the friction tests were made of two brass sheet metal M63 and M90. The M63 brass quality has hardening state of r, z4 and z6, while the M90 quality has hardening state of z4. Samples were prepared as a strip having 20 mm width and about 200

![Fig. 1  View of measuring position: 1-frame, 2-working rolls, 3-load cells, 4-specimen, 5-universal amplifier, 6-computer PC.](image-url)
mm length, cut along transverse directions of the sheet. The rolls were made of cold working tool steel hardened to 58 HRC. The tests were conducted under the following conditions:

- Surface roughness of rolls measured along the generating line: 0.32; 0.63; 1.25 and 2.5 µm;
- Clamping force: 0.4; 0.8; 1.2; 1.6 and 2 kN;
- Sliding velocity: 0.002 m/s, which is relatively high compared with the industrial values.

During the recording of the pulling and clamping forces, the sheet was drawn for a distance of about 10 mm. Next the clamping force value was increased simultaneously during the tests. To realize dry conditions both rolls and sheet specimens were degreased using acetone. The mean value of the friction coefficient is determined according to Eq. (1) for the stabilized range of values of \( F_p \) and \( F_c \):

\[
\mu = \frac{F_p}{2:F_c}
\]  

(1)

where: \( F_p \)—pulling force, \( F_c \)—clamping force.

To determine the mechanical properties, tensile test in universal testing machine was carried out. Surface roughness 3D parameters were measured by using Taylor Hobson Subtronic 3+ instrument. Table 1 presents the mechanical properties and selected spatial parameters of the sheets.

Friction tests were realized for all combinations of grade of brass, roughness of rolls and clamping forces. In this way eighty different data sets for training of the neural network were obtained.

### 3. Artificial Neural Networks

ANNs are tools to build and analyze linear and nonlinear models of complex regression and classification problems. These tools are used particularly when dependence between inputs and outputs are very complicated. The network consists of elements named neurons that are connected together and the processing data is supplied as input. In general, the work of neural networks is based on the parallel processing idea. Each input signal \( x_i \), where \( i = 1, \ldots, n \)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>M63 r</th>
<th>M63 z4</th>
<th>M63 z6</th>
<th>M90 z4</th>
</tr>
</thead>
<tbody>
<tr>
<td>C, MPa</td>
<td>581</td>
<td>570.3</td>
<td>615</td>
<td>400</td>
</tr>
<tr>
<td>n</td>
<td>0.365</td>
<td>0.143</td>
<td>0.043</td>
<td>0.040</td>
</tr>
<tr>
<td>Sa, µm</td>
<td>0.162</td>
<td>0.151</td>
<td>0.108</td>
<td>0.33</td>
</tr>
<tr>
<td>Sq, µm</td>
<td>0.205</td>
<td>0.196</td>
<td>0.137</td>
<td>0.279</td>
</tr>
<tr>
<td>Sp, µm</td>
<td>1</td>
<td>2.16</td>
<td>1.03</td>
<td>1.16</td>
</tr>
<tr>
<td>Ssk</td>
<td>0.262</td>
<td>0.0371</td>
<td>0.191</td>
<td>0.202</td>
</tr>
<tr>
<td>Sz, µm</td>
<td>1.78</td>
<td>2.24</td>
<td>1.34</td>
<td>2.14</td>
</tr>
<tr>
<td>Sv, µm</td>
<td>1.05</td>
<td>1.03</td>
<td>0.623</td>
<td>0.988</td>
</tr>
<tr>
<td>Sku, µm</td>
<td>3.37</td>
<td>4.72</td>
<td>3.74</td>
<td>4.24</td>
</tr>
<tr>
<td>Str, mm</td>
<td>0.287</td>
<td>0.153</td>
<td>0.109</td>
<td>0.175</td>
</tr>
<tr>
<td>SHTp, µm</td>
<td>0.337</td>
<td>0.309</td>
<td>0.223</td>
<td>0.31</td>
</tr>
<tr>
<td>Smmr, mm³/mm²</td>
<td>0.00105</td>
<td>0.00103</td>
<td>0.000623</td>
<td>0.000988</td>
</tr>
<tr>
<td>Sdq, µm/µm</td>
<td>0.0201</td>
<td>0.0183</td>
<td>0.0136</td>
<td>0.0184</td>
</tr>
<tr>
<td>Std, °</td>
<td>1.5</td>
<td>1</td>
<td>1</td>
<td>1.5</td>
</tr>
<tr>
<td>Sfd</td>
<td>2.5</td>
<td>2.32</td>
<td>2.4</td>
<td>2.41</td>
</tr>
<tr>
<td>Sdr, %</td>
<td>0.201</td>
<td>0.167</td>
<td>0.009</td>
<td>0.0169</td>
</tr>
<tr>
<td>Sbi</td>
<td>0.316</td>
<td>0.107</td>
<td>0.172</td>
<td>0.147</td>
</tr>
<tr>
<td>Sal, mm</td>
<td>0.074</td>
<td>0.149</td>
<td>0.088</td>
<td>0.123</td>
</tr>
</tbody>
</table>

is loaded to the neuron by weighted connections \( w_i \) (Fig. 2). Every neuron has a threshold value specified as its activation level. Sum of input signal values \( x_n \) multiplied by weight factors is calculated at \( k^{th} \) neuron. This value is then increased by external signal value which is referred to as a bias term \( \Theta_k \). Calculated in this way \( e \) value is the neuron activation value which is converted by established activation function \( f_k(e) \) of \( k^{th} \) neuron. The value determined by activation function is output neuron value and specifies the nonlinear relationship between resultant input signal and output signal \( y \) neurons. In some cases, the output of a unit can be a nonlinear function or a stochastic function of the total input of the unit.

![Fig. 2 Structure of nonlinear neuron k.](image-url)
Among nonlinear functions, a hyperbolic tangent function can be used with yielding output values in the range \([-1; +1]\). Mathematically, a formula describing an output value of neuron can be written as

\[
y = f\left(\sum_{j=1}^{g} w_j x_j\right)
\]

where:

\[
f(x) = 1 \text{ when } x \geq 0; \\
f(x) = 0 \text{ when } x < 0; \\
y = \text{output of the neuron;} \\
w_j = \text{the weight of the connections that feeds into neuron } j; \\
x_i = \text{input signal of the neuron } i; \\
\nu = \text{the threshold of the neuron.}
\]

A single layer neural network is characterized by the simplest structure. However, multilayer networks named multilayer perceptrons (MLP) are mostly utilized. A MLP with a suitable number of hidden layers and neurons is theoretically sufficient to approximate any nonlinear function \([7]\). In order to calculate the output value of neuron of MLP network the hyperbolic tangent function (Eq. (3)) is applied:

\[
f(a) = \tanh(x) = \frac{e^a - e^{-a}}{e^a + e^{-a}}
\]

To determine weighted sum and threshold activation value of separated neurons, it is necessary to prepare the training data set consisting of input signals and the corresponding values of output signals. In this study, the following input sets of variables were assigned as input signals:

1. Strength coefficient \(C\) [MPa] and strain hardening index \(n\);
2. Clamping force of rolls \(F_C\);
3. Roughness average parameter \(R_a\) of rolls surface [\(\mu m\)];
4. Surface roughness 3D parameters of the sheets:
   - Amplitude: \(S_a\) [\(\mu m\)], \(S_q\) [\(\mu m\)], \(S_p\) [\(\mu m\)], \(S_v\) [\(\mu m\)], \(S_t\) [\(\mu m\)], \(S_k\), \(S_u\), \(S_z\) [\(\mu m\)];
   - Superficial and volumetric: \(SHTp\) [\(\mu m\)], \(Smmr\) [\(mm^3/mm^2\)], \(Smmv\) [\(mm^3/mm^2\)];
   - Spatial: \(Sds\) [number of vertex/\(mm^2\)], \(Srt\), \(Sal\) [\(mm\)], \(Std\) [°], \(Sfd\);
5. Hybrid: \(Sdq\) [\(\mu m/\mu m\)], \(Ssc\) [1/\(\mu m\)], \(Sdr\) [%], and
6. Functional: \(Sbi\), \(Sci\), \(Svi\).

One of the main tasks necessary to build optimal model of neural network is selection of sufficient input variables that essentially influence the output variable value. Too large number of variables may cause noisy data whereas not taking into account even a single variable that essentially influences the output variable may lead to wrong results. Further, adding more input network results in excessive expansion of the network architecture and at the same time the value of training data is increased. In turn, omission of essential variables in input can cause decreasing quality of the network. This indicates that there are no universal criteria to select architecture of ANN. Selection of variables that essentially influence the friction coefficient value is difficult because of complex interactions of many factors particularly surface parameters which are additionally correlated with each other.

Application of genetic algorithms however allows selecting input variables without the necessity of having knowledge about physical interdependences between individual input variables and output variable.

4. Formulating Optimization Model in GA

Genetic algorithms are based on natural selection mechanisms as well as heredity; and operated on population of individuals that are potential solutions of the problem. Analogous to natural conditions individuals are subjected to reproduction. Mechanisms of natural selection depend on survival of the most adapted individuals in a specified environment. In other words, only the strongest individuals survive and are able to transmit genetic information to their offsprings. A data carrier about individual characteristics of the individuals is the chromosome, which is the gene set with specified size. Most often binary representation of chromosomes in computer calculation is used. The genetic encoding of a real or artificial organism is contained within their chromosomes. Each chromosome consists of a large
number of genes, each uniquely located on the chromosome. Each gene in turn is composed of several alleles that are encoded as either zero or one and represented by a single computer bit. A suitable representation of potential results should be able to be decoded in order to find solution for the input data structure. Set of many chromosomes is called population which is subjected to undergoing continuous changes and depends on moment \( t \) expressed by

\[
P(t) = \{v_1(t), v_2(t), \ldots, v_n(t)\} \tag{4}
\]

where:

- \( n \) — number of chromosomes;
- \( v \) — single chromosome.

Further fundamentals and details on the evolution process of chromosomes and the mechanisms of genetic operators is available in the open literature [8-11]. One of the important genetic processes in terms of engineering applications is reproduction whose task is to ensure that output of optimization procedure from local maxima of the fitness function by variability of chromosomes. The objective of the GA is thus finding solutions for which the value of the fitness function reaches maximum.

Several parameters such as size of initial population, crossover and mutation coefficients influence operation of GAs. The initial population of the GA in this study has 200 individuals with crossover coefficient \( p_c = 0.5 \), mutation coefficient \( p_m = 0.1 \). In addition, different values of coefficient of unit penalty (Table 2) are selected. The unit penalty coefficient is multiplied by a number chosen as a mask for each of the input variables, and then added to the value of validation error. For optimization of a number of input variables, classical Holland’s [9] genetic algorithm was used. The evaluation of the population was carried out with the help of a mechanism that, for each solution set, loads the best individuals so far found. This is done in accordance with the selection probability to new population with the help of a roulette wheel selection. The task of the genetic algorithm is thus to check the quality of the network that realizes the generalized regression for a given set of input variables to the NN resulting from the reproduction mechanism of initial population. The collection of the best sets of input data is the result of GA rules (Table 2).

As shown in Fig. 3, increasing the value of unit penalty causes the reduction of the number of input variables. With a large number of input variables determined by a small value of unit penalty appears a high value of a genetic algorithm error, which next decreases until it reaches local minimum for unit penalty value of 0.001. The error values increase with the value of the unit penalty. The high error value with a great number of variables can be explained by noise of variables which can be in certain range of values correlated with each other. Then a high value of unit penalty causes, from the viewpoint of the quality of algorithms, the number of variables more important [12]. Local increase of the error value with the value of unit penalty can be explained by the fact that removing two variables makes the correlation with other variables dominating. For further analysis a set of 12 input variables characterized by the smallest error value was chosen.

### Table 2  Influence of unit penalty value on the choice of input variables by genetic algorithm.

<table>
<thead>
<tr>
<th>Unit penalty</th>
<th>Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0001</td>
<td>+ + + +</td>
</tr>
<tr>
<td>0.0002</td>
<td>+ + + +</td>
</tr>
<tr>
<td>0.0005</td>
<td>+ + + +</td>
</tr>
<tr>
<td>0.001</td>
<td>+ + + +</td>
</tr>
<tr>
<td>0.002</td>
<td>+ + + +</td>
</tr>
<tr>
<td>0.004</td>
<td>+ + + +</td>
</tr>
<tr>
<td>IPS</td>
<td>+ + + +</td>
</tr>
</tbody>
</table>
The process of constructing the network on the basis of information contained in a chromosome and following the learning of a received model must be done each time by determining the quality of the chromosome.

During the operation of the GA optimization, the number of evaluated neuron networks is the product of chromosome numbers in the population and the number of considered generation. In the early phase of the genetic algorithm, set of input variables were investigated using a series of analyses of different ANN architecture in Statistica Neural Networks. The objective of these analyses was to find the network architecture that ensures the smallest value of standard deviation ratio in connection with high value of Pearson’s correlation coefficient $R$ [13].

Results of these analyses were compared with results of network model based on input variables determined by Intelligent Problem Solver (IPS) built-in Statistica Neural Networks as given in Table 2. “The best” set of neural network architecture was determined on the basis of the loading data of the IPS. In Statistica, there is no particular information about mathematical nature of input variable selection by using IPS.

Among all experimental sets of input data that correspond with the output signal, 20% were separated and assigned as test set (Ts). Data vectors from a test set did not participate in the training process and served for ANN prognostic evaluation purpose only. From the remaining set of experimental data belonging to training set (Tr), 10% was separated and assigned as validation set (V). Data from this group were used for independent check of back propagation (BP) training algorithm. The learning rate was equal to 0.1 [14]. One of the important parameters in GA operations is the stop criteria. This, among others, helps to prevent overlearning. In this case, the learning process was stopped when the value of verification root mean square error (Eq. 5) for validation set was stopped dropping [12].

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (z_i - y_i)^2}$$

where:

- $N$—number of vectors of training set;
- $y_i$—signal of output signal for $i$th standard;
- $z_i$—expected signal of output neuron for $i$th standard.

Table 3 shows the regression statistics of “the best” neural networks for input variables determined by genetic algorithm (MLP 12:12-12-1:1, MLP 12:12-14-1:1), Intelligent Problem Solver (MLP 15:15-10-1:1) and using entire variable set without preprocessing (MLP 25:25-12-1:1). The model with the lowest values of standard deviation ratio in connection with highest value of Pearson-R correlation.

<table>
<thead>
<tr>
<th>Set</th>
<th>T_r</th>
<th>V</th>
<th>T_s</th>
<th>T_r</th>
<th>V</th>
<th>T_s</th>
<th>T_r</th>
<th>V</th>
<th>T_s</th>
<th>T_r</th>
<th>V</th>
<th>T_s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error S. D.</td>
<td>0.005</td>
<td>0.005</td>
<td>0.008</td>
<td>0.006</td>
<td>0.007</td>
<td>0.011</td>
<td>0.005</td>
<td>0.010</td>
<td>0.009</td>
<td>0.008</td>
<td>0.900</td>
<td>0.013</td>
</tr>
<tr>
<td>Abs error mean</td>
<td>0.0009</td>
<td>0.002</td>
<td>0.0001</td>
<td>0.001</td>
<td>0.008</td>
<td>0.009</td>
<td>0.003</td>
<td>0.004</td>
<td>0.002</td>
<td>0.009</td>
<td>0.012</td>
<td>0.005</td>
</tr>
<tr>
<td>RMS error</td>
<td>0.005</td>
<td>0.006</td>
<td>0.009</td>
<td>0.004</td>
<td>0.005</td>
<td>0.010</td>
<td>0.007</td>
<td>0.009</td>
<td>0.012</td>
<td>0.008</td>
<td>0.013</td>
<td>0.011</td>
</tr>
<tr>
<td>S. D. ratio</td>
<td>0.149</td>
<td>0.253</td>
<td>0.250</td>
<td>0.153</td>
<td>0.248</td>
<td>0.170</td>
<td>0.196</td>
<td>0.298</td>
<td>0.171</td>
<td>0.275</td>
<td>0.351</td>
<td>0.285</td>
</tr>
<tr>
<td>Correlation R</td>
<td>0.989</td>
<td>0.968</td>
<td>0.973</td>
<td>0.997</td>
<td>0.976</td>
<td>0.996</td>
<td>0.980</td>
<td>0.992</td>
<td>0.993</td>
<td>0.923</td>
<td>0.918</td>
<td>0.937</td>
</tr>
</tbody>
</table>
is network MLP 12:12-14-1:1. The regression statistics of this network is slightly better when compared with network MLP 12:12-12-1:1 and considerably better than MLP 15:15-10-1:1, taking into account all analyzed input variables.

One of the methods to determine the importance of the influence of particular input variables on the value of the selected set of data is sensitivity analysis (Table 4), which also can be used to choose input variables. The criterion of sensitivity analysis is the value of the network error after removing this variable. Higher importance of the variable determines higher value of the network error. In the case of the training set, the clamping force $F_C$, the roughness average parameter $R_a$ of rolls surface and the maximum peak height $S_p$ have the highest influence on the value of the friction value.

Analyzing the networks with genetic selection of input variables, it can be noticed that increasing the number of neurons in the hidden layer causes the error value on the output network for each increase in the data set. Less number of neurons in the hidden layer should influence the network’s ability to eliminate the noises coming from input data thus gives greater ability of generalization, which reflects the quality of generalization of acquired knowledge in the learning process to cases from the learning set.

As depicted in Fig. 4, results of this research show excellent agreement between experimental data and outcomes of neuron models for all ranges of input variables used for the training set. This will allow reduction or elimination of expensive and time-consuming experimental tests in order to determine friction coefficient value. The application of the ANN also allows eliminating the search for complicated dependence between parameters influencing the friction and the friction coefficient value.

5. Conclusions

An approach to integrate genetic algorithms in the working process of neural networks to calculate the friction coefficient in metal forming is demonstrated in this article. This method allows avoiding the time-consuming testing of neuron models with different architecture in order to find the optimum network for specific task. Regression statistics of neuron model based on the genetic selection of input variables and using Intelligent Problem Solver are considerably better than models without preprocessing of input data. Proper selection of input variables is found crucial for the work of neural networks. Sensitivity analysis through study of the value of the output error is thus used to determine the importance of particular input variables. The comparison of the network results for selected input variables with experimental ones show that the optimization model gives results that agree with the experiment. Continuing works in this research involve simulation
of the network for different options of friction coefficient and radius of rolls of the tool.

Acknowledgment

The financial support for this research project was provided by Island, Liechtenstein and Norway and was co-financed by European Economic Area and Norwegian Financial Mechanism under the Scholarship and Training Fund. The authors would like to acknowledge this financial support.

References