Neural Intelligent Control for a Steel Plant

Gérard Bloch, Frank Sirou, Vincent Eustache, Philippe Fatrez

Neural Intelligent Control for a Steel Plant

Gérard Bloch, Frank Sirou, Vincent Eustache, and Philippe Fatrez

***Abstract*—The improvement of the performances of a complex production process such as the Sollac hot dip galvanizing line of Florange (France) needs to integrate various approaches, including quality monitoring, diagnosis, control, optimization methods, etc. These techniques can be grouped under the term of intelligent control and aim to enhance the operating of the process as well as the quality of delivered products. The first section briefly describes the plant concerned and presents the objectives of the study. These objectives are mainly reached by incorporating the skill of the operators in neural models, at different levels of control. In Section II, the low-level supervision of measurements and operating conditions are briey presented. The control of the coating process, highly nonlinear, is divided in two parts. In Section III, the optimal thermal cycle of alloying is determined using a radial basis function neural network, from a static database built up from recorded measurements. The learning of the weights is carried out from the results of a fuzzy C-means clustering algorithm. In Section IV, the control of the annealing furnace, the most important equipment, is achieved by mixing a static inverse model of the furnace based on a feedforward multilayer perceptron and a regulation loop. Robust learning criteria are used to tackle possible outliers in the database. The neural network is then pruned in order to enhance the generalization capabilities.**

***Index Terms*—Intelligent control, fault diagnosis, galvannealing, neural network, modeling, steel industry.**

I. Introduction

Worldwide use of metallic coated steel sheet in the car industry is continually increasing. Particularly, the market evolves toward the use of zinc iron alloy coating. These galvannealed sheets have excellent corrosion resistance and the surface quality is high after painting. As the production cost is moderate, these galvannealed sheets become of major interest for the car industry [1], [2].

This paper presents the works led by the French steel company Sollac and the Research Centre for Automatic Control of Nancy (CRAN) to improve the quantity and the quality of the galvannealed sheet produced on the galvanising line of Florange, France. The industrial use of the proposed techniques should be mentioned. The low-level supervision of measurements and operating conditions is currently used. The thermal cycle optimization and the annealing furnace control can be considered in β test.

A. The Galvannealing Process

In January 1990, Sollac started up a new hot dip galvanizing line. This line was designed for the production of galvanized steel sheets for the outside panels of automobiles with optimum surface quality. The line, which is 500 m long, is equipped with about 5000 sensors and produces 300 000 tons/yr of galvanized steel sheet, at a speed of 80 to 120 m/min. A layout of the galvannealing section of this plant is shown in Fig. 1, including the galvannealing furnace, the soaking furnace, and the air cooling section. The steel sheet, exiting the annealing furnace, is dipped into a bath of molten zinc allowing the zinc coating. The thickness of this coating is controlled by an air knives wiping system. The coated strip is then annealed to permit the diffusion of strip iron in the coating: the strip is reheated with an induction furnace until reaching a set point known as the induction temperature ($\theta_{induction}$). Next it goes through a soaking furnace, whose inner temperature is called holding temperature ($\theta_{holding}$) and then moves into the cooling section to stop the galvannealing reaction.

The quality of the product is related to the percentage of iron at the surface. An under-alloyed product (lack of iron in the coating) is caused by an insufficient alloying temperature while an over-alloyed product (excess iron in the coating) is obtained when the thermal cycle is too high. The control of the alloying is based on $\theta_{induction}$. The problem is to determine and to control the optimal temperature at this point, knowing the operating conditions which are the speed, width and thickness of the strip, and the thickness of the coating by acting on the total heating power applied to the furnace.

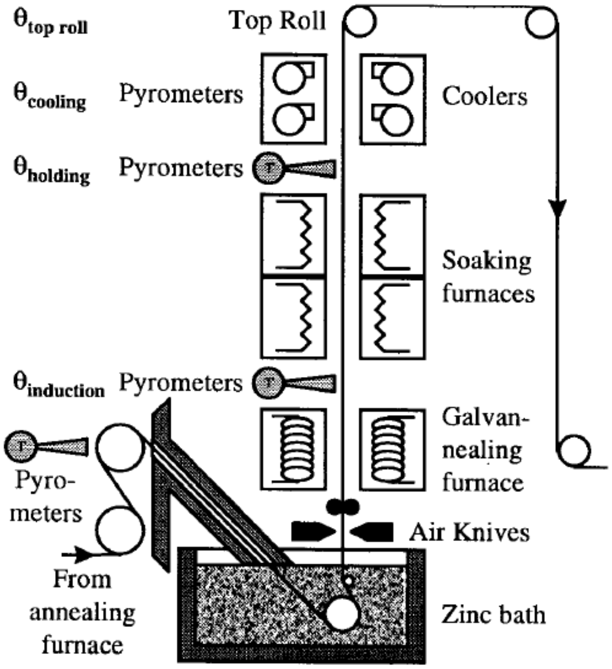

Fig. 1. The galvannealing section.

G. Bloch, F. Sirou, and V. Eustache are with the Centre de Recherche en Automatique de Nancy (CNRS URA 821), CRAN-ESSTIN, F-54500 Vandœuvre, France.

P. Fatrez is with the SOLLAC Ligne de galvanization Sainte Agathe, F-57191 Florange cedex, France.

B. Intelligent Control Approach

This brief description highlights some features currently found in steel industry:

- numerous and interconnected describing variables;
- complex physical phenomena, only partially known in an industrial production context;
- nonlinear relationships;
- importance of the skill of the operators.

As pointed out by Harris [3], many processes, too complex for direct modeling based on physical laws, are manually controlled by human operators before automatic controls are installed. The plant operator is able to cope with plant nonlinearities and slowly varying parameters. He is also able to respond to complex sets of noisy observations and poorly specified constraints and to satisfy multiple subjective-based performance criteria.

Thus, one of the basic ideas of the presented study, which can be termed an intelligent control application, is to incorporate the flexible and creative attributes of human controllers, while avoiding their associated characteristics of unreliability.

The global objective is to improve the overall equipment effectiveness (OEE). This indicator quantifies the proportion of the installed production capacity which is actually used to supply products of required quality. In the spirit of total productive maintenance (TPM) [4], the efficiency of a plant can be dened with respect to the different types and causes of losses: by stopping (for failures, tools changes, preparations, and setting), by slowing (light running, micro-failures, production rate lowering) and by product quality defects. The overall equipment effectiveness can be thus increased by analyzing the sources of losses and classifying them with respect to their economical impact. The means to eliminate or reduce these causes are then related to the hierarchical functional breaking up of the plant (computer integrated production, or CIP) given Fig. 2.

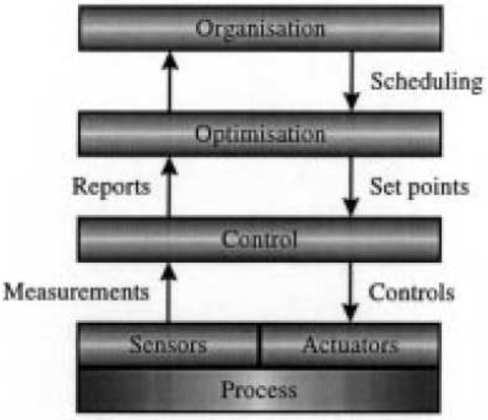

Fig. 2. CIP breaking up of industrial plant.

Leaving on one side the production management and planning aspects, the improvement of the galvannealing process control must tackle the three levels of the breaking up mentioned above: sensors, control, and optimization. The intelligent control approach must improve the performances over a wide range of operating conditions. It must also increase the fault tolerance and recongurability degree of the plant. Another important aspect is to reduce the design cost of the control procedures.

This paper is organized as follows. Section II deals with the monitoring of measurements and operating conditions of the alloying process. This supervision is based on redundancy of sensors and on analytical models. It uses detection tests of changes of mean in the residuals generated from measured signals.

Section III presents the optimization of the thermal cycle of the alloying process. A static database is first built up from recorded measurements. The operating points which have led to a good quality of the product are used to determine the optimal alloying thermal cycle. This estimation is achieved by using a radial basis function (RBF) neural network, which predicts, from the operating conditions and the features of the steel sheet, the thermal energy required for correct alloying. The centers of the RBF network are obtained with a fuzzy C-mean (FCM) algorithm. This neural model gives the optimal temperatures which will be used as set points to control the alloying process.

Described in Section IV, this control is based on the steady-state modeling of the induction furnace used in the line for galvannealing. Its nonlinear behavior is modeled by a feedforward multilayer perceptron. Robust learning criteria are used to take into account possible outliers in the database. The neural network obtained is then pruned to reduce the number of connections and enhance the generalization capabilities. The resulting model is finally used in association with a control loop to regulate, on-line, the temperature at the exit of the furnace.

II. Supervision of Measurements and Operating Conditions

The easiest way to validate the measurement of a physical quantity is to multiply the sensors (material redundancy) or the estimations (analytical redundancy) of this variable. The redundancy principle is then to compare these different measurements, direct or indirect, of the variable. Several approaches have been proposed, but it is possible to find a common structure represented at Fig. 3 [5], [6]. A detector/estimator takes into account information from sensors or models to validate the measured variable and to give a warning signal if a fault is detected.

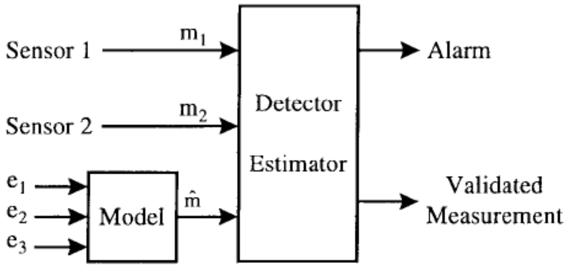

Fig. 3. General principle of measurement validation.

The detection procedure presents three main steps: measurement filtering, residual generation, residual evaluation. The *measurement filtering*, not always necessary, can be used to minimize the noise effect or to reject outliers. The *residual generation* consists, for material redundancy, of a simple difference between the filtered measurements. This residual without faults is assumed to be zero mean and with constant variance. When a failure occurs, the statistical properties of the residual are modified by a jump of mean, a change of variance, a modification of the statistical distribution or a variation of the spectral properties. The *residual evaluation* allows to detect this change. The nature of the statistical test used depends on how the failure modifies the residual.

In the present case, the possible faults transform the residuals by a jump of mean which is easily detected by the commonly used Page-Hinkley test.

A. Page-Hinkley Test

This test, proceeding from the works of Page [7] and Hinkley [8], has been amply studied [9], [10]. Only the recursive forms of the test are presented here. For a positive jump in the mean

$$U_0 = 0, \quad U_n = U_{n-1} + (x_n - \mu_0 - \frac{\delta_m}{2}), \quad n \geq 1, \quad \text{(1a)}$$

$$m_0 = 0, \quad m_n = \min(m_{n-1}, U_n), \quad n \geq 1, \quad \text{(1b)}$$

$$U_n - m_n \geq \lambda, \quad \text{(1c)}$$

and for a negative jump

$$T_0 = 0, \quad T_n = T_{n-1} + (x_n - \mu_0 + \frac{\delta_m}{2}), \quad n \geq 1, \quad \text{(2a)}$$

$$M_0 = 0, \quad M_n = \max(M_{n-1}, T_n), \quad n \geq 1, \quad \text{(2b)}$$

$$M_n - T_n \geq \lambda, \quad \text{(2c)}$$

where x_n is the nth sample of the signal to test, δ_m the minimum magnitude of the jump to detect, and λ the detection threshold. The initial mean μ_0 (before jump) is usually considered as zero in lack of fault.

Nevertheless, in the present industrial context, the residuals differ from the theoretical hypotheses. An example of residual is presented Fig. 4. This residual is not centered around zero before the jump and the mean μ_0 must be estimated during an initialization period of the algorithm. The example of residual includes normal fluctuations caused by the coming of a new roll of steel sheet (around 15 mn), but also includes an increase of mean after fluctuations. For the test, these fluctuations are considered as outliers.

The robustness to these outliers can be reached by two ways: before the test by filtering the rough signals; by using a robust form of the algorithm.

B. Prefiltering

The problem is to reduce the noise influence and above all to reject outliers. The median filter which is selected is nonparametric, does not use *a priori* considerations on the noise, and is more robust than the mean filter [11].

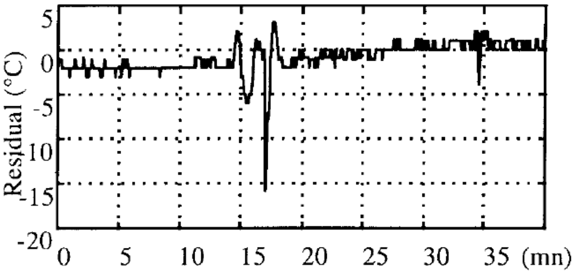

Fig. 4. Example of residual for $\theta_{induction}$.

On a moving window of size $2m + 1$ points, the signal values to be filtered are sorted in ascending order and denoted Ys_{n+j}. The expression of the median filter is then

$$Yf_n = \frac{1}{(2m+1) - r - s} \sum_{j=-m+r}^{m-s} Ys_{n+j}, \quad n > m. \quad \text{(3)}$$

In this filter parameterized by r and s, the r smallest values and the s greatest values are excluded of the moving mean calculation. When the r and s values go from zero to m, the filter moves from a mean filter to a median filter with the robustness properties of the median with respect to the mean.

C. Robust Page-Hinkley Test

The idea of this test, proposed by Huber [12], is to limit the outlier magnitude in order to reduce their influence. In the cumulative sum U_n (respectively, T_n), the contribution of the current value x_n is limited to the values d_1 and d_0 with $d_1 \leq 0 \leq d_0$. The recursive form of this robust test is then for a positive jump

$$U_0 = 0, \, U_n = U_{n-1} + \min\left(d_0, \max(d_1, x_n - \mu_0 - \frac{\delta_m}{2})\right), n \geq 1, \quad \text{(4a)}$$

$$m_0 = 0, \, m_n = \min(m_{n-1}, U_n), n \geq 1, \quad \text{(4b)}$$

$$U_n - m_n \geq \lambda, \quad \text{(4c)}$$

and for a negative jump

$$T_0 = 0, \, T_n = T_{n-1} - \min\left(d_0, \max(d_1, x_n - \mu_0 + \frac{\delta_m}{2})\right), n \geq 1, \quad \text{(5a)}$$

$$M_0 = 0, \, M_n = \max(M_{n-1}, T_n), n \geq, 1 \quad \text{(5b)}$$

$$M_n - T_n \geq \lambda. \quad \text{(5c)}$$

These three possibilities, standard, prefiltered, and robust Page-Hinkley tests are presented and tested in [13] and [14]. These monitoring tests are implemented on the plant supervision system. They provide the operators with an on-line validation of the pyrometer temperatures, the cooling pressures, and the operating conditions of the induction and soaking furnaces, and with fault detection of sensors. These modules improve the fault tolerance of the measurement systems at low level.

III. Optimization of the Alloying Thermal Cycle

After an on-line validation of the most important measurements and processes, the following step of the study concerns the improvement of the product quality, which needs the determination of the optimal set points of the thermal cycle, particularly $\theta_{induction}$, for different line speeds and product types.

To obtain a quality coating, with about 10% of iron, an optimal energy must be supplied to the coated strip. The energy (E) useful to the alloying reaction is given by means of an induction furnace and a soaking zone which heat the strip at the $\theta_{induction}$ and $\theta_{holding}$ temperatures, respectively (see Fig. 5).

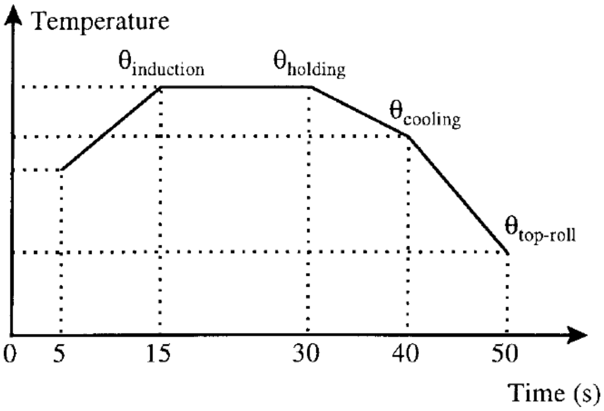

Fig. 5. Thermal cycle.

The metallurgy knowledge is not sufficient to explain and to know the optimal temperature to initiate the alloying reaction. The complexity of the reaction and the numerous nonlinear relationships between all variables lead to the use of learning algorithms from the operating points fixed by the control operators during several months.

The idea is first to model the energy supplied to the strip with respect to the features of the strip and to the line speed. Second, from this estimated energy, the induction temperature is calculated by imposing particular conditions on the thermal cycle. Fig. 6 summarizes the estimation of the $\theta_{induction}$ temperature.

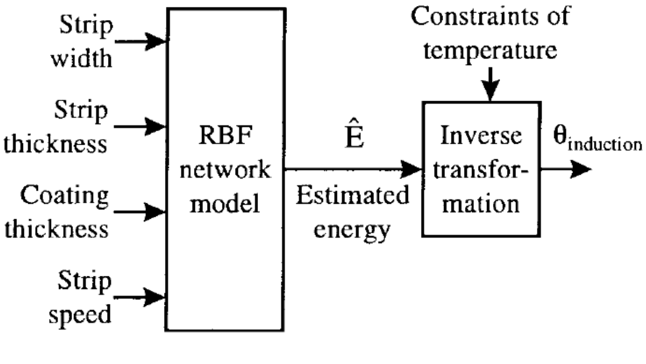

Fig. 6. Estimation of the induction temperature.

Before explaining the determination of the induction temperature from the estimated energy, the modeling of the energy supplied to the strip is described below.

Model learning requires a steady-state database. From dynamic measurements, operating points are extracted to generate this database, following the method described in [15]. Each operating point is validated as good or bad with respect to the quality of the product and only good points are kept. The criteria for quality which are used are the strip aspect, evaluated by a surface finish inspector, and the results of an off-line mechanical test of the coating adhesion (“powdering”). The remaining steady states are then separated in two sets, a modeling one of 260 points and a test one of 130 points. The variables included in the data sets are all the temperatures of the galvannealing section, the speed of the line and the features of the strip. A “measured” energy is also included. This measured energy is actually calculated from the temperatures of the thermal cycle. This energy is then used as target for the model learning, as presented in Fig. 7.

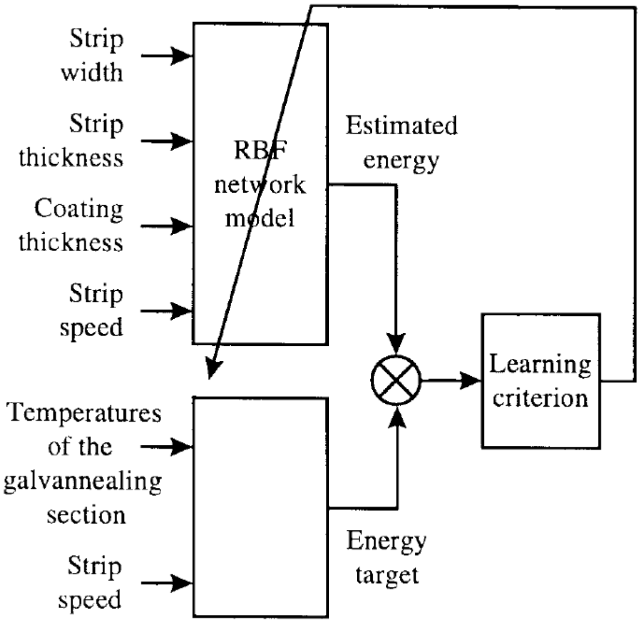

Fig. 7. Model learning.

The process typically operates around a finite number of operating points corresponding to the different strip formats and speeds. So, the most suitable neural model is an RBF network.

The network is composed of three layers. The input variables of the network are the features of the strip and the line speed. The hidden layer is composed with n_1 neurons whose the output is given by

$$h_i = \exp\left(-\frac{1}{2}(X - C_i)^T Q_i (X - C_i)\right), \qquad (6)$$

where X is the input vector, C_i is the center, and Q_i is the dispersion matrix for neuron i.

The output of the network represents the predicted energy and is given by

$$z = \sum_{i=1}^{n_1} w_i h_i(X, C_i, Q_i) + w_0. \qquad (7)$$

The choice of this neural structure involves to determine the position of the centers, the dispersion matrices, and to estimate the weights w_i between the hidden and output layers. Several methods have been used knowing that great care must be taken in deciding how the network centers are positioned.

The first method is the orthogonal least squares algorithm proposed by Chen [16]. This algorithm selects the centers among the database points and supposes an equal dispersion for each neuron. The estimation of the weights w_i then uses an orthogonal least squares algorithm. The method is useful to determine the number of hidden neurons, but the choice of the centers among the database is not the best.

The other method deals separately with the choice of centers of the hidden layer and with the weights estimation. Furthermore, contrary to the previous method, Q_i matrix, chosen diagonal, is supposed different for each neuron i. The determination of the centers is based on unsupervised clustering in the input space.

The first method used is the K-means method (KM) known as efficient for large data sets [17]. The purpose of this method is to obtain a final partition represented by n_1 clusters, each one composed of its own data points and center of gravity. The choice of the number of clusters is based on a criterion represented by the ratio W/B between the within-clusters variance W and the between-clusters variance B. This method allows to know how many neurons are necessary to build the RBF network.

In this algorithm, the belonging of each data point to the clusters is binary. A fuzzy clustering algorithm known as the fuzzy C-means (FCM), presented by Bezdek [18], can be applied. The major feature of this method is the use of a membership degree of each data point to each cluster. The final result provides n_1 clusters having the same shape. With the fuzzy partition, the boundaries between clusters are not obvious. As the previous method, the initialization of the partition must be performed with care. The ratio used to select the appropriate number of clusters is the ratio of two criteria. The first one, NFI, proposed by Libert and Roubens [19], checks the maxima of the membership degrees of the points to the clusters and must be maximized. The second one, S, proposed by Xie and Beni [20], tests the separability of the clusters and must be minimized.

Both of the methods have been tested with the same matrix of data to be clustered. In each case, the number of clusters has continually increased from ten to 45. Fig. 8 shows, for FCM, the ratio NFI/S. The appropriate partition is obtained with 38 clusters corresponding to a maximum ratio.

The partition obtained with the K-means method leads to non homogenous clusters, there are some empty clusters in the final result, while the FCM method creates a partition without empty cluster and no cluster with a strong number of data points.

Once the fuzzy partition obtained, all clusters with their own center are defined. Each data point is attributed to the nearest cluster (maximum membership degree) and the dispersion matrices Q_i can be estimated with respect to the dispersion of the data of each cluster. The hidden layer of the neural network is then completely defined.

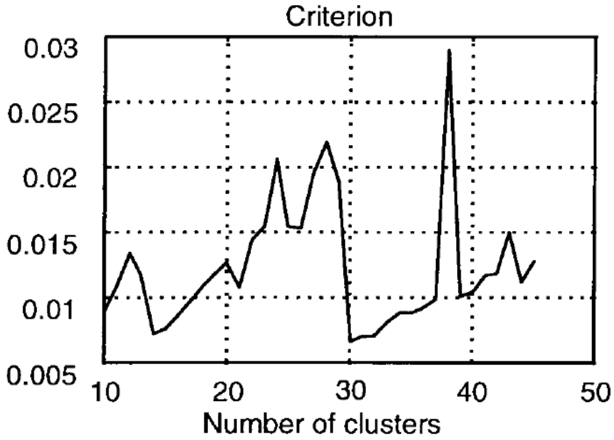

Fig. 8. Choice of the appropriate number of clusters.

The three results obtained with each algorithm and associated to an RBF network are summarized in Table I.

TABLE I
RBF NEURAL-NETWORK RESULTS

Methods	Identification RMSE	Validation RMSE	# hidden units
KM	2.26	3.10	38
FCM	1.29	2.0209	38
OLS	1.44	2.2055	29

The most accurate network is the one built with a FCM algorithm. This is verified for both identification and validation databases. It must be noticed that the RBF built with the OLS algorithm gets a good result with the validation database. This is due to the fact that the algorithm takes into account the performance of RBF network to select the appropriate number of neurons.

From the neural prediction of the energy with respect to the strip features and operating conditions, the next step consists in determining the induction temperature to be reached at the exit of the induction furnace. For confidentiality reasons, hypotheses about the thermal cycle cannot be exposed. The results of the prediction obtained for the induction temperature are shown on Figs. 9 and 10. The difference between the estimated and the measured temperature is shown on Fig. 9, for the validation database composed of 130 data points. Note that the accuracy of the corresponding pyrometer measure is about 5°C.

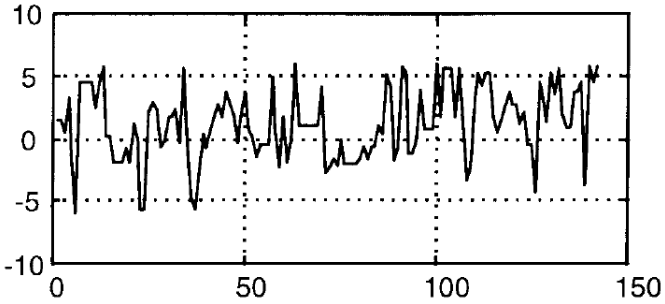

Fig. 9. Difference between measured and estimated exit temperature.

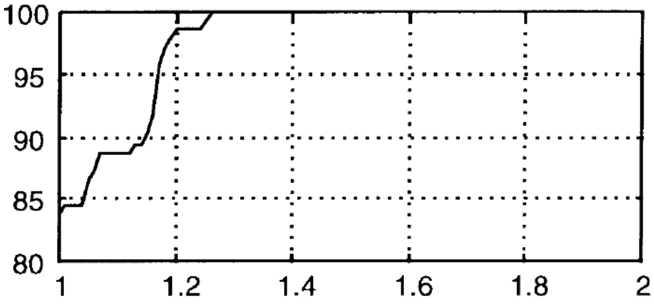

Fig. 10. Accuracy of predicted values.

Fig. 10 is another representation of the accuracy of the estimated values. X axis corresponds to the residual expressed in percent of corresponding measurement. Y axis gives the percentage of data points. To summarize, the model gives for at least 98% of the validation database points an estimated temperature equal to the measured value at 1.2% maximum.

The prediction of the induction temperature being accurate enough, the problem is then to obtain this temperature at the end of the induction furnace. This control problem is presented in the next part.

IV. Control of the Induction Furnace

This part is focused on the control of the alloying cycle and particularly the strip temperature at the exit of the induction furnace ($\theta_{induction}$). A power preset (Pgal) to apply to the furnace is determined with a steady-state inverse model to obtain a strip temperature close to the optimal temperature estimated previously (RBF neural network) (see Fig. 11). The behavior of the furnace being highly nonlinear, a neural network is used to build the inverse model.

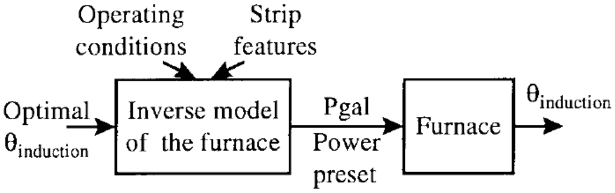

Fig. 11. Open loop control of $\theta_{induction}$ temperature.

Because of the small modeling errors and to take into account the weak fluctuations of the unknown strip temperature at the entrance of the furnace, a control loop is implemented on the process (see Fig. 12). Note that the chosen strategy includes the possibility to disconnect the control loop in case of measurement unavailability of $\theta_{induction}$. That allows for the use of only the neural inverse model in open loop in order to maintain a sufficient degree of fault tolerance.

The model given by the furnace designers provides the power preset required to obtain specified exit temperature, with respect to the strip dimensions (width and thickness), line speed (speed), and the entry temperature. This model is not used on-line because the entry temperature is not measured but it provides three useful points of information. First, this model is a static one. Second, the variables for possible inclusion in

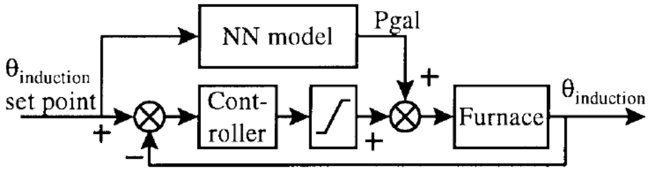

Fig. 12. Control architecture for $\theta_{induction}$.

the Pgal model: exit temperature $\theta_{induction}$, thickness, width, and speed, can be extracted from this model. And finally, it provides an analytical nonlinear description of the efficiency of the furnace. The power to be applied, predicted by the neural model, will be corrected by this efficiency coefficient.

The neural architecture used to determine the furnace model is a feedforward one hidden layer perceptron. The predicted power is given by the output z of the single output neuron

$$z = \sum_{i=1}^{n_1} w_i^2 x_i^1 + b^2 = \sum_{i=1}^{n_1} w_i^2 \, g\left(\sum_{h=1}^{n_0} w_{ih}^1 x_h^0 + b_i^1\right) + b^2 \quad (8)$$

where $x_i^1, i=1,\ldots,n_1$, are the outputs of the hidden neurons, $w_i^2, i=1,\ldots,n_1$, and b^2 are, respectively, the weights and bias associated with the output neuron, $x_h^0, h=1,\ldots,n_0$, are the n_0 inputs of the network, and $w_{ih}^1, i=1,\ldots,n_1, h=1,\ldots,n_0$, and $b_i^1, i=1,\ldots,n_1$, are, respectively, the weights and biases associated with the hidden neurons. Note that the activation functions g of the hidden neurons are hyperbolic tangent functions and the activation function of the output neuron is simply the identity function.

The learning method used is a constructive one. The algorithm starts from a minimal neural network, adds new hidden units, and learns the corresponding weights. To reduce the learning time, the number of hidden units is limited to 10 and the number of random initializations of the weights to 20. The best neural network is thus determined with respect to three criteria:

Mean error: $\mathrm{ME} = \frac{1}{n}\sum_{k=1}^{n}(y(k) - \hat{y}(k))$,

Root mean square error: $\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(y(k) - \hat{y}(k))^2}$,

Maximal error: $\mathrm{E}_{max} = \max_k(|y(k) - \hat{y}(k)|)$,

where y and $\hat{y}$ are, respectively, the measured and predicted outputs and n the number of points.

The prediction error can be introduced:

$$\varepsilon(k,\theta) = y(k) - \hat{y}(k,\theta), \quad (9)$$

where θ comprises all the unknown weights and biases of the network and the following criterion can be written:

$$V(\theta) = \frac{1}{n}\sum_{k=1}^{n} L(\varepsilon(k,\theta)), \quad (10)$$

where $L(.)$ is a scalar valued function, often called case cost function.

The minimization of the criterion (10) can therefore be performed using the Gauss-Newton algorithm

$$\hat{\theta}^{i+1} = \hat{\theta}^i - (H(\hat{\theta}^i))^{-1} V'(\hat{\theta}^i). \tag{11}$$

The gradient of the criterion (10) with respect to θ can be written

$$V'(\theta) = -\frac{1}{n}\sum_{k=1}^{n} \psi(k,\theta) L'(\varepsilon(k,\theta)), \tag{12}$$

where $\psi(k,\theta)$ is the gradient of $\hat{y}(k,\theta)$ with respect to θ and $L'(\varepsilon(k,\theta))$ is the first derivative of L with respect to ε, often called score function.

The second derivative of the criterion (10) with respect to θ, known as the Hessian matrix, is obtained by differentiating (12) and can be approximated classically by

$$H(\theta) = \frac{1}{n}\sum_{k=1}^{n} \psi(k,\theta) L''(\varepsilon(k,\theta)) \psi^T(k,\theta), \tag{13}$$

where $L''(\varepsilon(k,\theta))$ is the second derivative of L with respect to ε. In case the matrix $H(\theta)$ may be close to singular, the following approximate Hessian can be taken, leading to the Levenberg-Marquardt update rule:

$$H(\theta) = \frac{1}{n}\sum_{k=1}^{n} \psi(k,\theta) L''(\varepsilon(k,\theta)) \psi^T(k,\theta) + \beta I, \tag{14}$$

where I is the identity matrix and β a nonnegative small scalar.

The first criterion used is the classical least squares one corresponding to the following case cost function:

$$L(\varepsilon(k)) = \frac{1}{2}\varepsilon^2(k) \tag{15}$$

in the criterion (10), with its first derivative

$$L'(\varepsilon(k)) = \varepsilon(k) \tag{16}$$

in the gradient (12) and with its second derivative

$$L'(\varepsilon(k)) = 1 \tag{17}$$

in the approximate Hessian (14). This algorithm will be called Levenberg-Marquard neural network (LMNN).

Three other outlier-robust algorithms have been used. They are more precisely described in [21]. The first one, called weighted neural network (WNN), is based on the Huber's model [22] of measurement noise contaminated by outliers. The corresponding criterion involves the weighted robust norm as case cost function

$$L(\varepsilon(k)) = \frac{1}{2}\frac{\varepsilon^2(k)}{\sigma_n^2(k)} \tag{18}$$

with its first derivative

$$L'(\varepsilon(k)) = \frac{\varepsilon(k)}{\sigma_n^2(k)} \tag{19}$$

in the criterion gradient (12) and with its second derivative

$$L'(\varepsilon(k)) = \frac{1}{\sigma_n^2(k)} \tag{20}$$

in the approximate Hessian (14). The expression of the variance $\sigma_n^2(k)$ of the kth prediction error, calculated at each iteration of algorithm (11), from all the n errors, is given in [21].

The second robust algorithm, called exponential neural network (ENN), uses, as first derivative of the case cost function, the following expression:

$$L'_p(\varepsilon) = \begin{cases} p\,(|\varepsilon| - 3\hat{\sigma}(1-p/2))^{p-1}, & \varepsilon \geq 3\hat{\sigma}, \\ 2\,\varepsilon\,(p/2)^p\, 3\hat{\sigma}^{p-2}, & |\varepsilon| < 3\hat{\sigma}, \\ -p\,(|\varepsilon| - 3\hat{\sigma}(1-p/2))^{p-1}, & \varepsilon \leq -3\hat{\sigma}, \end{cases} \tag{21}$$

in the criterion gradient (12) and as second derivative in the approximate Hessian (14)

$$L''_p(\varepsilon) = \begin{cases} (p-1)p(|\varepsilon| - 3\hat{\sigma}(1-p/2))^{p-2}, & |\varepsilon| \geq 3\hat{\sigma}, \\ 2\,(p/2)^p\, 3\hat{\sigma}^{p-2}, & |\varepsilon| < 3\hat{\sigma}. \end{cases} \tag{22}$$

The expression of the robustly estimated variance $\hat{\sigma}$ of the prediction errors is given in [21]. When p varies from one to two, the corresponding L_p criterion goes from a least absolute values to a least squares criterion.

The third algorithm, called trimmed neural network (TNN), uses the following rather classical robust norm [23]:

$$L(\varepsilon(k)) = \begin{cases} \frac{1}{2}\varepsilon(k)^2, & |\varepsilon(k)| < \rho\hat{\sigma}, \\ \rho\hat{\sigma}\varepsilon(k), & \varepsilon(k) \geq \rho\hat{\sigma}, \\ -\rho\hat{\sigma}\varepsilon(k), & \varepsilon(k) \leq -\rho\hat{\sigma}, \end{cases} \tag{23}$$

where $\hat{\sigma}$ is the estimated standard deviation of the prediction errors and ρ is a scalar in the range $1 \leq \rho \leq 1.8$. $\hat{\sigma}$ should not be disturbed by outliers and can be thus estimated by

$$\hat{\sigma} = \frac{MAD}{0.7} \tag{24}$$

where MAD is the median of $\{|\varepsilon(k) - \tilde{\varepsilon}|\}$ with $\tilde{\varepsilon}$ as the median of $\{\varepsilon(k)\}$.

The first derivative of (23) can be written

$$L'(\varepsilon(k)) = \begin{cases} \varepsilon(k), & |\varepsilon(k)| < \rho\hat{\sigma}, \\ \rho\hat{\sigma}, & \varepsilon(k) \geq \rho\hat{\sigma}, \\ -\rho\hat{\sigma}, & \varepsilon(k) \leq -\rho\hat{\sigma}, \end{cases} \tag{25}$$

but the second derivative is not continuous and is equal to zero for the outliers, leading to convergence problems of the minimization algorithm. Thus this second derivative is simply always taken equal to one.

Table II presents the final results of the identification of the inverse model of the furnace. The number of hidden neurons of the chosen models has been selected from a tradeoff between model complexity and quality of estimation, particularly checked on the test set. As expected, the results of the classical LMNN algorithm are at least as good as those of the robust ones, for the learning set. However, for the test (validation) set, the results obtained with robust criteria are significantly better, leading to improved generalization capabilities. In the present case, the WNN model seems to be chosen.

In the preceding learning procedures, each hidden neuron is connected to all the inputs. This can rapidly lead to overtting which is harmful to the generalization capacity of the model. A pruning of the unprofitable connections is thus achieved

TABLE II
CRITERIA FOR THE FOUR LEARNING ALGORITHMS

Algorithms	# hidden neurons	Data set	ME	RMSE	E$_{max}$
LMNN	4	Learning	0.24	22.73	65.02
		Validation	18.51	38.79	92.71
WNN	4	Learning	-0.47	21.41	67.23
		Validation	10.11	33.01	77.54
ENN	4	Learning	0.14	24.83	85.67
		Validation	16.70	31.86	88.17
TNN	4	Learning	0.18	22.84	88.06
		Validation	6.07	26.87	85.07

with the OBS algorithm [24]. This method estimates which unit is the least important, deletes it, and trains the remaining weights. This procedure is iterated while the generalization capacity is improved.

V. CONCLUSION

Neural models have been intensively used in the framework of an intelligent control application, intended to improve the performances of a complex galvanizing line. The neural learning approach allows us to incorporate the skill of the control operators in automatic control and optimization systems with a moderate cost of design. Since the mastering of the galvannealing process is one of the most crucial tasks for the control of the galvanizing line, the neural tools are introduced very carefully in the real-time control systems. Presently, the former and neural means of control are implemented together in parallel, leaving the operator in a position to go back to former methods. Although being difficult to quantify, the use of neural tools leads to a decrease of the occurrence of the under-alloyed products. This use also permits a progressive reduction of operator intervention in furnace control. The adaptation of the neural models for new operating conditions (particular strip formats, higher line speed) is not an easy problem, which is currently under investigation.

REFERENCES

[1] M. Beguin, M. Dachelet, E. Montagna, Y. Hardy, J. P. Fischbach, P. Simon, and S. Wilmotte, “Revamping of SEGAL’s galveannealing section by high frequency (HF) induction and misting jets,” in *Proc. Conf. Use and Manufacture of Zinc and Zinc Alloy Coated Steel Sheet Products 21st Century (Galvatech95), Chicago, IL, USA*, Sept. 17-21, 1995, pp. 199–203.

[2] A. Stadlbauer, F. Rubenzucker, K. Zeman, and E. Furhmann, “A new galvannealling process controller,” in *Proc. Conf. Use and Manufacture of Zinc and Zinc Alloy Coated Steel Sheet Products 21st Century (Galvatech95), Chicago, IL, USA*, Sept. 17-21, 1995, pp. 81–85.

[3] C. J. Harris, *Advances in Intelligent Control*. London: Taylor and Francis, 1994.

[4] P. Willmott, *TPM the Western Way*. London: Butterworth-Heinemann, 1994.

[5] J. Brunet, D. Jaume, M. Labarrère, M. Rault, and M. Vergé, *Détection et diagnostic de pannes : Approche par modélisation*. Paris: Hermès, 1990.

[6] G. Zwingelstein, *Diagnostic des défaillances : Théorie et pratique pour les systèmes industriels*. Paris: Hermès, 1995.

[7] E. S. Page, “Continuous inspection schemes,” *Biometrika*, vol. 41, pp. 100–115, 1954.

[8] D. V. Hinkley, “Inference about the change point from cumulative sum tests,” *Biometrika*, vol. 58, no. 3, pp. 509–523, 1971.

[9] M. Basseville and A. Benveniste, “Detection of abrupt changes in signals and dynamical systems.” *Lecture Notes in Control and Information Sciences*, vol. 77. New York: Springer-Verlag, 1986.

[10] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[11] J. Ragot, M. Darouach, D. Maquin, and G. Bloch, *Validation de données et diagnostic*. Paris: Hermès, 1990.

[12] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[13] F. Sirou, *Amélioration de l’efficience globale d’une ligne de galvanisation*. Thèse de Doctorat de l’Université Henri Poincaré-Nancy 1, Spécialité Automatique, 1996.

[14] F. Sirou, V. Eustache, G. Bloch, T. Cecchin, and P. Fatrez, “Monitoring of sensors and processes in a galvanizing line,” in *Proc. IFAC Symp. Intelligent Components and Instruments for Control Applications (SICICA’97), Annecy, France*, June 9-11, 1997, pp. 193-198.

[15] S. Cao and R. R. Rhinehart, “An efficient method for on-line identification of steady state,” *J. Proc. Cont.*, vol. 5, no. 6, pp. 363–374, 1995.

[16] S. Chen, F. N. Cowan, and P. M. Grant, “Orthogonal least squares learning algorithm for radial basis function networks,” *IEEE Trans. Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.

[17] G. Celeux, E. Diday, G. Govaert, Y. Lechevallier, and H. Ralambondrainy, *Classication automatique des données*. Paris: Dunod informatique, 1989.

[18] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[19] G. Libert and M. Roubens, “Non metric fuzzy algorithms and their cluster validity,” in *Approximate Reasoning in Decision Analysis*, M. M. Gupta and E. Sanchez, Eds. Amsterdam, The Netherlands: North Holland, 1982, pp. 417–425.

[20] X. L. Xie and G. Beni, “A validity for measure for fuzzy clustering,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 8, pp. 841–847, 1991.

[21] P. Thomas and G. Bloch, “From batch to recursive outlier-robust identication of nonlinear dynamic systems with neural networks,” in *Proc. IEEE Int. Conf. Neural Networks (ICNN96), Washington, DC, USA*, vol. 1, June 3-6, 1996, pp. 178–183.

[22] P. J. Huber, “Robust estimation of a location parameter,” *Ann. Math. Stat.*, vol. 35, no. 1, pp. 73–101, 1964.

[23] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[24] B. Hassibi and D. G. Stork, “Second-order derivatives for network pruning: Optimal brain surgeon,” *Neural Inform. Processing Syst. 5*, pp. 164–171, 1992.