Neurocomputing ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Design of experiments and focused grid search for neural network parameter optimization

F.J. Pontes [a], G.F. Amorim [b,c,*], P.P. Balestrassi [b], A.P. Paiva [b], J.R. Ferreira [d]

[a] UNESP (Universidade Estadual Paulista)-Avenida Ariberto Pereira da Cunha, no. 333, Pedregulho, Guaratinguetá, SP CEP: 12516-410, Brazil
[b] Institute of Industrial Engineering and Management, UNIFEI (Universidade Federal de Itajubá)-Rua Dr. Pereira Cabral, no. 1303, Pinheirinho, Itajubá, MG CEP: 37500-903, Brazil
[c] University of Tennessee – Knoxville Through the Science Without Borders Program, United States
[d] Institute of Mechanical Engineering, UNIFEI (Universidade Federal de Itajubá)-Rua Dr. Pereira Cabral, no. 1303, Pinheirinho, Itajubá, MG CEP: 37500-903, Brazil

ARTICLE INFO

Article history:
Received 15 January 2015
Received in revised form
16 November 2015
Accepted 28 December 2015

Keywords:
Design of Experiment
Focused Grid Search
Artificial Neural Network
Machining
Tuning

ABSTRACT

The present work offers some contributions to the area of surface roughness modeling by Artificial Neural Networks (ANNs) in machining processes. It proposes a method for an optimized project of a Multi-Layer Perceptron (MLP) network architecture applied for the prediction of Average Surface Roughness (R_a). The tuning method is expressed in the format of an algorithm employing two techniques from Design of Experiments (DOE) methodology: Full factorials and Evolutionary Operations (EVOP). Datasets retrieved from literature are employed to form training and test data sets for the ANN. The proposed tuning method leads to significant reduction of roughness prediction errors in machining operations in comparison to techniques currently used. It constitutes an effective option for the systematic design models based on ANN for prediction of surface roughness, filling the gap reported in the literature on this subject.

1. Introduction

As newcomers to the use of Artificial Neural Networks (ANN), researchers on the field of manufacturing started to explore ways to apply networks to control or to foresee critical product quality features, and to optimize multiple objective production processes. The growing number of papers published during the past decade testifies this interest.

Machining processes, for example, generates surfaces or parts through removal of material. Production rate, cost, and product quality are conflicting objectives in this kind of process, posing additional challenges to its planning and optimization [1,2]. One feature particularly difficult to control in machined products is the surface roughness, a widely used index of product quality and a technical requirement for machined parts [3]. It affects properties such as fatigue behavior, corrosion resistance, friction, wear, light reflection, heat transmission, lubrication, electrical conductivity and coating [4,5].

The ability to accurately control surface quality can reduce machining costs by lessening the rework activities. It means that this is not just a defying issue, but also an area of research interest. The surface roughness cannot be controlled as accurately [6] because it is influenced by many variables like steel properties, tool material and geometry, vibration of cutting tool, cutting speed, feed, depth of cut, lubricant, and others [7].

Although online roughness control applications are found in literature, a more common approach is the application of ANNs to offline control based on process parameters. Off-line quality techniques are considered an effective approach to improve product quality at a relatively low cost [8]. A survey on practical efforts for network topology optimization reveals a drive towards parameter optimization. Jiménez et al. [9], for example, used Focused Grid Search (FGS) techniques for classification problems.

Despite the enthusiasm of using ANN for roughness control, the results obtained are mixed: in many cases, authors deem networks performance as equal or even worse in comparison to other modeling techniques [3,10,11]. However, a close examination on literature reveals some issues such as basics of neurocomputing being disregarded in many works. A broad review [12] found that in more

* Corresponding author at: University of Tennessee – Knoxville Through the Science Without Borders Program, United States.
Tel.: +55 35 9130 8293/+1 865 403 0028.
E-mail addresses: fabriciojosepontes@uol.com.br (F.J. Pontes), gabi@unifei.edu.br (G.F. Amorim), ppbalestrassi@gmail.com (P.P. Balestrassi), andersonppaiva@unifei.edu.br (A.P. Paiva), jorofe@unifei.edu.br (J.R. Ferreira).

http://dx.doi.org/10.1016/j.neucom.2015.12.061
0925-2312/

than 40% of papers dealing with surface roughness controlled by ANNs, networks are designed by trial and error and in less than 10% any effort to optimize network topologies could be positively identified. This paper proposes then a method for tuning optimized networks of Multi-Layer Perceptron (MLP) architecture applied here for Average Surface Roughness (R_a) control in machining processes. By combining two distinct techniques, Design of Experiments (DOE) and Focused Grid Search (FGS), this work manages to establish optimization decisions based on solid statistical criteria. It is innovative because of the sequential use of DOE arrangements for finding optimal model parameters. DOE is an applied statistical methodology whose use allows to plan experiments capable of generating appropriate data for an efficient statistical analysis, resulting in valid and objective conclusions [13].

The strategy adopted consists in the use of DOE arrangements to search for network configurations that benefits output control. This method addresses problems found with previous optimization attempts: (1) It imposes no restriction on the outer search limits; (2) It avoids the use of large intervals between levels of design factors adopted in experimental planning; (3) It addresses the simultaneous optimization of the selected design parameters; (4) It takes into consideration the effects of interaction among design factor levels; and (5) As an algorithm, it proposes a systematic design method for ANN practical use. The last one is pointed as a limiter and as a disadvantage in many works [14–17].

This paper is then organized as follows. Section 2 briefly reviews concepts of machining, surface roughness and the use of DOE for ANN optimization. Section 3 explains the work's reorientation toward Evolutionary Operations (EVOP). Section 4 presents the optimization method algorithm for ANN tuning in details. Section 5 shows the experimental strategy, and Section 6 approaches the two works selected for comparison. Section 7 shows and compares the results of the optimization method to the results of the works dataset were extracted for, and to the results of a software package intended to optimize ANN architectures. Conclusions and suggestions for further research are then presented in Section 8.

2. Background and literature review

Machining is a process that generates surfaces through removal of material, conferring form and dimension to a part. Turning is the most common machining operation [18], being characterized by simultaneous and continuous movement of part and tool. Turning is controlled by its movements, which are: feed, depth of cut and cutting speed. One of the main quality features resulting from machining process is the surface roughness, which can define functional behaviors of a part such as fatigue life, wear patterns, lubricant retention, or resistance to corrosion [10,19,20]. It is linked to machine tool errors, workpiece deformation, vibration, workpiece material inhomogenities, cutting edges shape and condition, chip formation, cutting parameters, and physicochemical mechanisms acting on workpiece grain and lattice structures [7]. As pointed out, it plays an important role in determining the quality of a machined product [21,22].

Roughness is then an indicator of process performance and must be controlled within proper limits for particular machining operations [23]. The process-dependent nature of roughness formation, along with many uncontrollable factors, makes difficult to keep it between desirable limits, i.e. to control it [7,19]. Operators use their own experience and machining guidelines in order to achieve the best possible surface finish [24].

Among the parameters to measure surface roughness, the most commonly used is Roughness Average (R_a). It is the arithmetic average of the absolute value of the heights of roughness irregularities from the mean value measured [25]. For discrete measurement, R_a can be defined as in Eq. (1) [26].

$$R_a = \frac{1}{n}\sum_{i=1}^{n} |y_i| \tag{1}$$

the roughness average (R_a) is typically measured in micrometers (μm), n is the number of samples in a given length, and $|y_i|$ stands for the absolute measured values of the peak and valley in relation to the center line average. According to international standards [27], machining processes can achieve roughness values ranging from 0.025 μm to 50 μm.

Efforts to model roughness involve analytical, experimental and AI techniques [28]. Theoretical and empirical models, however, suffer from a number of problems. Theoretical models take no account of imperfections in the real process, such as tool vibration or chip adhesion [20]. Empirical models have their application limited to very specific operational conditions. The experience in both cases is then poor, as stated in many works [29,30].

The use of ANNs in machining processes has been encouraged in a considerable number of papers. Authors sustain that ANNs are a good alternative to conventional empirical modeling based on linear regressions for surface roughness modeling [10], also maintain that neural networks are able to capture the turning characteristic of non-linearity [24]. In hard turning operations, some authors approaches the difficulty of generating explicit analytical models with the complex relationship among the parameters involved and, according to them, ANN pose a suitable and practical option for modeling [31].

There is no consensus, however, on the experience with ANN for roughness modeling. Some authors point to the lack of systematic design methods as a disadvantage [14]. Others claim that finding a good ANN architecture requires several modeling attempts, making it a time consuming activity [15,16]. Researchers also testify the need of large amounts of data for training and validation as restrictions to the practical application of ANN in machining processes [32].

The most popular approaches for ANN design are empirical search optimization (trial and error), pruning and constructive approach [33]. Trial and error is common practice in most works on the field of intelligent systems [17]. In more than 40% of the papers using ANN, network topologies are explicitly defined by trial and error; Clear optimization efforts are detected in less than 10% [12]. The application of statistics for network topology optimization is not widespread in literature. The few examples found shows that the full potential of this subject is not uncovered yet. This factor could contribute to such a mixed view of ANNs abilities controlling model roughness.

DOE technique is based on the concept of simultaneous variation of factors levels, in order to build forecasting models for relevant outputs [13]. An additional advantage is that DOE principles can be implemented in a well-defined and relatively low number of experiments [30]. It is one of the most important methodologies for researchers dealing with experiments in practical applications and its tools are incorporated in many statistical software packages that ease calculation and interpretation of results [34].

A DOE application for ANN optimization in machining process can be found in [31]. The authors employed a DOE arrangement called Taguchi to select the inputs for roughness prediction in CNC face milling process. In [35], the development of roughness prediction model for polymer blends machining using MLP trained by back-propagation is also proposed employing Taguchi. Besides roughness prediction, some DOE applications for network optimization can be found such as tool wear [36] or thickness prediction [37], robust design [38], assembly sequence planning [39], neuro-fuzzy systems [40], time series forecasting [41], electricity spot simulation [42], among many others.

The use of DOE is not widely spread in simulation as it should be due to reasons such as the lack of knowledge about the methods, the lack of access to DOE research and the misconception that DOE is applicable only to real-world experimentation rather than for simulation and numerical experimentation [43]. DOE is also a useful and necessary part of complex simulation analysis. The DOE-simulation concept can be extended to an ANN design when the process is defined as the network parameters initialization.

3. Work reorientation using Focused EVOP optimization

The initial goal of this work was to conduct an exploratory experiment to identify the most influential factors for the performance of MLP networks and then apply the Response Surface Methodology (RSM) to search for optimal network topologies. However, unsatisfactory results were obtained for the Steepest Descent and Central Composite Design phases of RSM. Those results testify to the complexity of MLP error surfaces, already known in literature [43], and disqualified RSM as an optimization approach.

The work was then re-oriented towards the use of Evolutionary Operations (EVOP), a DOE technique, in conjunction with Focused Grid Search (FGS). EVOP was intended from industrial processes optimization [44]. The basic concept is the successive application of DOE arrangements, and the displacement of the operational point to the optimal region, based on the results of statistical analysis. EVOP analysis makes no previous assumption on the nature of the surface being modeled, what made it suitable for the optimization.

Fig. 1 shows the resulting algorithm as a flow chart that summarizes the activities carried out for optimizing the ANN parameters, i.e. the ANN architecture. The proposed steps for the implementation of a Multi-phase EVOP are detailed in Section 4. The five quantitative factors subjected to optimization (H1, H2, E1, LR, and E2) are summarized in Table 1 and the seven qualitative factors set out before the optimization (P1, P2, OF, PU, PI, W1, and W2) are in Table 2.

4. Details on the optimization method

As shown in Fig. 1, the proposed method initiates by establishing the best settings for the qualitative design factors. It is based on the results of an exploratory experiment employing a Taguchi array [45] to identify levels of those factors that would

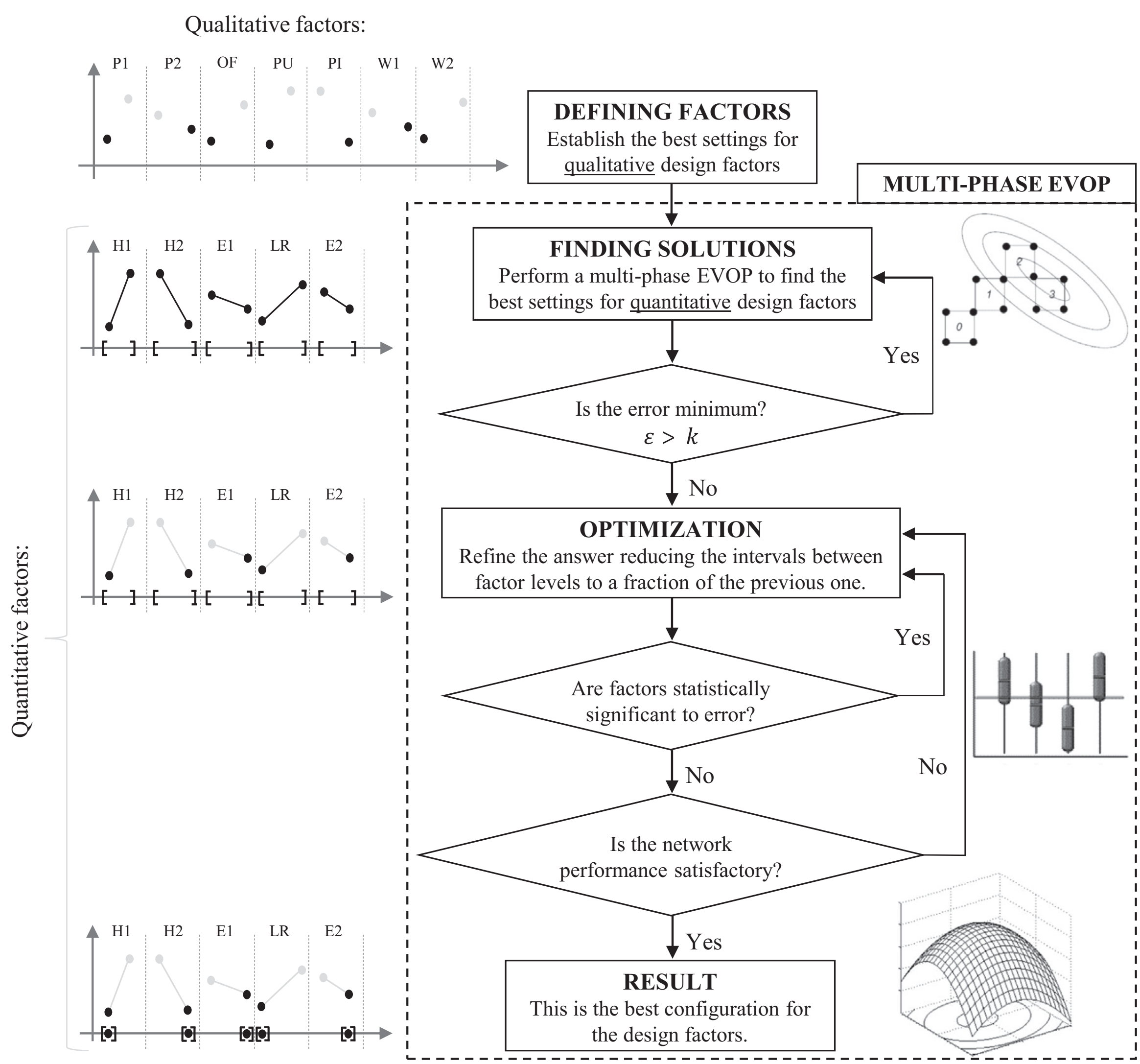

Fig. 1. Flowchart of the proposed algorithm for the optimal ANN parametrization.

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

Table 1
Quantitative design factors subject to optimization.

Factor code	Factor description
H1	Number of neurons hidden layer 1
H2	Number of neurons hidden layer 2
E1	Number of training epochs phase 1
LR	Learning rate training phase 1
E2	Number of training epochs training phase 2

Table 2
Qualitative design factors set out.

Factor code	Factor description	Option set
P1	Training algorithm phase 1	Back-Propagation algorithm
P2	Training algorithm phase 2	Conjugate Gradient algorithm
OF	Output Activation Function	Logistic Sigmoid
PU	Pruning of output neurons	Pruning of weights smaller than 0.05
PI	Pruning of inputs	No pruning
W1	Weights decay regularization phase 1	No regularization
W2	Weights decay regularization phase 2	No regularization

benefit the predictive ability of the network. A mixed L16 Taguchi array was employed based on [41].

Another essential step is the definition and interpretation of model outputs [46]. All possible performances of a network form a distribution of predictions due to random effects. The design goals then change from find networks 'with the smallest error averages' to 'capable of achieving minimal errors', or even 'to maximize the probability of finding networks capable of achieving minimal errors'. This subject is studied at the statistical research area of Extreme Distributions, in order to model the probability of extreme events.

In order to have a statistical estimate of the network performance, it is necessary to collect a significant sample of predictions. It is also important to express ANN model accuracy in statistical terms, instead of absolute error values. Adopting the Extreme Distribution concept, it is not the mean performance of a network topology that counts, but the probability of a network model to achieve an error of *X*% or less in the task proposed, when independently trained and tested over a given dataset for a number of times.

To meet with such statement, the process output for DOE was defined by the independent repetition of the sequence "training-selection-validation" of each network topology for 100 times. The variable to be minimized through the use of DOE was defined as the smallest possible *Mean Absolute Error* (MAE) value for prediction (in percentage) for which there is a 0.1 cumulative probability in the population of network predictions. In practice, the output variable was calculated as the average absolute errors of the bottom decile of the results obtained from each replica and referenced by MAE-DI%.

Once qualitative factors and output goals were defined, the optimization process of quantitative factors is initiated based in a multi-phase EVOP. It consists in an iterative algorithm beginning with broad differences between factor levels until it comes to a point where it shows no improvement. Then, a new phase is initiated, focusing on the best configuration of the previous phase and reducing the intervals to a fraction of the previous one. The phases continue up to a point where no factor is deemed statistically significant to network performance, or up to a point that the designer deems to be satisfactory. The next section shows the algorithm main routine to be implemented concerning the quantitative factors for ANN tuning.

4.1. Multi-phase EVOP main routine

The initial step is the initialization of variables. The outer loop corresponds to phases. Throughout each phase, intervals between factor levels remain constant. The inner loop corresponds to cycles within each phase, during which the center of experimental array moves toward a network configurations with minimal error.

On each cycle, an experimental array is assembled, performed and analyzed. If any main factor or interaction is considered to be statistically significant, the configuration is stored. It is compared to the best performance so far. In case of improvement, the network configuration of the cycle replaces the configuration previously pointed as the best one. The center point of the array moves toward the newly optimized network configuration and another cycle begins, with the experimental setup built around the new center point.

Cycles are executed up to the point no statistical improvement is detected or until the center point established for the next cycle corresponds to a point already employed as center point in the same phase, thus marking the phase end. Intervals between factor levels are then reduced, history of central points for the current phase is reset and another phase begins, with reduced intervals between the factor levels.

Phases are conducted up to the iteration number defined by user, or in case the statistic test for analysis of variance (ANOVA) performed in a given cycle indicates no factor is statistically significant for the reduction of network error. The best network configuration at the end is considered to be the best neural model for roughness control and its constructive and statistical parameters are stored for model characterization.

The main routine of the proposed algorithm for quantitative factors optimization is described as a pseudocode in Fig. 2. This pseudocode was implemented at the software Statistica in order to achieve this works results, but could be transcribed in other languages such as Matlab, R, C, C++, among others.

4.2. Establishment of initial values

At this work, the initial center point and initial intervals between experimental factor levels were defined as shown in Table 3. These values were established during exploratory experiments, based on fast convergence, amplitude of parameter search coverage and experimental resolution. After each phase, the intervals were reduced by fractions of 0.5.

4.3. Establishment of next center point

After the end of a cycle, the new center point shall be determined as specified on the pseudocode shown in Fig. 3. This algorithm states that levels of a factor should be reduced or increased as specified by the results of an ANOVA test. After the end of a phase, the intervals between factors levels are reduced and the cycle continues until the point no statistical improvement is detected.

5. Experimental strategy

Several planned experiments provide the factors levels and thus the 'direction' to the following one. The large number of independent repetitions was made possible due to the use of computational routines to conduct experimental cycles and to collect data. In order to reinforce research quality,

```
Begin
Initialize DOE Center Point;
Initialize Maximum Phase Number;
Best Topology Current Cycle, Best Topology Current Phase, Best Topology Overall ← DOE Center Point;
Best Performance Overall ← ∞;
For Phase Index ← 1 up to Maximum Phase Number repeat
    End Phase Flag ←FALSE;
    Set of DOE Center Points ← NULL;
    Best Performance Previous Cycle ← ∞;
    Best Performance Current Cycle ← ∞;
    Best Performance Current Phase ← ∞;
    Set Intervals between levels of factors;
    Cycle Index ← 1;
    While End Phase Flag = FALSE repeat
        Store DOE Center Point into Set of DOE Center Points;
        Set levels of factors on Experimental Array;
        Execute experimental Array;
        Format validation tests' results;
        Analyze validation tests' results;
        If at least one design factor is significant for network error reduction do
            Establish Best Topology Current Cycle;
            Best Performance Current Cycle ← performance of Best Topology Current Cycle;
            If Best Performance Current Cycle < Best Performance Previous Cycle do
                Best Topology Current Phase ← Best Topology Current Cycle;
                Best Performance Current Phase ← Best Performance Current Cycle;
                Establish new DOE Center Point;
                Compare new DOE Center Point to Set of DOE Center Points;
                If DOE Center Point ⊄ Set of DOE Center Points do
                    Best Performance Previous Cycle ← Best Performance Current Cycle;
                else
                    End Phase Flag ← TRUE;
                end If "If DOE Center Point (...)";
            else
                End Phase Flag ← TRUE;
            end If "If Best Performance Current Cycle(...)";
        else
            End Algorithm.
        End If "If at least one design factor (...)";
        Cycle Index ← Cycle Index + 1;
    End While "While End Phase Flag";
    If Best Performance Current Phase < Best Performance Overall do
        Best Topology Overall ← Best Topology Current Phase;
        Best Performance Overall ← Best Performance Current Phase;
    End If "If Best Performance Current Phase (...)";
    Phase Index ← Phase Index + 1;
End For "For Phase Index (...)";
End Algorithm.
```

Fig. 2. Pseudocode of the proposed algorithm main routine.

recommendations set from fields of neurocomputing [47,48] and DOE [13,49] were carefully considered. also, some recommendations for statistical validation were observed [50].

The experimental plan employed for Focused EVOP is a full factorial composed of five factors in two levels, with a central point. The DOE was executed in two replicates for each treatment and for the central point. Each replicate was the independent repetition for 100 times of the sequence "training-selection-testing" for each network configuration (treatment). Factor levels were indicated by the convention -1, 0 and $+1$, which represent inferior, central and superior, respectively.

Experiments were conducted for this work using the software Statistica®. A system of scripts and batch files was developed allowing automated and independent repetition of tests on each

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

network topology. Scripts were also used to perform complete experimental arrangements, greatly increasing productivity and thereby enabling timely coverage of the vast domain of the independent variables investigated during the project.

Statistical analysis was performed using the software Minitab®. After each cycle, results were formatted and the proper analysis carried out. The ANOVA test results contain the relative weight of each factor (whether main of interaction) on network performance. The establishment of all input factors at levels that entails the steepest minimization error indicates the optimal network (and thus, the array center) for the next cycle.

Special care was taken in the analysis of residuals. Those must be normal and uncorrelated [13,46]. Residuals of each experiment were analyzed through the application of an Anderson-Darling test, at a level of significance of 0.05. For experiments whose residuals did not follow the normal distribution, outliers were removed and the normality test repeated. Only after the normality of residuals had been achieved, the statistical test ANOVA was conducted. Effects were considered to be significant if their resulting *p*-values measured below the level of significance adopted, which was 0.05.

6. Selected works for comparing the surface roughness prediction methods

The proposed method for ANN tuning was validated through its application to datasets collected from selected papers on the subject of roughness optimization. Statistical comparisons were made between the results of the best network topology identified by the proposed method, results of the original studies, and results obtained by the best network topology identified by a specialist computer package available at the software Statistica®.

Table 3
Initial values adopted for the quantitative factors to be optimized.

Factor	Initial value
Number of neurons in hidden layer 1 (H1)	8
Number of neurons in the hidden layer 2 (H2)	8
Number of epochs training phase 1 (E1)	32
Learning rate training phase 1 (LR)	0.4
Number of epochs training phase 2 (E2)	64

Those papers were chosen because they also aimed to build a neural network applied for roughness prediction and they showed the explicit data they used to do it. In order to make a fair comparison for the results, it was used in this paper the same division of the data set that the authors of the original papers used: the same amount for training, validation and tests.

The first paper selected (Case Study no. 1) was the work of Sharma et al. [20], whose focus was the prediction of parameters such as cutting forces and surface roughness in hard turning process. The material specimens were composed of Adamite, with measured hardness of 467 HV.

Roughness values measured during the experiments were between 0.84 μm and 7.49 μm. The data collected was employed to build an ANN model for surface roughness prediction, following an author defined heuristic. The ANN input factors adopted in that study are shown in Table 4. The output variable is the *Average Roughness* (R_a) measured in micrometers (μm). The training set was composed of 34 examples, being 30 for training and 4 cases used for validation, randomly swapped on each training epoch. The test set comprised 17 examples.

The second paper employed for validation (Case Study no. 2) is the work of Sarkar et al. [51], which approaches a process of Electric Discharge Machining (EDM) of an alloy of aluminum and titanium. Hardness of the specimens measured is 148 HV. The goal was to propose a new strategy for process optimization, identifying an operation point of minimum roughness.

A neural network trained by back-propagation algorithm was developed to model the machining process and then employed to generate multidimensional models for identification of optimal process parameters. The input factors used to predict surface roughness are listed in Table 5. The output variable of the model is also the *Average Roughness* (R_a) measured in micrometers (μm). Roughness values measured during the experiments were between 2.31 μm and 3.16 μm. The training set contains 18 examples (15 examples for training and 3 for validation) and the test set contains 6 examples.

7. Proposed method application: results and comparisons to previous works

This section presents the results obtained applying the method proposed in this work for the two selected data set. The resulting

```
Begin
For each factor do
    If Analysis of variance indicates that factor should be decreased do
        If Cycle Index = 1 do
            Factor centerpoint level ← (Factor centerpoint level - Interval between factor levels)
        else
            If Factor centerpoint level - Interval between factor levels ≥ lowest possible value for
            factor do
                Factor centerpoint level ← (Factor centerpoint level - Interval between factor
                levels)
            end If "If Factor centerpoint (...)"
        end If "If Cycle Index (...)"
    else
        If analysis indicates that factor should be increased do
            Factor centerpoint level ← (Factor centerpoint level + Interval between factor levels)
        end If "If analysis (...)"
    end If "Analysis of variance (...)"
end For "For each factor (...)"
End Algorithm
```

Fig. 3. Pseudocode of the proposed algorithm to determine the new center point.

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

ANN performances are then compared to the ANN performances proposed at the original works and to the ANN models identified through IPS, an automated tool for optimizing ANN using the software Statistica®.

It is important to emphasize that the results presented here were obtained exclusively using the test set, i.e. not the data used for training or validation. Results were first analyzed using the software Minitab®: the traditional DOE approach for analyzing the results of the proposed method can be seen in Figs. 4–6.

Fig. 4 illustrates the main effects caused on network performance by changing levels of input factors. This figure shows, for each experimental factor on the x-axis, the encoded values it assumed on the experimental arrangement (-1, 0 or 1). The y-axis displays the average values of the output obtained for the experiments conducted with the array factor at the given level.

Fig. 5 illustrates the two-way interaction effects. Interactions can be understood as amplification or inhibition of the effect of varying a particular factor on the output due to the level assumed by another factor. The ability to detect interactions between factors is one of the strengths of DOE methodology. As for the main effects, the significance of interaction effects is measured by ANOVA. It is possible to visualize graphically the importance of an interaction effect by comparing the relative slope of the lines. The more significant the effect of an interaction the closer lines are to perpendicular.

For experiments involving more than two factors, it is possible to estimate effects of order greater than two. Fig. 6 shows the set of significant effects for the example in question. The Pareto chart indicates main and interaction effects considered statistically significant by ANOVA. Effects crossing the line are considered to be significant, at a significance level of 0.05. It may be noted that some triple or even quadruple interaction effects are significant to the performance of the network. Moreover, according to the principle of hierarchy [13], the main effect of a factor must be taken into account at the analysis if there is any significant interaction effect in which it is involved.

Table 4
Input factors of dataset collected from Sharma et al. [20].

Factor code	Factor description	Units
V_c	Cutting speed	m/min
f	Feed	mm/v
a_p	Depth of cut	mm
AA	Approaching angle	°

Table 5
Input factors of dataset collected from Sarkar et al. [51].

Factor code	Factor description	Units
T_{on}	Pulse on time	μs
T_{off}	Pulse off time	μs
I_P	Peak current	A
WT	Wire tension	g
SV	Servo reference voltage	V
FR	Dielectric flow rate (discharge pressure)	Kg/cm^2

7.1. Results for Case Study no. 1

7.1.1. Initialization and development

Applying the proposed method for tuning a specific ANN for the dataset from Case Study no. 1, a *Maximum Number of Phases* of four was adopted. The *Center Point* and the *Interval Between Factor Levels* were initialized according to the values specified on the algorithm in Fig. 3.

The optimization process required the execution of nine cycles. The first phase ended after the second cycle, when the analysis pointed that the *Center Point* for next cycle was a set of factor levels already used as a *Center Point* in the same phase. The second phase ended after the fifth cycle, for the same reason. Third and fourth phases ended after cycles 7 and 9, respectively, when results pointed to no error reduction. Results obtained are displayed on Table 6.

Table 6 details the factors levels corresponding to centers of factorial arrays executed every cycle. This table informs the combination of factors identified at the end of each cycle, under columns grouped as 'Best configuration obtained in the cycle'.

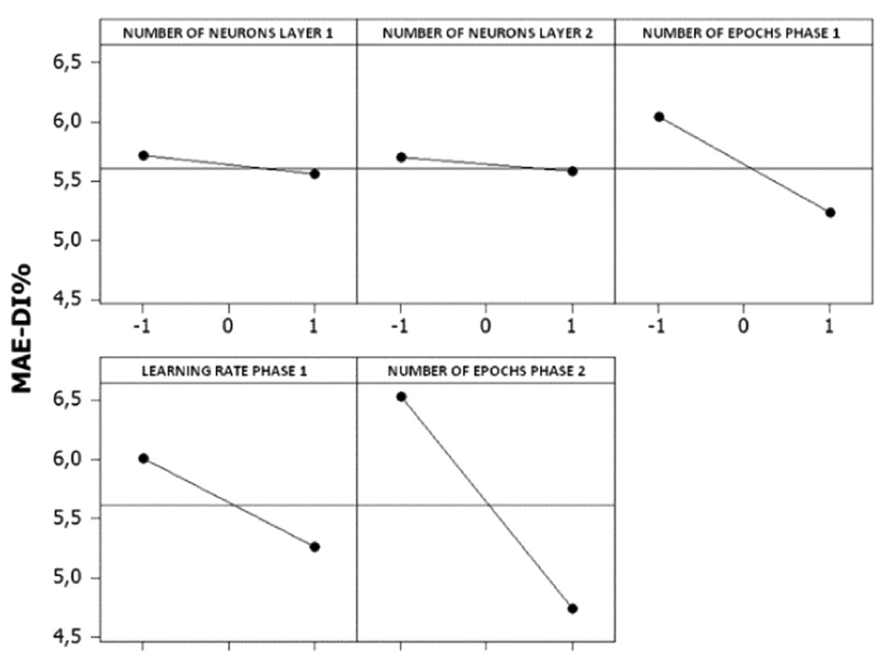

Fig. 4. Main effects plot (first optimization cycle using dataset from [41]).

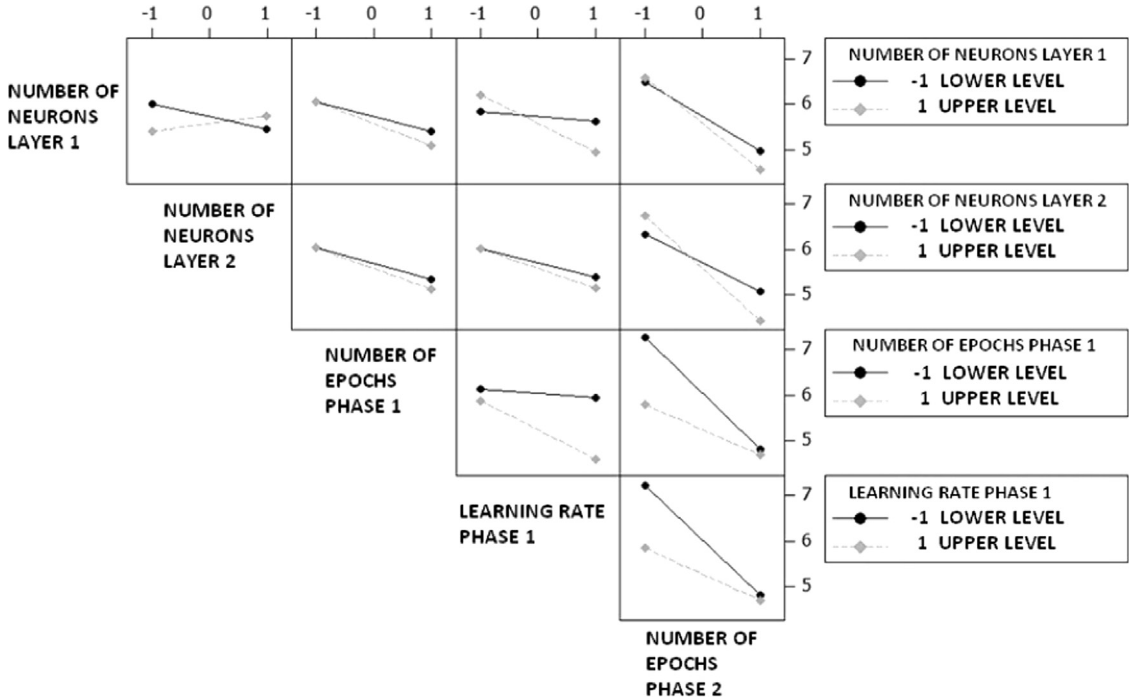

Fig. 5. Two-way interaction effect plot (first optimization cycle using dataset from [41]).

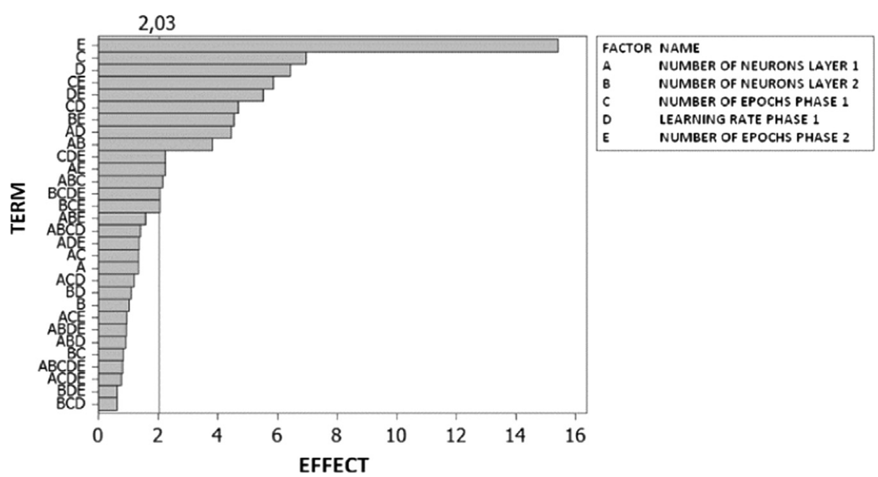

Fig. 6. Pareto chart of standardized effects (first optimization cycle using dataset from [41]).

Prediction performances achieved by that configuration are displayed under column 'Output (MAE-DI%)'.

After the final cycle, the best network topology was the one whose quantitative factors resulting from EVOP are detailed in Table 7 and the qualitative factors resulting from Taguchi are shown on Table 2. This ANN performance is summarized in Table 8. The independent repetition of the sequence "initialization-training-test" for 100 times made it possible to obtain statistically significant data on model performance associated to the ANN topology.

The optimal ANN model could generate Response Surfaces for the roughness process being studied. Figs. 7 and 8 are examples of Response Surfaces generated by the optimal ANN configuration. Fig. 7 displays the response surface of network predictions for a_p (*Depth of cut*) versus AA (*Approaching angle*), keeping V_c (*Cutting speed*) fixed at 90 [m/min] and f (*Feed*) fixed at 0.2[mm/turn]. Fig. 8 displays the response surface of network predictions for a_p (*Depth of cut*) versus f (*Feed*), keeping AA (*Approaching angle*) fixed at 30 [°] and V_c (*Cutting speed*) fixed at 90 [m/min].

7.1.2. Comparing results from the proposed method and the original work [20] used as Case Study no. 1

The performance of the best ANN tuning obtained through the application of the proposed method was compared to that achieved by the network topology adopted in the original work where this data set was used. The error in both cases was compared using a hypothesis test. A bilateral t-test for the null hypothesis of equal average error between the ANNs at a significance level of 0.05 was conducted. The value for the best network identified in this work is equal to 12.797%, while the corresponding value calculated for Case Study no. 1 is 72.108%.

The result was a p-value of 0.000, implying in a rejection of the null hypothesis of equal means. Therefore there is a strong statistical evidence to state that the ANN performance obtained through the proposed method is superior to that obtained at the

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

Table 6
Experimental array centers and best factor settings per cycle – proposed method application to dataset from Case Study no. 1.

		Array center (natural units)					Best configuration of the cycle (coded units)					Output (MAE-DI%)
Phase	Cycle	H1	H2	E1	LR	E2	H1	H2	E1	LR	E2	
1	1	8	8	32	0.400	64	1	−1	−1	1	−1	14.104
	2	16	8	32	0.800	64	−1	−1	−1	−1	−1	13.718
2	3	8	8	32	0.400	64	1	−1	−1	−1	−1	14.805
	4	12	4	16	0.200	32	1	−1	−1	−1	−1	14.198
	5	16	4	16	0.200	32	−1	−1	−1	−1	−1	14.127
3	6	12	4	16	0.200	32	1	−1	−1	−1	−1	13.635
	7	14	2	8	0.100	16	−1	−1	1	−1	−1	13.956
4	8	14	2	8	0.100	16	−1	−1	1	1	−1	12.797
	9	13	1	12	0.150	8	1	−1	−1	−1	−1	13.513

Table 7
Best ANN topology for quantitative factors using dataset from Case Study no. 1.

Design factor	Optimal factor level
Number of neurons hidden layer 1	13
Number of neurons hidden layer 2	1
Number of epochs training phase 1	12
Learning rate phase 1	0.150
Number of epochs training phase 2	8

Table 8
Performance of the best ANN topology for roughness prediction (Case Study no. 1).

	Mean value	Standard deviation	Minimum value
Prediction error (MAE-DI%)	34.299	15.230	10.470

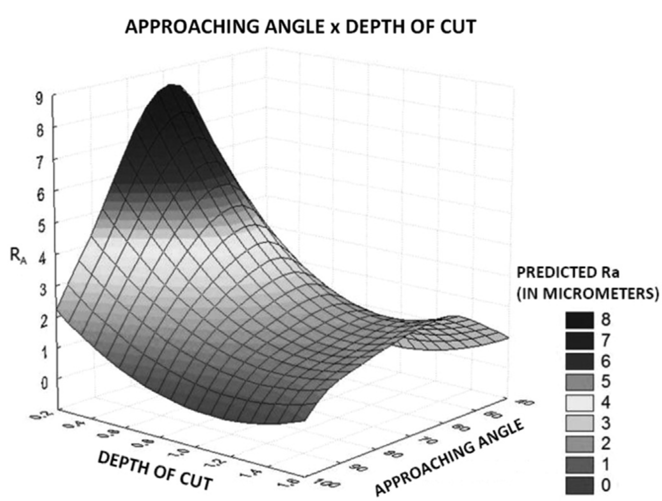

Fig. 7. Response surface for best network topology (data set of Case Study no. 1, Statistica®).

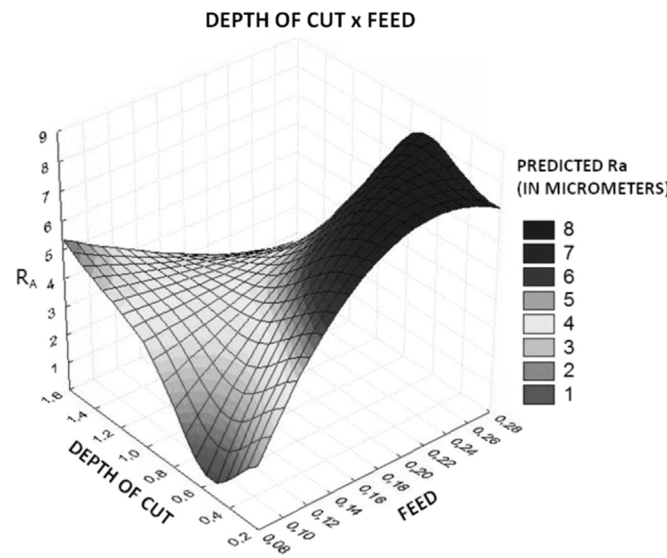

Fig. 8. Response surface for best network topology (data set of Case Study no. 1, (Statistica®).

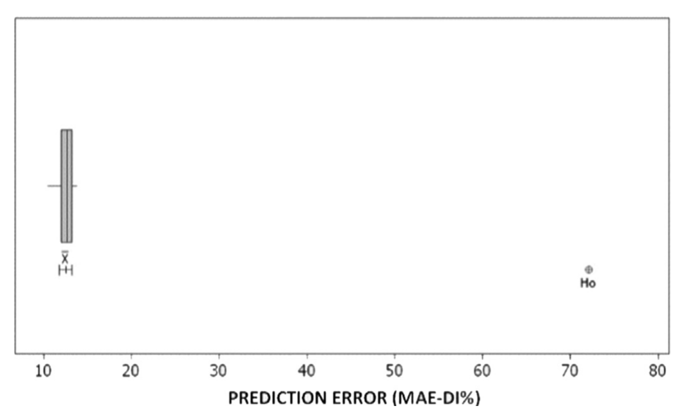

Fig. 9. Boxplot for *t*-test between best ANN topology and result from Case Study no. 1 with 95% t-confidence interval for the mean (Minitab®).

original paper. Fig. 9 additionally displays the boxplot for the outcome of this test. In this figure, X̄ represents the average of this work output variable (MAE-DI%) and H0 represents the error from the work used for comparison.

In addition to the statistical evidence, it could be observed that the same configuration presented even better results in some instances of the test. The overall best performance observed had an error of 10.470%, what gives an indication of this work potential for roughness control.

7.1.3. Comparing results from the proposed method and a network optimization tool using data set from Case Study no. 1

An additional comparison was made between the best MLP network configuration obtained by the proposed method and an optimized topology identified by an automated tool for network optimization, the Statistica IPS®. The performance of the two models on the prediction of surface roughness for the test cases was compared using ANOVA. IPS was employed with extremely elevated high end search limits (up to 100.000 network topologies and up to 1000 neurons in hidden layers) in order to maximize the

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

probability of finding an optimal architecture tool. The output (MAE-DI%) for the ANN proposed by the tool was equal to 14.170%.

Another ANOVA test was employed to compare the prediction error of the two network configurations. The ANOVA result indicates that, at a level of significance of 0.05, there is evidence of statistical difference between the prediction errors of the network configurations under comparison. The prediction error of the network obtained through the method is smaller than that obtained using the software package. The comparison is further illustrated in Fig. 10, which displays a box-plot diagram of the prediction error for the two network configurations. A clear difference can be observed between the ANN performances under comparison.

7.2. Results for Case Study no. 2

7.2.1. Initialization and development

The same initialization values adopted for the previous case were employed for Case Study no. 2. The optimization process required the execution of eleven cycles. Phases ended after the cycles 2, 5, 8 and 11, when results pointed to no error reduction. Obtained results are displayed on Table 9.

Table 9 details the levels of factors corresponding to centers of factorial arrays executed every cycle. This table informs the best combination of factors identified at the end of each cycle, under columns grouped as ‘Best configuration of the cycle’. Prediction performances achieved by that configuration are displayed under column ‘Output (MAE-DI%)’. After the final cycle, the best network topology was the one whose quantitative factors, resulting from EVOP, are detailed in Table 10 and Table 7 the qualitative factors, resulting from Taguchi, are shown on Table 2. This ANN performance is summarized in Table 11.

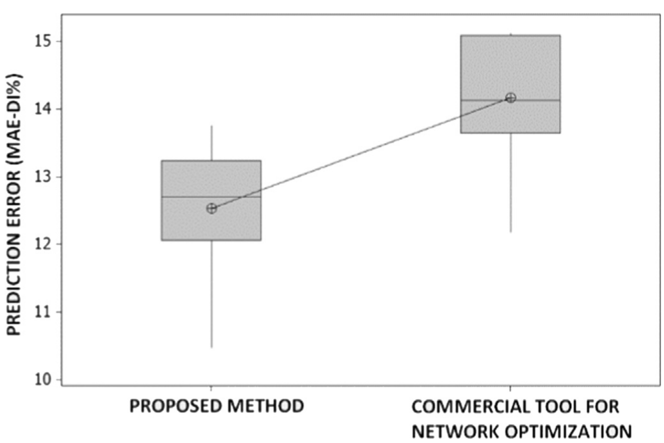

Fig. 10. ANOVA Boxplot for comparison between the proposed method and Statistica IPS® using the data set of Case Study 1 (Minitab®).

7.2.2. Comparing results from the proposed method and the original work [51] used as Case Study no. 2

The performance of the best ANN configuration obtained through the application of the proposed method was compared to that achieved by the ANN topology used in the original work. As for Case Study no. 1, errors were compared by means of a bilateral t-test using the same parameters. The value for the proposed method is equal to 3321% while the corresponding value calculated for Case Study no. 2 is equal to 4899%.

The resulting p-value was 0.000, implying the rejection of the null hypothesis of equal means. Once again, there is strong statistical evidence in favor of the ANN configuration obtained through the proposed optimization method. The boxplot for the hypothesis test is illustrated in Fig. 11. In this figure, $\bar{X}$ represents the average of this work output error (MAE-DI%) while H0 represents the error from the original work. As observed for the first case, the best ANN configuration presents even better results in some instances of the test. The overall best performance observed had an error of 1.399%.

7.2.3. Comparing results from the proposed method and a network optimization tool using data set from Case Study no. 2

The comparison between the best MLP network configuration obtained by the proposed method and an optimized network topology identified by an automated tool for network optimization, the Statistica IPS®, was repeated for Case Study no. 2. Their

Table 10
Best ANN topology for roughness prediction using dataset from Case Study no. 2.

Design factor	Optimal factor level
Number of neurons hidden layer 1	24
Number of neurons hidden layer 2	0
Number of epochs training phase 1	64
Learning rate phase 1	0.600
Number of epochs training phase 2	96

Table 11
Performance of the best ANN topology for roughness prediction (Case Study no. 2).

	Mean value	Standard deviation	Minimum value
Prediction error (MAE-DI%)	6.758	2.095	1.399

Table 9
Experimental array centers and best factor settings per cycle – proposed method application to dataset from Case Study no. 2.

		Array center (natural units)					Best configuration of the cycle (coded units)					Output (MAE-DI%)
Phase	Cycle	H1	H2	E1	LR	E2	H1	H2	E1	LR	E2	
1	1	8	8	32	0.400	64	1	−1	1	1	1	3.570
	2	16	8	64	0.800	128	−1	1	1	−1	−1	4.037
2	3	16	8	64	0.800	128	1	−1	−1	−1	−1	3.807
	4	20	4	48	0.600	96	1	−1	1	1	1	3.707
	5	24	4	64	0.800	128	1	−1	−1	−1	1	3.724
3	6	24	4	64	0.800	128	1	−1	1	−1	−1	3.877
	7	26	2	72	0.700	112	−1	−1	−1	−1	−1	3.321
	8	24	2	64	0.600	96	−1	−1	−1	−1	−1	3.697
4	9	24	2	64	0.600	96	−1	−1	1	1	1	3.867
	10	23	1	68	0.650	104	−1	−1	−1	1	1	3.796
	11	22	1	64	0.700	112	−1	−1	1	1	−1	3.832

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

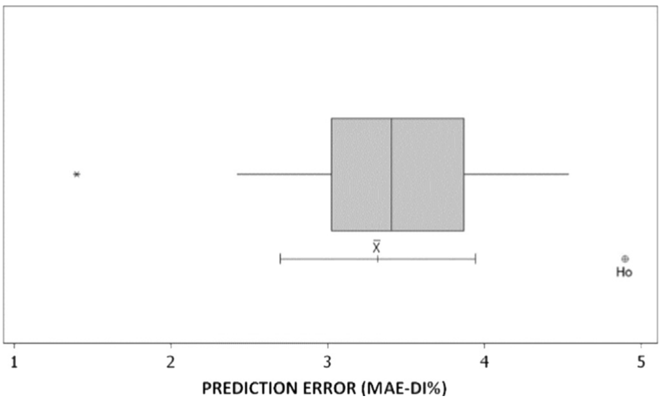

Fig. 11. Boxplot for *t*-test between best ANN topology and result from Case Study no. 2 with 95% *t*-confidence interval for the mean (Minitab®).

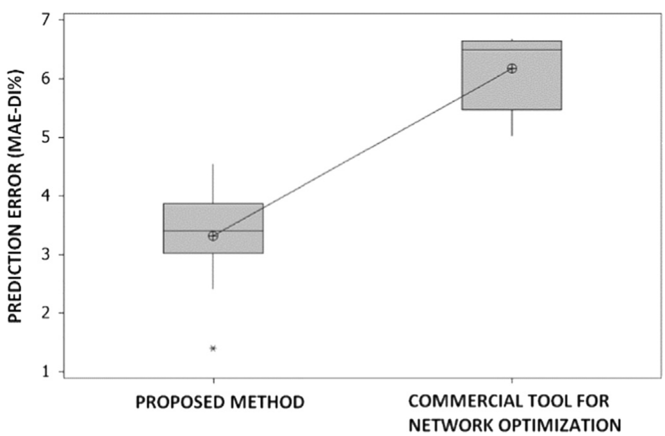

Fig. 12. ANOVA Boxplot for comparison between the proposed method and Statistica IPS® using the data set of Case Study 2 (Minitab®).

performances were compared using ANOVA. As before, IPS was employed with extremely elevated high end search limits (up to 100.000 network topologies and up to 1000 neurons in hidden layers) in order to maximize the probability of finding an optimal architecture tool. The output (MAE-DI%) for the ANN proposed by the tool was equal to 6.189%.

The prediction error of ANN obtained through the proposed method is also lower than the result obtained using the computational package. At a level of significance of 0.05, the results of the ANOVA test evidences the superiority of the network obtained through the proposed method. Fig. 12 displays the box-plot diagram of the ANOVA for the two network configurations.

8. Discussion and conclusions

In this work, a multi-phase EVOP was conducted for the quantitative factors of ANN design. The first EVOP phase begins with broad differences between factor levels until it came to a point where the analysis shows no improvement. A new phase is then initiated, focusing on the best configuration for the previous phase and reducing the intervals between factor levels to a fraction of the previous one. The EVOP phases continued up to a point where no factor is deemed statistically significant to network performance by ANOVA, or to a point that the designer deems to be satisfactory.

The proposed method for the design of ANN was applied to control surface roughness in two different machining processes. It led to the identification of network topologies presenting reductions for training and testing [20,51]. Networks designed according to the proposed method presented also reduction of prediction error of 82.3% and 71.5% in comparison with the ANN model results reported on the original papers.

In addition, statistical prediction error reduction of 9.7% and 46.3% was achieved when comparing the ANNs designed by the proposed method with ANNs topologies identified by a computational tool intended to optimize network topologies, using the same data set. Results obtained also reveal that the dispersion of prediction errors is significantly reduced in comparison with the networks proposed by the papers used as case studies and by computer package.

The use of DOE techniques allows identifying and quantifying each design factor impacts the most on network performance. Moreover, the proposed method takes into account the effect of interactions between factor levels that could affect the performance. It encompasses also other advantages from DOE methodology, such as the realization of hypothesis tests that can prove the differences statistically and the assessment of the extent to which factors left out of the design can influence model performance.

The numerical experiments were conducted on an Intel® Core 2 Quad machine with 1GB RAM. Script execution for each experimental array took about 10 min. Time required for execution increases, as expected, with the number of training cases. Data processing and analysis for each cycle took around 30 min. It can be said that computational cost per cycle is approximately one hour. Such a time slice may be reduced by using more powerful machines or investing in the development of software tools aimed at implementing the whole algorithm of the analytic process.

Based on the results showed here, the proposed algorithm constitutes an effective option for the systematic design of ANNs of MLP architecture for controlling the surface roughness, filling the gap reported in previous literature on the subject. The conclusions obtained should not be extrapolated to architectures of ANN not addressed in this work. On the other hand, the results achieved and the generality of the proposed method does not prevent its application to other network architectures as well as data collected from other processes or from problems of other fields of knowledge.

8.1. Further research

The method proposed in this paper can be employed for optimization of distinct network architectures, such as RBF or SOM, for example. The method could as well be applied to the optimization of machine learning models, such a Support Vector Machines. It could also be applied to datasets obtained from other processes, or from other class of problems (such as classification).

Augmented version of the proposed evolutionary algorithm, covering a larger number of factors, might lead to further reductions in prediction error of the roughness. The list of additional optimization may include the kind of training algorithms of phases 1 and 2, the inputs and pruning of neurons, the weight decay regularization in phases 1 and 2 of training, the use of transfer functions and other pre-processing of data, among others.

Another interesting point worth further investigation is the representation of an ANNs model generalization capability in terms of extreme values statistical distribution.

Acknowledgments

The authors would like to express their gratitude to FAPEMIG (Fundação de Amparo à Pesquisa do estado de Minas Gerais,

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061

Brazil), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil), and CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil) for their support in this research.

References

[1] D. Singh, P.V. Rao, A surface roughness prediction model for hard turning process, Int. J. Adv. Manuf. Technol. 32 (2007) 1115–1124, http://dx.doi.org/10.1007/s00170-006-0429-2.

[2] F. Cus, U. Zuperl, Approach to optimization of cutting conditions by using artificial neural networks, J. Mater. Process. Technol. 173 (2006) 281–290, http://dx.doi.org/10.1016/j.jmatprotec.2005.04.123.

[3] U. Çaydaş, S. Ekici, Support vector machines models for surface roughness prediction in CNC turning of AISI 304 austenitic stainless steel, J. Intell. Manuf. 23 (2012) 639–650, http://dx.doi.org/10.1007/s10845-010-0415-2.

[4] E. Kilickap, M. Huseyinoglu, A. Yardimeden, Optimization of drilling parameters on surface roughness in drilling of AISI 1045 using response surface methodology and genetic algorithm, Int. J. Adv. Manuf. Technol. 52 (2011) 79–88, http://dx.doi.org/10.1007/s00170-010-2710-7.

[5] P. Muñoz-Escalona, P.G. Maropoulos, Artificial neural networks for surface roughness prediction when face milling Al 7075-T7351, J. Mater. Eng. Perform. 19 (2010) 185–193, http://dx.doi.org/10.1007/s11665-009-9452-4.

[6] D. Karayel, Prediction and control of surface roughness in CNC lathe using artificial neural network, J. Mater. Process. Technol. 209 (2009) 3125–3137, http://dx.doi.org/10.1016/j.jmatprotec.2008.07.023.

[7] K.S.S. Lopes, W.F. Sales, E.S. Palma, Influence of machining parameters on fatigue endurance limit of AISI 4140 steel, J. Braz. Soc. Mech. Sci. Eng. 30 (2008) 77–83, http://dx.doi.org/10.1590/S1678-58782008000100011.

[8] J. Antony, R.B. Anand, M. Kumar, M.K. Tiwari, Multiple response optimization using Taguchi methodology and neuro-fuzzy based model, J. Manuf. Technol. Manag. 17 (2006) 908–925, http://dx.doi.org/10.1108/17410380610688232.

[9] Á.B. Jiménez, J.L. Lázaro, J.R. Dorronsoro, Finding optimal model parameters by deterministic and annealed focused grid search, Neurocomputing 72 (2009) 2824–2832, http://dx.doi.org/10.1016/j.neucom.2008.09.024.

[10] M. Correa, C. Bielza, J. Pamies-Teixeira, Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process, Expert Syst. Appl. 36 (2009) 7270–7279, http://dx.doi.org/10.1016/j.eswa.2008.09.024.

[11] U. Çaydaş, A. Hasçalık, A study on surface roughness in abrasive waterjet machining process using artificial neural networks and regression analysis method, J. Mater. Process. Technol. 202 (2008) 574–582, http://dx.doi.org/10.1016/j.jmatprotec.2007.10.024.

[12] F.J. Pontes, J.R. Ferreira, M.B. Silva, A.P. Paiva, P.P. Balestrassi, Artificial neural networks for machining processes surface roughness modeling, Int. J. Adv. Manuf. Technol. 49 (2010) 879–902, http://dx.doi.org/10.1007/s00170-009-2456-2.

[13] D.C. Montgomery, Design and Analysis of Experiments, 5th. ed., John Wiley & Sons, Inc., 2001.

[14] V.G. Dhokia, S. Kumar, P. Vichare, S.T. Newman, R.D. Allen, Surface roughness prediction model for CNC machining of polypropylene, J. Eng. Manuf. 222 (2008) 137–153, http://dx.doi.org/10.1243/09544054jem884.

[15] C. Cervellera, A. Wen, V.C.P. Chen, Neural network and regression spline value function approximations for stochastic dynamic programming, Comput. Oper. Res. 34 (2007) 70–90, http://dx.doi.org/10.1016/j.cor.2005.02.043.

[16] S.R. Karnik, V.N. Gaitonde, J.P. Davim, A comparative study of the ANN and RSM modeling approaches for predicting burr size in drilling, Int. J. Adv. Manuf. Technol. 38 (2008) 868–883, http://dx.doi.org/10.1007/s00170-007-1140-7.

[17] E. Bagci, B. Işik, Investigation of surface roughness in turning unidirectional GFRP composites by using RS methodology and ANN, Int. J. Adv. Manuf. Technol. 31 (2006) 10–17, http://dx.doi.org/10.1007/s00170-005-0175-x.

[18] E.M. Trent, P.K. Wright, Metal Cutting, 4th. ed., Butterworth Heinemann, Woburn, 2000.

[19] P.G. Benardos, G.C. Vosniakos, Predicting surface roughness in machining: A review, Int. J. Mach. Tools Manuf. 43 (2003) 833–844, http://dx.doi.org/10.1016/S0890-6955(03)00059-2.

[20] V.S. Sharma, S. Dhiman, R. Sehgal, S.K. Sharma, Estimation of cutting forces and surface roughness for hard turning using neural networks, J. Intell. Manuf. 19 (2008) 473–483, http://dx.doi.org/10.1007/s10845-008-0097-1.

[21] H. Oktem, T. Erzurumlu, F. Erzincanli, Prediction of minimum surface roughness in end milling mold parts using neural network and genetic algorithm, Mater. Des. 27 (2006) 735–744, http://dx.doi.org/10.1016/j.matdes.2005.01.010.

[22] C.-K. Chang, H.S. Lu, Study on the prediction model of surface roughness for side milling operations, Int. J. Adv. Manuf. Technol. 29 (2006) 867–878, http://dx.doi.org/10.1007/s00170-005-2604-2.

[23] S.K. Pal, D. Chakraborty, Surface roughness prediction in turning using artificial neural network, Neural Comput. Appl. 14 (2005) 319–324, http://dx.doi.org/10.1007/s00521-005-0468-x.

[24] J.P. Davim, V.N. Gaitonde, S.R. Karnik, Investigations into the effect of cutting conditions on surface roughness in turning of free machining steel by ANN models, J. Mater. Process. Technol. 205 (2008) 16–23, http://dx.doi.org/10.1016/j.jmatprotec.2007.11.082.

[25] S.H. Yang, U. Natarajan, M. Sekar, S. Palani, Prediction of surface roughness in turning operations by computer vision using neural network trained by differential evolution algorithm, Int. J. Adv. Manuf. Technol. 51 (2010) 965–971, http://dx.doi.org/10.1007/s00170-010-2668-5.

[26] International Organization for Standardization, ISO 4287:1997 / Cor 2:2005, 1997.

[27] International Organization for Standardization, ISO 1302:2002(E)-Geometrical Product Specifications (GPS)-Indication of surface texture in technical product documentation, 2002.

[28] A.M. Zain, H. Haron, S. Sharif, Prediction of surface roughness in the end milling machining using artificial neural network, Expert Syst. Appl. 37 (2010) 1755–1768, http://dx.doi.org/10.1016/j.eswa.2009.07.033.

[29] Z.W. Zhong, L.P. Khoo, S.T. Han, Prediction of surface roughness of turned surfaces using neural networks, Int. J. Adv. Manuf. Technol. 28 (2006) 688–693, http://dx.doi.org/10.1007/s00170-004-2429-4.

[30] T. Özel, Y. Karpat, Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks, Int. J. Mach. Tools Manuf. 45 (2005) 467–479, http://dx.doi.org/10.1016/j.ijmachtools.2004.09.007.

[31] P.G. Benardos, G.C. Vosniakos, Prediction of surface roughness in CNC face milling using neural networks and Taguchi' s design of experiments, Robot. Comput. Integr. Manuf. 18 (2002) 343–354, http://dx.doi.org/10.1016/S0736-5845(02)00005-4.

[32] Y. Karpat, T. Özel, Multi-objective optimization for turning processes using neural network modeling and dynamic-neighborhood particle swarm optimization, Int. J. Adv. Manuf. Technol. 35 (2007) 234–247, http://dx.doi.org/10.1007/s00170-006-0719-8.

[33] C. Zanchettin, T.B. Ludermir, Global Optimization Methods for Designing and Training Feedforward Artificial Neural Networks, Adv. Neural Netw. 14 (2007) 328–337.

[34] D.E. Coleman, D.C. Montgomery, A. Systematic, Approach to planning for a designed industrial experiment, Technometrics 35 (1993) 1–12, http://dx.doi.org/10.2307/1269285.

[35] C.-F.J. Kuo, Y.-S. Wu, Application of a Taguchi-based neural network prediction design of the film coating process for polymer blends, Int. J. Adv. Manuf. Technol. 27 (2006) 455–461, http://dx.doi.org/10.1007/s00170-004-2215-3.

[36] R. Quiza, L. Figueira, J.P. Davim, Comparing statistical models and artificial neural networks on predicting the tool wear in hard machining D2 AISI steel, Int. J. Adv. Manuf. Technol. 37 (2008) 641–648, http://dx.doi.org/10.1007/s00170-007-0999-7.

[37] J.A.V. Lasheras, F.S. Vilán, J.J.C. García Nieto, P.J. Díaz, The use of design of experiments to improve a neural network model in order to predict the thickness of the chromium layer in a hard chromium plating process, Math. Comput. Model. 52 (2010) 1169–1176, http://dx.doi.org/10.1016/j.mcm.2010.03.007.

[38] J.M. Ortiz-Rodriguez, M.R. Martínez-Blanco, H.R. Vega-Carrillo, Robust design of artificial neural networks applying the taguchi methodology and DoE, In: Proc. Electron. Robot. Automot. Mech. Conf. CERMA 2006, pp. 131–136 http://dx.doi.org/10.1109/CERMA.2006.83.

[39] W.-C. Chen, Y.-Y. Hsu, L.-F. Hsieh, P.-H. Tai, A systematic optimization approach for assembly sequence planning using Taguchi method, DOE, and BPNN, Expert Syst. Appl. 37 (2010) 716–726, http://dx.doi.org/10.1016/j.eswa.2009.05.098.

[40] C. Zanchettin, L.L. Minku, T.B. Ludermir, Design of experiments in neuro-fuzzy systems, Int. J. Comput. Intell. Appl. 09 (2010) 137–152, http://dx.doi.org/10.1142/S1469026810002823.

[41] P.P. Balestrassi, E. Popova, A.P. Paiva, J.W. Marangon Lima, Design of experiments on neural network's training for nonlinear time series forecasting, Neurocomputing 72 (2009) 1160–1178, http://dx.doi.org/10.1016/j.neucom.2008.02.002.

[42] A.R. Queiroz, F.A. Oliveira, J.W.M. Lima, P.P. Balestrassi, Simulating electricity spot prices in brazil using neural network and design of experiments, In: Proceedings of the IEEE Lausanne PowerTech, 2007, pp. 2029–2034 http://dx.doi.org/10.1109/PCT.2007.4538630.

[43] J.P.C. Kleijnen, S.M. Sanchez, T.W. Lucas, T.M. Cioppa, State-of-the-art review: a user's guide to the brave new world of designing simulation experiments, Inf. J. Comput. 17 (2005) 263–289, http://dx.doi.org/10.1287/ijoc.1050.0136.

[44] G.E.P. Box, Evolutionary operation: a method for increasing industrial productivity, J. R. Stat. Soc. Ser. C (Appl. Stat.) 6 (1957) 81–101, http://dx.doi.org/10.2307/2985505.

[45] G. Taguchi, E. Elsayed, T. Hsiang, Quality Engineering in Production Systems, Mcgraw-Hill College, New York, 1988.

[46] G.E.P. Box, J.S. Hunter, W.G. Hunter, Statistics for Experimenters: Design Innovation and Discovery, 2nd. ed., John Wiley & Sons Inc (2006) http://dx.doi.org/10.1198/tech.2006.s379.

[47] S. Haykin, Neural Networks and Learning Machines, 3rd ed., Pearson Prentice Hall, Hamilton, Ontario, Canada, 2009.

[48] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[49] D.C. Montgomery, C.L. Jennings, M. Kulahci, Introduction to Time Series Analysis and Forecasting, John Wiley & Sons Inc, 2008.

[50] B. Sick, On-line and indirect tool wear monitoring in turning with artificial neural networks: a review of more than a decade of research, Mech. Syst. Signal Process. 16 (2002) 487–546, http://dx.doi.org/10.1006/mssp.2001.1460.

[51] S. Sarkar, S. Mitra, B. Bhattacharyya, Parametric optimisation of wire electrical discharge machining of γ titanium aluminide alloy through an artificial neural network model, Int. J. Adv. Manuf. Technol. 27 (2006) 501–508, http://dx.doi.org/10.1007/s00170-004-2203-7.

Fabrício J. PONTES graduated from Universidade Federal de Minas Gerais – UFMG, Brazil (1996), had his master's degree in Industrial Engineering from Universidade Federal de Itajubá – UNIFEI, Brazil (2006), and his D.Sc. in Mechanical Engineering, in the area of Management and Optimization from Universidade Estadual Paulista Julio de Mesquita Filho – UNESP, Brazil (2011). He works as a Product Development Engineer at Embraer. Themes of academic interest: Neural Networks, Applied Statistics, Design of Experiments and Manufacturing Processes.

Gabriela da F. de AMORIM holds a degree in Control and Automation Engineering and a master's degree in Industrial Engineering, both from Universidade Federal de Itajubá – UNIFEI, Brazil. She is currently a Research Scholar at the University of Tennessee (Knoxville), and is pursuing a D.Sc. degree at the Industrial Engineering program from Universidade Federal de Itajubá – UNIFEI, Brazil. Her research interests are in Statistics, Design of Experiments and Six Sigma.

Pedro P. BALESTRASSI received the D.Sc. degree in Industrial Engineering from Universidade Federal de Santa Catarina - UFSC, Brazil, in 2000. Since 1994, he has been at Universidade Federal de Itajubá, Brazil, as Professor of Industrial Engineering. In 1998/1999, he was a visitor at Texas A&M University (TX, USA). In 2005/2006, he was a visiting professor at the University of Texas (Austin) and in 2010/2011, he was a visiting professor at the University of Tennessee (Knoxville), USA. His areas of interest lie in time series forecasting, ANN and statistics.

Anderson P. de PAIVA is Professor of Quantitative Methods in the Industrial Engineering Institute of Universidade Federal de Itajubá, Brazil. He received the D.Sc. degree in Industrial Engineering in 2006 at Universidade Federal de Itajubá, Brazil. His research interests are in Multi-variate Statistical Analysis, Nonlinear Optimization and Neural Networks applied to manufacturing processes.

João R. FERREIRA graduated in Mechanical Engineering from Escola Federal de Engenharia de Itajubá - EFEI (1986), Brazil, had his Master's and D.Sc. degrees in Mechanical Engineering from Universidade Estadual de Campinas - UNICAMP (1992 and 1999), Brazil. He is currently a Professor at Universidade Federal de Itajubá, Brazil, who has experience in Machining Manufacturing Processes area, as well as Design and Analysis of Experiments and Optimization of processes.

Please cite this article as: F.J. Pontes, et al., Design of experiments and focused grid search for neural network parameter optimization, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.12.061