Hybrid Incremental Modeling Based on Least Squares and Fuzzy K-NN for Monitoring Tool Wear in Turning Processes

Francisco Penedo, Rodolfo E. Haber, *Member, IEEE*, Agustín Gajate, and Raúl M. del Toro

***Abstract*—There is now an emerging need for an efficient modeling strategy to develop a new generation of monitoring systems. One method of approaching the modeling of complex processes is to obtain a global model. It should be able to capture the basic or general behavior of the system, by means of a linear or quadratic regression, and then superimpose a local model on it that can capture the localized nonlinearities of the system. In this paper, a novel method based on a hybrid incremental modeling approach is designed and applied for tool wear detection in turning processes. It involves a two-step iterative process that combines a global model with a local model to take advantage of their underlying, complementary capacities. Thus, the first step constructs a global model using a least squares regression. A local model using the fuzzy k-nearest-neighbors smoothing algorithm is obtained in the second step. A comparative study then demonstrates that the hybrid incremental model provides better error-based performance indices for detecting tool wear than a transductive neurofuzzy model and an inductive neurofuzzy model.**

***Index Terms*—Fuzzy k-nearest-neighbors, hybrid model, machining processes, tool wear.**

I. Introduction

MAJOR advances in techniques for modeling complex, large-scale systems are currently under development. Effective understanding, control and optimization of such systems may only be achieved through a model or some similar representation. The modeling process consists of finding a mathematical representation of its behavior (differential equations, integral equations, etc.). However, the application of conventional techniques to the complexity and nonlinearity of certain processes is cumbersome and costly.

In general, one of the main shortcomings when modeling a system is the need for prior knowledge of the structure that the model will take (parametric model) before an approximation is made. Unfortunately, in most real cases, it is not easy to define the structure or functional form of the model, and any decision in this regard can subjectively influence the nature of the problem. This situation has sparked an interest in the study and use of nonparametric techniques [1].

Nowadays, efficient modeling strategies are a key issue in the development of a new generation of monitoring systems. Statistics and soft-computing techniques have been widely applied to process modeling and monitoring over past decades. A wide variety of methods based on principal component analysis [2], partial least squares [3], fuzzy and neural computing [4]–[6], evolutionary computation and machine learning [7], and probabilistic reasoning have been developed for process monitoring and fault diagnosis [8]. A complete review of current research and development in this area goes beyond the scope of this paper. Nevertheless, it is of interest to assess how this topic has been addressed from the perspective of industrial informatics. Odiowei and Cao [2] recently reported an interesting example of current research based on principal component analysis (PCA). They propose and verify a new monitoring technique combining canonical variate analysis and kernel density estimation in a simulated plant.

Among soft-computing techniques, fuzzy clustering is one of the most intensively used strategies for process modeling and monitoring [9]. Fuzzy c-means, Gustafson–Kessel (G–K), Gath–Geva (G–G), and fuzzy k-nearest neighbor are the most commonly applied methods. Filev *et al.* [10] have recently presented a practical framework for autonomous monitoring of equipment to enable autonomous diagnostics and prognostics. The kernel of this approach is an "evolving" model based on unsupervised learning methods and the application of a Greedy expectation maximization (EM) clustering algorithm [11], in which multiple fuzzy regions serve to represent machine signatures under different operating conditions. Lo *et al.* [12] proposed a fuzzy-genetic algorithm as the foundation for automatic failure detection systems in aircraft. A fuzzy-based classifier is employed to estimate time of occurrence and the type of actuator failure. They demonstrated viability of the suggested approach, which combines the strengths of fuzzy reasoning and heuristic optimization, through various simulations.

Notable among the various model-free methods in the literature are the incremental models proposed by Pedrycz and Kwak [1]. This approach exploits the principle of model incrementality, in the sense that any model has to start with the simplest,

Manuscript received October 08, 2011; revised February 02, 2012; accepted May 18, 2012. Date of publication June 22, 2012; date of current version October 18, 2012. This research was supported in part by the Spanish Ministry of Science, and Innovation DPI2008-01978 COGNETCON and DPI2011-24447 C2MICRO Projects. Paper no. TII-11-623.

F. Penedo is with the C4Life Group, Escuela Politecnica Superior, Universidad Autonoma de Madrid, 28049 Madrid, Spain (e-mail: francisco.penedo@estudiante.uam.es).

R. Haber is with the C4Life Group, Escuela Politecnica Superior, Universidad Autonoma de Madrid, and also with Centre for Automation and Robotics, Spanish Council for Scientific Research (CSIC), 28500 Madrid, Spain (e-mail: rodolfo.haber@car.upm-csic.es).

A. Gajate and R. M. del Toro are with Centre for Automation and Robotics, Spanish Council for Scientific Research (CSIC), 28500 Madrid, Spain (e-mail: agustin.gajate@car.upm-csic.es; raul.deltoro@car.upm-csic.es).

Digital Object Identifier 10.1109/TII.2012.2205699

most generic (global) form imaginable. If necessary, it is then iteratively tuned by invoking some more refined (and localized) technique to model a particular region of the input space. The present work applies that principle to the development of a hybrid incremental model inspired in [13] and then applies the overall strategy to the detection of tool wear. Since the basic or global level has to be as simple as possible, linear regression techniques are clearly a viable solution because of their straightforward application. The counterpart technique we selected for the subsequent refinement of the basic model in the form of hybrid incremental models is the fuzzy k-nearest-neighbors smoothing algorithm [14].

Here, we apply hybrid incremental modeling to manufacturing technologies that are widely used in industrial processes throughout the world [5], in order to illustrate the benefits of the procedure. Among these technologies, machining processes are of particular importance, because of their relevance in such key sectors as the aeronautics, aerospace, and automotive industries. A machining process consists of various subprocesses: drilling (30%), turning (20%), milling (16%), thread-cutting (15%), engraving (6%), and others (13%) [8]. All these processes require expensive equipment and materials and skilled operators. Hence, a model that can efficiently describe the physical processes that occur within them would be essential for their improvement and optimization. In particular, it is very difficult to optimize productivity and minimize the risk of failures and machine and tool breakages without the aid of models.

For example, the complex characteristics of an electromechanical process such as turning places constraints on the use of conventional mathematical tools for its modeling. When conventional techniques fail to produce the expected results, because there is either no exact mathematical model, or it is excessively complex, hybrid modeling techniques can play an essential role. In particular, they can overcome certain limitations: imprecise exact knowledge of nonlinear aspects of a process and difficulties in the linear or quasi-linear representation of a nonlinear process.

The main contributions of this work are threefold. First, the design of a computational efficient algorithm, implemented on the basis of work by Roh *et al.* [13]. Second, the development of an algorithm that enables the exploration of model parameters, on the basis of an error criterion to achieve an optimal setting of the modeling parameters. This means that the choice of the most appropriate model parameters (order of the polynomial m, neighborhood size k, and fuzzy strength parameter p) may be enabled in a user-friendly environment. Finally, the third contribution of this paper is the application of hybrid incremental modeling to the manufacturing industry for tool wear monitoring in a turning process. This industrial case study is selected, because tool wear reduces the surface quality of a machined workpiece and increases the power consumption of the machine tool [15], [16].

The rest of this paper is organized as follows. Section II presents a brief description of hybrid incremental modeling. Section III describes its application to the modeling of tool wear in turning processes, and the results are compared with those of neurofuzzy models reported in the literature [17], [18]. Finally, Section IV presents the conclusions.

II. Hybrid Incremental Modeling

Hybrid incremental modeling (HIM) aims to provide an approximation of the behavior of a locally nonlinear system [13]. To that end, the strategy is to use a global model that captures the basic or general behavior of the system and then superimposes a local model on it that captures the local behavior. The idea arises from the concept of an incremental model as proposed by Pedrycz [1]: "*Adopting a construction of linear regression as a first-principle global model, refine it through a series of local fuzzy rules that capture the remaining and more localized nonlinearities of the system*."

A. Local Model

When no prior knowledge of the system is available, a generic model, such as a linear regression or second-order polynomials, can conveniently represent the global behavior of the system. If, however, some prior knowledge of the system is available, it might be possible to use a function that better matches the global behavior.

In our case, it was decided to use a generic strategy to obtain the global model by applying least squares to fit a polynomial of degree m. The output of the global model would therefore be as follows:

$$\hat{y}_B(x_i) = f_B\left(x_i, g(x_i)\right) \tag{1}$$

where x_i is the ith input data, and $g(x_i)$ is its output (target) value.

The procedure to obtain the global model consists of computing and storing the parameters of the fitting function (in this case the polynomial). The algorithm is evaluated at a query data q from the function with the parameters obtained during training.

B. Local Model

The development of the local model is based on the fuzzy k-nearest neighbors $(\mathrm{F}-k\mathrm{NN})$ approach. Compared to other techniques, $\mathrm{F}-k\mathrm{NN}$ is simple, easily interpretable and can achieve an acceptable accuracy rate [14], [19], [20]. The fuzzy version of k-NN averages the value of the points closest to the query point, on the assumption that points close to each other have similar values [21]. However, standard k-nearest neighbor methods place equal weights on all the selected neighbors, regardless of their distances from the query data [22]. In this work, the problem is partially addressed by data normalization. Learning in fuzzy k-NN is simple, in the sense that, at this stage, it is only necessary to store the known data. The set is the neighborhood of a query point q

$$N = \{d_i \in D\} \tag{2}$$

where, D is the set of input points for the algorithm, and d_i is one of the k-nearest neighbors of q.

The similarity between q and the points of N is given by

$$S(n_i,q)=\begin{cases} 1, & \text{if } \|n_i-q\|=0 \\ \left[\|n_i-q\|^{\frac{2}{p-1}} \cdot \sum_{j=1}^{k}\left(\frac{1}{\|n_j-q\|}\right)^{\frac{2}{p-1}}\right]^{-1}, & \text{if } \|n_i-q\|\neq 0 \end{cases} \tag{3}$$

where n_i is the ith neighbor of the query point q, and p is the fuzzy strength parameter.

The target value of query point q is now calculated as the mean of the target values of the points of the set N, weighted by the similarity S

$$\hat{g}(q) = \sum_{i=0}^{k} S(n_i, q) \cdot g(n_i). \tag{4}$$

C. Incremental Model

The incremental model integrates the two models described above. As previously made clear, $\mathrm{F}-k\mathrm{NN}$ locally adjusts the output value of the polynomial, as it does not capture the nonlinear localized characteristics of the system. The basic model is first trained by obtaining the coefficients of the polynomial of order m that best fit the data.

Thus, let $\hat{y}_B(x)$ be the function that is the output of the basic model. Then the prediction error of the basic model is

$$\varepsilon(x) = g(x) - \hat{y}_B(x). \tag{5}$$

These errors are the target values of the samples which are passed to the local model for training. It means that $\mathrm{F}-k\mathrm{NN}$ does not use original input-output data, but the errors resulting from the global modeling strategy. Thus, the local modeling done by $\mathrm{F}-k\mathrm{NN}$ is responsible for refining the global model output in regions with localized nonlinear behavior or where the polynomial can not properly represent the system. It is important to remark that neighborhood size k, and fuzzy strength parameter p are not necessary at this stage in the training, which considerably reduces the complexity of the modeling.

The incremental model evaluates a sample of data input q, by adding to the output of the basic model the compensation term calculated by the local model according to (4)

$$\hat{y}(q) = \hat{y}_B(q) + \hat{g}(q). \tag{6}$$

A theoretical example for illustrative purposes of hybrid incremental modeling is given below in Fig. 1. A theoretical function is generated from a straight line (linear region) and a Gaussian function, choosing $m = 1$. The residuals (dotted line) are the target values for training the local model. In the evaluation, the basic model generates an output (straight line) and the local model uses the knowledge of local errors taken from the polynomial to refine the output. For the sake of clarity, the estimated output of the HIM is depicted in grey.

The detailed algorithm that supports the design of the proposed hybrid incremental models is outlined as a sequence of the steps listed in Figs. 2 and 3. The first part trains the model on the target data (see Fig. 2), and the second evaluates the resulting model with the new data (see Fig. 3).

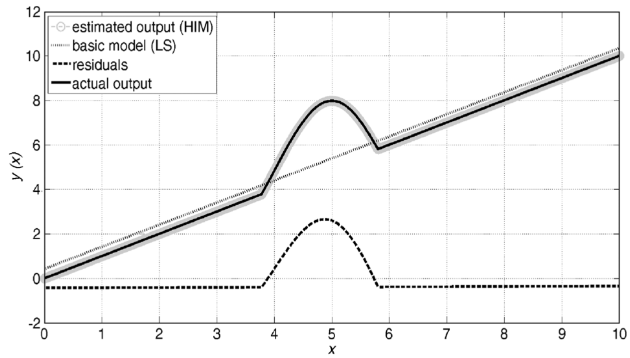

Fig. 1. Example application of HIM representing a theoretical function.

Training:

1. Obtain the training parameters:
   a) Basic global model parameters (m).
   b) Local model parameters (k and p).
   c) Input data and output values x_i and $g(x_i)$.
2. Normalization of x_i and $g(x_i)$ using (7).
3. Training the global model:
   a) Calculation of the coefficients of the polynomial.
4. Calculation of the errors using (5).
5. Training the local model using the errors:
   a) Memoriazation of input data and the corresponding errors.

Fig. 2. Steps in training the incremental model.

Evaluation:

1. Obtain the query points.
2. Normalization of the query points using (7).
3. Evaluation of the basic model at each query point
4. Evaluation of the local model at each query sample with (4).
5. Evaluation of the incremental model at each query sample with (6).
6. De-normalization of the data using the inverse of (7)

Fig. 3. Steps in evaluating the incremental model.

D. Data Normalization

The data variables are usually spread over very different ranges. This means that both the global and the local algorithms have to work under extreme conditions in which some of the input variables can be discarded, and weights are only given to variables with a broader domain. If the standard normalization (mean and standard deviation) is applied to the local model, then the selection of neighbors will discard variables with narrow ranges because of their negligible effect on the norm.

The performance of clustering algorithms is influenced by input-output data characteristics and data normalization or standardization is necessary in many real problems. There are many types of normalization, in order to scale data to fit in a specific range. For the sake of simplicity, only standard normalization is considered in this work.

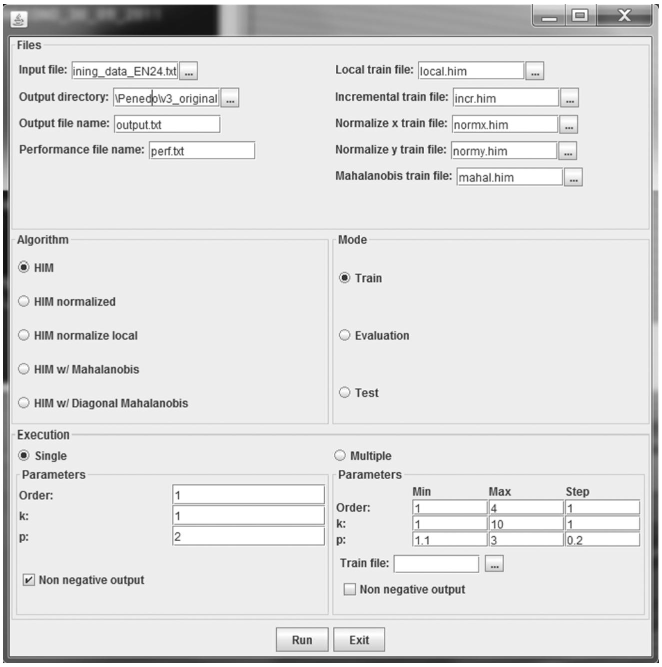

Fig. 4. User interface for aiding HIM development.

Denoting variable j of input point i as x_i^j yields the normalized points in the following form, also known as standard score or standard normalization:

$$z_i^j = \frac{x_i^j - \hat{\mu}(j)}{\hat{\sigma}(j)} \tag{7}$$

$$\hat{\mu}(j) = \frac{\sum\limits_{i=0}^{n} x_i^j}{n} \tag{8}$$

$$\hat{\sigma}(j) = \sqrt{\frac{\sum_{i=0}^{n}\left(x_i^j - \mu\right)^2}{n-1}} \tag{9}$$

where $\hat{\mu}(j)$ is the mean of the jth variable, $\hat{\sigma}(j)$ its standard deviation, and n the number of input points.

We used the C/Java programming language to implement the algorithm, and all the tests were performed on both Linux 2.6 kernel and Windows XP. The overall view of program interface is depicted in Fig. 4.

E. Algorithm Complexity

The complexity of the algorithm is related to the two algorithms that constitute the hybrid incremental training. With regard to the training step, let D be the number of training samples, V the number of evaluation samples and M the number of input variables. In the case of the above-mentioned algorithms (least squares and $\mathrm{F}-k\mathrm{NN}$), floating point operations $O(D^2 \cdot V \cdot m)$ (assuming an algorithm based on SVD) and the construction of the extended matrix $O(D \cdot V \cdot m)$ are used to consider the complexity of the HIM. $\mathrm{F}-k\mathrm{NN}$ simply stores information on the hard disk. The operations of the HIM consist of polynomial evaluation, which were implemented in the floating-point operations $O(D \cdot V \cdot m)$.

In the case of the evaluation step, the overall cost is the cost of evaluating the polynomial for the V samples plus the heavy computing load due to local model computing. In this implementation, for the sake of simplicity, we performed the calculation of neighborhood (N), sorting the data by proximity to the target. The complexity for each data set under evaluation is therefore $O(D \cdot \log(D))$ comparisons for sorting, where each comparison calculates two distances (B floating-point operations for each Euclidean distance). The computational cost of calculating the performance index or figure of merit should be included after obtaining N.

For example, considering an Intel (R) Core (TM) i7-2600 CPU@3.40 GHz, a study of the computing time is shown in Table I. These processing times (in seconds) are obtained with the polynomial degree $m = 2$, $M = 11$, and evaluation databases of different sizes.

TABLE I
COMPUTING TIME FOR TRAINING AND EVALUATION IN RELATION TO THE SIZE OF FILES

Size (MBytes)	Processing Time Training (s)	Processing Time Evaluation (s)
0.1	0.01	0.64
0.5	0.04	3.11
1.2	0.08	6.77
8.2	1.16	47.25

III. EXPERIMENTAL STUDIES

We applied our technique to a real problem of vital importance in the world of industrial machining: tool wear monitoring in turning processes. The hybrid incremental modeling strategy was then compared with other techniques used to deal with problems of this type in order to demonstrate its benefits.

A. Tool Wear Monitoring

Tool wear monitoring is a crucially important factor in turning processes. The accuracy of tool wear monitoring is a key issue with high economic costs in industrial sectors. We address this problem through the hybrid incremental model that will serve to improve too wear monitoring in turning operations. The models used in the comparative study, are created from a series of input-output data. The inputs to the model are time (t), cutting force (F_z), tool vibrations (acceleration, a_t), and the process's acoustic emission signals (*AES*). The output is tool wear (T'_w, wear at the flank). The data set used to create the models and the other data used to test them were obtained from an experimental platform described in [18]

$$T'_w = H(t, F_z, a_t, \mathrm{AES}). \tag{10}$$

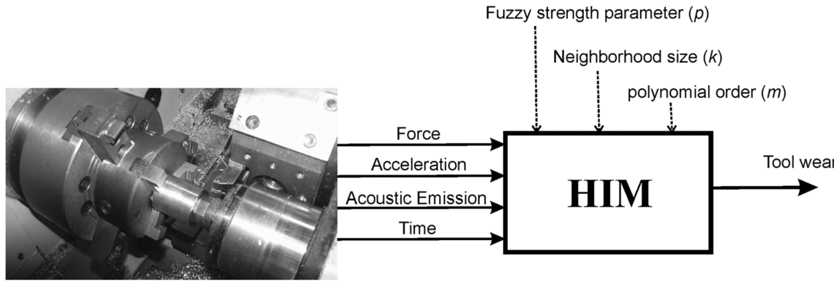

Fig. 5. Overall system using the black-box approach.

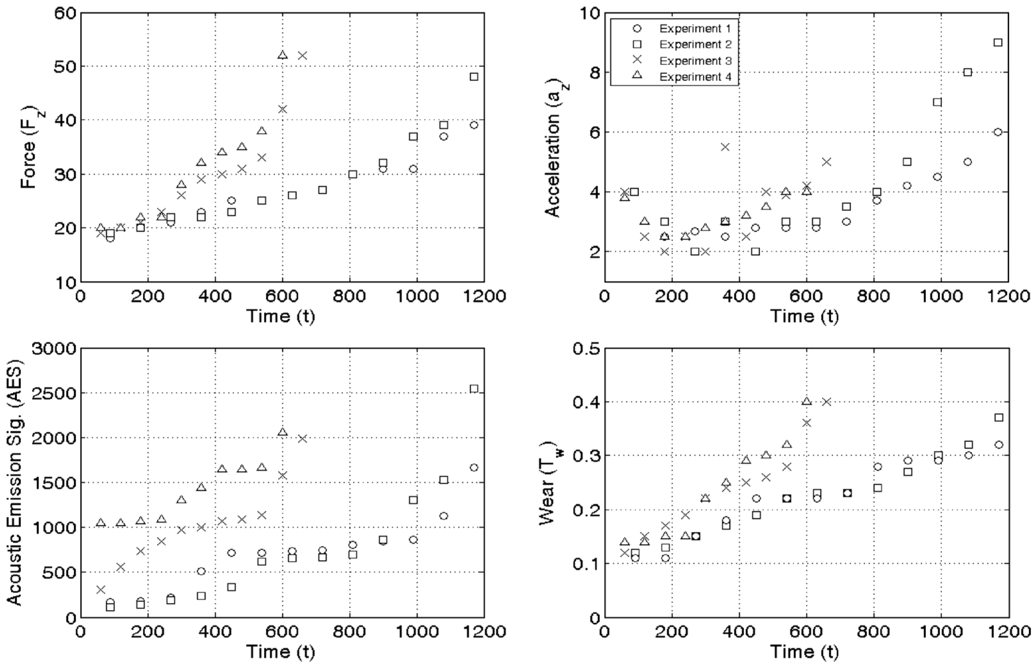

Fig. 6. Experimental data for modeling tool wear.

The overall view of the system using the black box approach is depicted in Fig. 5. The diagram includes the above-mentioned inputs and output and also the parameters of the hybrid incremental model. For every operation carried out with the lathe, the acoustic emission signals (number of pulses), tool vibration (acceleration), cutting force, process time, and information on the state of the cutting tool (tool wear data) are logged, processed, and stored on a PC. The measurement devices are a dynamometer, an accelerometer, an acoustic emission sensor, and a microscope to measure tool wear.

The data used to create the incremental model were obtained from turning operations on workpieces of two different materials: cast iron (gray cast iron FG15) and a steel alloy (EN24), in order to test the validity of the tool wear model for different materials. A Widia CCMT 060204 TTS uncoated carbide insert tool was used for both materials.

Four experiments were performed on each material. The same cutting depth, 0.7 mm, was used for all operations on both materials. The tool wear, in particular, the flank wear, was measured offline under a microscope. Table II summarizes the experimental condition for each run.

Only the FG15 experimental data are depicted in Fig. 6, for the sake of clarity and conciseness.

TABLE II
EXPERIMENTAL CONDITIONS FOR EACH RUN

Run	Material	Cutting Speed (m/min)	Cutting feed (mm/rev)
1	FG15	94	0.06
2	FG15	94	0.08
3	FG15	188	0.06
4	FG15	188	0.08
5	EN24	94	0.06
6	EN24	94	0.08
7	EN24	188	0.06
8	EN24	188	0.08

The figure-of-merit or performance index used as an total average error (*TAE*), which gives an idea of overall model behavior

$$TAE = \frac{1}{n}\sum_{N}\frac{|T_w - T'_w|}{T_w}\cdot 100 \qquad (11)$$

TABLE III
MAIN PARAMETERS OF EACH MODELING STRATEGY CONSIDERED IN THIS STUDY

Algorithm	ANFIS [18]	TWNFIS [17]	HIM
Model	MISO	MISO	MISO
Clustering	Substractive	QT_{Clust}	Fuzzy k-NN
Structure	Fixed	Variable	Fixed
Inference	Global	Local	Global/Local
Main parameters to adjust	- Number of MFs - Type of MFs - Iterations of BP - Learning rate - Error tolerance	- Neighborhood size - Type of MFs - Clustering threshold - Minimum number of data in a cluster - Iterations of BP - Learning rate - Error tolerance	- Neighborhood size - Fuzzy strength - Polynomial order
Training algorithms	Backpropagation + LSE	Backpropagation	LSE + IBL
Training data set (FG 15 / EN24)	24 / 36 samples	24 / 36 samples	24 / 36 samples

BP= Backpropagation. MF=Membership Function. LSE=Least Square Error. IBL= Instance Based Learning.

where, T_w is the actual tool wear (measured by microscopy), T'_w is the wear obtained through the model, and N is the number of data points in each experiment.

This error-based performance index is selected because it is widely applied in industry for assessing the actual performance of manufacturing systems. Moreover, this figure of merit has been applied in previous case studies [17], [18] evaluating the performance of other modeling strategies.

However, on occasions there is a need to examine the local behavior of the model. For this purpose, we selected as a second figure-of-merit (accuracy index) the number of data points in each experiment with an (individual) average error (AE) higher than 10%. This figure was chosen because, at an industrial level (especially in a context of process monitoring), certain margins of error are acceptable due to noise in the signals and sensor inaccuracy. Accordingly, errors of below 10% are usually more or less acceptable (on a case-by-case basis), whereas above this level the information has to be regarded with caution.

The resulting tool wear data were modeled using the suggested HIM procedure. First, the most appropriate set of parameters (m, k, p) was estimated by exploring the region $m \in [1, 6]$, $k \in [1, 5]$, $p \in [1, 3]$ where m, $k \in$ Z; $p \in \mathbf{R}$. It means that six integer values were considered for m, five integer values for k, and thirty real values (step 0.1) were considered for p. Finally, nine hundred combinations were verified to find the most appropriate set of parameters (m, k, p). As shown in Section II-D, the processing time was very short. However, an optimization strategy based on simulated annealing is under design to improve the efficiency of the algorithm [23].

The minimum $TAE = 2.48\%$ was obtained for $(m, k, p) = (2, 1, 1.2)$ and for FG15 (average value for run 1 to run 4). The minimum TAE for EN24 was $(m, k, p) = (3, 2, 1.8)$, which produced an error of 3.69% (average value for run 5 to run 8). For the sake of computational simplicity the set of parameters $(m, k, p) = (1, 1, 2)$ was selected. This implies a first order polynomial (linear), only one neighbor for the neighborhood size, and 2 for the fuzzy strength parameter.

A comparative study with an adaptive neural-fuzzy inference system (ANFIS) [18] and a transductive-weighted neuro-fuzzy inference system (TWNFIS) [17], [24] was conducted, in order to evaluate the performance of the proposed strategy for modeling tool wear. The reader can find further details on the topology and training parameters of the neurofuzzy models in [17]. The ANFIS parameters are: seven (FG15) and ten (EN24) membership functions, three iterations in the backpropagation training algorithm with a 0.001 learning rate and zero error tolerance. TWNFIS parameters are basically four neighbors (FG15 and EN24) in the algorithm, a clustering threshold value of 0.32 (maximum diameter), the minimum number of elements for the clustering algorithm is 1 and the training algorithm uses the same parameters as ANFIS. Table III summarizes the main parameters and characteristics of the three methods considered in this study.

The results obtained by the three models considered in this study (ANFIS, TWNFIS and HIM) are shown in Table IV. HIM outperformed both ANFIS and TWNFIS. TWNFIS yielded slightly better accuracy than HIM in only three runs (3, 5, and 7). Moreover, the overall TAE (3.36%) of the HIM-based model is less than the average TAE of the TWNFIS-based model (4.55%). Therefore, HIM improves the TAE by about 25% with regard to TWNFIS and reduces the TAE of ANFIS threefold. Likewise, tool wear estimation T'_w when higher than

TABLE IV
COMPARATIVE STUDY OF THE THREE MODELS (ANFIS, TWNFIS, AND HIM)

Run	ANFIS [18]	*TAE* > 10%	TWNFIS [17]	*TAE* > 10%	HIM	*TAE* > 10%
1	7.12	3	4.27	1	3.64	2
2	40.40	4	5.04	3	3.36	1
3	3.46	1	2.55	3	2.78	1
4	1.97	0	5.71	0	1.76	1
Average value	**13.24**		**4.39**		**2.84**	
5	3.20	1	4.62	2	5.84	1
6	3.99	2	5.70	1	2.15	1
7	10.06	2	2.00	0	5.30	2
8	10.50	3	6.50	3	2.22	2
Average value	**6.94**		**4.71**		**3.88**	

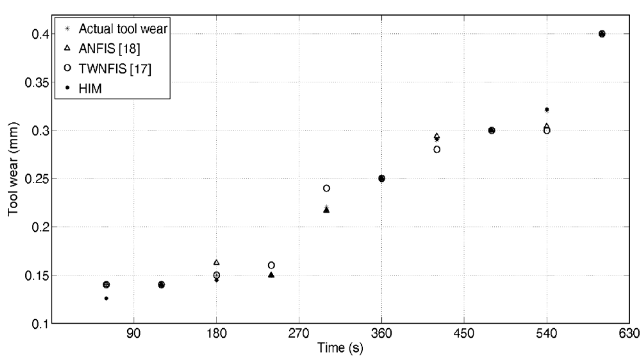

Fig. 7. Measured and estimated tool wear for the three models in experiment 4.

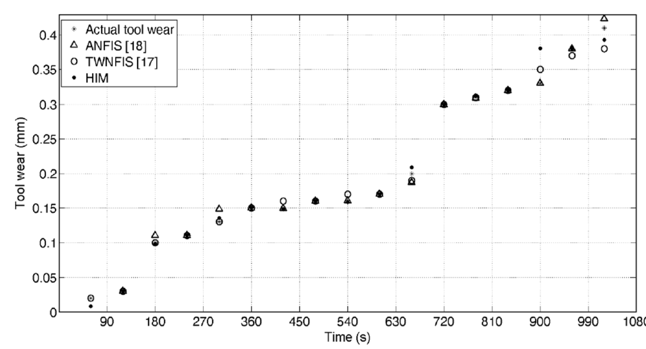

Fig. 8. Measured and estimated tool wear for the three models in experiment 7.

a threshold value $TAE > 10\%$ was also recorded and analyzed. The number of data points surpassing the threshold was higher in the TWNFIS-based model than in the HIM-based model. It is important to remark that the main advantage of HIM with regard to ANFIS and TWNFIS is that training may be done without assuming two different material properties (FG15 and EN24). This difference meant that the results of ANFIS and TWNFIS were achieved for each material separately, training and validating each material independently.

Two experiments are shown in Figs. 7 and 8, in order to visually assess the behavior of the model. The best result obtained with HIM, corresponding to experiment 4, is depicted in Fig. 7. The worst result obtained by HIM in experiment 7 is shown in Fig. 8.

IV. CONCLUSION

In order to improve the efficiency of modeling for process monitoring, a hybrid incremental modeling strategy has been designed and implemented. The hybrid incremental model has been applied to detect tool wear in turning processes.

The procedure for designing and implementing hybrid incremental models is simple and computationally efficient. Our study has demonstrated how we can capture the essential characteristics of processes with only a few parameters and a basic configuration (first order polynomial, one neighbor and the easiest fuzzy strength parameter). Moreover, the determination of the parameters for the hybrid incremental model (order of the polynomial, number of neighbors and fuzzy strength parameter) has shown how obtain computationally efficient models may be obtained with few neighbors and a very low-order polynomial.

Finally, the comparative study with other soft-computing techniques has demonstrated the potential of hybrid incremental modeling. The hybrid incremental model provides better accuracy than a transductive neurofuzzy model and an inductive neuro-fuzzy model. Likewise, the hybrid incremental model also provides better error-based performance indices for detecting tool wear than above-mentioned neuro-fuzzy models.

REFERENCES

[1] W. Pedrycz and K. C. Kwak, "The development of incremental models," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 3, pp. 507–518, Sep. 2007.

[2] P. E. P. Odiowei and Y. Cao, "Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations," *IEEE Trans. Ind. Inf.*, vol. 6, no. 1, pp. 36–45, Feb. 2010.

[3] Y. Zhang, H. Zhou, S. J. Qin, and T. Chai, "Decentralized fault diagnosis of large-scale processes using multiblock kernel partial least squares," *IEEE Trans. Ind. Inf.*, vol. 6, no. 1, pp. 3–10, Feb. 2010.

[4] R. H. Abiyev, O. Kaynak, T. Alshanableh, and F. Mamedov, "A type-2 neuro-fuzzy system based on clustering and gradient techniques applied to system identification and channel equalization," *Appl. Soft Comput. J.*, vol. 11, pp. 1396–1406, 2011.

[5] D. Tanikic, M. Manic, G. Radenkovic, and D. Mancic, "Metal cutting process parameters modeling: An artificial intelligence approach," *J. Sci. Ind. Res.*, vol. 68, pp. 530–539, 2009.

[6] S. Yu, X. Yu, and Z. Man, "A fuzzy neural network approximator with fast terminal sliding mode and its applications," *Fuzzy Sets Syst.*, vol. 148, pp. 469–486, 2004.

[7] Q. Wu, "Car assembly line fault diagnosis model based on triangular fuzzy Gaussian wavelet kernel support vector classifier machine and genetic algorithm," *Expert Syst. Appl.*, vol. 38, pp. 14812–14818, 2011.

[8] J. V. Abellan-Nebot and F. R. Subirón, "A review of machining monitoring systems based on artificial intelligence process models," *Int. J. Adv. Manufact. Technol.*, vol. 47, pp. 237–257, 2010.

[9] R. E. Haber, R. H. Haber, J. R. Alique, and S. Ros, "Application of knowledge-based systems for supervision and control of machining processes," in *Handbook of Software Engineering and Knowledge Engineering*. Singapore: World Scientific, 2002, vol. 2, pp. 673–710.

[10] D. P. Filev, R. B. Chinnam, F. Tseng, and P. Baruah, "An industrial strength novelty detection framework for autonomous equipment monitoring and diagnostics," *IEEE Trans. Ind. Inf.*, vol. 6, no. 4, pp. 767–779, Nov. 2010.

[11] J. J. Verbeek, N. Vlassis, and B. Kröse, "Efficient greedy learning of gaussian mixture models," *Neural Comput.*, vol. 15, pp. 469–485, 2003.

[12] C. H. Lo, E. H. K. Fung, and Y. K. Wong, "Intelligent automatic fault detection for actuator failures in aircraft," *IEEE Trans. Ind. Inf.*, vol. 5, no. 1, pp. 50–55, Feb. 2009.

[13] S. B. Roh, T. C. Ahn, and W. Pedrycz, "The refinement of models with the aid of the fuzzy k-nearest neighbors approach," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 3, pp. 604–615, Mar. 2010.

[14] C. B. Cheng and E. S. Lee, "Nonparametric fuzzy regression—k-NN and kernel smoothing techniques," *Comput. Math. Appl.*, vol. 38, pp. 239–251, 1999.

[15] K. Zhu, S. H. Geok, and S. W. Yoke, "Multiscale singularity analysis of cutting forces for micromilling tool-wear monitoring," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2512–2521, Jun. 2011.

[16] Z. Jun-Hong, P. C. Khiang, F. L. Lewis, and Z. Zhao-Wei, "Intelligent diagnosis and prognosis of tool wear using dominant feature identification," *IEEE Trans. Ind. Inf.*, vol. 5, no. 4, pp. 454–464, Nov. 2009.

[17] A. Gajate, R. Haber, R. del Toro, P. Vega, and A. Bustillo, "Tool wear monitoring using neuro-fuzzy techniques: A comparative study in a turning process," *J. Intell. Manufact.*, In Press (Online First, 2 August 2010).

[18] V. S. Sharma, S. K. Sharma, and A. K. Sharma, "Cutting tool wear estimation for turning," *J. Intell. Manufact.*, vol. 19, no. 1, pp. 99–108, 2008.

[19] J. M. Keller, M. R. Gray, and J. A. Givens, "Fuzzy k-nearest neighbor algorithm," *IEEE Trans. Syst., Man, Cybern.*, vol. 15, no. 4, pp. 580–585, Apr. 1985.

[20] Z. Li, X. Yan, C. Yuan, and L. Li, "Gear multi-faults diagnosis of a rotating machinery based on independent component analysis and fuzzy k-nearest neighbor," *Adv. Mater. Res.*, vol. 108–111, pp. 1033–1038, 2010.

[21] K. Hattori and M. Takahashi, "A new nearest-neighbor rule in the pattern classification problem," *Pattern Recognit.*, vol. 32, pp. 425–432, 1999.

[22] P. Tian-Hong, S. Bi-Qi, D. S. H. Wong, and J. Shi-Shang, "A virtual metrology system for predicting end-of-line electrical properties using a MANCOVA model with tools clustering," *IEEE Trans. Ind. Inf.*, vol. 7, no. 2, pp. 187–195, May 2011.

[23] R. E. Haber, R. Haber-Haber, A. Jiménez, and R. Galán, "An optimal fuzzy control system in a network environment based on simulated annealing. An application to a drilling process," *Appl. Soft Comput.*, vol. 9, pp. 889–895, 2009.

[24] A. Gajate, R. E. Haber, P. I. Vega, and J. R. Alique, "A transductive neuro-fuzzy controller: Application to a drilling process," *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1158–1167, Jul. 2010.

Francisco Penedo was born in Carballiño Galicia, Spain, in 1991. He is currently working toward the B.Sc. degree in computer science and mathematics at the Autonomous University of Madrid, Madrid, Spain.

In 2010, he joined the research group C4LIFE at Escuela Politécnica Superior. His main interests are intelligent systems, self-adaptive systems, and self-organized, advanced strategies of software engineering.

Rodolfo E. Haber (M'11) received the Ph.D. degree in industrial engineering from the Universidad Politécnica de Madrid, Madrid, Spain, in 1999.

He has been with the Centre of Automation and Robotics-CAR, CSIC-UPM (formerly Institute for Industrial Automation-IAI) since 1996, and has actively participated in a number of National, European, and International RTD projects in the area of control systems. In 1999, he also joined the Computer Engineering Department at the Universidad Autónoma de Madrid as an Adjunct Professor. He has published over 100 technical papers in the areas of the application of intelligent systems and complex processes. He has taught several seminars on the same theme in the Georgia Institute of Technology and University of Central Florida. His research interests include networked and embedded intelligent control systems, artificial cognitive control systems, modeling, control, and supervision of complex processes.

Dr. Haber was an IPC Member for several IEEE and IFAC conferences from 2000 to 2008. Since 2002 he has belonged to IFAC's Technical Committee 3.1 *Computers for Control*. Since 2005, he has been Member of the ASME Technical Committee for *Model Identification and Intelligent Systems*. Since 2003, he has served as expert evaluator for the European Commission in the 6th and 7th Framework Programme. Since 2007, he has been an Associate Editor of the *Transactions on Computational Science* and the *International Journal of Mechatronics and Manufacturing Systems*.

Agustín Gajate received the B.Eng. degree in industrial engineering, the M.S. degree in intelligent systems, and the Ph.D. degree in informatics and automation from the Universidad de Salamanca, Salamanca, Spain, in 2006, 2009, and 2010, respectively.

In 2007, he joined with the Spanish Council for Scientific Research (CSIC), Madrid, Spain, where he is currently a Researcher with the Centro de Automática y Robótica (CSIC-UPM) working on several research and development projects. His research interests include automation of machining process, and intelligent control with an emphasis in fuzzy and neurofuzzy systems.

Raúl M. del Toro received the B.Eng. degree in automation engineering from the Universidad de Oriente, Santiago de Cuba, in 1997, the M.Sc. degree in automation and robotics from the Polytechnic University of Madrid, Madrid, Spain, in 2009, and the Ph.D degree in computer science and telecommunication engineering from the Universidad Autónoma de Madrid, Madrid, Spain, in 2011.

In 2006, he joined to the Spanish Council for Scientific Research (CSIC), where he is currently a Researcher at the Centre of Automation and Robotics-CAR (formerly Institute for Industrial Automation) working on several research and development projects. His research interests include intelligent control systems, modelling, control, and monitoring of complex electromechanical processes.