Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Intelligent non-linear modelling of an industrial winding process using recurrent local linear neuro-fuzzy networks

Hasan ABBASI NOZARI, Hamed DEHGHAN BANADAKI, Mohammad MOKHTARE, Somayeh HEKMATI VAHED

(*Department of Mechatronics, Faculty of Engineering, Islamic Azad University, Science and Research Branch, Tehran, Iran*)

E-mail: {nozari.amirali, hmd.dehghan}@gmail.com; {m.mokhtare, s.hekmati}@srbiau.ac.ir

Received Oct. 17, 2011; Revision accepted Feb. 10, 2012; Crosschecked Apr. 9, 2012

Abstract: This study deals with the neuro-fuzzy (NF) modelling of a real industrial winding process in which the acquired NF model can be exploited to improve control performance and achieve a robust fault-tolerant system. A new simulator model is proposed for a winding process using non-linear identification based on a recurrent local linear neuro-fuzzy (RLLNF) network trained by local linear model tree (LOLIMOT), which is an incremental tree-based learning algorithm. The proposed NF models are compared with other known intelligent identifiers, namely multilayer perceptron (MLP) and radial basis function (RBF). Comparison of our proposed non-linear models and associated models obtained through the least square error (LSE) technique (the optimal modelling method for linear systems) confirms that the winding process is a non-linear system. Experimental results show the effectiveness of our proposed NF modelling approach.

Key words: Non-linear system identification, Recurrent local linear neuro-fuzzy (RLLNF) network, Local linear model tree (LOLIMOT), Neural network (NN), Industrial winding process

doi:10.1631/jzus.C11a0278 **Document code:** A **CLC number:** TP183

1 Introduction

Nowadays, physical process modelling has become one of the most attractive and challenging issues in control theory and applications. There are some factors that make conventional modelling a tedious and time consuming task and one that may even yield unpromising results if there is a lack of precise, formal knowledge about the system or a high degree of uncertainty. However, extracting a suitable model for a winding process seems to be necessary for model-based control and diagnosis trials. Thus, in this paper, data-driven modelling methods are applied instead of analytical methods to identify an accurate and reliable model of a real winding process.

Winding systems are major components of a wide variety of industrial plants. For example, rolling mills in the steel industry (Hussein *et al.*, 2001; Babuska and Verbruggen, 2003) and plants involving web conveyance including coating, paper making, and polymer film extrusion processes (Ebler *et al.*, 1993; Braatz *et al.*, 1996; SISTA, 1999). In the last few decades, researchers have studied the issue of reducing the computational burden associated with the design, analysis, and implementation of control techniques and active fault tolerant control in web conveyance systems (Sievers *et al.*, 1988), sheet and film processes (Braatz *et al.*, 1996), aluminium industries (Hoshino *et al.*, 1988), and steel industries (Parant *et al.*, 1989).

In addition to studies employing analytical methods in the modelling of winding machines (Hoshino *et al.*, 1988; Sievers *et al.*, 1988; Parant *et al.*, 1992), recently, a few attempts have been made to utilize data-based techniques to identify an appropriate model of the winding process. A linear model of a winding machine, such as an auto-regressive with external inputs (ARX) structure, was built for model-

based fault diagnosis applications (Noura *et al.*, 2009). Subspace methods were exploited for multivariable identification of a winding process with the purpose of tension control (Bastogne *et al*., 1998). Genetic programming (GP) was used to construct a non-linear model of a winding process (Hussian *et al.*, 2000). A non-parametric neural network (NN) predictor model of a winding process was developed (Hussein *et al.*, 2001). It is apparent that there are few studies that deal with intelligent non-linear modelling of a winding process on the basis of soft computing techniques. To our knowledge, no attempt has been made to apply neuro-fuzzy (NF) modelling to a winding process. The main drawback of NN models is that systems cannot be expressed in them because they are usually considered as black boxes. NF modelling can be regarded as a grey-box technique on the boundary between neural networks and qualitative fuzzy models, in which a system is expressible through fuzzy rules using fuzzy modelling. The most common NF systems are based on two types of fuzzy models, Takagi–Sugeno (TS for short, data-based) and Mamdani (knowledge-based), combined with NN learning algorithms. The TS-type NF model is preferable when the accuracy of the model is the main concern (Razavi-Far *et al.*, 2009).

2 Linear dynamic system identification

The procedure for the linear identification of dynamic multi-input single-output (MISO) systems from input-output sequences ($u(t)\in\mathbb{R}^p$, $y(t)\in\mathbb{R}^r$, with r=1) is described in this section. According to system identification theory, the simplest solutions should be tried first to build the model of a plant when no prior knowledge about its intrinsic behaviour is available. It has been mathematically proven that the least square error (LSE) method is the optimal modelling method in the case of linear systems (Ljung, 1987; Nelles, 2001). To achieve the linear model, a finite sequence of the input-output variables observed with a constant sampling interval is considered. If dynamic linear relations exist among these variables, they can be described by the following model:

$$y_{\mathrm{M}}(k)=\sum_{l=1}^{n}a_l y_{\mathrm{M}}(k-l)+\sum_{i=1}^{p}\sum_{l=1}^{m_i}b_{il}u_i(k-l). \qquad (1)$$

This describes linear MISO discrete-time systems (and is simply convertible to its corresponding discrete-time transfer function) where a_l and b_{il} are parameters, m_i denotes the order of the numerator of the ith input, and n is the denominator order, usually of the same value (Nelles, 1996). The parameters can be estimated by the least squares techniques since the prediction error is linear in parameters (Ljung, 1987; Nelles, 2001).

One of the most important parts of modelling a system is to select the proper model structure. The purpose of order selection is to identify a model that best fits a given data set. This issue is so critical that imperfection in this section may lead to some severe problems in other parts of the modelling. Several information criteria can be used to accomplish this task (Ljung, 1987), such as the Akaike information criterion (AIC). For simplicity, the obtained orders based on linear models can then be applied to non-linear approximators in the case of non-linear identification. In the given order selection technique, the order of the linear model presented in Eq. (1) is increased and the AIC index is calculated in each step. Finally, after some iteration, the order corresponding to the lowest AIC value is selected as the model order:

$$J_{\mathrm{AIC}}=\lg \mathrm{SSE}+\frac{2P}{Q}, \qquad (2)$$

where Q is the number of samples used to compute J_{AIC}, P is the number of parameters, and $2P/Q$ is a penalty term. The sum of square errors (SSE) is given by

$$\mathrm{SSE}=\sum_{N=1}^{Q}(y_{\mathrm{P}}(N)-y_{\mathrm{M}}(N))^2, \qquad (3)$$

where y_{P} and y_{M} are the outputs of the process and model, respectively.

3 Non-linear dynamic system identification

Non-linear system identification is more appropriate when linear methods do not present satisfactory results in modelling of physical systems. Thus, in this section, the process of coping with the nonlinearity as well as the dynamism of a winding process is described on the basis of recurrent local linear neuro-

fuzzy (RLLNF) techniques. Other NN architectures used for comparative purposes are also introduced in the subsequent sections.

3.1 Recurrent local linear neuro-fuzzy networks

To identify a non-linear simulator model of a winding process, an RLLNF network is utilized, and a local linear model tree (LOLIMOT) algorithm is employed to find the best structure and parameters for the network. There are a host of advantages of an RLLNF model trained with a LOLIMOT algorithm, including low computational cost due to local estimation, robustness with respect to noise due to regularization effect, high accuracy, fast training and evaluation capabilities, and online adaptation.

More advantages of RLLNF models were described by Nelles (1996).

The structure of the RLLNF network is shown in Fig. 1. Each neuron realizes a local linear model (LLM) and an associated fuzzy validity function that determines its region of validity. The LLNF models have basically the same interpretation as TS models, where each neuron represents one fuzzy rule, the validity functions represent the rule premise, and the LLMs represent the rule consequents.

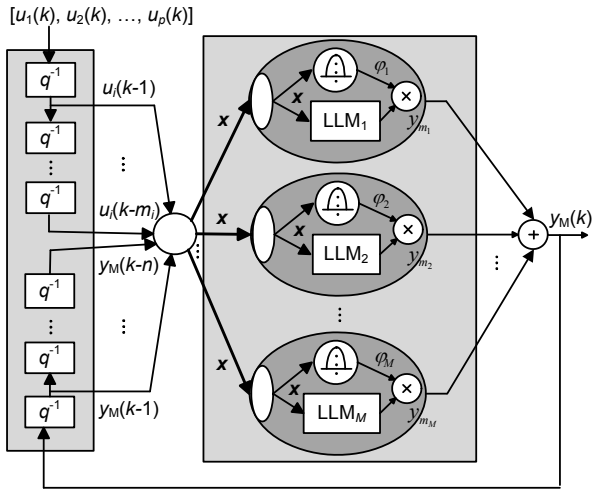

Fig. 1 Architecture of a recurrent local linear neuro-fuzzy (RLLNF) network

To create a simulator model, the delayed inputs of the process and past samples of the LLNF model output are injected into the model as inputs. Hence, in the case of a recurrent (dynamic) LLNF network the input of the model is given as

$$\begin{aligned}\boldsymbol{x}=&\left[u_1(k-1),\ldots,u_1(k-m_1),u_2(k-1),\ldots,\right.\\&u_2(k-m_2),\ldots,u_p(k-1),\ldots,u_p(k-m_p),\\&\left.y_{\mathrm{M}}(k-1),\ldots,y_{\mathrm{M}}(k-n)\right]^{\mathrm{T}},\end{aligned}\tag{4}$$

where n and m_i (i=1, 2, …, p) are the denominator and numerator orders of the ith input, respectively. The parameters $\boldsymbol{w}_j$ of the jth rule consequence is

$$\boldsymbol{w}_j=\left[b_{j_{11}},b_{j_{12}},\ldots,b_{j_{1m_1}},\ldots,b_{j_{p1}},b_{j_{p2}},\ldots,b_{j_{pm_p}},\ldots,a_{j1},\ldots,a_{jn}\right]^{\mathrm{T}}.\tag{5}$$

These parameters are estimated using the weighted least square (WLS) solution (Nelles, 2001). Thus, the global output of the model is calculated as the weighted summation of the output of all LLMs as follows:

$$\begin{aligned}y_{\mathrm{M}}(k)=\sum_{j=1}^{M}\sum_{i=1}^{p}\Big[&b_{j_{i1}}u_i(k-1)+b_{j_{i2}}u_i(k-2)+\ldots\\&+b_{j_{im_i}}u_i(k-m_i)-a_{j1}y_{\mathrm{M}}(k-1)-a_{j2}y_{\mathrm{M}}(k-2)\\&-\ldots-a_{jn}y_{\mathrm{M}}(k-n)+\zeta_j\Big]\varphi_j(\boldsymbol{x}),\end{aligned}\tag{6}$$

where $b_{j_{im_i}}$ and a_{in} represent the numerator and denominator coefficients respectively, ζ_j is the offset of LLM$_j$, and $\varphi_j(\boldsymbol{x})$ are the operating point-dependent weighting factors. In other words, the network interpolates between different LLMs using the validity functions (Nelles, 1996). The validity functions on $\boldsymbol{x}$ are typically chosen as normalised Gaussians, so they form a partition of unity as

$$\sum_{j=1}^{M}\varphi_j(\boldsymbol{x})=1.\tag{7}$$

In the case of axis-orthogonal Gaussians, the validity functions are defined as

$$\varphi_j(\boldsymbol{x})=\mu_j(\boldsymbol{x})\Big/\sum_{j=1}^{M}\mu_j(\boldsymbol{x}).\tag{8}$$

Thus, $\mu_j(\boldsymbol{x})$ can be given as

$$\mu_j(\boldsymbol{x})=\exp\left\{-\frac{1}{2}\left[\left(\frac{x_1-c_{j1}}{\sigma_{j1}}\right)^2+\ldots+\left(\frac{x_D-c_{jD}}{\sigma_{jD}}\right)^2\right]\right\},\tag{9}$$

where $D = n + \sum_{i=1}^{p} m_i$ is the total number of input channels of the LLNF model, and $\boldsymbol{c}$ and $\boldsymbol{\sigma}$ are the centre coordinates and the individual standard deviations, respectively.

Local linear modelling approaches are based on a divide-and-conquer strategy. A complex winding process model is broken into a number of smaller and thus simpler sub-problems, which are solved independently by identifying piecewise linear models (Nelles and Isermann, 1996; Nelles, 2001). The most important factor for the success of this model using an LLM method is the division strategy for the original complex problem. This is determined by an algorithm named LOLIMOT. The LOLIMOT algorithm consists of an outer loop in which the rule premise structure is determined, and a nested inner loop in which the rule consequent parameters are optimized by local estimation. This loop can be summarised in five steps as follows (Nelles, 1996; 2001; Nelles and Isermann, 1996):

Step 1: Start with an initial single LLM which is globally optimal based on least-squares estimation over the whole input space.

Step 2: Find the worst performing LLM, i.e., the one that has the maximum local loss function, e.g., the mean squared error (MSE) among the M LLMs.

Step 3: The worst model (neuron) selected in Step 2 is considered for further division. The validation hypercube of this neuron is broken into two halves with an axis-orthogonal split. Divisions in all dimensions are carried out, and for each of the divisions the following four steps are performed:

(i) Construction of the multi-dimensional fuzzy membership functions for both hyper-rectangles.

(ii) Construction of all validity functions.

(iii) Local estimation of the rule consequent parameters for both newly generated local linear neurons.

(iv) Calculation of the global loss function for the current overall model.

Step 4: Find the best division (the best of the alternatives checked in Step 3), and increment the number of neurons: $M \rightarrow M+1$.

Step 5: Test for convergence.

Remark 1 For the termination criterion various options exist including a maximal model complexity, i.e., a maximal number of LLMs, statistical validation tests, or information criteria.

3.2 Multilayer perceptron neural network

Multilayer feedforward neural networks are very suitable tools for data-based modelling of real plants due to their general function approximation capabilities. Detailed information concerning neural networks was given by Nelles and Isermann (1996) and Nelles (2001). The architecture of an MLP neural network suitable for MISO identification is shown in Fig. 2. This network has one non-linear hidden layer with L tangent hyperbolic activation functions. The output layer also has one neuron with linear activation functions with a slope of 1.

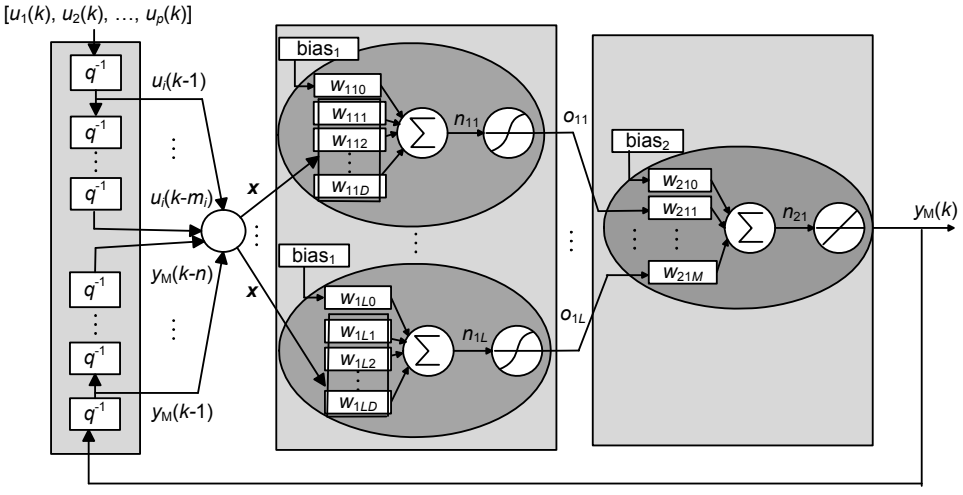

Fig. 2 Network structure of a recurrent multilayer perceptron (MLP)

Here, just a brief overview of the learning algorithm used in this network is given. Neural networks aim to update their defined parameters with learning algorithms to give proper outputs for proper inputs and to minimize the error. There are many learning algorithms for this purpose, such as Gauss-Newton (GN), gradient descent (GD), and Levenberg-Marquardt (LM). We decided to use the LM algorithm to update the MLP network parameters because it has the following advantages: (1) faster convergence than the GD method, (2) more robustness than the GN method, (3) an interpolation between the GN and GD methods (i.e., it has the speed of GN and the convergence of GD).

The LM algorithm updates the parameters of the MLP network according to

$$\begin{cases} \boldsymbol{W}_{n+1} = \boldsymbol{W}_n + \Delta \boldsymbol{W}, \\ \Delta \boldsymbol{W} = -(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{W})\boldsymbol{J}(\boldsymbol{W}) + \mu \boldsymbol{I})^{-1}(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{W})\boldsymbol{e}), \end{cases} \tag{10}$$

where $\boldsymbol{e}$ is the error function, $\boldsymbol{J}$ is a Jacobian matrix, and μ is a scalar that makes LM closed to either GD or GN. $\boldsymbol{W}$, which contains the weights of the network, is defined as follows:

$$\begin{aligned} \boldsymbol{W} = [&w_{110}, w_{111}, \ldots, w_{11D}, \ldots, w_{1L0}, w_{1L1}, \\ &\ldots, w_{1LD}, w_{210}, w_{211}, \ldots, w_{21M}]. \end{aligned} \tag{11}$$

Regarding the feedforward multilayer neural network, note that a network with one hidden layer is sufficient for most approximation tasks. More layers can give a better fit, but the training takes longer.

3.3 Radial basis function network

A radial basis function (RBF) network is a two-layered neural network with the topology depicted in Fig. 3. This network is represented by the following function:

$$y_{\mathrm{M}}(k) = f(\boldsymbol{x}) = \sum_{j=1}^{L} v_j \mu_j(\boldsymbol{x}), \tag{12}$$

where the usual choice for the basis functions $\mu_j(\boldsymbol{x})$ is the Gaussian function given in Eq. (9). The parameters of RBF networks are the output weights v_j and the parameters of the basis functions (centres $\boldsymbol{c}_j$ and

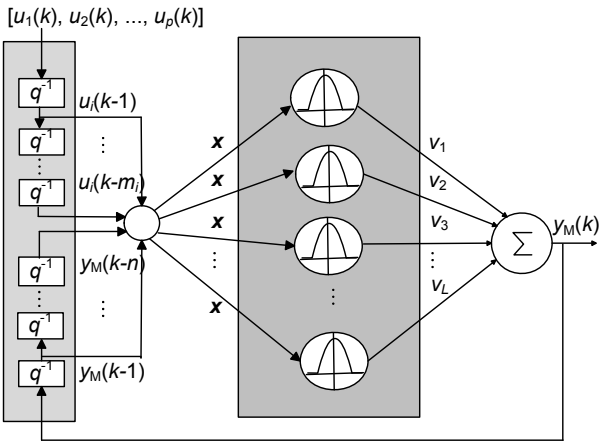

Fig. 3 Recurrent radial basis function (RBF) network with external dynamics

standard deviation $\boldsymbol{\sigma}_j$). Since the network's output Eq. (12) is linear in the weights v_j, these weights can be estimated by least-squares methods, whereas $\boldsymbol{c}$ and $\boldsymbol{\sigma}$ are network non-linear parameters which can be updated via an optimization problem solved using the GD method (Ljung, 1987; Nelles, 1996). First-order gradient methods are based on the following general update rule for the parameters $\boldsymbol{c}_j$, $\boldsymbol{\sigma}_j$:

$$\begin{cases} \boldsymbol{c}_{n+1} = \boldsymbol{c}_n + \mu_{cn} \nabla J(\boldsymbol{c}_n), \\ \boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mu_{cn} \nabla J(\boldsymbol{\sigma}_n), \end{cases} \tag{13}$$

where $\boldsymbol{c}_n$ and $\boldsymbol{\sigma}_n$ are the centre and standard deviation vectors, respectively. At iteration n, μ is the learning rate, and J is the cost function defined as

$$J = \frac{1}{2} \sum_{N=1}^{Q} \left(y_{\mathrm{P}}(N) - y_{\mathrm{M}}(N) \right)^2. \tag{14}$$

$\nabla J(\cdot)$ is the Jacobian of the network, given as

$$\begin{cases} \nabla J(\boldsymbol{c}) = \left[\dfrac{\partial J(\boldsymbol{c})}{\boldsymbol{c}_1}, \dfrac{\partial J(\boldsymbol{c})}{\boldsymbol{c}_2}, \ldots, \dfrac{\partial J(\boldsymbol{c})}{\boldsymbol{c}_l} \right]^{\mathrm{T}}, \\ \nabla J(\boldsymbol{\sigma}) = \left[\dfrac{\partial J(\boldsymbol{\sigma})}{\boldsymbol{\sigma}_1}, \dfrac{\partial J(\boldsymbol{\sigma})}{\boldsymbol{\sigma}_2}, \ldots, \dfrac{\partial J(\boldsymbol{\sigma})}{\boldsymbol{\sigma}_l} \right]^{\mathrm{T}}. \end{cases} \tag{15}$$

4 Case study: winding process description

The winding plant introduced in the first subsection represents the major subsystem often met in

web conveyance systems. Modelling of a winding process using theory-based methods such as Hooke's equation is a complicated task and represents a challenge for both system theory and applications (Hussian *et al.*, 2000). To overcome these problems, non-analytical techniques presented in previous sections are experimentally applied for the identification of winding process models.

4.1 Process description

The case study is a test setup of an industrial winding process (SISTA, 1999). Fig. 4 shows the layout of the winding process under consideration, which consists of multi-variable and coupled systems with process parameters varying during operation. The main role of the winding process is to control the web conveyance in order to avoid the effects of friction and sliding. The solution consists of maintaining a traction effort on the strip and controlling the tension at different points along the web (Braatz *et al.*, 1996). The winding machine is composed of three reels driven by DC motors denoted as M_1, M_2, and M_3, gear reduction coupled with the reels, and a plastic strip. Motor M_1 corresponds to the unwinding reel, M_3 to the rewinding reel, and M_2 to the traction reel. The angular velocity of motor M_2 (Ω_2) and the strip tensions between the reels (T_1, T_3) are measured using a dynamo tachometer and tension meters, respectively. Each motor is driven by a local controller. Torque control is achieved for motors M_1 and M_3, while speed control is realized for motor M_2. Significant process variables are measured by sensors at pre-chosen points of the process and then the acquired data are recorded at a sampling rate of 0.1 s in the monitoring system.

Note that the parameter variation which occurs in this process is due to variation in the reel radius during unwinding. This non-measurable variation of the reel radius significantly modifies the dynamic behaviour of the system during the overall process of unwinding (Noura *et al.*, 2009). Hence, owing to these uncontrolled effects to which the winding process is subjected, this type of process is challenging in modelling, identification, and control. To identify the winding process, the input and output variables are listed in Table 1.

Table 1 Input and output variables in the winding process

Variable name		Variable description
Input	S_1	Angular speed of reel 1
	S_2	Angular speed of reel 2
	S_3	Angular speed of reel 3
	RI_1	Set point current at M_1
	RI_3	Set point current at M_2
Output	T_1	Tension in the web between reels 1 and 2
	T_3	Tension in the web between reels 2 and 3

4.2 Data pre-processing for system identification

Data pre-processing methods are required to extract valid data from the available experimental data. Peak shaving and smoothing of the intensive changes in real data are very important in the pre-processing phase. Toward these objectives, the real data are passed through a first-order digital Butterworth low-pass filter with a bandwidth of 0.3 Hz. From a signal processing standpoint, a suitable filter should not change or affect the original signal shape

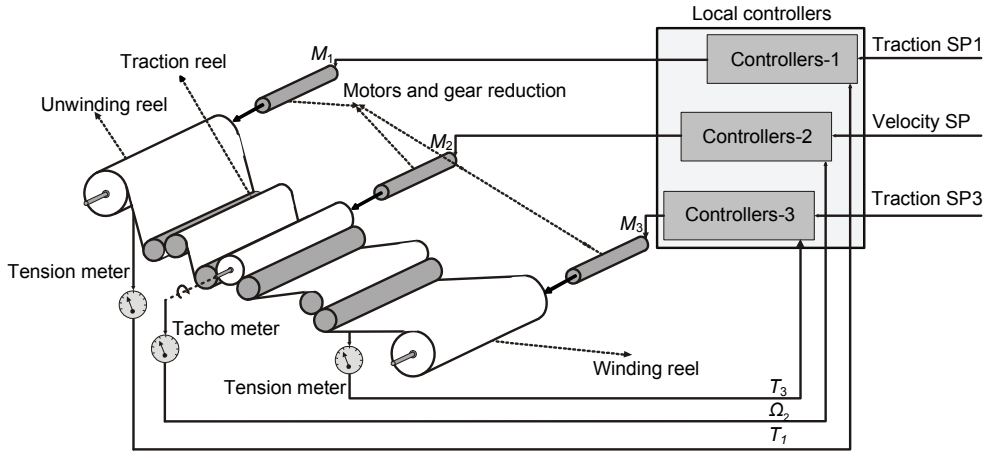

Fig. 4 Schematic of an industrial winding process with monitored sensors

while eliminating noise and disturbing signals. Furthermore, since the input and output data have different ranges, the filter may yield errors in data quantization and lead to poor identification of the system (Banadaki *et al.*, 2011; Sadeghian and Fatehi, 2011). Data normalisation must be performed as an essential step of data mining for system identification. Experiments have proven that more promising modelling results will be revealed using data normalisation. The original signal S can be mapped to the normalised signal S_n as follows:

$$S_n = \frac{S - S_{min}}{S_{max} - S_{min}}, \tag{16}$$

where S_{max} and S_{min} are the maximum and minimum values of S, respectively. The effects of the Butterworth filter and normalisation on the signal S_2 are shown in Fig. 5.

4.3 Experimental modelling results

To evaluate modelling performance, we used the mean squared error (MSE) and the variance accounted for (VAF). MSE is defined as

$$\mathrm{MSE} = \frac{1}{Q}\sum_{N=1}^{Q}\left(y_P(N) - y_M(N)\right)^2. \tag{17}$$

The percentile VAF was also used to measure the performance of the obtained model:

$$\mathrm{VAF} = 1 - \frac{\mathrm{cov}(y_P - y_M)}{\mathrm{cov}(y_P)} \times 100\%, \tag{18}$$

where cov(·) is the covariance of the respective vector.

As discussed in Section 2.1, the order of the model should be determined to improve performance through the system identification procedure. For the purpose of order selection, an AIC information index was exploited in this study. Table 2 shows the variation in the AIC index with respect to the increase in model order for both outputs of the winding process, denoted as T_1 and T_3.

Table 2 Variation in the Akaike information criterion (AIC) information index with respect to order increment

Order	JAIC-T_1 (N/m)	JAIC-T_3 (N/m)	Order	JAIC-T_1 (N/m)	JAIC-T_3 (N/m)
1	0.0209	0.1611	6	0.0178	0.2229
2	0.0145	0.1464	7	0.0169	0.2034
3	0.0177	0.2273	8	0.0170	0.2076
4	0.0188	0.2250	9	0.0173	0.2101
5	0.0183	0.2260	10	0.0176	0.2089

As the orders of the models increased from 1 to 2, the values of the AIC index for both linear dynamic models decreased. However, when increasing the order from 2 to 10, there were no significant improvements in AIC values for either model. Thus, we decided to choose the second-order models which yield the lowest AIC values. The selected orders for linear models were then also applied as the orders of the non-linear models.

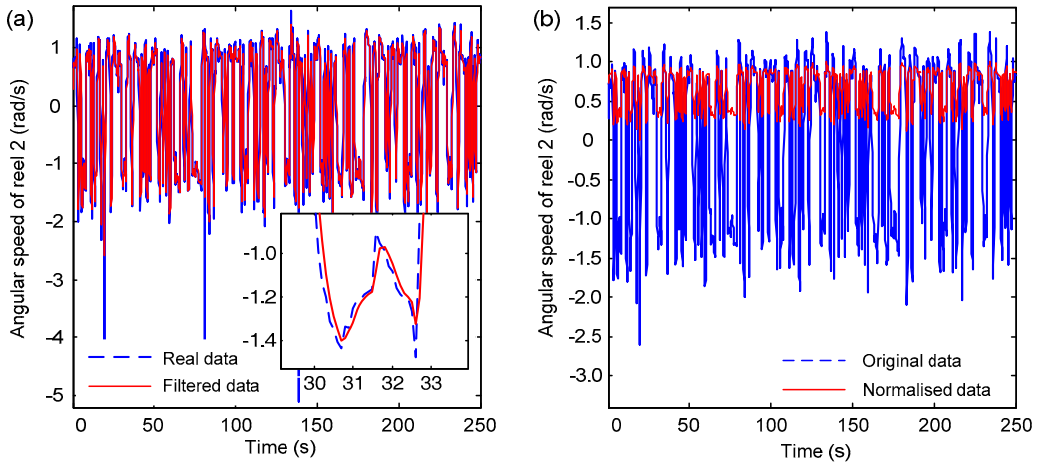

Fig. 5 Data pre-processing of the angular speed of reel 2: (a) Butterworth effects; (b) normalisation effects

In the case of both LLNF and NN models, the most crucial concern is the number of neurons, which should be as small as possible. The number of neurons for all the networks presented in this work was based on the MSE curve. A typical MSE curve for an MLP network is shown in Fig. 6 for the simulated web tension between reels 2 and 3 (T_3). In this regard, the ideal neuron number was determined by increasing the number of neurons until more neurons did not have a significant effect on the reduction of the MSE for the test data. As the neuron number increased from 1 to 5, MSE values for the training and test data sets declined. When the number of hidden neurons increased past 5, there were no marked improvements in the MSE for the test data. Hence, an MLP network with five hidden neurons was selected based on the MSE curve. The same procedure was performed to select the optimal neuron number for other networks, namely RBF and LLNF.

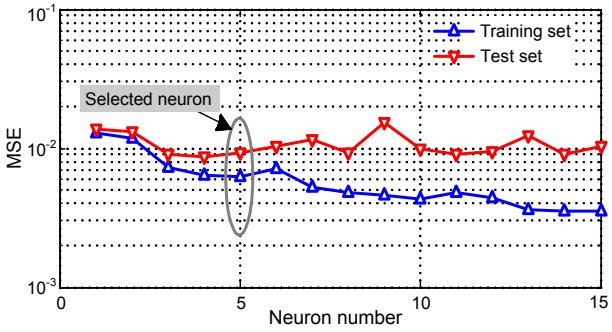

Fig. 6 The variation of the mean square error (MSE) of the web tension model between reels 2 and 3 using the multilayer perceptron (MLP) model

Figs. 7 and 8 show the responses of the proposed NF, MLP, and RBF models, and the response of the real winding process to the tensions in the web between reels 1 and 2 (T_1) and between reels 2 and 3 (T_3). The proposed NF model was best able to track the corresponding outputs of the real winding process. Other models had severe problems in tracking the responses of the real systems, especially in some time intervals. Such a case can be observed in the time interval between 220 s and 225 s for the second output of the real winding process (T_3). The real response of the system was effectively tracked by the response of the RLLNF model, whereas other intelligent models, namely MLP and RBF, had some tracking problems, especially in intervals in which a set of fluctuations are observed within the response of the system (e.g., [235, 245] s). The LSE linear models had severe tracking problems, especially with large magnitudes of system responses. Moreover, since LSE is the optimal modelling method for linear systems, it can be concluded that the winding process belongs to the non-linear systems category.

Table 3 shows the accuracy results achieved on the basis of the proposed modelling accuracy criteria for different winding machine models. Taking into account the obtained MSE and VAF values, the proposed RLLNF models showed almost the highest accuracies throughout the modelling of the non-linear winding process compared with the other modelling approaches. Besides, in relation to the neuron number of the non-linear networks presented in Table 3, note that the neuron (rule) numbers of the RLLNF networks are meaningfully large as a result of the fast training and evaluation capabilities of the LOLIMOT algorithm.

Table 3 Accuracy results for linear and non-linear models of the winding process*

Model output	Neuron number	MSE		VAF (%)	
		Training set	Validation set	Training set	Validation set
LSE−T_1	–	0.1247	0.1359	87.3062	86.7406
MLP−T_1	12	8.93e-4	8.61e-4	98.4746	98.6752
RBF−T_1	10	8.76e-4	8.54e-4	98.5999	98.6899
LLNF−T_1	20	1.44e-4	7.72e-4	99.6915	98.8234
LSE−T_3	–	0.2547	0.2430	75.2313	73.9855
MLP−T_3	5	9.27e-4	1.10e-3	93.5317	89.9052
RBF−T_3	10	8.37e-4	2.20e-3	94.5855	86.7895
LLNF−T_3	25	8.86e-4	1.10e-3	94.0865	92.2721

* Values in this table are approximate. LSE: least square error; MLP: multilayer perceptron; RBF: radial basis function; LLNF: local linear neuro-fuzzy. MSE: mean square error; VAF: variance accounted for

All the non-linear simulator models presented could simulate winding plant behaviour after training without the need of real winding plant output data, but only by feeding inputs into the models. This advantage of the proposed intelligent simulator models makes it possible to simulate and design model-based controllers over the whole operating ranges of a non-linear winding system. Furthermore, the local linear modelling strategy employed in constructing an NF simulator model allows for exploiting the obtained LLM parameters for designing or re-designing the local controllers of a winding process owing to the representation of a non-linear system by several NF piecewise linear models.

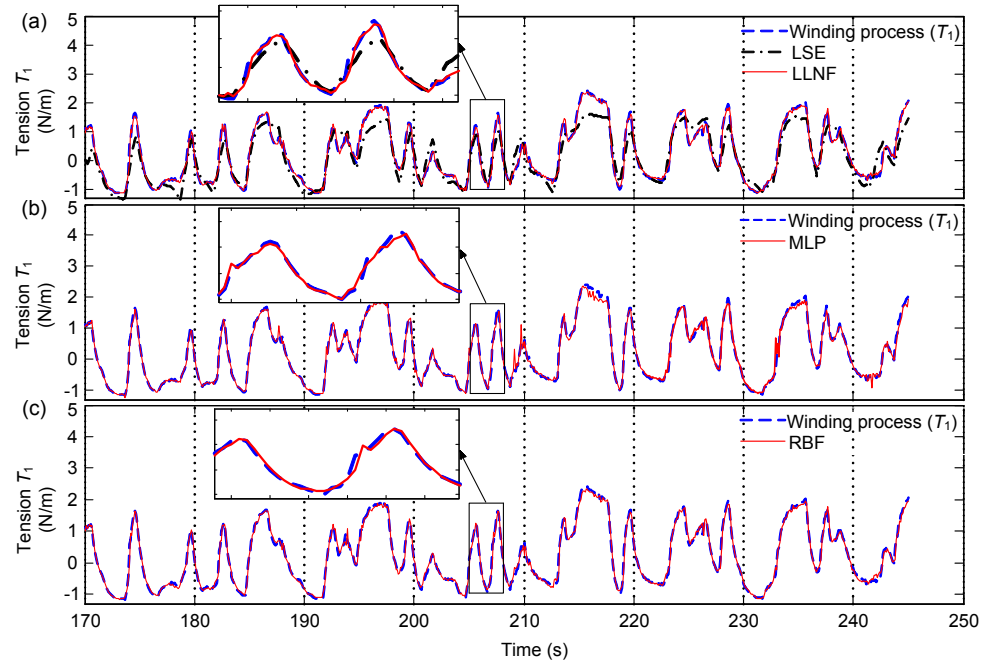

Fig. 7 Modelling performance for the web tension model between reels 1 and 2

(a) Responses of local linear neuro-fuzzy (LLNF) and least square error (LSE) models; (b) Response of the multilayer perceptron (MLP) model; (c) Response of the radial basis function (RBF) model

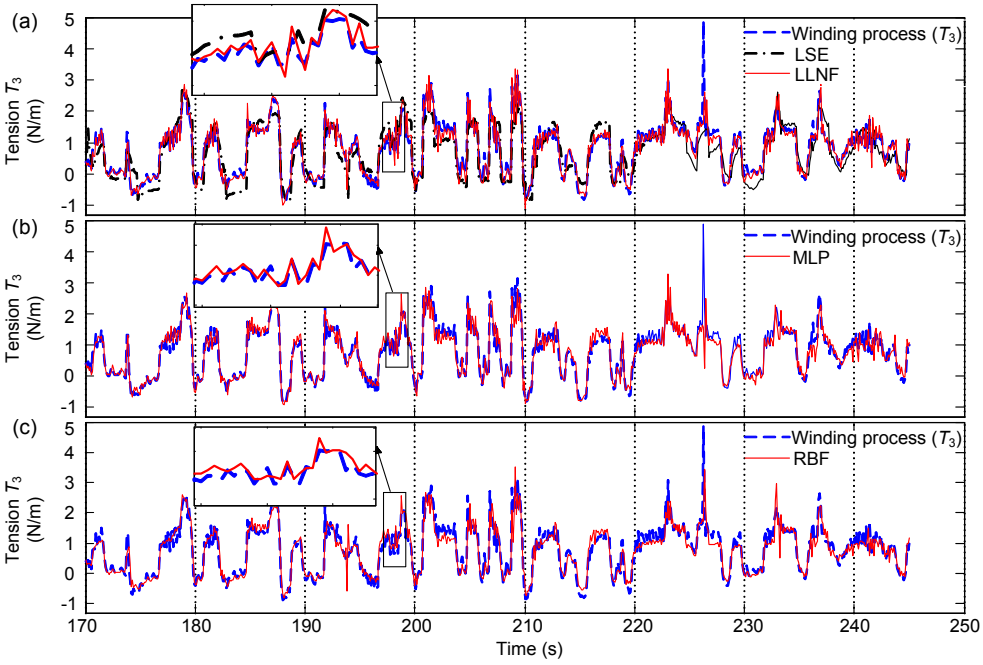

Fig. 8 Modelling performance for the web tension model between reels 2 and 3

(a) Responses of local linear neuro-fuzzy (LLNF) and least square error (LSE) models; (b) Response of the multilayer perceptron (MLP) model; (c) Response of the radial basis function (RBF) model

5 Conclusions

In this paper, the identification of non-linear simulator models using an RLLNF technique was carried out for a real industrial winding process. For data-based modelling, noise elimination, data normalisation, and order selection are essential. After pre-processing the data and obtaining a rich data set, one linear LSE-based and three non-linear recurrent intelligent models were proposed for all the winding machine outputs by LLNF, MLP, and RBF. The LLNF simulator model for a real winding process has been presented for the first time.

In terms of the experimental results, all proposed NF models demonstrated the highest accuracies in that they effectively created a faithful replica of their corresponding MISO sub-system. Furthermore, the comparisons made between linear and non-linear modelling methods proved that the winding process is a non-linear system. Employing this LLNF simulator model in developing model-based fault diagnosis and fault tolerant control systems of winding processes seems to be a worthwhile direction for future research. Moreover, developing NF piecewise linear predictor models of the winding process could be considered as another direction for further research toward designing a model predictive controller.

Acknowledgements

The authors gratefully acknowledge Thierry BASTOGNE with the University of Henri Point Care, who provided us with the winding process data.

References

Babuska, R., Verbruggen, H., 2003. Neuro-fuzzy methods for nonlinear system identification. *Ann. Rev. Control*, **27**(1): 73-85. [doi:10.1016/S1367-5788(03)00009-9]

Banadaki, H.D., Nozari, H.A., Kakahaji, H., 2011. Nonlinear simulator model identification of a walking beam furnace using recurrent local linear neuro-fuzzy network. *Int. J. Control Autom.*, **4**(4):123-134.

Bastogne, T., Noura, H., Sibille, P., Richard, A., 1998. Multivariable identification of a winding process by subspace methods for tension control. *Control Eng. Pract.*, **6**(9): 1077-1088.

Braatz, R.D., Ogunnaike, B.A., Featherstone, A.P., 1996. Identification, Estimation and Control of Sheet and Film Processes. 13th IFAC World Congress, p.319-324.

Ebler, N.A., Arnason, R., Michaelis, G., D′Sa, N., 1993. Tension control: dancer rolls or load cells. *IEEE Trans. Ind. Appl.*, **29**(4):727-739. [doi:10.1109/28.231986]

Hoshino, I., Maekawa, Y., Fujimoto, T., Kimura, H., Kimura, H., 1988. Observer-based multivariable control of the aluminum cold tandem mill. *Automatica*, **24**(6):741-754. [doi:10.1016/0005-1098(88)90050-7]

Hussein, E.L., Sheta, A., El Wahab, A.A., 2001. Modeling of a Winding Machine Using Non-parametric Neural Networks. WSEAS Int. Conf. on Scientific Computation and Soft Computing, p.528-533.

Hussian, A., Sheta, A., Kamel, M., Telbaney, M., Abdelwahab, A., 2000. Modeling of a Winding Machine Using Genetic Programming. Proc. Congress on Evolutionary Computation, p.398-402.

Ljung, L., 1987. System Identification Theory for the User. Prentice Hall, Upper Saddle River, NJ.

Nelles, O., 1996. Local Linear Model Tree for On-line Identification of Time Variant Non-linear Dynamic Systems. Int. Conf. on Artificial Neural Networks, p.115-120.

Nelles, O., 2001. Nonlinear System Identification. Springer Verlag, Berlin.

Nelles, O., Isermann, R., 1996. Basis Function Networks for Interpolation of Local Linear Models. IEEE Conf. on Decision and Control, p.470-475.

Noura, H., Theilliol, D., Ponsart, J.C., Chamseddine, A., 2009. Fault-Tolerant Control Systems: Design and Practical Applications. Springer-Verlag London Limited.

Parant, F., Iung, C., Bello, P., 1989. Traction and Speed Control. Third E.P.E. Conf., p.1417-1419.

Parant, F., Coeffier, C., Iung, C., 1992. Modeling of web tension in a continuous annealing line. *Iron Steel Eng.*, p.46-49.

Razavi-Far, R., Davilu, H., Palade, V., Lucas, C., 2009. Model-based fault detection and isolation of a steam generator using neuro-fuzzy networks. *Neurocomputing*, **7**(13-15):2939-2951. [doi:10.1016/j.neucom.2009.04.004]

Sadeghian, M., Fatehi, A., 2011. Identification, prediction and detection of the process fault in a cement rotary kiln by locally linear neuro-fuzzy technique. *J. Process Control*, **21**(2):302-308. [doi:10.1016/j.jprocont.2010.10.009]

Sievers, L., Balas, M.J., von Flotow, A., 1988. Modeling of web conveyance systems for multivariable control. *IEEE Trans. Autom. Control*, **33**(6):524-531. [doi:10.1109/9.1247]

SISTA, 1999. DaISy: Database for the Identification of Systems. Available from http://homes.esat.kuleuven.be/~smc/daisy/ [Accessed on Mar. 24, 2011].