Time series modelling of a radial-axial ring rolling system

Oscar Bautista Gonzalez* and Daniel Rönnow

Department of Electrical Engineering, Mathematics and Science,
University of Gävle,
80176 Gävle, Sweden
Email: oscar.bautista.gonzalez@hig.se
Email: daniel.ronnow@hig.se
*Corresponding author

Abstract: In the present work, a digital twin of a radial-axial ring rolling machine was built by modelling the time series of the positions of the tools and control signals rather than the metrics of the produced rings, as performed in previous studies. Real data from the industry was used for modelling. The used model selection methodology is shown in detail to replicate such work for similar systems in the steel industry. The modelling results of ARX, ARMAX and orthonormal basis model structures are shown; additionally, they were validated considering SISO and MIMO systems. The modelling results were better when the subsystems considered were ARMAX and MISO than when ARX and SISO were taken into consideration. The best modelling results were obtained when physical knowledge was included in the model structure. Lastly, it was found that the model error of the horizontal subsystem could be used for predictive maintenance.

1 Introduction

Digital twins are used for modelling the behaviour of different machines. There is an abundance of methods for modelling various types of behaviour in machines in different industries. This paper is about modelling time-series of a radial-axial ring rolling (RARR) machine in steel industry using a behavioural model (or black box) rather than a physical one.

Digital twins are virtual or digital representations of physical entities (Roy et al., 2020), though there are some variations in the definition (He and Bai, 2021). Digital twins are foreseen to play an essential role in manufacturing for, e.g., process simulation and equipment status monitoring (He and Bai, 2021). In Xiang et al. (2018), it is discussed how digital twins could be used in interactions in the steel industry for the entire product life cycle. Different types of digital twins are, thus, needed for different parts of the

product life cycle. Digital twins can be used for modelling physical processes in steel production. For example, in Samodurova et al. (2020), a digital twin was used for modelling the physical parameters that affect steel quality in a rolling process for various rod profiles. The physical process and the mechatronics of rolling-drawing wire mills were modelled by a digital twin (Karandaev et al., 2021).

Digital twins and machine learning are used in the steel industry in applications related to maintenance. In Björsell and Dadash (2021), a model of interconnected mills is presented. It was suggested that health parameters could be monitored and that the workload on the different subsystems could be controlled to increase the lifetime. Bampoula et al. (2021) used a deep learning approach for assessing the condition and predict the need for the maintenance of a rolling milling machine. The prediction of the performance of an air blast system in the steel industry is presented in Raducan et al. (2020). A deep learning technique was used in Karagiorgou et al. (2020) to predict the remaining useful lifetime in a use case where data came from various sensors in the steel industry. Health parameters were modelled for machinery in steel sheet production using machine learning in Ruiz-Sarmiento et al. (2020). A predictive model was used for estimating the mechanical properties of hot-rolled strips (Chen and Fan, 2).

Ring mills are used for producing steel rings at high temperatures. Several aspects of ring-forming have been investigated scientifically. The physics of the ring forming process has been studied. In Quagliato et al. (2018), a finite element method (FEM) was used for estimating radial and axial forces and, in Zhao and Qian (2010), an FEM method was used to study the effect of the rolling ratio on ring forming. The energy consumption was estimated by a machine learning method in Mirandola et al. (2021). Neural networks were used to predict geometrical distortion (Bruschi et al., 2005). Fahle et al. (2020) investigated several methods for preprocessing time-series data from a radial axis ring mill. Subsequently, Fahle et al. (2020) compared several machine learning methods to classify ring ovality (Fahle et al., 2021).

This work is about modelling a ring mill using different models. Model selection and identification is therefore a crucial part of the study. According to Söderström and Stoica (1989), the models for dynamic systems can be white-box or black-box models. White-box models are based on physical principles, e.g., Newton's second law for modelling mechanical systems (Qian et al., 2009). The design of such models is time-consuming and assumptions are quite restrictive for systems in complex environments. Black-box models require data for estimating their parameters. Deductions are still needed but are less stringent than white-box models. The time spent by the user is relatively low, and it is invested in data preprocessing, model structure selection and validation (Mattsson et al., 2019).

Black-box models can be parametric or non-parametric. A parameter vector defines parametric models, e.g., linear regression models, which can be estimated through different methods. Methods such as least squares, maximum likelihood and maximum a posteriori are well-known. Curve or functions define non-parametric models, e.g., impulse-responses (Söderström and Stoica, 1989). Parametric models assume a model structure that can be different from the true one. Non-parametric models do not consider any model structure, but the number of parameters is much higher than that in parametric ones, in general (James et al., 2013). Additionally, more data are required. This work is focused on black-box parametric models.

ARX, ARMAX and ANN models are commonly used for modelling the time series of dynamical systems (Akouemo and Povinelli, 2017; Moeeni and Bonakdari, 2018). A model's output is a linear combination of the parameters for ARX models, whereas it is a nonlinear function of the inputs for ANN. Therefore, interpreting the weight of each input to the output is much easier in ARX models. Choosing models with higher flexibility also means more parameters to tune and an increase in the risk of overfitting. The parsimony framework can counter these aspects, i.e., if a linear model describes the measured data well enough, it is better than a nonlinear model. On the other hand, if nonlinearities and correlated noise sources characterise measured data, ANN outperforms ARX due to its higher flexibility. Regularisation techniques can reduce the overfitting risk when in need of flexible models (Song l., 2022).

Although time series modelling of physical system has been a field of research for some time, it is still a field in progress. Recently developed algorithms based on fractional calculus allows identifying input and nonlinear ARX models (Chaudhary et al., 2021). New identification methods based on stochastic information gradient algorithms have been developed in the last decade, where the probability density function of the errors is used instead of the standard least square criteria (Jing, 2022). A method for automatic determination of the model order by general spectral decomposition before parameter estimation of ARX systems was developed recently (Maurya et al., 2021). A new study motivated by having unknown contacting forces between a robot and the workpiece has succeeded in estimating the unknown forces by implementing a Kalman filter after identifying an ARX system (Nguyen et al., 2022).

This work is the first to model RARR systems using time-series regression models of the dynamical systems theory. Whereas previous studies on RARR systems have focused on modelling the ring forming process, this paper gives a model of position of the machine tools, see Quagliato et al. (2018), Zhao and Qian (2010), Bruschi et al. (2005) and Fahle et al. (2021). To our knowledge, modelling the tools position has not been adressed before, neither has black or white box mdodels. Black box models have the advantage of low complexity – compared to, e.g., FEM methods – and could be used in online machine monitoring, c.f., online monitoring at a real continuous steel casting production line in the steel industry (Zhang et al., 2016). Here, the reader can find a systematic method to build up a digital twin and a comparison of black-box

models' results. By encoding system knowledge within the modelling, grey-box models are also considered in the comparison. The resultant models can be used to detect when the system deviates from normal conditions and could be helpful in the estimation of the remaining useful lifetime in predictive maintenance. Additionally, the derived models could be used as digital twins in applications in the steel industry.

The layout of the paper is as follows. In Section 2, the RARR machine and the analysed signals are explained. In Section 3 the model selection is described in detail. In Section 4, the different models considered on this work are described. The identification results for each subsystem and model are indicated in Section 5. Section 7 includes the evaluation of such models for some potential uses. Section 8 contains the conclusions.

2 System

The radial–axial ring rolling (RARR) machine from Banning (1971) is shown in Figure 1. It can be divided into five subsystems (see Figure 2). The mandrel and the primary roll are components of the horizontal subsystem (1); the lower and upper canonical rolls make up the axial subsystem (2); both canonical rolls are attached to the vertical subsystem (3). Finally, the left (4) and right arms (5), with their corresponding guiding rolls, complete the system.

Figure 1 Ring forming using a radial axial ring rolling machine (see online version for colours)

Hundreds of different ring types are manufactured with this system. Each ring type has different dimensions and shapes. The process operates in a sequence of three operations.

1 During the initialisation, the tools move towards the ring at high speed.

2 A lower speed movement towards the ring characterises the pre-rolling step.

3 In rolling, ring forming occurs due to the contact between the tools and the ring.

Figure 2 Mechanical scheme of the system (see online version for colours)
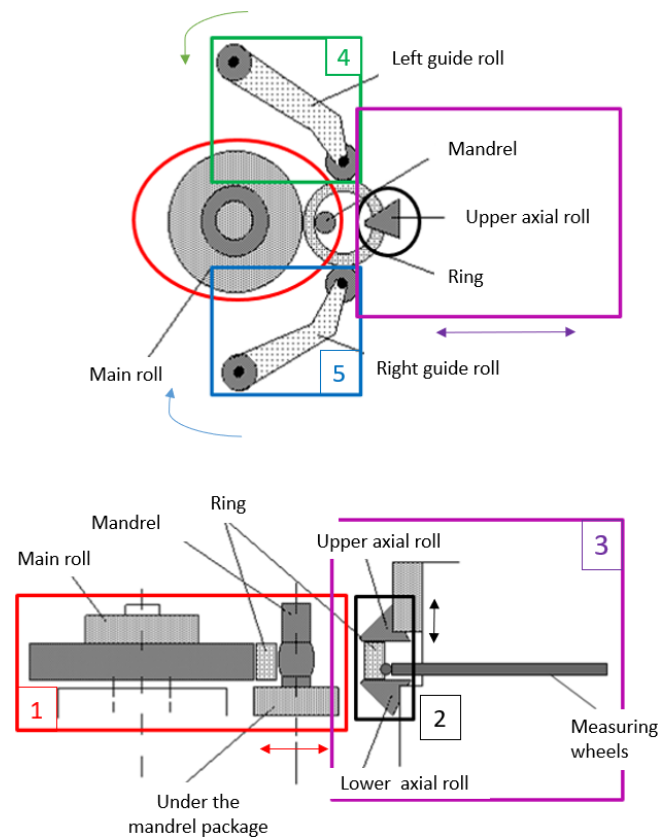
Notes: Five subsystems:
1 horizontal tool
2 axial rools
3 vertical tower
4 left arm
5 right arm.

Changing ring type requires the calibration of the subsystems and different tolling of the subsystems. Maintenance activities are performed regularly.

For each subsystem, there is a reference signal, a control signal and an output signal, which are the measured positions of the respective tools. Here, data from sensors of the different subsystems were collected during production, using an Iba-DAQ acquisition system with a 0.01 s sampling time. Despite being given by default, the sampling time is assumed to be fast enough based on the signals' characteristics, see Section 3.

3 Model selection

This work is in the field of analysis of the time series of physical systems. We followed the methodology described in Figure 3 for model selection, estimation and validation, based on the work by Söderström and Stoica (1989).

Real data from the production time of one ring type were obtained. In collaboration with the system's experts, the control and position signals were selected as input and output, respectively, for each of the five subsystems. The

signals corresponding to the production of one ring, i.e., one batch, were selected as training data for estimating the parameters (see Section 2) and the data for the next produced ring were selected as test data for validation. In both cases, the rolling part of the signals was considered.

1   *Data analysis:* The time invariance, stability, memory and causal properties of the system were analysed in the time domain (see Figure 4). The system's linearities were studied in the frequency domain (see Figure 5). After the last two analyses, the system was assumed to be a linear time-invariant (LTI) system. Then, a correlation analysis was performed to investigate the linear dependency among the subsystems.

   ●   *Time analysis:* The input and output signals shown in Figure 4 suggested a system that could be described as time invariant, bounded-input and bounded-output (BIBO) stable, dynamic and causal.

Figure 3   Modelling methodology flowchart (see online version for colours)
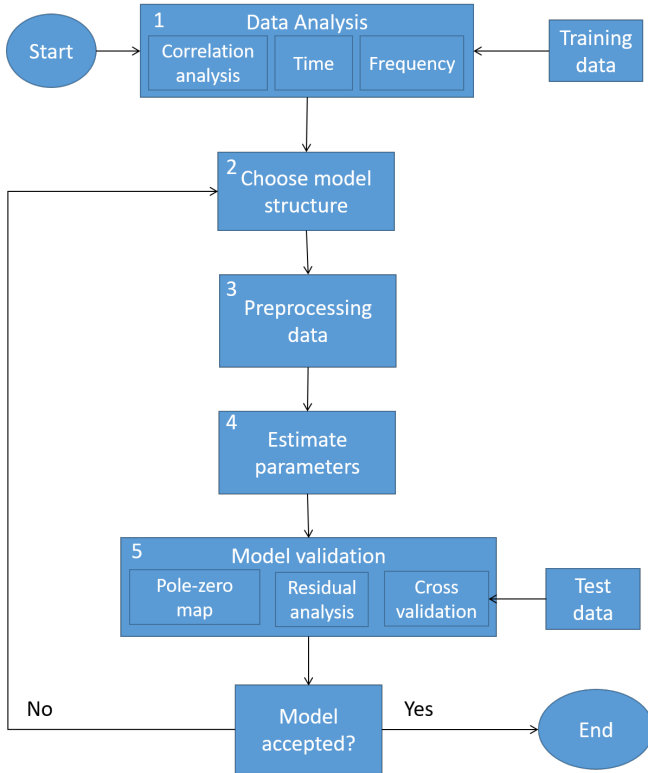
   ●   *Frequency analysis:* There were no new frequency components in the output signal compared to the input (see Figure 5). The system could be considered to be linear.

   ●   *Correlation analysis:* The linear dependence among the outputs of the different subsystems was analysed. Additionally, the correlation among the signals from each subsystem for different

produced rings was studied. After the analyses, similarities among the variations in the dynamics among subsystems were observed. Three similar model structures were expected after this analysis, one for the horizontal and axial tools, one for the vertical tower and one for both arms.

Figure 4   Signals from different rings in time-domain from subsystem 4, (a) output (b) input (see online version for colours)
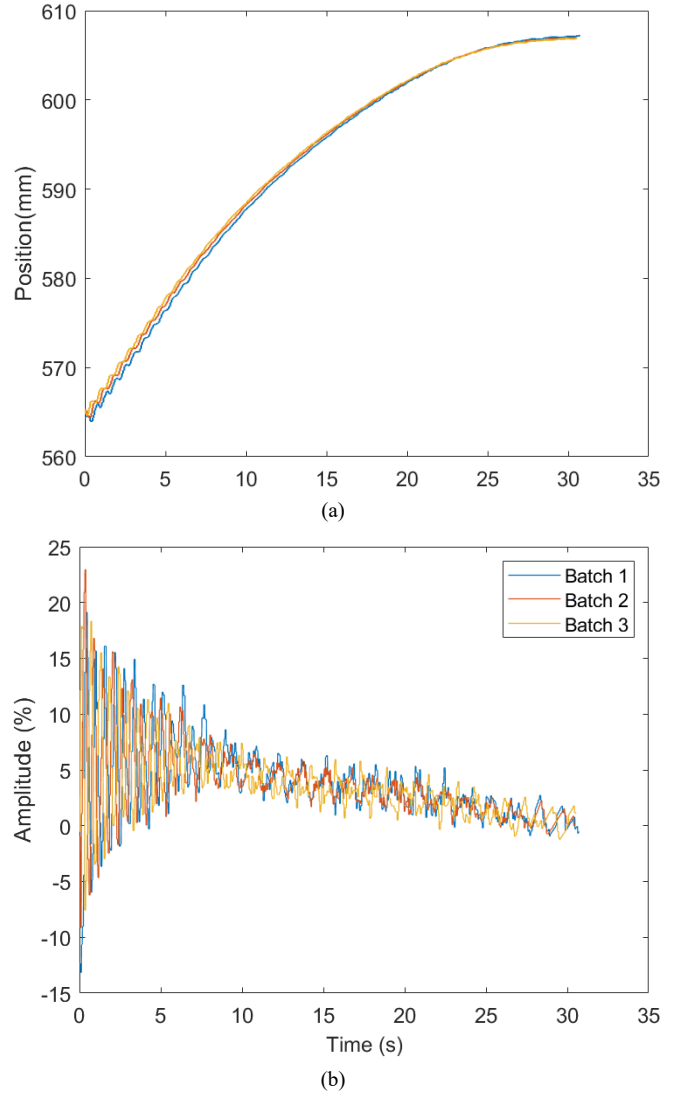
2   *Model structure selection:* After establishing that the time series of an LTI system were under study, ARX and ARMAX were considered as possible model structures of the various subsystems. This choice gave good modelling results for all the subsystems but not as good as expected in subsystems 4 and 5 (see Section 5). For subsystems 4 and 5, modelling using orthonormal bases was also considered after some iterations following the schematic flowchart in Figure 3. The correlation analysis indicated dependency among the signals from different subsystems. Therefore, we considered SISO and MISO structures.

Figure 5 Signals of subsystem 1 in frequency domain (see online version for colours)
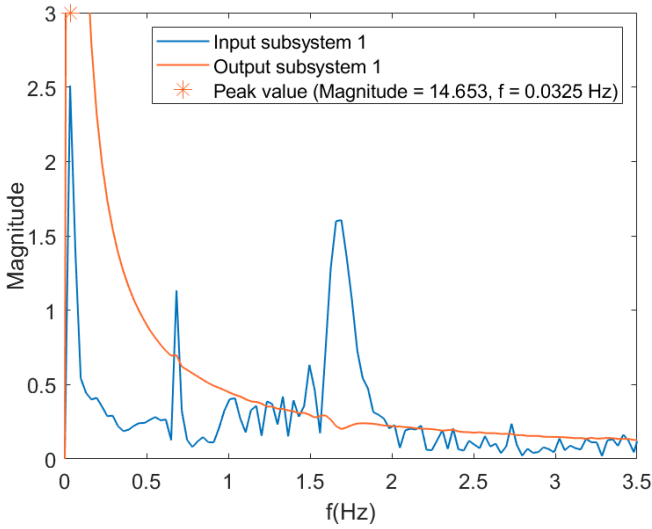
3 *Data cleaning:* After checking the samples for the training and test datasets in the time domain, the datasets were considered clean. The preprocessing of the data was performed by removing the mean value of the input and output signals for modelling the subsystems based on prediction error methods.

4 *Parameter estimation:* The least-squares criterion was chosen to estimate the parameters as rule of thumb. The least-squares method was used to estimate ARX models' parameters and the parameters of the models based on orthonormal bases. The System Identification Toolbox's function *arx* in MATLAB was used for estimating ARX models' parameters and the ones of the models based on orthonormal bases. Iterative methods were used for ARMAX, which is a nonlinear model in terms of its parameters [see more information in Subsection 10.2 in Ljung (1987)]. The function *armax* of the System Identification Toolbox in MATLAB was used (Ljung, 1995).

5 *Model validation:* The cross-validation with test data, which is different from that used for estimating the parameters, pole-zero maps and a residual analysis were considered for model validation. Cross-validation was used for checking the accuracy of the model with data from a different batch through the indicator

$$NRMSE = 1 - \frac{||\mathbf{y} - \hat{\mathbf{y}}||}{||\mathbf{y} - \overline{\mathbf{y}}||}, \qquad (1)$$

where $\mathbf{y}$ is an $(N \times 1)$ vector containing the real output of the subsystem, $\hat{\mathbf{y}}$ is an $(N \times 1)$ vector containing the model's output, $\overline{\mathbf{y}}$ is an $(N \times 1)$ vector containing the mean value of $\mathbf{y}$ at all the entries and $||.||$ denotes the Euclidean norm. The cancellation between the zeros and poles of the difference equations was analysed with pole-zero maps, besides the stability of the models. A good selection of the order and model structure was also analysed with a

residual analysis, expecting white noise as residual for the best case.

After model validation, a decision upon the model was performed as is indicated in the flowchart diagram of Figure 3. If there was no overlapping between the zeros and poles of the model, the residual analysis indicated relatively small errors between the real output of the subsystem and the model's output; the performance of the model with the test data was relatively similar to that obtained with the training data; the model was accepted and the modelling of the subsystem was finished as consequence. In another case, the modelling was restarted by changing the order of the model or the identification method, e.g., considering ARMAX instead of ARX.

4 Models

A LTI set of models for MIMO systems with $\mathbf{y}(t)$ as the $(m \times 1)$ output vector, $\mathbf{u}(t)$ as the $(n \times 1)$ input vector and $\mathbf{e}(t)$ as the $(m \times 1)$ vector containing white-noise processes of finite variance on each entry is given, in Ljung (1987), by

$$\mathbf{y}(t) = G(q, \theta)\mathbf{u}(t) + H(q, \theta)\mathbf{e}(t), \qquad (2)$$

where $G(q, \theta)$ and $H(q, \theta)$ are $(m \times n)$ and $(m \times m)$ matrices, respectively, with rational functions of q as entries; θ is the parameter vector which is an element of a subset of $\mathbb{R}^d$. The matrix transfer functions can be factorised as

$$G(q, \theta) = A^{-1}(q)B(q), H(q, \theta) = A^{-1}(q)C(q) \qquad (3)$$

by matrix fraction description (MFD), where

$$\begin{aligned}
A(q) &= I + A_1 q^{-1} + A_2 q^{-2} + ... + A_{na} q^{-na}, \\
B(q) &= B_1 q^{-1} + B_2 q^{-2} + ... + B_{na} q^{-nb}, \\
C(q) &= I + C_1 q^{-1} + C_2 q^{-2} + ... + C_{na} q^{-nc}.
\end{aligned} \qquad (4)$$

The model structure given by equations (2)–(4) is identifiable under the conditions described in Theorem 4A.1 and Corollary 4A.1 in Ljung (1987). θ includes all the coefficients of the matrix polynomials in equation (4), where $A(q)$ is an $(m \times m)$, $B(q)$ is an $(m \times n)$ and $C(q)$ is an $(m \times m)$ matrix polynomial.

The identification of the parameters in θ was performed using the given training dataset $D = \{(\mathbf{u}(1), \mathbf{y}(1)), ..., (..., \mathbf{u}(N), \mathbf{y}(N))\}$ (see below for the specific models).

4.1 ARMAX

The model structure in equations (2)–(4) is the general ARMAX model. The estimation of the parameters in θ was made using iterative methods, since the model is nonlinear in θ. Considering the theory in Ljung (1987) and Söderström and Stoica (1989), as well as a MISO or SISO model structure, the optimal linear predictor can be written as

$$\hat{y}(t|\theta) = \boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{\varphi}(t,\theta), \qquad (5)$$

where

$$\boldsymbol{\theta} = \begin{pmatrix} a_1 \dots a_{na} \, b_{11} \dots b_{nb,n} \, c_1 \dots c_{nc} \end{pmatrix}^{\mathsf{T}} \qquad (6)$$

is the $((na + n \cdot nb + nc) \times 1)$ parameter vector and $\boldsymbol{\varphi}(t)$ is the $((na + n \cdot nb + nc) \times 1)$ regressor vector. Then,

$$\hat{\boldsymbol{\theta}} = \arg\min_{\Theta} \frac{1}{N} \sum_{t=1}^{N} (y(t) - \hat{y}(t|\theta))^2 \qquad (7)$$

becomes a minimisation problem of a nonlinear function in $\boldsymbol{\theta}$, which is a specific value of the parameter space $\Theta \subseteq \mathbb{R}^{(na+n\cdot nb+nc)}$, where iterative methods are needed.

4.2 ARX

In the case of ARX, equation (3) becomes

$$G(q, \theta) = A^{-1}(q)B(q), H(q, \theta) = A^{-1}(q). \qquad (8)$$

which is linear in the parameters

$$\boldsymbol{\theta} = \begin{pmatrix} A_1 \dots A_{na} \, B_1 \dots B_{nb} \end{pmatrix}^{\mathsf{T}} \qquad (9)$$

which with the $((na + nb) \times 1)$ vector $\boldsymbol{\varphi}(t)$, gives the least-squares criterion

$$\hat{\boldsymbol{\theta}} = \arg\min_{\Theta} \frac{1}{N} \sum_{t=1}^{N} ||\boldsymbol{y}(t) - \boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{\varphi}(t)||^2 \qquad (10)$$

where (Ljung, 1989)

$$\hat{\boldsymbol{\theta}} = \left(\frac{1}{N} \sum_{t=1}^{N} \boldsymbol{\varphi}(t)\boldsymbol{\varphi}^{\mathsf{T}}(t) \right)^{-1} \frac{1}{N} \sum_{t=1}^{N} \boldsymbol{\varphi}(t)\boldsymbol{y}^{\mathsf{T}}(t). \qquad (11)$$

4.3 Orthonormal bases

The model structure in equation (2) was considered for this case, where the dynamics between input and output are described by $G(q, \theta)$, which is an $(m \times n)$ matrix, where each entry is a rational scalar transfer function $G_{ij}(q, \theta)$ in q and describes the dynamics between input i and output j. $H(q, \theta)$ is an $(m \times m)$ identity matrix.

When using orthonormal basis functions, the aim is to express $G(q, \theta)$ as a truncated series expansion of a finite number of bases.

$$G(q, \theta) = \sum_{k=0}^{p-1} \boldsymbol{\theta}_k^{\mathsf{T}} \mathcal{B}_k(q), \qquad (12)$$

where $\mathcal{B}_k$ is a scalar rational transfer function. Then, the model structure in equation (2) can be expressed as

$$\boldsymbol{y}(t) = \begin{pmatrix} \boldsymbol{\theta}_0^{\mathsf{T}} \, \boldsymbol{\theta}_1^{\mathsf{T}} \dots \boldsymbol{\theta}_{p-1}^{\mathsf{T}} \end{pmatrix} \begin{pmatrix} D_0(q) \\ D_1(q) \\ \vdots \\ D_{p-1}(q) \end{pmatrix} \boldsymbol{u}(t) = \theta^{\mathsf{T}}\phi(t) \,(13)$$

where $D_k(q) = I_n\mathcal{B}_k(q)$, $\boldsymbol{\theta}_k$ is a $(n \times m)$ matrix, θ an $(np \times m)$ matrix and $\phi(t)$ an $(np \times 1)$ vector.

Prior information about the system was used in the basis functions $\mathcal{B}_k(q)$, the construction of which is

$$\mathcal{B}_k(q) = \left(\frac{\sqrt{1 - |\xi_k|^2}}{q - \xi_k} \right) \prod_{i=0}^{k-1} \left(\frac{1 - \overline{\xi_i}q}{q - \xi_i} \right). \qquad (14)$$

Prior system information was used in the selection of the poles $\{\xi_0, \xi_1, ..., \xi_{p-1}\}$, which should be close to the true poles of the system $G(q)$ (Ninness, 1994).

Then, the least-square method

$$\hat{\theta} = \arg\min_{\Theta} \frac{1}{N} Tr \left(\sum_{t=0}^{N-1} \varepsilon(t, \theta)\varepsilon^{\mathsf{T}}(t, \theta) \right), \qquad (15)$$

gives

$$\hat{\theta} = \Phi^{\dagger}Y, \qquad (16)$$

where

$$\begin{aligned} Y &= \begin{pmatrix} \boldsymbol{y}(1) \, \boldsymbol{y}(2) \dots \boldsymbol{y}(N) \end{pmatrix}^{\mathsf{T}}, \\ \Phi &= \begin{pmatrix} \boldsymbol{\phi}(1) \, \boldsymbol{\phi}(2) \dots \boldsymbol{\phi}(N) \end{pmatrix}^{\mathsf{T}}, \end{aligned} \qquad (17)$$

and

$$\varepsilon(t, \theta) = \boldsymbol{y}(t) - G(q, \theta)\boldsymbol{u}(t). \qquad (18)$$

5 Results

This section is divided into three subsections: the first section reports the results of the modelling of the horizontal and axial subsystems; the second one, the results of the modelling of the vertical subsystem; in the last one, we describe the modelling results for the arms of the RARR. The grouping of the subsystems was made based on similarities in the model structure of the respective subsystems. This conclusion was drawn from the correlation analysis results of the data analysis described in Section 3.

Table 1 Training and validation results (NRMSE in %) for models used for subsystem 1

Model	Type	Train	Val.	#p
ARX	SISO	0.470	0.507	40
	MISO (1-1,2)	0.485	0.527	60
ARMAX	SISO	0.991	0.991	12
	MISO (1-1,2)	0.994	0.990	16
	Reciprocity	0.988	0.987	12

Notes: The number of parameters of each case is included in column #p. (1-1,2) indicates that output of subsystem 1 and inputs of subsystems 1 and 2 are considered in the model structure.

Considering the ARX SISO and ARMAX SISO cases in Tables 1, 2, 4, 6 and 7, A_i and B_i in equation (4) became scalar. For ARX SISO, $C(q) = 1$, whereas, for ARMAX SISO, C_i is scalar. ARX MISO and ARMAX MISO also had a scalar A_i, but B_i is a $(1 \times n)$ vector. $C(q) = 1$ for ARX MISO and C_i is scalar for ARMAX MISO. The derivation of the computational cost in Tables 3, 5 and 8 is explicitly described.

Table 2 Training and validation results (NRMSE in %) for models used for subsystem 2

Model	Type	Train	Val.	#p
ARX	SISO	0.225	0.224	40
	MISO (2-1,2)	0.346	0.392	60
ARMAX	SISO (unstable)	0.990	0.987	12
	MISO (2-1,2)	0.988	0.989	16
	Reciprocity	0.990	0.991	12

Notes: The number of parameters for each case is included at column #p. (2-1,2) indicates that output of subsystem 2 and inputs of subsystems 1 and 2 are considered in the model structure.

Table 3 Computational cost for models used for subsystems 1 and 2 in terms of multiplications and additions

Model – type	Mult.	Additions	#p
ARX	1,640 N + 21,360	1,640 N + 19,680	40
SISO	40	39	
ARX	3,660 N + 72,040	3,660 N + 68,320	60
MISO	60	59	
ARMAX	216 kN + 908 k	216 kN + 428 k	12
SISO	12	15	
ARMAX	336 kN + 1,936 k	336 kN + 1,104 k	16
MISO	16	19	
ARMAX	220 kN + 908 k	220 kN + 428 k	12
Reciprocity	16	19	

Notes: The first row of each model-type pair corresponds to the cost of estimating the parameters, and the second is the cost of computing the optimal one-step predictor. k denotes the number of iterations and N the number of samples.

Table 4 Training and validation results (NRMSE in %) for models used for subsystem 3

Model	Type	Train	Val.	#p
ARX	SISO	73.76	77.08	8
	MISO (3-3,2)	76.06	79.25	21
ARMAX	SISO	75.16	78.1	17
	SISO	80.92	83.71	20
	MISO (3-3,2)	80.18	83.16	32
	MISO (3-3,2)	98.39	98.1	62

Notes: The number of parameters for each case is included at column #p. (3-3,2) indicates that output of subsystem 3 and inputs of subsystems 3 and 2 are considered in the model structure.

The ARMAX MISO model becomes [cf. equation (2)], for subsystems 1 and 2,

$$y_1(t) = \big(G_{11}(q)\; G_{12}(q)\big)\, \boldsymbol{u}(t) + H_1(q)e_1(t),$$

and

$$y_2(t) = \big(G_{21}(q)\; G_{22}(q)\big)\, \boldsymbol{u}(t) + H_2(q)e_2(t),$$

respectively. In the case of reciprocity, subsystem 1 becomes

$$y_1(t) = \big(G_{11}(q)\; G_{12}(q)\big)\, \boldsymbol{u}(t) + H_1(q)e_1(t)$$

and subsystem 2 becomes

$$y_2(t) = \big(G_{12}(q)\; G_{22}(q)\big)\, \boldsymbol{u}(t) + H_2(q)e_2(t).$$

Table 5 Computational cost for models used for subsystem 3 in terms of multiplications and additions

Model – type	Mult.	Additions	#p
ARX	72 N + 176	72 N + 96	8
SISO	8	7	
ARX	462 N + 3,110	462 N + 2,618	21
MISO	21	20	
ARMAX	408 kN + 2,278 k	408 kN + 1,343 k	17
SISO	22	21	
ARMAX	560 kN + 3,540 k	560 kN + 2,260 k	20
SISO	26	25	
ARMAX	1,312 kN + 13,088 k	1,312 kN + 9,888 k	32
MISO	39	38	
ARMAX	4,836 kN + 87,358 k	4,836 kN + 75,578 k	62
MISO	76	75	

Notes: The first row of each model-type pair corresponds to the cost of estimating the parameters, and the second is the cost of computing the optimal one-step predictor. k denotes the number of iterations and N the number of samples.

Table 6 Training and validation results (NRMSE in %) for models used for subsystem 4

Model	Type	Train	Val.	#p
ARX	SISO	0.427	0.487	2
	MISO (4-2,4)	0.432	0.492	9
ARMAX	SISO	0.884	0.849	12
	MISO (4-2,4)	0.961	0.924	12
Orth. bases	SISO	0.956	0.868	12 (2)
	MISO (4-2,4)	0.944	0.921	12 (2)

Notes: The number of parameters for each case is included at column #p. (4-2,4) indicates that output of subsystem 4 and inputs of subsystems 2 and 4 are considered in the model structure.

Table 7 Training and validation results (NRMSE in %) for models used for subsystem 5

Model	Type	Train	Val.	#p
ARX	SISO	0.427	0.516	2
	MISO (5-2,5)	0.414	0.508	9
ARMAX	SISO	0.768	0.822	12
	MISO (5-2,5)	0.799	0.872	12
Orth. bases	SISO	0.962	0.763	12 (2)
	MISO (5-2,5)	0.941	0.906	12 (2)

Notes: The number of parameters for each case is included at column #p. (5-2,5) indicates that output of subsystem 5 and inputs of subsystems 2 and 5 are considered in the model structure.

Tables 1 and 2 present the modelling results of subsystems 1 and 2, respectively. ARX showed relatively small NRMSE values and a high number of parameters in comparison to ARMAX in both Tables 1 and 2. ARMAX SISO showed good training and validation results in both

cases, but it was unstable for subsystem 2. ARMAX MISO gave a stable model for subsystem 2. The comparison of ARMAX SISO and MISO in Table 1 showed no improvements of the model by considering information of subsystem 2. Considering reciprocity gave no improvements of subsystem 1, unlike subsystem 2. The number of parameters was also lower than for ARMAX MISO and it was stable as in the case of no reciprocity.

Table 8 Computational cost for models used for subsystems 4 and 5 in terms of multiplications and additions

Model – type	Mult.	Additions	#p
ARX	6 N + 4	6 N – 4	2
SISO	2	1	
ARX	90 N + 249	90 N + 150	9
MISO	9	8	
ARMAX	216 kN + 908 k	216 kN + 428 k	12
SISO	12	15	
ARMAX	336 kN + 1,936 k	336 kN + 1,104 k	16
MISO	16	19	
Orth.bases	324 N + 584	312 N + 416	12
SISO	168	155	
Orth.bases	252 N + 584	240 N + 416	12
MISO	96	83	

Notes: The first row of each model-type pair corresponds to the cost of estimating the parameters, and the second is the cost of computing the optimal one-step predictor. k denotes the number of iterations and N the number of samples.

Figure 6 Measured and modelled output signals of subsystem 1 for the investigated models (see online version for colours)
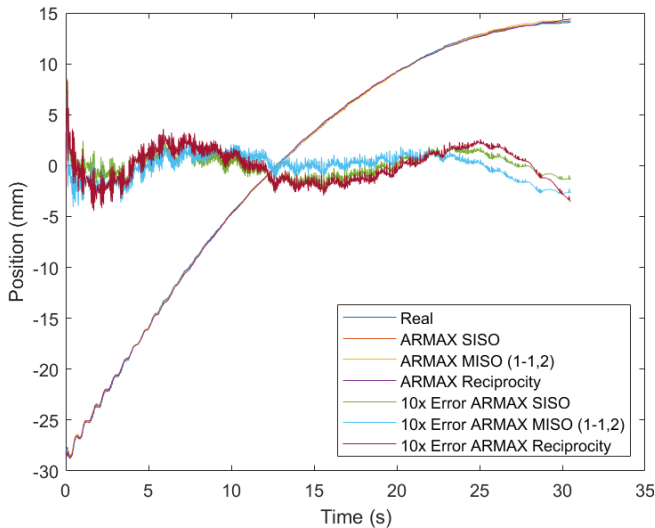
Note: Also show the respective error signals.

Figure 6 shows a comparison of measured and modelled output signals of subsystem 1. It can be seen that the model outputs were relatively similar to the measured ones. On the other hand, the error signals did not resemble white noise for the three models; the error signals suggested residues with 2 s and 10 s of periodicity. Thus, some information about the dynamics of the subsystem was

still in the residuals. This fact could have been due to the fact that we did not include the true model of the subsystems in the considered models when choosing the model structure. Another possibility is that the subsystems operated in feedback mode; we did not consider this fact in our identification process.

Table 3 presents the computational cost for estimating the parameters of each model used for subsystems 1 and 2 and the cost of computing the optimal one-step predictor for each method. The number of operations for ARX SISO and MISO increases with the number of parameters, but for ARMAX, it also increases with the number of iterations. In this case, the number of multiplications and additions is higher for the ARMAX methods than ARMAX considering that it takes 20 iterations to obtain the results in Tables 1 and 2. The number of operations for the ARMAX models is higher for the MISO case, and there is a smaller number of operations for ARMAX reciprocity compared to ARMAX MISO. For computing the one-step predictor, the number of operations is more significant for ARX models than ARMAX models. ARMAX reciprocity presents no advantage in computing the one-step predictor with respect to the ARMAX MISO.

5.1   System 3

The modelling results of subsystem 3 are presented in Table 4. The ARX models gave poorer results than the ARMAX models. In the case of the ARMAX models, the MISO models outperformed the SISO models in accuracy but at the cost of a higher number of parameters. For the MISO case, including information of subsystem 2 through the input signal $u_2(t)$, i.e., MISO models, increased the NRMSE of the training and validation sets. The difference between the real output of the subsystem and the model's outputs is shown in Figure 7. The error signals were not white noise and had a structure that suggested residues with time scales of 1 and 10 s.

Figure 7 Measured and modelled output and error signals of subsystem 3 (see online version for colours)
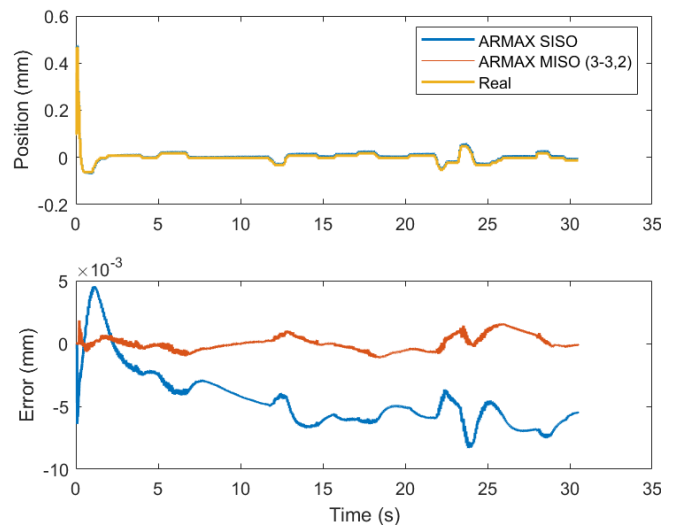
Comparing the modelling of subsystem 3 to that of subsystems 1 and 2, i.e., Tables 1 and 2, we noticed that the required number of parameters to obtain good models was approximately five times higher in this case. The error signals in Figure 7 had relatively small amplitudes compared to those of subsystems 1 and 2.

Table 5 indicates the computational cost of estimating the parameters for each of the models used for subsystem 3 and the cost of computing the optimal one-step predictor for each method. The number of operations for ARX SISO and MISO increases with the number of parameters, but the cost related to ARMAX methods also increases with the number of iterations. The number of operations for estimating the parameters of the ARMAX models is considerably bigger than the ARX ones considering that we needed 20 iterations to achieve the results indicated in Table 4. The case of ARMAX MISO with 62 parameters is the most computational demanding. The number of operations for computing the one-step predictor is also bigger for ARMAX methods, particularly for the ARMAX MISO with 62 parameters.

Comparing the computational requirements of subsystem 3 with subsystems 1 and 2, i.e., Table 3, we notice that the number of operations for the best model of each subsystem is bigger for subsystem 3 in both, estimating the parameters and computing the one-step predictor.

5.2 Systems 4 and 5

The modelling results of subsystems 4 and 5 are presented in Tables 6 and 7, respectively. ARX and ARMAX structures were used in the modelling. In addition to what reported in the previous subsections, orthonormal basis models were used. In the orthonormal basis case for SISO, the elements of θ_i in equation (13) are scalar, θ is a $(p \times 1)$ vector and $D_i(q) = \mathcal{B}_i(q)$ and ϕ_i are $(p \times 1)$ vectors. For the MISO case, the elements of equation (13) are as follows: θ_i is a (2×1) vector, θ is a $(2p \times 1)$ vector, $D_i(q) = \mathbb{I}_2 \mathcal{B}_i(q)$ and ϕ_i is a $(2p \times 1)$ vector. Notice that, with such configurations, equation (13) becomes scalar.

ARMAX MISO outperformed ARMAX SISO in both Tables 6 and 7. Modelling using orthonormal bases gave an increase in the model accuracy of subsystem 5, but not in that of subsystem 4, compared to the ARMAX models. Figure 8 shows the measured and modelled output signals of subsystem 4 and the corresponding error signals. The use of the orthonormal basis model introduced high-frequency components in the model's output. These high-frequency components appeared because the chosen poles were complex conjugates instead of real poles, making the basis function described in equation (14) transform, according to Nyness (1994), into a basis function which places a pair of complex conjugate poles defined by the user in the opposite real sides of the complex plane.

In comparison to the previous cases, the modelling by ARX and ARMAX of subsystems 4 and 5 gave poorer results than previous subsystems in terms of NRMSE,

but the number of parameters was in the same order. The use of orthonormal bases reduced the difference in model performance between the results of subsystem 5 and the previous ones. The number of parameters for the orthonormal basis model structure was 12. Additionally, two parameters were set by the user, i.e., the modulus and argument of the complex pole, ξ.

Figure 8 Measured and modelled output and error signals from subsystem 4 (see online version for colours)
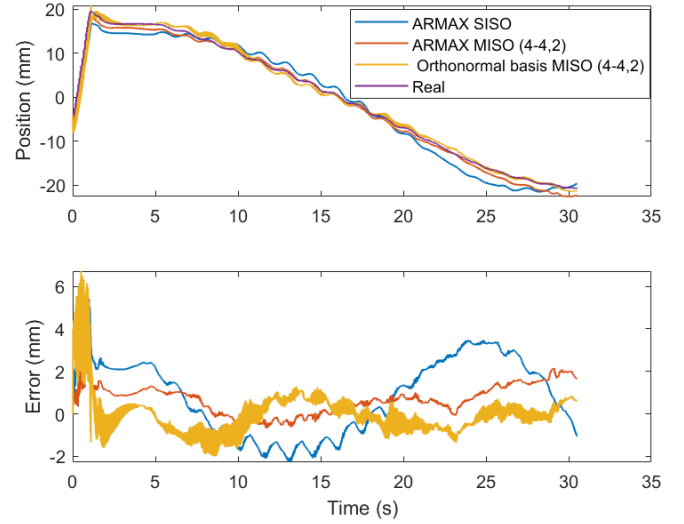

Table 8 indicates the computational cost of estimating the parameters for each model used for subsystems 4 and 5 and the cost of computing the optimal one-step predictor for each method. As for subsystems 1, 2 and 3, the number of operations for ARX SISO and MISO increases with the number of parameters, but for ARMAX methods, it also increases with the required number of iterations for solving equation (7). The number of operations for estimating the parameters of ARMAX models is considerably bigger than the ARX ones considering that 20 iterations are needed to achieve the results in Tables 6 and 7. The number of operations to estimate the parameters of models based on orthonormal bases is of the same order of magnitude as ARX models. However, the number of operations for computing the one-step predictor is bigger considering 12 parameters. The computation of 12 different IIR filters is needed to obtain the one-step predictor, where each filter requires $2o$ multiplications and $2(o - 1)$ additions, being o the order of the basis function.

Comparing the computational requirements of subsystems 4 and 5 to 1, 2 and 3, i.e., Tables 3 and 5, we notice that the number of operations for the more accurate model for each of the subsystems is bigger for subsystem 3 in terms of estimating the parameters but, for the recursion, it is subsystem 4 and 5 the one that requires more computations when selecting the models based on orthonormal bases.

6 Application

This section evaluates some potential uses of the resulting models of the previous section in ring manufacturing. Production is carried out in sequential cycles, where one ring is manufactured in each cycle. Additionally, the RARR is used to manufacture hundreds of different ring types. Therefore, the modelling performance over production cycles, as well as for different ring types, is analysed in this section.

Figure 9   Residuals evolving over cycles, (a) ARMAX MISO of subsystem 1 (horizontal tool) (b) ARMAX MISO of subsystem 2 (axial tool) (see online version for colours)


(a)

(b)
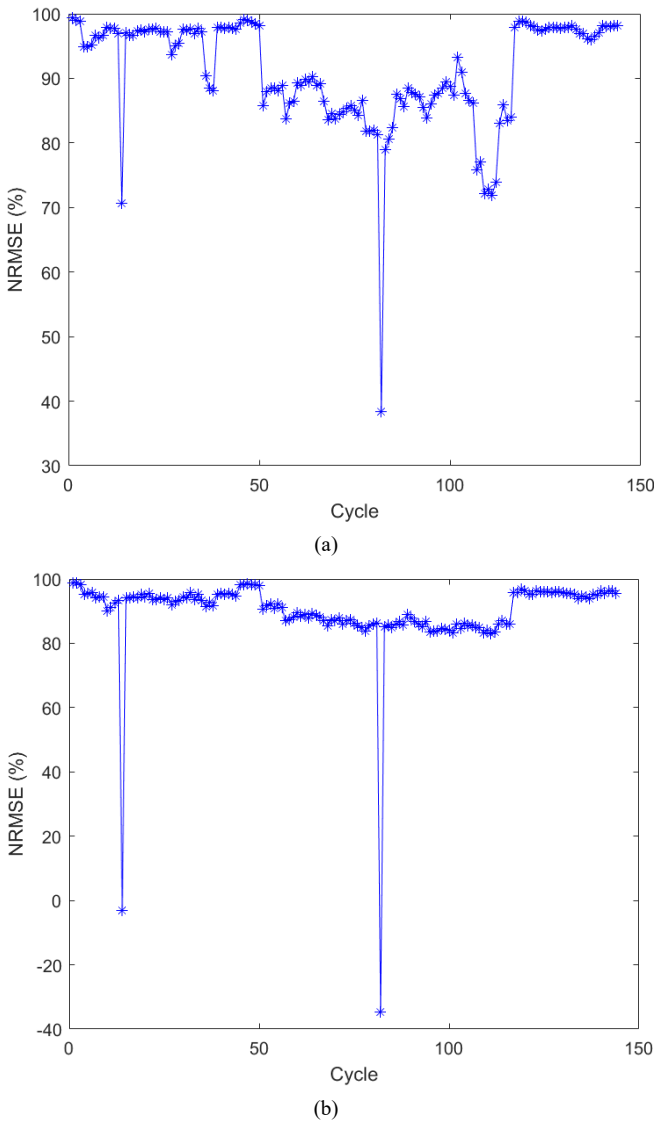
6.1 Residuals

The evolution of the NRMSE of the model's output when the same ring type is produced is presented in this section. This analysis was performed with the ARMAX MISO model for subsystems 1 and 2. The modelling results for the training and test data are given in Tables 1 and 2.

Figure 9 shows the NRMSE in percentage vs. cycle. Around cycles 10 and 80, there was a drop in the NRMSE of the models. The system's main mandrel was broken at this stage. Figure 9(a) – the ARMAX MISO's NRMSE for subsystem 1 – shows a decrease and larger variation in the NRMSE over cycles, specifically from cycle 50 to cycle 120. Before and after that, it was relatively stable. Figure 9(b), which shows the results for subsystem 2, shows a fairly clear decreasing trend in comparison with Figure 9(a). Thus, an application could be to use the model of this subsystem for the prognostics of the machine health status.

6.2 Validation with different ring types and after calibration

The model was also analysed after system's maintenance, change in ring type and calibration of the tools. Figure 10 shows the measured and modelled output of subsystem 1 for the ARMAX SISO case, which is shown in Table 1. In this case, the test data were from another ring type than the training data. The error signal indicated a worse performance than that reported in Figure 6, having, in this case, an NRMSE of 92.5%, whereas Table 1 shows 99% for ARMAX SISO.

Figure 10   Model validation using data from a different ring type, after calibration, and several maintenance actions upon subsystem 1 (see online version for colours)
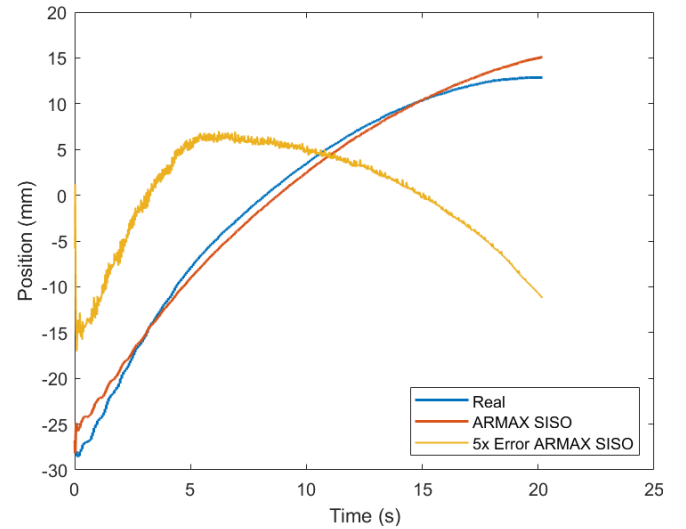
7 Conclusions

A rigorous methodology process was followed for model selection. After analysing the frequency content of the position and control signals of the different subsystems, the system was found to be linear. Moreover, comparing the signals of the same subsystem for different cycles showed that each subsystem was close to being time invariant. We conclude that the theory of LTI systems can be used to describe the dynamics of RARR systems under similar operating conditions.

Whereas RARR was modelled by considering a MIMO model structure, the correlation analysis between the position signals showed, during model selection, three groups of subsystems, with one group being the horizontal and axial tool, another being the vertical subsystem and, lastly, both arms forming a third subsystem type. This observation was also corroborated in the modelling results. ARMAX MISO presented the best NRMSE results for the first group, also having the reciprocity case being the best particular case for the axial subsystem. ARMAX MISO showed the best results for the vertical tool but with a higher number of parameters than the first group. Finally, using orthonormal bases, in which the signals' frequency content was considered, gave the best results for the arms. Therefore, we conclude that ARMAX and orthonormal basis functions are the best choice to model RARR systems under similar conditions.

The NRMSE of the ARMAX MISO model of subsystem 2 vs. cycle showed a decreasing trend. Therefore, this feature could be used for the prognostics of the machine health status. Analyses considering more data should be carried out to determine such a claim. The performance of models for different ring types from the one considered in the training data showed to be relatively poor. One model for each ring type is expected for future work.

Further analysis of time-series data will be carried out for predicting the remaining useful lifetime of the machine by studying trends such as in Figure 9(b).

References

Akouemo, H.N. and Povinelli, R.J. (2017) 'Data improving in time series using ARX and ANN models', *IEEE Transactions on Power Systems*, Vol. 32, No. 5, pp.3352–3359.

Bampoula, X., Siaterlis, G., Nikolakis, N. and Alexopoulos, K. (2021) 'A deep learning model for predictive maintenance in cyber-physical production systems using LSTM autoencoders', *Sensors*, Vol. 21, No. 3, p.972.

Björsell, N. and Dadash, A.H. (2021) 'Finite horizon degradation control of complex interconnected systems', *IFAC-PapersOnLine*, Vol. 54, No. 1, pp.319–324.

Bruschi, S., Casotto, S., Dal Negro, T. and Bariani, P.F. (2005) 'Real-time prediction of geometrical distortions of hot-rolled steel rings during cooling', *CIRP Annals*, Vol. 54, No. 1, pp.229–232.

Chaudhary, N.I., Raja, M.A.Z., Khan, Z.A., Cheema, K.M. and Milyani, A.H. (2021) 'Hierarchical quasi-fractional gradient descent method for parameter estimation of nonlinear ARX systems using key term separation principle', *Mathematics*, Vol. 9, No. 24, p.3302.

Fahle, S., Kneißler, A., Glaser, T. and Kuhlenkötter, B. (2020) 'Research on preprocessing methods for time series classification using machine learning models in the domain of radial-axial ring rolling', in *Congress of the German Academic Association for Production Technology*, Springer, Berlin, Heidelberg, September, pp.487–496.

Fahle, S., Glaser, T. and Kuhlenkötter, B. (2021) 'Investigation of machine learning models for a time series classification task in radial-axial ring rolling', in *Forming the Future*, pp.589–600, Springer, Cham.

He, B. and Bai, K.J. (2021) 'Digital twin-based sustainable intelligent manufacturing: a review', *Advances in Manufacturing*, Vol. 9, No. 1, pp.1–21.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013) *An Introduction to Statistical Learning*, Vol. 112, p.18, Springer, New York.

Jing, S. (2022) 'Multierror stochastic gradient algorithm for identification of a Hammerstein system with random noise and its application in the modeling of a continuous stirring tank reactor', *Optimal Control Applications and Methods*.

Karagiorgou, Rountos, C., Chatzimarkaki, G., Vafeiadis, G., Ntalaperas, D., Vergeti, D. and Alexandrou, D. (2020) 'On making factories smarter through actionable predictions based on time-series data', *Procedia Manufacturing*, Vol. 51, pp.1207–1214.

Karandaev, A.S., Gasiyarov, V.R., Radionov, A.A. and Loginov, B.M. (2021) 'Development of digital models of interconnected electrical profiles for rolling-drawing wire mills', *Machines*, Vol. 9, No. 3, p.54.

Ljung, L. (1987) 'Theory for the user', *System Identification*.

Ljung, L. (1995) *System Identification Toolbox: User's Guide*, MathWorks Incorporated, Natick, MA.

Mattsson, P., Zachariah, D. and Björsell, N. (2019) 'Flexible models for smart maintenance', in *2019 IEEE International Conference on Industrial Technology (ICIT)*, IEEE, February, pp.1772–1777.

Maurya, D., Tangirala, A.K. and Narasimhan, S. (2021) 'ARX model identification using generalized spectral decomposition', *IFAC-PapersOnLine*, Vol. 54, No. 9, pp.690–695.

Mirandola, I., Berti, G.A., Caracciolo, R., Lee, S., Kim, N. and Quagliato, L. (2021) 'Machine learning-based models for the estimation of the energy consumption in metal forming processes', *Metals*, Vol. 11, No. 5, p.833.

Moeeni, H. and Bonakdari, H. (2018) 'Impact of normalization and input on ARMAX-ANN model performance in suspended sediment load prediction', *Water Resources Management*, Vol. 32, No. 3, pp.845–863.

Nguyen, Q.C., Vu, V.H. and Thomas, M. (2022) 'A Kalman filter based ARX time series modeling for force identification on flexible manipulators', *Mechanical Systems and Signal Processing*, Vol. 169, p.108743.

Ninness, B., Gomez, J.C. and Weller, S. (1995) 'MIMO system identification using orthonormal basis functions', in *Proceedings of 1995 34th IEEE Conference on Decision and Control*, IEEE, December, Vol. 1, pp.703–708.

Ninness, B. and Gustafsson, F. (1997) 'A unifying construction of orthonormal bases for system identification', *IEEE Transactions on Automatic Control*, Vol. 42, No. 4, pp.515–521.

Qian, D.S., Hua, L. and Pan, L.B. (2009) 'Research on gripping conditions in profile ring rolling of raceway groove', *Journal of Materials Processing Technology*, Vol. 209, No. 6, pp.2794–2802.

Quagliato, L., Berti, G.A., Kim, D. and Kim, N. (2018) 'Slip line model for forces estimation in the radial-axial ring rolling process', *International Journal of Mechanical Sciences*, Vol. 138, pp.17–33.

Raducan, E., Nicolau, V., Andrei, M., Petrea, G. and Vlej, G.M. (2020) 'Prediction algorithms using specialized software tools for steel industry equipment', in *2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, IEEE, October, pp.174–177.

Roy, R.B., Mishra, D., Pal, S.K., Chakravarty, T., Panda, S., Chandra, M.G., Pal, A., Misra, P., Chakravarty, D. and Misra, S. (2020) 'Digital twin: current scenario and a case study on a manufacturing process', *The International Journal of Advanced Manufacturing Technology*, Vol. 107, No. 9, pp.3691–3714.

Ruiz-Sarmiento, J.R., Monroy, J., Moreno, F.A., Galindo, C., Bonelo, J.M. and Gonzalez-Jimenez, J. (2020) 'A predictive model for the maintenance of industrial machinery in the context of Industry 4.0', *Engineering Applications of Artificial Intelligence*, Vol. 87, p.103289.

Samodurova, M.N., Karandaeva, O.I., Khramshin, V.R. and Liubimov, I.V. (2020) 'Calculating power parameters of rolling mill based on model of deformation zone with four-roll passes', *Machines*, Vol. 8, No. 4, p.73.

Söderström, T. and Stoica, P. (1989) *System Identification*, Prentice-Hall International.

Xiang, F., Zhi, Z. and Jiang, G. (2018) 'Digital twins technolgy and its data fusion in iron and steel product life cycle', in *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC)*, IEEE, March, pp.1–5.

Zhang, F., Liu, M., Zhou, Z. and Shen, W. (2016) 'An IoT-based online monitoring system for continuous steel casting', *IEEE Internet of Things Journal*, Vol. 3, No. 6, pp.1355–1363.

Zhao, Y.M. and Qian, D.S. (2010) 'Effect of rolling ratio on groove-section profile ring rolling', *Journal of Mechanical Science and Technology*, Vol. 24, No. 8, pp.1679–1687.

Appendix

The derivations for calculating the computational cost for estimating the parameters of each model is derived here. The case of SISO is considered for ilustration.

ARX

The complexity related to estimate the parameters of an ARX model can be investigated by considering

$$\left(\sum_{t=1}^{N}\varphi(t)\varphi(t)^{\mathsf{T}}\right)\theta = \sum_{t=1}^{N}\varphi(t)y(t). \tag{19}$$

Constructing $\varphi(t)\varphi(t)^{\mathsf{T}}$ takes $(na+nb)^2$ multiplications and zero additions. $\sum_{t=1}^{N}\varphi(t)\varphi(t)^{\mathsf{T}}$ takes $N(na+nb)^2$ multiplications and $(N-1)(na+nb)^2$ additions. In a similar way, $\sum_{t=1}^{N}\varphi(t)y(t)$ requires $N(na+nb)$ multiplications and $(N-1)(na+nb)$ additions. For solving the system of linear equations in (19), we consider Gauss elimation and back substitution. The creating of zeros below each pivot takes $(p-i)^2$ multiplications and additions, where $(i \times i)$ is the pivot position and $p = na+nb$. So, Gaussian elimination takes $\sum_{t=1}^{p-1}(p-i) = \frac{1}{6}(p-1)p(2p-1)$ multiplications and additions, by sum of powers. Back substitution of each element requires $p-i+1$ multiplications and $p-i$ additions. A total of $\sum_{i=1}^{p}(p-i+1) = \frac{1}{2}(p+1)p$ multiplications and $\sum_{i=1}^{p-1}(p-i) = \frac{1}{2}(p-1)p$ additions, by sum of powers.

OBF

The number of operations required for the identification of θ in the case of OBF is summarised in

$$(\Phi^{\mathsf{T}}\Phi)\theta = \Phi Y \tag{20}$$

where Φ, θ and Y are defined in Subsection 4.3. Constructing Φ requires filtering $u(t)$ for each basis function. Each basis function is an IIR filter. $B_o(q)$ is an IIR filter of order 1, $B_1(q)$ has order 2, and so on until reaching $B_{p-1}(q)$, which is order p. In the case of choosing complex conjugate poles as prior information, $B_0(q)$ and $B_1(q)$ are order 2, $B_2(q)$ and $B_3(q)$ are order 4, and so on (Ninness, 1994). By choosing complex conjugate poles, each row of Φ takes $8\sum_{i=1}^{p/2} i = 2p(\frac{p}{2}+1)$ multiplications and $8\sum_{i=1}^{p/2} i - \sum_{i=1}^{p/2} 1 = 2p(\frac{p}{2}+1) - p$ additions, by sum of powers. In total, it takes $2Np(\frac{p}{2}+1)$ multiplications and $2Np(\frac{p}{2}+1) - Np$ additions. Similarly to ARX, $\Phi\Phi^{\mathsf{T}}$ requires Np^2 multiplications and $(N-1)p^2$ additions, $\Phi^{\mathsf{T}}Y$ needs Np multiplications and $(N-1)p$ additions. The number of operations for Gauss elimination and back substitution is the same as in ARX.

ARMAX

There are many methods to solve the nonlinear optimisation problem defined in equation (7). Here,

$$\hat{\theta}^{(k+1)} = \hat{\theta}^{(k)} - \alpha_k[V_n''(\theta^{(\hat{k})})]^{-1}]V_N'(\theta^{(\hat{k})})^{\mathsf{T}} \tag{21}$$

is used as suggested in Ljung (1987) and Söderström and Stoica (1989), which is known as the Newton-Rapshon algorithm, where

$$V_N'(\theta) = -\frac{2}{N}\sum_{t=1}^{N}\varepsilon(t,\theta)\psi^{\mathsf{T}}(t,\theta) \tag{22}$$

and

$$V_N''(\theta) = \frac{2}{N}\sum_{t=1}^{N}\psi(t,\theta)\psi^{\mathsf{T}}(t,\theta), \tag{23}$$

where $\psi(t,\theta) = -\left(\frac{\partial\varepsilon(t,\theta)}{\partial\theta}\right)^{\mathsf{T}}$. Therefore, computing equation (21) requires computing equations (22), (23), $\varepsilon(t,\theta^{(k)})$ and $\psi(t,\theta^{(k)})$ for each iteration.

$$\varepsilon(t,\theta) = \frac{A(q^{-1})}{C(q^{-1})}\left(y(t) - \frac{B(q^{-1})}{A(q^{-1})}u(t)\right) \tag{24}$$

takes $(na+nb+nc)$ multiplications and additions. $\frac{\partial\varepsilon(t,\theta)}{\partial\theta}$ is a $((na+nb+nc) \times 1)$ vector requires nc multiplications and additions for computing each entry. The entry can be of the form $\frac{\partial\varepsilon(t,\theta)}{\partial a_i} = \frac{1}{C(q^{-1})}y(t-i)$, $\frac{\partial\varepsilon(t,\theta)}{\partial b_i} = -\frac{1}{C(q^{-1})}u(t-i)$ or $\frac{\partial\varepsilon(t,\theta)}{\partial c_i} = \frac{1}{C(q^{-1})}\varepsilon(t-i,\theta)$ depending on the entry i. Therefore, each iteration requires $N(na+nb+nc)$ for constructing the N samples of $\varepsilon(t,\theta)$ and $N(na+nb+nc)nc$ for the N samples of $\psi(t,\theta)$. Computing equation (22) requires a

total of $N(na + nb + nc) + N(na + nb + nc)nc + (N + 2)(na + nb + nc)$ multiplications and $N(na + nb + nc) + N(na + nb + nc)nc + (N - 1)(na + nb + nc)$ additions. Computing equation (23) takes $(N + 2)(na + nb + nc)^2$ multiplications and $(N - 1)(na + nb + nc)^2$ additions. Instead of computing the inverse on equation (21), the system of linear equations defined by $V_N''(\theta^{(k-1)})d_k = V_N'(\theta^{(k-1)})$ is solved by Gauss elimination and back substitution with the same requirements as previous subsections. Finally, $\hat{\theta}^{(k+1)} = \hat{\theta}^{(k)} - \alpha_k d_k$ takes $na + nb + nc$ multiplications and additions.