A bio-inspired computational high-precision dental milling system

Vicente Vera, Alvaro Enrique Garcia
Maria Jesus Suarez, Beatriz Hernando
Departamento de Estomatología I & II
Facultad de Odontologia, UCM
Madrid, Spain
vicentevera@odon.ucm.es; aegarcia@odon.ucm.es

Emilio Corchado, Maria Araceli Sanchez, Ana Gil
Departamento de Informática y Automática
Universidad de Salamanca
Salamanca, Spain
escorchado@usal.es

Raquel Redondo
Department of Civil Engineering
University of Burgos
Burgos, Spain
rredondo@ubu.es

Javier Sedano
Dept. of A.I. & Applied Electronics
Castilla y León Technological Institute
Burgos, Spain
javier.sedano@itcl.es

***Abstract*— A novel bio-inspired computational high-precision dental milling system is proposed in this interdisciplinar research. The system applies several bio-inspired models, based on unsupervised learning, that analyse and identify the most relevant features of high-precision dental-milling data sets and their internal structures. Finally, a supervised neural architecture and certain identification techniques are applied, in order to model and to optimize the high-precision process. This is done by empirically testing the model using a real data set taken from a dynamic high-precision machining centre with five axes.**

Keywords-Artificial Neural Networks; Exploratory Projection Pursuit; Industrial and Medical Applications; Identification Systems; Control System

I. INTRODUCTION

Bio-inspired computing represents a set of various technologies based on biological principles, characterized by their adaptive, reactive, and distributed properties. It investigates, simulates and analyzes very complex issues and phenomena in order to solve real-world problems. In this case, bio-inspired models are used in a high-precision medical process (dental milling process) which requires an interdisciplinary approach.

The marginal adjustment of dental prostheses to the dental structure (i.e. tooth enamel) without invading tissue and other anatomical spaces is an essential aspect of dentistry, as the success of any dental care treatment is based on an adjustment of about 15-25 micrometers between the dental prosthesis and the contact tissue. This close fit helps to reduce and prevent bacteria and germs that filter into oral fluid that could eventually lead to further dental treatment.

In the past, dental structures were hand moulded by technicians, using casting wax. This is still a valid technique, but the behaviour of certain variables that play an important role in this process is not fully under control, which can affect the precision of the moulding process.

It is for this reason, among others, that artificial intelligent and bio-inspired models are now applied to the field of dental prosthestics, in order to achieve a higher precision dental adjustment process that is around 15-25 micrometers.

New dental milling machines permit the manufacture of individual crowns or complex bridges from data collected directly from a dental scanner.

The proposed model was tested and validated using a three-step procedure: firstly, the dataset is analyzed using projection methods such as Principal Component Aanalysis (PCA) and Cooperative Maximum-Likelihood Hebbian Learning (CMLHL) [1] to extract the dataset structure and the key relations between its variables and to establish whether the data set is sufficiently informative. It means that the initial collected data set once is analysed, it shows a certain degree of clustering, which is a sign that there is no any problem related to any sensor when collecting the information. Also means that as there is a certain degree of clustering, there is any kind of correlation and the process is well defined by such data set. Then, we can apply the following steps of the process. Finally, a model is produced at the modelling stage to estimate production time errors by modeling techniques.

This paper is organized as follows. Section II introduces the unsupervised projection techniques for analysing the datasets, in order to extract their relevant internal structures. Section III deals with classical identification techniques used in the system modelling. Section IV describes the case study and Section V describes the bio-inspired model that is presented to solve the case study. Section VI presents the experimentation and its results and finally, the conclusions are set out and some comments on future lines of work are outlined.

II. BIO-INSPIRED COMPUTING

Bio-inspired computing is a set of several technologies devoted to solve complex problems using computational methods modeled on design principles drawn from the natural world [2]. It investigates, simulates, and analyzes very complex issues and phenomena in order to solve real-world problems [3]. There are many algorithms [4][5] reported in the literature for artificial neural networks [7][12][18], which constitute one area under the umbrella of

bio-inspired computing that has been successfully applied to feature selection.

A. Data structure analysis using connectionist techniques

1) Unsupervised models.

The Negative Feedback Network [13] is defined as follows. Consider an N-dimensional input vector, (x), and a M-dimensional output vector, (y), with W_{ij} being the weight linking input j to output i and let η be the learning rate.

The initial situation is that there is no activation at all in the network. The input data is feedforward via the weights from the input neurons (the x-values) to the output neurons (the y-values) where a linear summation is performed to get the output neuron activation value. We can express this as:

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \tag{1}$$

The activation is fed back through the same weights and subtracted from the inputs (where the inhibition takes place):

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j \tag{2}$$

After that, simple Hebbian learning is performed between input and outputs:

$$\Delta W_{ij} = \eta e_j y_i \tag{3}$$

The effect of the negative feedback is the network learning stability. This network is capable of finding the principal components of the input data [13] in a manner that is equivalent to Oja's Subspace algorithm [14], and so the weights will not find the current Principal Components but a basis of the Subspace spanned by these components.

Maximum Likelihood Hebbian Learning [11][12][15][17] is based on the previous PCA-type rule and can be described as a family of learning rules based on the following equations: a Feedforward step (1) followed by a Feedback step (2) and then a weight change, which is as follows:

$$\Delta W_{ij} = \eta . y_i . sign\left(e_j\right) | e_j |^{p-1} \tag{4}$$

It is expected that for leptokurtotic residuals (more kurtotic than a Gaussian distribution), values of $p<2$ would be appropriate, while for platykurtotic residuals (less kurtotic than a Gaussian), values of $p>2$ would be appropriate.

By maximising the likelihood of the residual with respect to the current distribution (tuning p parameter), the learning rule is matched to the probability density function of the residual. Maximum Likelihood Hebbian Learning (MLHL) [11][12][15][17] has been linked to the standard statistical method of Exploratory Projection Pursuit (EPP) [16][11].

2) Cooperative Exploratory Projection Pursuit

Cooperative Maximum-Likelihood Hebbian Learning (CMLHL) [1] is based on the standard Maximum Likelihood Network [11][12][15][17] but has a lateral connection (which acts after the feedforward but before the feedback) derived from the Rectified Gaussian Distribution [17][1][18] and using cooperative distribution. The final architecture is described as follows. There is a Feedforward step (1), then there is a lateral activation passing:

$$y_i(t+1) = \left[y_i(t) + \tau\left(b - Ay\right)\right]^+ \tag{5}$$

A Feedback step (2) and finally a weight updating (4) takes place. Where:

The parameter τ represents the strength of the lateral connections. The cooperative distribution in the case of N variables is defined by:

$$A_{ij} = \delta_{ij} + \frac{1}{N} - \frac{4}{N}\cos\left(\frac{2\pi}{N}\left(i-j\right)\right) \tag{6}$$

and

$$b_i = 1 \tag{7}$$

Where δ_{ij} is the Kronecker delta and i and j represent the identifiers of the output neuron. Finally, $[\,]^+$ is necessary to ensure that the y-values remain within the positive quadrant.

B. Feature selection and extraction

Feature Selection and extraction [7][8] entails feature construction, space dimensionality reduction, sparse representations and feature selection among others. They are all commonly used pre-processing tools in machine learning tasks, which include pattern recognition. Although researchers have grappled with such problems for many years, renewed interest has recently surfaced in feature extraction.

The feature selection approach in this study is based on the dimension reduction issue. Initially, some projection methods such as PCA [9][10], MLHL [11][12] and CMLHL [1] are applied. In a first step they aim to analyse the internal structure of a representative data set of a case of study. If after applying these models a clear internal structure can be identified, it means that the recorded data is sufficiently informative -the system has been excited in their operating ranges and it can be modeled successfully-. Otherwise, data must be properly collected.

III. System Identification

System identification (SI) aims to obtain mathematical models to estimate one or more behaviours from a physical process the dynamic equations of which are unknown. Classic SI refers to the parametrical literature, which has its origin in the linear system analysis [19]. Nevertheless, increased computational capability and the availability of bio-inspired techniques have widened research into SI. Artificial Neural Networks (ANNs) are one of the most interesting bio-inspired paradigms used in SI. In some ways, the SI procedure invariantly takes advantage of the modelling technique in use.

The SI procedure comprises several steps [20][21]: the selection of the models and their structure, the learning methods [22][23], the identification and optimization criteria and the validation method. Validation ensures that the selected model meets the necessary conditions for estimation and prediction. Typically, validation is carried out using three different methods: the residual analysis $\varepsilon(t, \hat{\theta}(t))$ -by means of a correlation test between inputs, their residuals and their combinations-; the mean squared error (MSE) and

the generalization error value -normalized sum of squared errors (NSSE)- and finally a graphical comparison between the desired outputs and the model outcomes through simulation.The remaining parts of this section describe the use of ANN in SI.

A. *The ANN in the identification process*

The identification criterion consists in evaluating the group of candidate models that best describes the dataset gathered for the experiment; i.e., given a certain model $M(\theta_*)$, its prediction error may be defined as in (8). The aim is to obtain a model that meets the following premise [20]: a good model is one that makes good predictions and which produces small errors when the observed data is applied.

$$\varepsilon(t,\theta_*) = y(t) - \hat{y}(t \mid \theta_*) \tag{8}$$

The use of ANN in the process of identification requires the selection of several parameters: the number of layers, the number of neurons per layer and the activation functions [24][25]. It has been established that two feedforward layers using sigmoidal or hyperbolic functions in the hidden layer can learn any input-output relationship, nevertheless, more layers might learn complex relationships [26]. A feedforward network with two layers is shown in "Fig. 1".

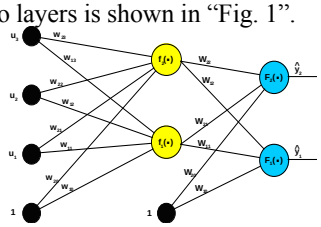

Figure 1. A feedforward network with two layers, with two nodes per layer, and three inputs. *W* is the weight matrix between the hidden and output layer, while *w* is the weight matrix between the inputs and the hidden layer. The network has two bias nodes with value 1.

The number of neurons per layer is also a relevant design parameter. It should be analyzed in order to avoid over fitting [27][28][29]. Each algorithm will introduce some restrictions in the weight matrix. The most widely used training algorithms are the Lenvenberg-Marquardt method [30], the recursive Gauss-Newton method [20], the quasi-Newton algorithm [31] and the batch and the recursive versions of the back-propagation algorithm [32].

When using ANN, the purpose of an identification process is to determine the weight matrix based on the observations Z^t, so as to obtain the relationships between the network nodes. The weight matrix is usually referred as *w*, *W* or θ.

The supervised learning algorithm is then applied to find the estimator θ, so as to obtain the identification criterion. In this case, the minimization of the mean square error criterion as defined in (9) and (10) is used. The iterative minimization scheme is defined in (11), where *f(t)* represents the search direction and $\mu(t)$ the step size.

$$V_N(\theta, z^t) = \frac{1}{N}\sum_{t=1}^{N}\left[y(t) - \hat{y}(t \mid \theta)\right]^{\mathrm{T}}\left[y(t) - \hat{y}(t \mid \theta)\right] \tag{9}$$

$$\hat{\theta} = \arg\min_{\theta} V_N(\theta, Z^t) \tag{10}$$

$$\theta(t+1) = \theta(t) + \mu(t) f(t) \tag{11}$$

IV. CASE STUDY:

CAD/CAM in the field of dental applications may be structured into three steps: digitization (for instance, of a tooth stump in the mouth), followed by computer software design, specific to each system, allowing the design of the abutment of the prosthetic structure.

Finally, there is a mechanized process, in which milling instruments work on different materials such as ceramics, titanium, chromo-cobalt and compound resins. Therefore, at present, the latest dental milling machines manufacture individual crowns and complex bridge structures from the data collected by the dental scanner.

This study describes the way in which a bio-inspired system can be applied to optimize the last step of a Computer Aid Design/Computer Aid Manufacturing (CAD/CAM) system, by optimizing the time error detection for manufacturing metal dental pieces as shown in "Fig.2".

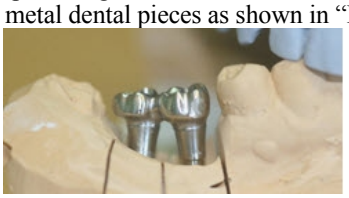

Figure 2. Example of metal pieces obtained by a high precision laser milling

The case study is described by an initial data set of 34 samples obtained by the dental scanner in the manufacturing of chromo-cobalt dental pieces with a toric tool. This data set has 6 input variables (Number of pieces, Radius, Revolutions, Feed rate X, Y and Z) and 2 output variables (Erosion and Real time of work) as shown in "TABLE I". Time errors for manufacturing are the difference between the estimated time by the machine itself and real time for production.

TABLE I. VALUES OF EACH VARIABLE USED IN THE PROCESS.

Variable (Units)	Range of values
Number of pieces	1 to 4
Radius	0.4 to 1.5
Revolutions per minute	10600 to 22000
Feed rate X	200 to 3000
Feed rate Y	200 to 3000
Feed rate Z	200 to 2000
Erosion (mm.)	0 to 0.1195
Real time of work (s.)	48 to 1468
Time errors for manufacturing (s.)	24 to 550

A dynamic high-precision machining centre with five axes ("Fig.3") was applied in this experiment.

Figure 3. A dynamic high precision machining center with five axes

V. OPTIMIZING A DENTAL MILLING PROCESS

A time-errors detection method is proposed for manufacturing dental pieces by comparing the material properties evolution of the manufacturing process –time errors- in normal operation with respect to its estimated time behaviour. Firstly, the dental manufacturing process is parameterised and its dynamic performance in normal operation is obtained by real manufacturing of dental pieces. Then, the gathered data is processed using PCA and CMLHL to recognize data set structures in order to determine the ability of the data set to be modeled and to identify the most relevant variables. This allows a third step, knowing a priori, that the model to obtain, can be achieved.

Once the model has been obtained –in the third step-, it is then used as a reference model to calculate the best conditions under normal operating conditions in a dental milling process for manufacturing dental pieces, so if the operator wants to make a dental piece, he may determine the best machining conditions to minimize manufacturing time errors compared to the estimated manufacturing time which is given by the machine itself. Election of the operating conditions was performed manually; a new step is under development that will automatically include this task in the process.

This section deals with the description of each step once the data set is collected (see Section IV). In the next subsection, the generation of the data set is described that will be used in the process. Sub-Section A presents the PCA and CMLHL steps, while in Sub-Section B the procedure to obtain the time error model is detailed.

A. Identification of the relevant features

As detailed in Section II, PCA and CMLHL, both of which were applied to this real-life case study, are techniques for identifying the internal structure of a data set and also to identify the most relevant variables. Then, by means of projection methods we analyse whether the data set is sufficiently representative of a case study, and we identify the most relevant variables to reduce the computational cost in the third and last step.

B. Modelling a normal dental milling operation

Once the relevant variables and their transformations have been extracted from the production data, then a model to fit the normal manufacturing operation should be obtained. This is done to identify bias in the estimated production time, which, in the end, is used for time error detection in the manufacturing of dental pieces. The different model learning methods used in this study were implemented in Matlab© [33].

The experiment followed the identification procedure detailed in Section III: the model structures were analyzed in order to obtain the models that best suited the dataset.

Moreover, several different indexes were used to validate the models. The indexes are well-known and widely used measures in system identification [20][21]: the percentage representation of the estimated model; the graphical representation for the prediction – $\hat{y}_1(t \mid m)$ – versus the measured output – $y_1(t)$ –; the loss function or error function (V) and the generalization error value.

The percentage representation of the estimated model is calculated as the normalized mean error for the prediction (FIT1). The loss function or error function (V) is the numeric value of the mean squared error (MSE) that is computed with the estimation data set. Finally, the generalization error value is the numeric value of the normalized sum of squared errors (NSSE) that is computed with the validation data set (NSSE1) and with the test data set (NSSE2).

VI. RESULTS

This initial data set was analyzed in order to select the features that best describe the relationships with manufacturing time errors [34].

CMLHL is a powerful technique for identifying internal dataset structures. It is applied to a data set, in order to select the features that best describe the relationships between variables, and determine whether the dataset is sufficiently informative. The axes forming the projections (Fig. 4. and Fig.5) are combinations of the features contained in the original datasets. The X and Y axes of the projections are not associated to a unique original feature in general. In the case of PCA, the model is looking for those directions with the biggest variance, when CMLHL is looking for those and index which measures how interesting is a dimension/direction. In this case, directions which are as less Gaussian as possible, (analyzing the kurtosis) [1] [11] .

As may be seen in "Fig. 4", PCA "Fig. 4.a" and CMLHL "Fig. 4.b" both found a clear internal structure in the dataset. Both methods identified 'revolutions' and ‘feed rate’ as relevant variables. CMLHL projection gives us more information because it has recognized the 'number of pieces' as another important variable. CMLHL provides a sparser representation than PCA.

An analysis of the results obtained with the CMLHL model, "Fig. 4.b", leads to the conclusion that CMLHL has identified four different clusters ordered by 'number of pieces'. A further conclusion is that CMLHL identified different clusters ordered by 'revolutions' or ‘feed rate’. It has identified fifteen clusters ordered by 'number of pieces' and 'revolutions'.

Inside each cluster there are further classifications by 'real time of work' and the dataset can be said to have an interesting internal structure.

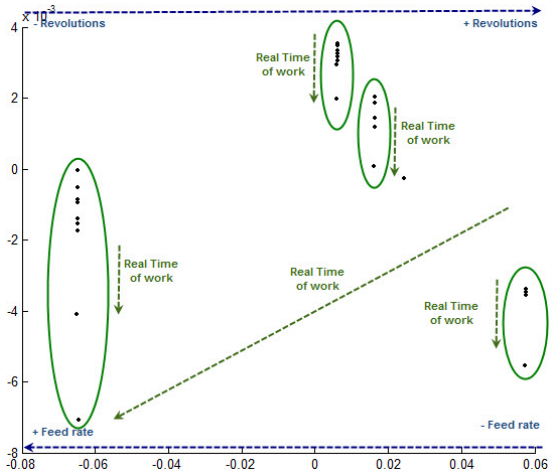

Figure 4 (a). PCA projections.

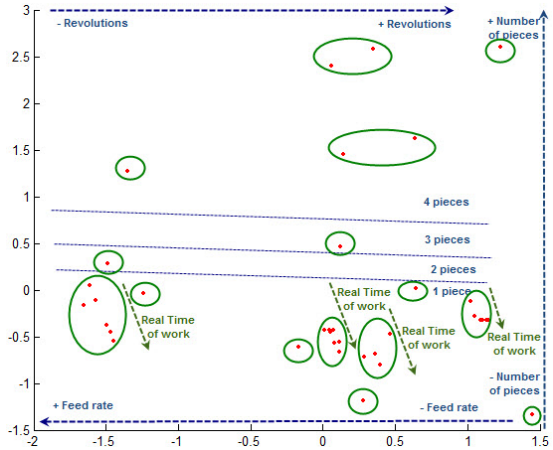

Figure 4 (b). CMLHL projections after 100000 iterations using a learning rate of 0.01, p=0.5 and τ=0.05.

Figure 4. PCA projections (Fig. 4.a) and CMLHL projection (Fig. 4.b)

When the dataset is considered sufficiently informative, the step for modelling the relations between inputs and production time errors in the process begins, through the application of several conventional ANN modelling systems.

Thus, an ANN was used to monitor time error detection in the manufacturing of dental pieces by using the preprocessed data set from the input and output normalization step –zero mean and unity standard deviation-, the reduction of the input vectors dimension –the data set gathered in the previus step- and the use of early stopping and Bayesian regularization techniques [29], a strategy to generalize new situations. "TABLE II" shows the features of the chosen ANN, training, structures, etc. and the characteristics and qualities for estimation and prediction, along with its indexes.

The graphic representations of the prediction - $\hat{y}_1(t \mid m)$ - of time error detection in the manufacturing of dental pieces versus the real measured - $y_1(t)$ - is shown in "Fig. 5" and "Fig. 6" for a feedforward network structure. These figures were used to validate the models. In "Fig. 5", the x-axis shows the number of total samples and the y-axis represents the normalized output variable range: which are the normalized time errors for manufacturing. In "Fig. 6", the y-axis represents the unnormalized output. For the early stopping technique, the estimation, validation and test data sets include 22, 6 and 6 samples, respectively. For the Bayesian regularization technique, the estimation data set includes 34 samples.

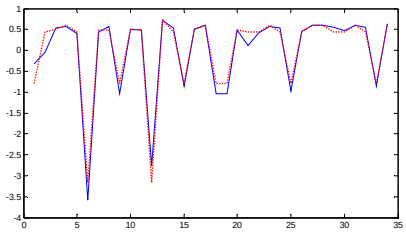

Figure 5 (a) Feedforward network with early stopping.

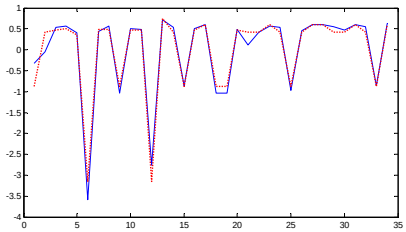

Figure 5 (b) Feedforward network with Bayesian regularization.

Figure 5. Normalized output response of the model: The current output (solid line) is graphically presented with prediction (dotted line). In Fig. 5.a is used the quasi-Newton algorithm for training, while in Fig. 5.b in training the Lenvenberg-Marquardt method has been used

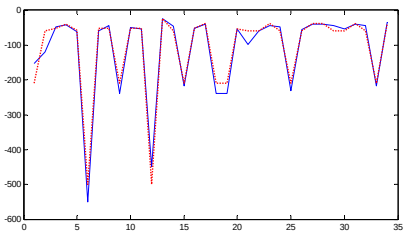

Figure 6 (a). Feedforward network with early stopping.

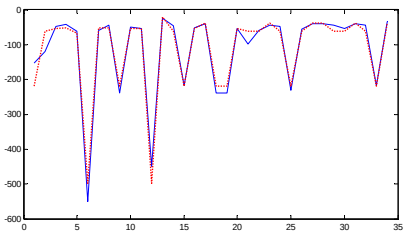

Figure 6 (b) Feedforward network with Bayesian regularization

Figure 6. Unnormalizaed output response of the model: The current output (solid line) is graphically presented with prediction (dotted line). In Fig. 6.a is used the quasi-Newton algorithm for training, while in Fig. 6.b in training the Lenvenberg-Marquardt method has been used.

From "Fig. 5", it may be concluded that the feedforward networks are able to simulate and predict the behaviour of time errors for the manufacturing of dental pieces –as a consequence of the production process-. They are capable of modelling more than 81% of the actual measurements.

In "TABLE II" the value of others indexes to validate the models obtained are shown. These models not only present a lower loss function (V) and error values (NSSE1 and NSSE2), but also a higher system representation index value (FIT1). The model thus obtained may not only be used to predict time errors for the manufacture of dental pieces, but also to determine the normal operating conditions of dental milling processes.

TABLE II. THE VALUE OF THE QUALITY INDEXES OBTAINED FOR THE PROPOSED MODEL

Model	Indexes
Feedforward network for the milling dental process. The model was obtained using the Bayesian regularized criterion. The ANN structure has 20 hyperbolic tangent units -layer 1-, 30 hidden hyperbolic tangent units -layer 2- and 1 linear output unit. The network is estimated using the Lenvenberg-Marquardt method.	FIT1:81.38% V: 0.036
Feedforward network for the milling dental process. The model was obtained using the early stopping technique. The ANN structure has 20 hyperbolic tangent units -layer 1-, 30 hidden hyperbolic tangent units -layer 2- and 1 linear output unit. The network is estimated using the quasi-Newton method.	FIT1:81.11% V: 0.0409 NSSE1:0.0117 NSSE2:0.0268

VII. CONCLUSIONS AND FUTURE WORK

The novel dental milling process described in this paper is for use in the manufacture according to medical specifications of precisely molded dental pieces such as implants.

The dental milling presents an important error rate for manufacturing about the 63%. This is because the difference between the estimated time by the machine itself and real time for production. The model obtained is capable of modelling more than 81% of the actual measurements. This helps to reduce the error and the variability rate of manufacturing processes up to 12%, compared to 63% initially. Acceptable error rate in planning work for dental milling.

In this research, the proposed methodology –for time error detection in chromo-cobalt dental pieces manufactured with a toric tool– in a first step aims to detect the ability of the data set to be modeled. In this step, both PCA and CMLHL have been used to show that the data sets were sufficiently informative and can be modeled in the following step with a greater guarantee of success. Finally, different techniques -supervised neural models- were applied to obtain a suitable model that is able to detect the time errors in the manufacturing of dental pieces.

Future lines of research include modelling erosion -errors in length or tooth wear-, which is a measure of the quality of the dental milling process. The optimization of manufacturing conditions from the model is also proposed, by using genetic algorithms -posting error values to achieve optimal manufacturing conditions-. Finally, an algorithm will be developed to automatically identify the best operating conditions: minor time errors for the manufacturing of dental pieces and minor erosion. The resulting model will moreover be applied to different metals used in prosthetic dentistry.

ACKNOWLEDGEMENT

This research is partially supported through projects of the Junta of Castilla and León (JCyL): [ref: BU006A08] and the Spanish Ministry of Science and Innovation [ref: TIN2010-21272-C02-01, CIT-020000-2009-12 and PID 560300-2009-11], the Spanish Ministry of Education and Innovation [ref: CIT-020000-2008-2] and the Spanish Ministry of Science and Technology [ref: TIN2008-06681-C06-04]. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolin Ingenieria, S.A., within the framework of project MAGNO2008 - 1028.- CENIT also funded by the Spanish Ministry of Science and Technology and also to ESTUDIO PREVIO (Madrid-Spain) for its collaboration in this research.

REFERENCES

[1] E. Corchado and C. Fyfe: Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. Int. Journal of Pattern Recognition and Artificial Intelligence, 17(8), pp. 1447-1466, (2003).

[2] T. Kohonen: The self-organizing map. Neurocomputing 21(1-3): 1-6 (1998)

[3] J. Sedano, E. de la Cal, L. Curiel, J. R. Villar and E. Corchado. Soft Computing for detecting thermal insulation failures in buildings. Proceedings of the 9thInternational Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE2009, Gijon, Spain. 2009.

[4] P. Leray, P. Gallinari, Feature selection with neural networks. Behaviormetrika, Vol. 26 (January 1999)

[5] A. Verikas and M. Bacauskiene, Feature selection with neural networks . Pattern Recognition Letters. Volume 23, Issue 11, September 2002, Pages 1323-1335

[6] C. Fyfe, R. Baddeley, and D.R. McGregor. *Exploratory Projection Pursuit: An Artificial Neural Network Approach*. Research report/94/160. University of Strathclyde, 1994.

[7] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research, Special Issue on variable and Feature Selection*, 3:1157–1182, 2003.

[8] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Knowledge and Data Engineering, IEEE Transactions*, 17(4):491–502, 2005.

[9] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[10] H. Hotelling. Analysis of a complex of statistical variables into Principal Components. *Journal of Education Psychology*, 24:417–444, 1933.

[11] E. Corchado, D.MacDonald, and C. Fyfe. Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit. *Data Mining and Knowledge Discovery*, 8(3):203–225, 2004.

[12] C. Fyfe and E. Corchado. Maximum Likelihood Hebbian rules. In *Proceedings of the 10th European Symposium on Artificial Neural Networks(ESANN 2002)*, pp. 143–148, Bruges, Belgium, 2002. D-side Publishers.

[13] C. Fyfe. Neural network for PCA and beyond. *Neural Processing Letters*, 6:33–41, 1996.

[14] E. Oja. Neural networks, principal components and subspaces. *International Journal of Neural Systems*, 1:61–68, 1989.

[15] E. Corchado and C. Fyfe. Orientation selection using Maximum Likelihood Hebbian Learning. *International Journal of Knowledge-Based Intelligent Engineering*, 7(2), 2003.

[16] J.H. Freedman and J.W. Tukey. Projection Pursuit algorithm for Exploratory data-analysis. *IEEE Transactions on Computers*, 23(9):881–890, 1974.

[17] E. Corchado, Y. Han, and C. Fyfe. Structuring global responses of local filters using lateral connections. *Journal of Experimental & Theoretical Artificial Intelligence*, 15(4):473–487, 2003.

[18] H.S. Seung, N.D. Socci, and D. Lee. The rectified gaussian distribution. *Advances in Neural Information Processing Systems*, 10:350–356, 1998.

[19] J. Sedano, E. Corchado, L. C. Herrera et al., "The application of a two-step AI model to an automated pneumatic drilling process," International Journal of Computer Mathematics, Vol. 86, No. 10-11, pp. 1769-1777, 2009.

[20] L. Ljung. System Identification. Theory for the User. Prentice-Hall, Upper Saddle River, N.J., USA, 2nd edition.

[21] M. Nørgaard, O. Ravn, N.K. Poulsen, and L.K. Hansen. Neural Networks for Modelling and Control of Dynamic Systems. Springer-Verlag, London U.K., 2000.

[22] P. Stoica and T. Söderström. A useful parametrization for optimal experimental design. In IEEE Trans. Automatic. Control, AC-27, 1982.

[23] X. He and H. Asada. A new method for identifying orders of input-output models for nonlinear dynamic systems. In Proc. of the American Control Conf., S.F., pp. 2520–2523, California, 1993.

[24] M.M. Gupta and D.H. Rao. On the principles of fuzzy neural networks. Fuzzy Sets and Systems, 61(10):1 – 18, January.

[25] S.D. Bekiros and D.A. Georgoutsos. Evaluating direction-of-change forecasting: Neurofuzzy models versus neural networks. In Mathematical and Computer Modelling, Vol. 46, Italy. Eds. M. Marinaro & P.G. Morasso.

[26] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal aproximators. Neural Networks, 2(5):359–366, 1989.

[27] L.K. Hansen and M.W. Pedersen. Controlled growth of cascade correlation nets. In International Conference on Artificial Neural Networks, pp. 797–800, Italy, 1994. Eds. M. Marinaro and P.G. Morasso.

[28] B. Hassibi and D.G. Stork. Second order derivatives for network pruning: Optimal Brain Surgeon. In Advances in neural Information Processing System 5, pp. 164–171, San Mateo, CA, USA, 1993. Eds. S.J. Hanson et al.

[29] D.J.C. Mackay, "Bayesian interpolation", Neural Computation, Vol. 4, No. 3, 1992, pp. 415-447.

[30] R. Fletcher. Practical Methods of Optimization. Wiley & Sons, Chichester, UK, 2nd edition, 1987.

[31] J.E. Dennis and R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Englewood Cliffs, NJ: Prentice-Hall, 1983.

[32] J. Hertz, A. Krogh, and R.G. Palmer. Introduction to the Theory of Neural Computation, Vol. 1. Addison-Wesley, Santa Fe Institute Studies In The Sciences Of Complexity, 1991.

[33] H. Demuth, M. Beale and M. Hagan, Neural Network Toolbox User's Guide, The Mathworks, Inc.,2010.

[34] J. Sedano, L. Curiel, E. Corchado et al., "A soft computing method for detecting lifetime building thermal insulation failures," Integrated Computer-Aided Engineering, Vol. 17, No. 2, pp. 103-115, 2010.