



# Artificial neural networks training via bio-inspired optimisation algorithms: modelling industrial winding process, case study

Malik Braik<sup>1</sup> · Hussein Al-Zoubi<sup>2</sup> · Heba Al-Hiary<sup>3</sup>

Published online: 23 November 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

This research provides a study on how the weights of artificial neural networks (ANNs) can be automatically updated by applying bio-inspired algorithms, particularly using the particle swarm optimisation (PSO) algorithm, grasshopper optimisation algorithm (GOA) and grey wolf optimisation (GWO). These evolutionary computation algorithms were used to evolve the synaptic weights of ANNs to find a particular architecture of ANNs. The developed nonlinear models were targeted to the identification of a particular nonlinear prediction system, an industrial winding process, as a case study. These new models were referred, respectively, to as ANN-PSO, ANN-GOA and ANN-GWO. The proposed models were compared with other linear and nonlinear conventional models including least square error and multiple nonlinear regression methods, respectively, as well as other state-of-the-art models including multilayer perceptron-type NNs, radial basis function and recurrent local linear neuro-fuzzy. The performance of the developed models was assessed using several metric criteria. Comparison of the proposed ANN-PSO, ANN-GOA and ANN-GWO models with other traditional and state-of-the-art models asserts the efficacy of the proposed modelling approaches.

**Keywords** Artificial neural network · Particle swarm optimisation · Grasshopper optimisation algorithm · Grey wolf optimisation · Industrial winding process · Multiple nonlinear regression

## 1 Introduction

Machine learning (ML) algorithms have progressed substantially through the past two decades, from laboratory

Communicated by V. Loia.

**Electronic supplementary material** The online version of this article (<https://doi.org/10.1007/s00500-020-05464-9>) contains supplementary material, which is available to authorized users.

✉ Hussein Al-Zoubi  
hussein.alzoubi@ju.edu.jo  
Malik Braik  
mbraik@bau.edu.jo  
Heba Al-Hiary  
hhiary@bau.edu.jo

<sup>1</sup> Department of Computer Science, Al-Balqa Applied University, Salt 19117, Jordan

<sup>2</sup> Computer Science Department, School of Electrical Engineering and Information Technology, German Jordanian University, Amman 11180, Jordan

<sup>3</sup> Department of Computer Information System, Al-Balqa Applied University, Salt 19117, Jordan

curiousness to a workable technology in prolonged commercial use. Artificial neural networks (ANNs), support vector machines (SVMs), fuzzy logic (FL) and neuro-fuzzy are well-known ML techniques that have proliferated as preferred methods of solving a broad range of problems in many areas. The research interest in ANN is related to its attractive benefits it manifests, like adaptation ability, learning capacity and its aptitude to generalise. As a score of this interest, ANN has been extensively used to solve an extensive set of problems such as in colour recognition Al-Azzeh et al. 2019, classification problems Arunkumar et al. 2020 and many other applications Chang et al. 2019. In effect, classification using ML methods is one of the most widespread and preferred applications in varied fields of study suchlike decision-making tasks that face human activity. For instance, ANNs have been effectively used to solve a large assortment of real-world classification problems in industry Wang et al. 2012, engineering Pirdashti et al. 2013 and medicine Saritas and Yasar 2019. Essentially, SVM is a binary classifier that builds a linear separating hyperplane to categorise data instances. SVMs have been appreciably used for classification Wang et al. 2012 as well as speech recognition Jain

et al. 2020 due to its appealing features in various applications Khosravi et al. 2018. FL, as one of the popular ML techniques, has gained a large popularity by researchers because of its benefits such as effective performance, ease of implementation and easy interpretation of the outcomes. Its impact has been felt widely across scientific and engineering fields and across a range of industries concerned with data-intensive issues, such as classification of faults in mechanical differential Asadi Asad Abad et al. 2015 and modelling of industrial systems Sheta et al. 2009. Neuro-fuzzy computing, which is a wise combination of FL and ANNs, enables the construction of more intelligent decision-making systems. Basically, this system integrates the general features of ANNs such as robustness and adaptive learning with the strengths of FL with the aim of covering the weaknesses of these two techniques. These two techniques have been applied together to solve several types of engineering problems Babuška and Verbruggen 2003; Sheta et al. 2013 and have achieved fruitful accomplishments in medical imaging Omotosho et al. 2018 and classification problems Rajasekaran and Sri Meena 2012. An outstanding benefaction of soft computing and neuro-fuzzy is the evolution of adaptive neuro-fuzzy inference system (ANFIS) Ansari and Gupta 2011, where it has been appropriately applied to address several applications in engineering and industrial fields Khosravi et al. 2018 and has been adopted to prediction problems Naderpour and Mirrashid 2019.

With the evolution of industrial process plants, it has become more complicated and burdensome to control them. In this context, increased global competition in the control and monitoring of industrial processes has driven to the development of new approaches for the design and analysis of industrial processes Wang et al. 2015. This is to fulfil the high request for product quality process safety and even the productivity of all working equipments. Thus, industrial modelling techniques are increasing in approval and have earned a notable development in practical sites and academic community Schlei-Peters et al. 2018.

Numerous applications in engineering and scientific fields have been suggested to control and monitor industrial processes based on the inference of mathematical models Sheta et al. 2013. The rotor system is one of base components in web conveyance systems, which have a significant impact on the dynamics of the systems. Many models have been proposed in the literature to control and monitor the operational processes of the rotor systems, which have been thoroughly explored in many research works and have played a crucial role in various industrial machines. As one of comprehensive and precise approaches, Liu and Shao (2018) presented a thorough investigation of dynamic modelling and simulation system for foretelling the vibration attributes of rolling element bearings with and without local and distributed flaws. El-Thalji and Jantunen (2015) provided an elabo-

rate review of the predictive health controlling methods, the corresponding abilities, drawbacks and merits in detecting and identifying the localised and distributed flaws in rolling element bearings. However, some industrial processes form a challenge in creating such mathematical models. Development of mathematical models is essential to automate and increase the possibility of simulating complex industrial operations. This imposes a great demand to evolve high quality models for the industrial systems to assure productivity and quality of service. There are two common ways to establish a relationship between the specified input and output variables for a nonlinear system.

- **Traditional empirical modelling methods**, which depend on building an automated physical model on the basis of raw data registered in empirical or industrial system characteristics. There are some circumstances where traditional modelling methods require prolonged time to fulfil a task that may lead to unfavourable results. This takes place if there is a shortage of precise or systematic knowledge about the system or the experimental data are subject to a high degree of uncertainty. These models may produce imprecise outcomes for many situations that do not represent the data so quite. Sometimes, the developed models are not incomprehensible and will not be well interpreted Ljung 1987.
- **Model-based methods**, which derive a mathematical relationship between the observed data and the true data of the system with a primary aim to reduce the variations between the target and the original datasets Braik et al. 2008; Sheta et al. 2009. Consequently, there is a difficulty in building a model based for a challenge industrial system with sufficient suitability that allows the model to efficiently identify the data measurements. Extracting an appropriate model for an industrial system is essential for model-based systems. The best design in industry is often associated with multiple design goals under nonlinear constraints. Diverse objectives often clash with each other, and occasionally, there are no really optimal solutions, and so often there is a need to some concessions and approximations Liu and Shao 2018.

Many mathematical models have been applied presented in the literature to model plant-wide industrial processes. Among these processes, the real winding process (WP) Nozari et al. 2012 was one of the challenging processes used extensively in research works for both academic and practical purposes. It has been widely used for judging and exploring the efficiency of the development of new mathematical models. To proffer an obviously and in-depth understanding of modelling industrial processes, we present a model-based method to simulate the WP using ANNs trained based on a set of bio-inspired algorithms (BIAs). The winding process

Nozari et al. 2012 is an important and appropriate process in all disciplines of system theory. This includes conducting comparative studies, verification of new algorithms and evaluation of control systems. However, to model the WP is a challenge because it is a nonlinear process and has inconstant conditions. Thus, extracting an appropriate model for a WP sounds necessary for model-based control and diagnosis experiments. Here, particle swarm optimisation (PSO) algorithm Kennedy 2011, grasshopper optimisation algorithm (GOA) Saremi et al. 2017 and grey wolf optimiser (GWO) Mirjalili et al. 2014 were used as a new type of learning approach to adjust the weights of the ANN.

The rest of the paper is organised as follows: In Sect. 2, several methods of modelling manufacturing processes are reviewed. Section 3 describes the linear and nonlinear procedures of modelling the dynamic systems. In Sect. 4, we briefly present the evolutionary optimisation algorithms of our interest. Then, in Sect. 5, we briefly describe the winding process case study. After that, Sect. 6 presents a description of data used in our experimentation followed by a discussion of the preliminary operations performed on the data. The proposed system identification procedure is presented in Sect. 7. The evaluation metrics are given in Sect. 8. The simulation and evaluation outcomes are given in Sect. 9, and finally, the paper is concluded in Sect. 10 accompanied with insights for future research directions.

## 2 Previous works

Today, there is a swift development in the modelling, monitoring and identification of industrial processes. In this sense, there are a substantial number of methods reported in the literature for modelling many types of manufacturing processes as well as the winding process Sheta et al. 2019; Chang et al. 2019; Naderpour and Mirrashid 2019. However, it is not feasible to study and analyse all the research methods used for modelling and monitoring of the WP. Besides the studies that have used empirical methods to model the winding machines Sievers et al. 1988; Parant et al. 1992, there are many methods used data-based methods to develop suitable models for industrial systems Babuška and Verbruggen 2003; Sheta et al. 2013; Khosravi et al. 2018.

Earlier, genetic programming (GP) was adopted to create a nonlinear model for a winding machine Hussian et al. 2000, and a nonparametric ANN model was suggested for modelling an industrial winding process Hussian et al. 2001. A linear model was created for faults diagnosis for a winding machine using an Auto-Regressive with eXternal inputs (ARX) structure Noura et al. 2009. Sheta et al. (2013) have used ANFIS with Takagi–Sugeno technique to create models for the dynamics of a hot rolling industrial for datasets gathered from the Eregli Iron and Steel factory in Turkey. Nozari

et al. (2012) have presented a technique for modelling an industrial winding process. Their proposed approach used an incremental tree-based algorithm, referred to as LOLIMOT, to train a recurrent local linear neuro-fuzzy (RLLNF) network. The performance of the RLLNF modelling approach was compared to least square error (LSE) method, multilayer perceptron (MLP) and radial basis function (RBF) identifiers.

The sharp increase in empirical data across industries provides unrivaled chances for data-driven decision making, where data-driven models can address experimental data to control, monitor and improve industry performance. Sadati et al. (2018) presented a modelling method that uses data from synthetic experiments to identify control variables during simplifying process parameters simultaneously. The method was tested on a real situation of a tire manufacturing company. Torres et al. (2018) used probabilistic boolean networks (PBNs) to model a manufacturing system, and they studied the relationship between components of a real industrial machine. The machine was modelled as a PBN, by specifying the regulatory nodes. The predictors, selection probabilities, simulation and property verification were used to assess the accuracy of the PBN model. They used simulation results to create the data needed to make inferential statistical tests to determine the degree of correspondence between the forecasts and real machine data.

Currently, many researchers tend to use ANN or one of its variations to model industrial processes due to its potentiality of generalisation and fitting adaptability Chang et al. 2019; Sheta et al. 2019; Wang et al. 2020. However, fitting ability of an ANN is typically affected by the configuration used, particularly number of hidden neurons, number of input parameters and the learning algorithm utilised to fine-tune its weights Crone and Kourentzes 2009. An iterative process of adding hidden neurons and inputs to the ANN model should result in a systematic decrease in the modelling error. The learning algorithm is a fundamental process for deducing the ANN model parameters that fit with the specified data set. This has encouraged the development of many learning algorithms such as gradient-based techniques, like Levenberg–Marquardt (LM) Ranganathan 2004 and back propagation (BP) Li et al. 2012 algorithms. These classical algorithms were preferred to use by many researchers due to their merits such as their efficient implementation, low computational burden and good at tuning the weights of the ANN model. On top of that, the rapid convergence is a key feature of gradient-based techniques, as the adequate exploitation of gradient information can significantly increase the convergence speed compared to a technique that does not calculate gradients. However, these algorithms are highly dependent on the initial weights when searching for optimal weights, are prone to falling into local optima with non-desirable solutions, may fail to explore multimodal and non-continuous surfaces and sometimes lead to poor perfor-

mance. Another potential fragility of these classical methods is that they are comparatively inflexible of obstacles suchlike noisy target function spaces Zingg et al. 2008.

Many BIAs have been satisfactorily for modelling different types of industrial systems. Nikabadi and Naderi (2016) have proposed a hybridisation algorithm of simulated annealing (SA) and GA for scheduling unconnected parallel machines with sequence-based setup times, standby times, variable due dates as well as some antecedence connections between the jobs. Ayough and Khorshidvand (2019) have evolved a model for a cellular manufacturing process to reduce the costs associated with a limited number of cells to be built by allocating workforce using PSO and SA. Mousavi et al. (2015) utilised GA, SA and a combination of SA and GA to simulate how generators endeavour in the spot electricity market to increase profits according to the work plans of other generators. In another aspect of BIAs in industry, cuckoo search algorithm (CSA) was exercised to a vehicle routing problem Santillan et al. 2018. In Dixit et al. (2019), GA and ANN were used to address a modelling problem and optimisation study on dimensional irregularities of square-shaped microgroove in laser microfabrication of aluminium oxide ceramic substance. Because assembly forms a large fraction of the cost of any product in any modern production system, a study offered in Dao et al. (2017) used GA for an optimum global solution to model a virtual computer-integrated manufacturing system. An incorporation of clonal selection algorithm (CSA) and SA, referred as CSA-SA, along with a mathematical model based on quadratic assignment was combined to address a stochastic dynamic facility problem Moslemipour 2018. An outstanding use of BIAs in industry was made by Yıldız (2008) who has presented a hybrid harmony search method to handle many manufacturing and engineering design problems.

Although reasonable performance has been achieved in the modelling of industrial systems using ANNs and BIAs, the performance of modelling industrial systems that are subject to variable conditions and uncontrolled factors lacks high precision and is not persuasive either modelled using ANNs Nozari et al. 2012 or BIAs Nikabadi and Naderi 2016. Therefore, development of reliable and compact models to simulate an industrial process is essential to generate a sensible estimate of the process parameters. In this paper, we have turned our attention to BIAs as learning algorithms for ANNs, looking for a suitable source of inspiration for learning ANNs, which can overcome the handicaps of classical methods in the modelling of an industrial system. Here, three BIAs, namely PSO Kennedy 2011, GOA Saremi et al. 2017 and GWO Mirjalili et al. 2014, were used to training an ANN model to model an industrial winding machine Nozari et al. 2012. The aim of these algorithms is to get an optimal set of weights for the ANN model, where an optimal model would be obtained. These optimisation algorithms have outstanding

exploratory search features and can avert getting caught into local minimum by exploring the search space, which are then well suited for adjusting the weights of ANNs and getting an optimal model structure for the winding process under study.

## 2.1 Challenges of the winding system

Winding systems are key elements of a broad range of industrial plants. For instance, steel rolling mills Babuška and Verbruggen 2003 and plants involve web conveyance that includes paper-making, coating and polymer film extrusion Braatz et al. 1996. Some researchers have considered limiting the computational efforts correlated with the analysis, design and active fault-tolerant control in web conveyance systems Sievers et al. 1988, steel industries Parant et al. 1992 and film and sheet processes Braatz et al. 1996. Developing accurate and reliable models to describe complex industrial processes is substantial to create a robust approximation of the output parameters. This is a specific requirement to create purposeful industrial process models. The winding systems have been largely used by control community as a data source to compare various approaches and evaluate the adequacy of process control methods Haddad et al. 2017. To sum up, a great number of literary articles have been directed to process modelling, performance control and winding process identification. Nevertheless, these works are short of accuracy about the winding process and in-depth study concerning of the many uncontrolled factors that occur in the actual winding process. In fact, due to these uncontrollable factors, modelling and monitoring methods of the winding process encountered a difficulty. The variations in the parameters of this process are further a challenge. Furthermore, the performance achieved in modelling and control of the winding process needs to be considered. So, we believe that there is still scope to do better. How to select suitable system architecture along with an appropriate algorithm for estimating system parameters and appropriate assessment criteria are considered in this paper.

The originality and relevance of this work are related to: (1) development of three intelligent models to model a real winding process based on ANNs and PSO, GOA and GWO with an aim to meet the expected theoretical behaviour aspects of the presented models and (2) assessment of the theoretical aspects of the proposed models like error reduction and performance expansion in additive process parameters. Adding more hidden neurons to the proposed models to achieve the desired degree of performance is a sensible process. The primary objective of this work encompasses obtaining a specific structure of an ANN with optimum values of the weights. In addition, it is anticipated that this will allow it practicable to compare the alternative models of the case study as a second objective.



### 3 Dynamic system identification

There are two popular procedures for the identification of the dynamic multi-input–single-output (MISO) systems from input/output sequences:

#### 3.1 Linear dynamic system identification

To build a model for a plant-wide process, the simplest solutions should be tried at outset based on the theory of system identification, particularly when no beforehand knowledge is available of its fundamental behaviour. To arrive at a linear model, a specific sequence of the input and output parameters perceived with a constant sampling interval is taken into account. Generally, it is possible to identify a problem to be solved given a set of input patterns,  $u(t) = \{u_1, u_2, \dots, u_p\}$ ,  $u \in \mathbb{R}^p$  and a set of desired patterns,  $y(t) = \{y_1, y_2, \dots, y_r\}$ ,  $y \in \mathbb{R}^r$ , where  $t$  is the sampling time and  $p$  and  $r$  are the number of samples in the input and output, respectively. So, it is potential to find a suitable mathematical model that represents the data model based on a relationship between input and output patterns, so the error is reduced between the true data and the estimated data. In the case of dynamic linear relationships between input and output variables, the relationship among them can be described by the following model Nozari et al. 2012:

$$y_M(k) = \sum_{l=1}^n (a_l y_M(k-l)) + \sum_{i=1}^p \sum_{l=1}^p b_{il} u_i(k-l) \tag{1}$$

where  $a_l$  and  $b_{il}$  are scale parameters,  $m_i$  represents the order of the numerator of the  $i$ th input and  $n$  represents the denominator order.

Equation 1 represents linear MISO discrete-time systems, which can simply be converted to the corresponding discrete-time transfer function. The parameters in Eq. 1 can be computed by the LSE technique because the prediction error is linear in parameters Ljung 1987. Choosing an appropriate model structure is an important component in any modelling a system. The purpose of selecting an order for the model is to select a model that fits a particular data set. This issue is so important that deficiencies in this selection process may result in poor accuracy in some key procedures of the modelling scheme. The Akaike Information Criterion (AIC) Ljung 1987 is an information criterion that can be used to fulfil this process. In such a context, the obtained orders on the basis of linear models can then be tested on nonlinear approximators in the case of nonlinear identification. In the particular order selection strategy, the linear model order displayed in Eq. 1 is augmented and the AIC index is computed in each step. At last, after some iterations, the corresponding order of the lowest AIC value is chosen as a typical model order:

$$J_{AIC} = \log SSE + \frac{2P}{Q} \tag{2}$$

where  $Q$  identifies the number of samples utilised to calculate the AIC index,  $P$  denotes the number of parameters,  $2P/Q$  stands for a penalty term and  $SSE$  is the sum of square errors set by Nozari et al. (2012):

$$SSE = \sum_{N=1}^Q (y_p(N) - y_M(N))^2 \tag{3}$$

where  $y_p$  and  $y_M$  are the output parameters of the process and model, respectively.

#### 3.2 Nonlinear dynamic system identification

The identification of a nonlinear system is more appropriate when linear methods do not provide satisfactory outcomes in the modelling of physical systems. Further, the potential adaptability, generality and capability to implement a nonlinear relationship between input and output parameters can be demonstrated using nonlinear regression techniques. The algorithm selection does not effect on the adjustment statistics value of the model. In many situations of the regression models, the fact that the change in output,  $Y$ , relies on input,  $X$ , is what turns out the relationship between  $X$  and  $Y$  to be nonlinear although the model is linear in its estimated parameters Chang et al. 2010. Yet the model is still considered a linear regression model in view that it is linear in terms of the regression coefficients,  $\beta_1, \beta_2, \dots, \beta_i$ . One strategy to demonstrate such a relationship is through a polynomial regression model. The relationship between the independent variable  $X$  and the dependent variable  $Y$  would be modelled as an  $m$ th degree polynomial in  $X$ . Such a model for a single regressor,  $X$ , is:

$$Y = \beta_0 + \beta_1 X^1 + \beta_2 X^2 + \dots + \beta_m X^m + \Upsilon \tag{4}$$

where  $m$  stands for the degree of the polynomial and  $\Upsilon$  represents a vector of model errors.

The computational problems of polynomial regression can be entirely handled using multiple nonlinear regression (MNLr) models. This can be achieved using  $X^1, X^2, \dots, X^m$  as distinct independent parameters Chang et al. 2010. The polynomial regression model can be described as:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_m x_i^m + \epsilon_i \tag{5}$$

where  $i = 1, 2, \dots, n$  and  $n$  is the number of coefficients of the model.

Equation 5 can be expressed by a matrix  $\mathbf{X}$  of input values, a response vector  $\mathbf{y}$  of output values, a coefficient vector  $\mathbf{f}$ ,

and a vector  $\mathbf{7}$  of model errors. To sum up, the regression model can be represented as shown in Eq. 6:

$$\mathbf{y} = \mathbf{X}\mathbf{f} + \mathbf{7} \quad (6)$$

The reality that the behaviour of most dynamical systems is nonlinear has made ANNs suitable for the identification tasks. MLP neural networks are generic nonlinear model structures that have proved to be adequate for black-box modelling and control problems. Several studies have recommended that MLP neural networks are a proper selection for defining a nonlinear system and that a variety of ANN architectures and training algorithms have been suggested for control problems Braik et al. 2008. Thus, the process of handling non-linearity as well as the dynamism of a WP was addressed on the basis of ANNs trained with BIAs as described below.

## 4 Evolutionary optimisation algorithms

We have chosen the well-known evolutionary algorithm, PSO, and two recent evolutionary algorithms, GOA and GWO, to train ANNs. These SI-based algorithms are well accepted by the artificial intelligence (AI) community for their potent optimisation in solving complex real-world problems. A brief background about each of these algorithms is given in the following subsections.

### 4.1 Particle swarm optimisation

The PSO was proposed by Kennedy (2011) to optimise continuous nonlinear functions by modelling the social behaviour of swarms of animals like bird flocking and fish schooling. The term ‘‘particle’’ was adopted because particles experience velocities and accelerations. In addition, it also indicates diffuse objects such as clouds. Basically speaking, PSO is mathematically simple and computationally inexpensive. In PSO, particles explore probable solutions of the hyperspace and accelerate towards optimum solutions. The PSO algorithm adjusts velocities and positions of the swarm particles as per the following equations, respectively Kennedy 2011:

$$\begin{aligned} V_i^{(k+1)}(j) &= \omega \times V_i^{(k)}(j) \\ &+ \alpha \times r_1 \times (pbest_i^{(k)}(j) - X_i^{(k)}(j)) \\ &+ \beta \times r_2 \times (gbest_i^{(k)}(j) - X_i^{(k)}(j)) \end{aligned} \quad (7)$$

$$X_i^{(k+1)}(j) = X_i^{(k)}(j) + V_i^{(k+1)}(j) \quad (8)$$

where  $i = 1, 2, \dots, P$ , is the particle’s index from a population of size  $P$ .  $pbest_i$  is the  $i$ th particle’s best known position and  $gbest_i$  is the best position known to the swarm.

$\omega$  is the inertia weight.  $\alpha$  and  $\beta$  are two positive acceleration constants, called the cognitive and social parameters, respectively, which control the influence of  $pbest_i$  and  $gbest_i$  on the search process.  $r_1$  and  $r_2$  are two random numbers uniformly distributed within the range  $[0, 1]$ .  $X_i^{(k)}(j)$  identifies the  $j$ th element of the current position of the  $i$ th particle in  $k$ th step.  $X_i^{(k+1)}(j)$  identifies the  $j$ th element of the new position of the  $i$ th particle in  $(k + 1)$ th step.  $V_i^{(k)}(j)$  identifies the  $j$ th element of the current velocity vector of the  $i$ th particle in  $k$ th step and  $V_i^{(k+1)}(j)$  stands for the  $j$ th element of the new velocity vector of the  $i$ th particle in  $(k + 1)$ th step of the PSO model.

PSO has a virtuous approval by AI because it is robust in finding optimum or near optimum solution. Based on the intuitive understanding of PSO model, there is a faction within the PSO research community that supports the use of PSO in training the weights of ANNs. This is due to its robust exploration capabilities, diminished susceptibility to being trapped in local minimum and because it does not suffer from early convergence as is the case with the global best ( $gbest$ ).

### 4.2 Grasshopper optimisation algorithm

The GOA, proposed by Saremi et al. (2017) is a gradient-free, nature-inspired stochastic optimisation algorithm that highly avoids local optima and considers the optimisation problem as a black box. GOA mimics grasshopper swarms (attraction and repulsion forces between grasshoppers in the swarm) with a mathematical model represented as Saremi et al. 2017:

$$X_i = r_1 \times S_i + r_2 \times G_i + r_3 \times A_i, \quad (9)$$

where the subscript  $i$  is the grasshopper’s index from a population of size  $N$ .  $X_i$  stands for the position.  $S_i$  represents the social interaction, which is a function of distance between grasshoppers.  $G_i$  represents the force caused by gravity and is directed towards the Earth’s centre.  $A_i$  represents wind advection and follows the direction of the wind, and  $r_1$ ,  $r_2$  and  $r_3$  are random numbers  $\in [0, 1]$ . Accordingly, as shown in Saremi et al. (2017), (9) can be rewritten as:

$$X_i = \sum_{j=1, j \neq i}^N s(|X_j - X_i|) \frac{X_j - X_i}{d_{ij}} - g\hat{e}_g + u\hat{e}_w \quad (10)$$

where  $g$  represents the gravitational constant,  $\hat{e}_g$  represents a unity vector towards the centre of the earth,  $u$  represents a constant drift,  $\hat{e}_w$  represents a unity vector in the direction of wind and  $s$  is a social forces function that was defined as:  $s(r) = fe^{\frac{-r}{r_0}} - e^{-r}$ .

Then, authors proceed to show that Eq. (10) becomes Saremi et al. 2017:

$$X_i^d = c \left( \sum_{j=1, j \neq i}^N c \frac{ub_d - lb_d}{2} s(|X_j^d - X_i^d|) \frac{X_j - X_i}{d_{ij}} \right) + \hat{T}_d \tag{11}$$

where  $ub_d$  and  $lb_d$  are the upper and lower limits in the  $d$ th dimension,  $\hat{T}_d$  represents the value of the  $d$ th dimension in the best target solution and  $c$  is a lessening factor to shrink the comfort area, repulsion area and attraction area.

GOA is different from PSO in that GOA mandates all search agents to contribute to the calculation of the position of every search agent. Moreover, GOA keeps a good balance between exploration and exploitation by adopting the so called adaptive comfort zone. At the beginning, grasshoppers experience high repulsion rate; this leads to high exploration and thus avoids local optima. Then, when the search proceeds and approaches final steps, the attraction between the grasshoppers in the swarm manifests and takes exploitation to its extent resulting in better search accuracy. In Saremi et al. (2017), it was shown that GOA excels in solving challenging real problems albeit with unknown search spaces. Based on the appealing features of GOA with high exploration and exploitation capabilities, there is a faction emanated to use GOA to adjust the weights of ANNs during the training process to model the industrial case study process.

### 4.3 Grey wolf optimisation

The GWO is a flexible metaheuristic optimisation method that avoids stagnation in local optima spots of the search space. This algorithm was proposed by Mirjalili et al. (2014); it imitates the social behaviour of grey wolves in the aspects of their hierarchical leadership and hunting manoeuvres. Mathematically, this algorithm models the leadership hierarchy of grey wolves by categorising them into four sets in accordance to their superiority from top to bottom as  $\alpha$ ,  $\beta$ ,  $\delta$  and  $\omega$ . In terms of hunting, the algorithm models prey (target) encircling by the hierarchical grey wolves pack as Mirjalili et al. 2014; Masadeh et al. 2018:

$$\mathbf{D} = |\mathbf{C} \cdot \mathbf{X}_p(t) - \mathbf{X}(t)| \tag{12}$$

$$\mathbf{X}(t + 1) = \mathbf{X}_p(t) - \mathbf{A} \cdot \mathbf{D} \tag{13}$$

where  $t$  defines the current iteration,  $\mathbf{A}$  and  $\mathbf{C}$  define coefficient vectors and  $\mathbf{X}(t)$  and  $\mathbf{X}_p(t)$  are the position vectors of the grey wolf and prey, respectively.

The vectors  $\mathbf{A}$  and  $\mathbf{C}$  were computed as follows:

$$\mathbf{A} = 2\mathbf{a} \cdot \mathbf{r}_1 - \mathbf{a} \tag{14}$$

$$\mathbf{C} = 2 \cdot \mathbf{r}_2 \tag{15}$$

where the items of  $\mathbf{a}$  are linearly diminished from 2 to 0 over the course of iterations and  $\mathbf{r}_1$  and  $\mathbf{r}_2$  are random vectors within the range from 0 to 1.

Then, the following equations are used for hunting and attacking the prey Mirjalili et al. 2014; Azizivahed et al. 2018:

$$\begin{aligned} \mathbf{X}_1 &= \mathbf{X}(t) - \mathbf{A}_1 \cdot \mathbf{D} \\ \mathbf{X}_2 &= \mathbf{X}_{fi}(t) - \mathbf{A}_2 \cdot \mathbf{D}_{fi} \\ \mathbf{X}_3 &= \mathbf{X}(t) - \mathbf{A}_3 \cdot \mathbf{D} \end{aligned} \tag{16}$$

where  $X_\alpha$ ,  $X_\beta$  and  $X_\delta$  are the best, second best and third best search agents, respectively, and  $D_\alpha$ ,  $D_{fi}$  and  $\mathbf{D}$  are calculated using Eq. 17.

$$\begin{aligned} \mathbf{D} &= |\mathbf{C}_1 \cdot \mathbf{X} - \mathbf{X}| \\ \mathbf{D}_{fi} &= |\mathbf{C}_2 \cdot \mathbf{X}_{fi} - \mathbf{X}| \\ \mathbf{D} &= |\mathbf{C}_3 \cdot \mathbf{X} - \mathbf{X}| \end{aligned} \tag{17}$$

As  $\mathbf{X}_\alpha$ ,  $\mathbf{X}_\beta$  identify the three best solutions obtained so far Gupta and Deep 2018; Mirjalili et al. 2014, other wolves are committed to update their positions in the whole population as shown in Eq. 18:

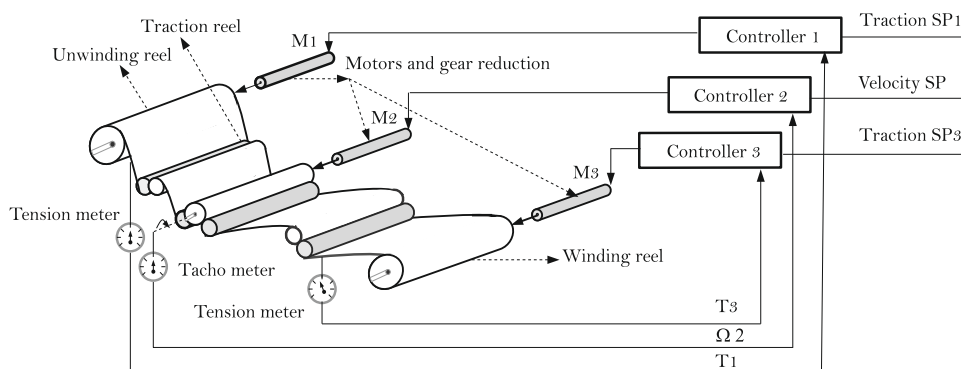
$$\mathbf{X}(t + 1) = \frac{\mathbf{X}_1 + \mathbf{X}_2 + \mathbf{X}_3}{3} \tag{18}$$

GWO has achieved competitive results in solving several types of complex real problems with unknown search spaces Mirjalili et al. 2014; Masadeh et al. 2018. The statistical analysis presented in Mirjalili et al. (2014) shows that the general statements formed about the ability of exploration, convergence and even the solution accuracy of GWO are superior, and shows that the algorithm can be considered outright, not even for particular problem sets. Therefore, GWO was conducted here to train the weights of ANNs due to its attractiveness in which it has only few parameters to set and can work well in a wide range of applications as well as for certain applications focusing on clearly defined requirements.

## 5 Winding process

The winding process is a test setup of a real industrial winding plant process, which often met in web conveyance systems Bastogne et al. 1998. This process is a well-presented benchmark problem for analysis and control design in the control community. Figure 1 shows the schematic diagram of the winding process that we are targeting in this work. This process consists of multivariate and correlated systems with process parameters varying during operation. The key role of this process is to control the web transferring to avert

**Fig. 1** Schematic diagram of a winding process with control sensors



the impacts of sliding and friction. The solution relies on conserving a traction effort on the strip and monitoring the tension at various points over the web Braatz et al. 1996. The winding machine considered in this study is composed of a plastic strip, three reels, referred to as reels 1, 2 and 3 or unwinding reel, traction reel and rewinding reel, respectively, and gear reduction connected with the three reels. The reels are controlled by three DC motors denoted as  $M_1$ ,  $M_2$ , and  $M_3$ . Motor  $M_1$  corresponds to the unwinding reel,  $M_2$  corresponds to the traction reel and  $M_3$  to the rewinding reel. Reels 1 and 3 are coupled using the DC motors driven via set-point currents  $I_1$  and  $I_3$ . Moreover, tension meters are placed to measure strip tensions in the web between reels 1 and 2, referred to as  $T_1$ , and between reels 2 and 3, referred to as  $T_3$ , in addition to the dynamo tachometer that measures the angular speeds of each reel ( $S_1$ ,  $S_2$  and  $S_3$ ). Each motor is run by a local controller as displayed in Fig. 1. Speed control is reached for motor  $M_2$ , while torque control is attained for motors  $M_1$  and  $M_3$ , given that the angular velocity of motor  $M_2$  ( $\Omega_2$ ) is measured using a dynamo tachometer. The data measurements of this process contain the process input variables and controlled output variables. The measured data were used to generate the training patterns for the developed model structures. The key process variables were measured through sensors at pre-selected points of the process, and the obtained data were registered at a sampling rate of 0.1s in the monitoring system. The input and output parameters for the WP of this case study are described in Table 1.

The output parameters for this case study describe the behaviour of the WP and show how it responds to various input sets. At any instant, to estimate the value of future response of the process, it is substantial to use both input and output parameters. The variations in the parameters that occurred in this process are due to the variation in the reel radius during the unwinding process. This non-measurable change of the reel radius remarkably amends the dynamic behaviour of the system throughout the overall process of unwinding Noura et al. 2009. Moreover, the performance and reliability of the monitoring system can be significantly affected by the system support stiffness, such

as bearing, housing and rotor Liu and Shao 2017. Interfacial frictional moments are further parameters impact on the accuracy of control and monitor of dynamical systems. Thus, many researchers have discussed the effect of housing support stiffness and interfacial frictional in the construction of their dynamic models Liu et al. 2014; Zhang et al. 2016. As a result of the uncontrolled influences to which the WP is subject; this type of process poses a challenge in modelling, identification and control. To overcome such problematic issues, non-analytical techniques as presented below are experimentally applied for the identification of the winding process models.

## 6 Model preparation process

Model development process requires some necessary preparation stages, which must be completed as a pre-step to yield a good modelling process. This involves data collection and data pre-processing.

### 6.1 Training and test datasets

Typically, the success of modelling an industrial process depends at first on the amount of data that must be large enough to study the behaviour of the process well. This assumes a large computational time during training. Thus, it is useful to use an adequate number of data measurements to train the model until a highly qualified model is created. Here, the number of data measurements for each input variable and output variable of the WP consisted of 2500 data samples, which are publicly available at<sup>1</sup>. The dataset for each web tension was divided equally into a training set to train the model and a test set for evaluating the performance of the developed model. Consequently, the number of data samples for each input parameter and output parameter in the training and testing processes consists of 1250 samples. This number of data values has the potential to yield an effective

<sup>1</sup> <https://homes.esat.kuleuven.be/~smc/daisy/daisydata.html>.



**Table 1** Input and output variables of the winding process

Data	Variable name	Variable description
Inputs	$S_1$	Angular velocity of reel 1
	$S_2$	Angular velocity of reel 2
	$S_3$	Angular velocity of reel 3
	$I_1$	Set point current at $M_1$
	$I_2$	Set point current at $M_2$
Outputs	$T_1$	Web tension between reel 1 and reel 2
	$T_3$	Web tension between reel 2 and reel 3

modelling process which could improve the performance of the identification process. The input variables of the WP in this study were indicated in the scope of the modelling problem as  $S_1(t)$ ,  $S_2(t)$ ,  $S_3(t)$ ,  $I_1(t)$  and  $I_3(t)$ . The models  $T_1(t)$  and  $T_3(t)$  of the winding problem as defined in Sect. 5 correspond to the tension in the web between reels 1 and 2 and reels 2 and 3, respectively. The measurements  $I_1(t)$  and  $I_3(t)$  were included as the fourth and fifth input variables of the model to improve the performance of the developed models in estimating the tension in the web between the unwinding and rewinding reels and between the rewinding and the traction reels.

## 6.2 Data pre-processing

The data employed to build a model-based for a nonlinear system must be strictly selected to ensure that it is rich enough to avoid trapping into an overfitting problem or early convergence. Data pre-processing methods have to be handled before the learning process to augment the quality of training patterns to reinforce the acquisition of a high reasonable modelling scheme. In essence, ANNs can perform an arbitrary nonlinear mapping between input and output data values. The simplest strategies of pre-processing are data filtering and scaling.

### 6.2.1 Data filtration

Homogenisation and smoothing of intensive changes are needed to apply to the raw data measurements to extract righteous input and output variables from the empirical datasets. In this paper, the collected raw data are passed through a first-order digital Butterworth low-pass filter with a sampling rate of 3 Hz and a bandwidth rate of 0.3 Hz.

### 6.2.2 Data scaling

The input and output datasets of the winding process have diverse ranges, which may lead to unsatisfactory modelling process accompanied with relatively high error rates in the evaluation process. Furthermore, the dataset is not usually

used immediately in the creation of models for industrial systems, as in many cases there is a variation in the magnitude of the variables of the systems. Thus, data scaling is an important issue concerned with high performance of the system identification. Data scaling should be conducted in a fixed range to prohibit data with larger magnitudes from overriding smaller magnitudes and impeding the early learning process. In this paper, the input and output data are scaled in the range between 0.1 and 0.9. The original data  $X$  were mapped to the scaled data  $X'$  as given in Eq. (19):

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \times s + o \quad (19)$$

where  $X_{max}$  and  $X_{min}$  identify, respectively, the maximum and minimum values of the original data,  $X$ ,  $s$  is the scale parameter that is equal to 0.9 and  $o$  is the offset parameter,  $o = 0.1$ .

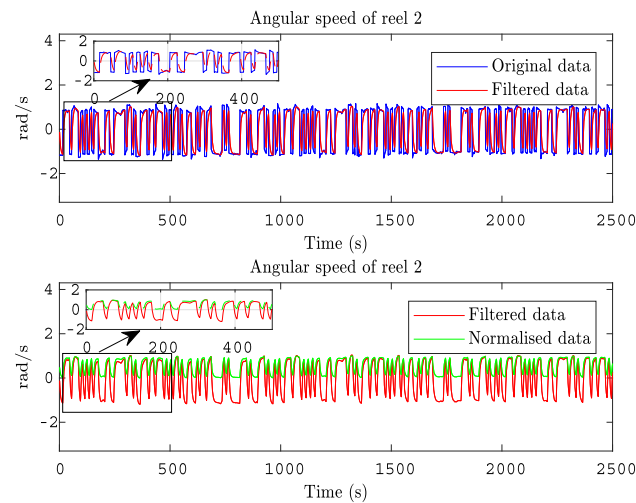
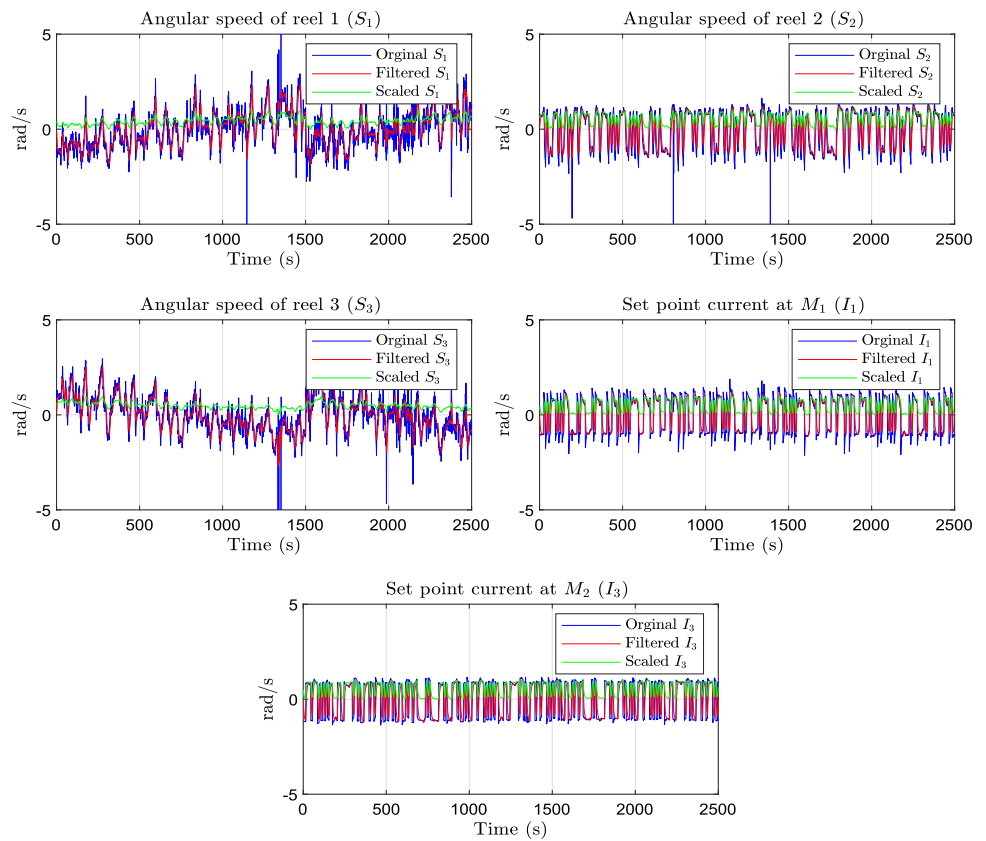
The effects of the Butterworth filter and the subsequent scaling for the inputs of the winding process are shown in Fig. 2. The raw inputs are shown in blue, and the filtered and scaled inputs are shown in red and green colours, respectively.

Figure 3 displays a filtration of the angular speed of reel 2 followed by normalisation.

## 7 Proposed system identification procedure

System identification often uses statistical methods to construct mathematical models for identifying dynamical systems using sets of measured data. In addition to the model preparation process described above, there are further three key issues to be addressed in the design of models for nonlinear industrial processes: model structure selection, model training process and model validation process. These issues should be handled accurately throughout the entire procedure to achieve high performance with a minimum modelling error. The mathematical structure of the proposed model for the web tensions of the winding process and the training algorithms of ANNs is described below:

**Fig. 2** Effects of applying the Butterworth filter on the original signals, followed by scaling. Raw signals are in blue, filtered signals are in red, and scaled signals are in green (color figure online)



**Fig. 3** Filtering and scaling the angular speed of reel 2 to the range between 0.1 and 0.9

- Proposed model structure:** the choice of a model structure with a limited number of parameters to model the WP is difficult due to its non-linearity. The general class of the developed model structure used to predict the target values of the tension webs of the winding process,  $T_1(t)$  and  $T_3(t)$ , follows the pattern given in Eq. (20).

$$y(t) = \phi(S_1(t), S_2(t), S_3(t), I_1(t), I_2(t), \theta)^T \quad (20)$$

where  $y(t)$  represents the tension reels  $T_1$  or  $T_3$ ,  $\phi$  represents the system model,  $\theta$  is the model parameters and  $t$  is the time instances.

To create  $T_1$  and  $T_3$  models of the WP with explainable structures that link inputs and output parameters; we use ANN trained based on PSO, GOA and GWO. These  $T_1$  and  $T_3$  models implement the web tensions between reels 1 and 2 and between reels 2 and 3, respectively. The ANN was organised in three layers; input and output layers in addition to a hidden layer. The ANN is then expanded, where hidden neurons are added to a hidden layer, one by one, until the ANN model is capable of achieving its functionality with a minimal realizable error. This depends primarily on the complexity of the modelling problem which is associated with the complexity of the dataset. The optimum size of ANN was obtained in this case study through train and error process, where the appropriate number of neurons was identified by an adaptive process that added or deleted neurons as required during the training process. In the feedforward (FF) process: the external inputs are initially fed to the input neurons of the input layer; the outputs from the input neurons are fed to the hidden neurons of the hidden layer; and finally the outputs of hidden layer are fed to the output neurons

of the output layer. The first step in training the ANNs is to initialise the weight parameters  $\mathbf{w}$ . Then, during the FF computation, the ANN weights,  $\mathbf{w}$ , were optimised using the proposed BIAs. The learning process consists of adjusting the synaptic weights  $\mathbf{w}$  until they reach the desired behaviour. The output is evaluated to measure the ANN performance; if the output is not as desired, the weights have to be adapted in terms of the input patterns. Here, supervised learning where the goal is to generate an output approximation with the desired patterns of input–output samples set  $p$  as described in Eq. (21) is applied:

$$\mathbf{T}^k = \left\{ \left( x^k \in \mathbb{R}^N, d^k \in \mathbb{R}^M \right) \right\}, \quad k = 1, 2, \dots, p \quad (21)$$

where  $\mathbf{T}^k$  represents the training sample set,  $x$  represents the input patterns,  $d$  is the desired response,  $N$  and  $M$  are the number of samples in the input and output patterns.

The requirement is to design and calculate the NN parameters so that the actual output  $y^k$  of the NN due to  $x^k$  is close statistically to the required degree to  $d^k$  for all  $k$ . The use of the classic BP algorithm to adjust the weights of the ANN as this algorithm, like other traditional algorithms, is based on the descendant gradient technique, which can remain stuck in a local minimum. Also, a BP algorithm cannot solve non-continuous problems. For this reason, other techniques that can address non-continuous and nonlinear problems are crucial to reach a better performance of the ANN and solve complex problems. Here, PSO, GOA and GWO, as described below, were used to adjust the synaptic weights of ANN to obtain a minimum error.

- Proposed training methods:** the ability of an ANN model is affected by the configuration used, particularly the hidden neurons, input and output variables numbers; as the number of model parameters rises, it favours the network learning, and hence, the fitting is effective. In principle, adding more hidden neurons to the models in a systematic strategy should result in regular reductions in the fit error. A proper training process for estimating the ANN parameters is the initial point for defining the model. Prior to feeding the data to the models, an ANN design was determined as described above. This is followed by a selection process to estimate the model parameters. A FF-NN was configured using the input and output datasets of the industrial WP with a number of neurons identified through train and error process. These design matters are illustrated above, and the behaviour of the models using train and error process that resulted in a sensible number of hidden neurons for the models is presented in the results section. Using train and error process appears feasible to develop a model that achieves the desired degree of performance. During training,  $\mathbf{w}$  is

updated through the use of BIAs until the mean square error (MSE) function defined in Eq. (22) is small enough.

$$e = \frac{1}{M \cdot p} \sum_i^M (y_i(x_k, \mathbf{w}) - d_{ik})^2 \quad (22)$$

where  $y_i$  that defines the output value at  $i$ th index was calculated overall  $p$  pattern samples and  $d_{ik}$  is the desired result.

While the effectiveness of GOA and GWO has been confirmed in reliably solving many complex engineering design problems, there is still a lack of research regarding their values in the area of optimising the weights of ANNs, particularly in the modelling of complex nonlinear industrial problems. In this work, we explored their usage in the optimisation of the weights of ANNs in the modelling of the winding process.

The PSO, GOA and GWO were proposed as learning search algorithms to train the ANN in order to: (1) prevent the error function from oscillating around a set of weights without any improvement, (2) estimate the model parameters and, (3) obtain the optimal weights for which the optimal model structures could be created for the winding problem. These optimisation algorithms used the MSE as a fitness function to measure the closeness of the estimated output to the actual output. These modelling approaches using ANN with training on the basis of PSO, GOA and GWO have many merits in generating models with interpretable structure that can relate input and output variables from a given dataset without specifying the key parameters. Further, each proposed modelling scheme has a property to produce a highly robust model and has played an important role in achieving a high level of performance when modelling a nonlinear system for a real industrial process. In the training phase, the input set given to the particular modelling system is fed to the ANN and the learning algorithms. Then, the output of the modelling system is also fed back to the ANN and BIAs, which act as the target patterns. The conduction weights of all interconnections between neurons of ANN are updated based on the proposed optimisation algorithms, namely PSO, GOA and GWO, to reach a predetermined number of iterations or to meet the MSE criterion, in which the specified inputs produce the desired outputs. The estimated output vector of each model and the target vector are iteratively compared during the training process to optimise the modelling error value. This value is used by PSO, GOA and GWO at each iteration to reduce the MSE value along with adjusting the ANN weights by varying the parameters of each optimisation algorithm.

Through these activities, the structured ANN-BIAs mod-

els learn the behaviour of the output response, where each independent model,  $T_1$  and  $T_3$ , of each web tension of the WP is learned with a compact set of parameters. The proposed BIAs are expected to evade the local minimum by exploring a large area of the search domain. Each algorithm has exploratory search features, making it suitable to optimise the weights of ANNs and providing a better scope to get an optimum solution. To sum up the above descriptions of optimising the weight parameters of ANN using PSO, GOA and GWO, they are summarised by the iterative procedural codes in Algorithms 1, 2 and 3, respectively.

---

**Algorithm 1:** A pseudo-code of the proposed PSO tuning algorithm for the ANN model.

---

```

1:  $MaxIter \leftarrow$  Maximum number of iterations.
2:  $Iter \leftarrow$  Current iteration
3:  $d \leftarrow$  A predefined threshold value
4:  $P \leftarrow$  Population size
5: The fitness function (MSE)  $\leftarrow f()$ 
6: Initialise the particles with random position and velocity vectors
7: Initialise the structure of the feedforward NN
8: repeat
9:   for each particle  $i = 1, 2, \dots, P$  do
10:     Feed ANN model structure with the training patterns
11:     if ( $f(X_i) > f(pbest_i)$ ) then
12:        $pbest_i \leftarrow X_i$ 
13:     end if
14:     if ( $f(pbest_i) > f(gbest_i)$ ) then
15:        $gbest_i \leftarrow pbest_i$ 
16:     end if
17:   end for
18:   for each particle  $i = 1, 2, \dots, P$  do
19:     Update particle velocity vector using Eq. (7)
20:     Update particle position vector using Eq. (8)
21:   end for
22:    $MaxIter \leftarrow MaxIter - Iter$ 
23: until ( $MaxIter = 0$  OR  $MSE < d$ )
24: return  $gbest$  (the optimised ANN structure)

```

---

The final optimal solutions for each optimisation algorithm, as described in Algorithms 1, 2 and 3, were used to get the optimum weights of the final model structures of the web tensions,  $T_1$  and  $T_3$ , of the winding process. During the evolution cycle of each optimisation algorithm, the connection weights of the ANN are updated based on the optimisation algorithms so that they can be used to constrain the modelling process to identify the unfamiliar data at the verification processes. In nutshell, the modelling procedures in Algorithms 1, 2 and 3 are iterated until a predefined maximum number of iterations is reached or the improvement in the MSE measure between the target output and the estimated output as measured by Eq. (22) falls below a defined threshold value  $d$ . The model of each proposed algorithm was trained several times to solve the problem of local min-

---

**Algorithm 2:** A pseudo-code of the GOA for adjusting ANN weights.

---

```

1:  $MaxIter \leftarrow$  Maximum number of iterations.
2:  $Iter \leftarrow$  Iteration counter
3:  $d \leftarrow$  A predefined threshold value
4:  $P \leftarrow$  Population size (number of grasshoppers)
5: The fitness function (MSE)  $\leftarrow f()$ 
6:  $c \leftarrow$  Parameter to balance the exploration and exploitation of the entire swarms (grasshoppers).
7:  $c_{max} \leftarrow$  Maximum value of coefficient  $c$ .
8:  $c_{min} \leftarrow$  Minimum value of coefficient  $c$ .
9:  $T \leftarrow$  The best search agent value.
10:  $X_i \leftarrow$  Grasshoppers positions;  $i = 1, 2, \dots, P$ .
11: Initialise  $X_i$ .
12: Configure feedforward NN structure taking into account input and output datasets
13: Initialise  $c_{max} = 1$ ,  $c_{min} = 0.00001$  and  $MaxIter = 1000$ 
14: Calculate the fitness of each searchagent
15: while ( $Iter < MaxIter$ ) OR  $MSE > d$  do
16:    $c \leftarrow c_{max} - Iter(c_{max} - c_{min})/Iter$ 
17:   for each searchagent,  $i = 1, 2, \dots, P$  do
18:     Feed the ANN model with the training patterns
19:     Evaluate the objective function (MSE)
20:     Normalise the distances between grasshoppers  $\in [1, 4]$ 
21:     Update the positions of the current search agents as described in Eq. (11)
22:     Fetch the current search agent back if it goes outside of its bounds
23:   end for
24:   Update  $T$  if is there is a better solution
25:    $Iter \leftarrow Iter + 1$ 
26: end while
27: Return  $T$  (the optimised ANN model structure)

```

---

ima. In these iterative procedures, to create a model based on ANNs trained by the proposed optimisation algorithms, the MSE measure was varied at each iteration to identify the web tension of the WP that best matched the estimated web tension values.

In this case study, the goal is not only to generate models that approximate the values of the underlying web tensions of the winding system, but also to give insight onto the behaviour of these web tensions. The PSO, GOA and GWO summarise the interaction between input and output variables of the WP and also identify the significant variables in this process because these variables will survive and appear in the best individuals at the end of the evolutionary process. The  $T_1$  and  $T_3$  models were designed to capture the main characteristics of the correct output responses of the WP along with the variability of the tuned weight parameters. The results of these iterative algorithms are the models that determine the output vectors representing the web tensions  $T_1$  and  $T_3$  of the winding process. A schematic flowchart of the proposed modelling method for the winding process is shown in Fig. 4.



---

**Algorithm 3:** A pseudo-code of GWO tuning algorithm for tuning the ANN weights.
 

---

```

1:  $MaxIter \leftarrow$  Maximum number of iterations.
2:  $Iter \leftarrow$  Iteration counter
3:  $d \leftarrow$  A predefined threshold value
4:  $P \leftarrow$  Population size (number of grey wolfs)
5: The cost function (MSE)  $\leftarrow f()$ 
6:  $\mathbf{A} \leftarrow$  A coefficient vector.
7:  $\mathbf{C} \leftarrow$  A coefficient vector.
8:  $\mathbf{a} \leftarrow$  A coefficient vector.
9:  $X_\alpha \leftarrow$  The best search agent.
10:  $X_\beta \leftarrow$  The second best search agent.
11:  $X_\delta \leftarrow$  The third best search agent.
12:  $X_i \leftarrow$  Swarm (grey wolfs) positions ;  $i = 1, 2, \dots, P$ .
13: Initialise  $X_i$ 
14: Initialise the feedforward NN structure taking into account
    input and output datasets
15: Initialise  $\mathbf{a}$  to [2, 0]
16: Initialise  $\mathbf{A}$  as defined in Eq. (14)
17: Initialise  $\mathbf{C}$  as defined in Eq. (15)
18: Calculate the fitness of each searchagent
19: while ( $Iter < MaxIter$  OR  $MSE < d$ ) do
20:   for each searchagent,  $i = 1, 2, \dots, P$  do
21:     Feed the ANN model structure with the training patterns
22:     Evaluate the MSE function
23:     Update the position of the current searchagent by Eq.
        (18)
24:   end for
25:   Update  $\mathbf{a}$ ,  $\mathbf{A}$ , and  $\mathbf{C}$ 
26:   Calculate the fitness of all searchagents
27:   Update  $X_\alpha$ ,  $X_\beta$ , and  $X_\delta$  using Eq. (16)
28:    $Iter = Iter + 1$ 
29: end while
30: Return  $X_\alpha$  (the optimised ANN model structure)
  
```

---

The first step in Fig. 4 is to read the collected data of the winding process. The second step is data pre-processing, which poses a crucial step before building the model. This step consists of data filtration and normalisation. Data filtering step aims to ameliorate data quality and transform it to a more suitable form. Data normalisation aims to get rid of the scale differences between the variables of the process. The third step of the proposed modelling method before model creation is identifying the training dataset, which is associated with sample and variables selection. In this step, half of the datasets were used for training, and all process variables were chosen to create the model. After preparing the dataset, the next step was building and configuring the ANN model, where PSO, GOA and GWO were used to update the weights of the ANN model. The model parameters are then updated iteratively until a maximum number of iterations is arrived at or the enhancement in the error measure computed as given in Eq. (22) is small enough. The green arrow represents the feedback of the modelling method. This results in the models, ANN-PSO, ANN-GOA and ANN-GWO, which are used to test the validity and suitability of the

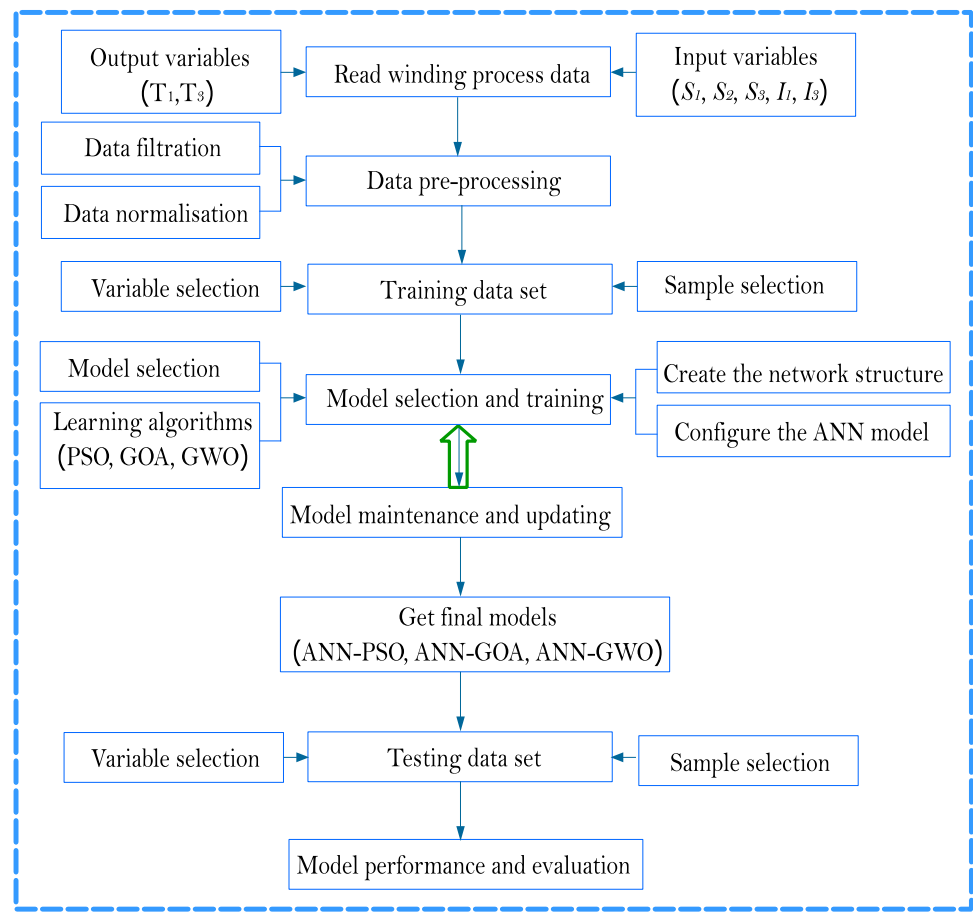
proposed modelling method using a test dataset independent of that used in the training dataset. The last step is to evaluate the performance of the modelling method for both training and test datasets.

According to the pre-processing phase, the proposed model structure and the training methods used to optimise the ANN weights, the developed models, ANN-PSO, ANN-GOA and ANN-GWO, are expected to be effective in describing and simulating the dynamics behaviour of the winding process. It is anticipated that these models can achieve very modest training and evaluation errors. This result is expected due to the coveted features of the adopted algorithms, PSO, GOA and GWO, which can largely avoid local solutions in finding the optimal weights for the ANN model. Other fascinating features of these optimisation algorithms include: 1) their ability to learn ANNs much faster than traditional learning algorithms suchlike BP and LM methods and 2) their efficiency in locating the global optimum and obtaining the optimal set of weights for the ANNs, where an optimal model structural will be obtained. In sum, there are a host of benefits of an ANN model trained by PSO, GOA and GWO, including robustness in term of noise resistance and disturbing signals, rapid training and evaluation capabilities, avoiding tripping in an undesirable solution and arriving at a high level of reliability in understanding and describing the process under study. The use of the algorithms, PSO, GOA and GWO, provides a stable optimisation for the weights of the ANN model that will present fast convergence with very low error rates in modelling. This is due to that these algorithms: (1) search for optimal weights without relying on the initial weights and (2) enhance the diversity of solutions and are not susceptible to converge at local optima. So, these models are powerful in optimising the ANN weights and have a high potentiality for capturing the important characteristics of the web tensions of the winding process with relatively simple mathematical formulas.

## 8 Model acceptance and evaluation

Model selection with a suitable evaluation measure is critical to identify how the model pursues during the evaluation process. This process is required to validate the model and illustrate the power of the model-based approach to identify T<sub>1</sub> and T<sub>3</sub> data. The capability of the developed models to characterise the behaviour of the web tensions of the WP was verified through the computation of several metric measures. If the identified models fail to reach a predefined accuracy with an acceptable degree of performance, the modelling process returns back to its training phase. Therefore, perfor-

**Fig. 4** A scientific diagram showing the key procedures of the proposed modelling approach (color figure online)



mance criteria are needed to identify the level of similarity between the data generated by real experiments and the data produced from the developed models.

The performance of the developed models of the sub-systems,  $T_1$  and  $T_3$ , of the winding problem was assessed based upon several criteria as defined below.

1. The mean absolute percentage error (MAPE), as defined below, is a measure of the percentage foretelling precision of the developed models.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (23)$$

where  $n$  stands for the number of data values used in the experiments and  $y$  and  $\hat{y}$  represent the  $n$  observed and modelled values, respectively.

2. The MSE or  $E$  as defined below assesses the accuracy of difference between the actual data produced by the measurement tools and their corresponding predicted data obtained from the models.

$$E = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (24)$$

where  $(y_i, \hat{y}_i)$  represents a single data web tension value for observation  $i$  of the measured value acquired from the experiment and the estimated web tension value created by the modelling procedure, respectively. An accurate estimation of a MSE value is achieved if the difference between the expected and the measured values is within tolerance.

3. The Pearson product–moment correlation coefficient ( $R$ ) as described below was used to test the potentiality of the developed models:

$$R = \frac{\sum_{i=1}^n (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2 \sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})^2}} \quad (25)$$

where  $y_i$  is the  $i$ th actual value of the web tension output measured by a real experiment,  $\hat{y}_i$  is the  $i$ th predicted web tension value generated from the model,  $\bar{y}$  and  $\bar{\hat{y}}$  are the mean values of the actual and foreseen web tension outputs,  $y$  and  $\hat{y}$ , respectively.

4. The coefficient of determination ( $R^2$ ), as given below, measures the differences between the observed and predicted means and variances of the observed data that can be explained by the model.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (26)$$

5. The variance-accounted-for (VAF) measure, as defined in Eq. (27), measures the proximity of the measured web tension values obtained by real experiments to the estimated web tension values created by the developed models.

$$VAF = \left[ 1 - \frac{\text{var}(y - \hat{y})}{\text{var}(y)} \right] \times 100\% \quad (27)$$

where  $\text{var}(y)$  is the variance of the actual web tension data and  $\text{var}(y - \hat{y})$  is the variance of the difference between the actual web tension data ( $y$ ) and the model data ( $\hat{y}$ ).

The aforementioned evaluation measures are relevant and compelling to quantify the accuracy of the developed models. These criteria help to assess the potential of the developed models for the prediction of  $T_1$  and  $T_3$  and to single out the degree of agreement between the real output data and the estimated output data. A statistical analysis was performed using Friedman statistical test Friedman 1937 to verify the efficiency of each presented model. This test is able to rank each model along with other conventional and state-of-the-art models.

## 9 Experimental results and discussion

### 9.1 Machine and software specifications

Speed of computation is influenced by the volume of data to be processed, the time taken to perform the computation of each model and the complexity of the algorithm. Here, the developed models are implemented in Java under Microsoft Windows 10 platform. All experiments were run on an Intel Machine with Core *i5* processor running at 2.50 GHz with 6.0 GB of RAM. The experiments were designed to assess the effect of tuning the weights of ANN using PSO, GOA and GWO. Each experiment was reiterated 10 times. The results of the proposed modelling methods are recorded in terms of the performance criteria defined above and compared with a set of popular traditional and state-of-the-art methods reported in the literature. The programming language, software and hardware specifications were presented to provide an indication of the efficiency of the developed models in terms of the computational burden. The average speed of operations for the 10 experiments using ANN-PSO, ANN-GOA and ANN-GWO are, respectively, about 79.453, 72.283 and 59.172 seconds, each with the controlling parameters given in Table 2. It is observed that the average computational

time of the developed modelling schemes is relatively low. This confirms that the computational effort of each developed model is persuaded and assures that the proposed modelling schemes are computationally efficient to address the WP. Indeed, the computational time is only intended to provide some allusions about the time requirements for the future researchers. In short, the elapsed computational time affirms that proposed models are capable of handling WP at a very high speed on a computer with modest specifications such as the one described above.

### 9.2 Experimental setup of ANN-based algorithms

The feedforward ANN was configured with a number of neurons identified as described along with the input and output data sets of the WP. The ANN weights were controlled through each of the PSO, GOA and GWO to obtain the optimised models for  $T_1$  and  $T_3$  tensions. The control parameters of these tuned optimisation algorithms are displayed in Table 2.

The control parameters in Table 2 were adjusted by the use of design of experiments to fit each swarm-based algorithm to the underlying nature of the modelling problem. The trial-and-error process was used to find the best parameters for each tuned algorithm. A reasonable wide range of each control parameter was initially defined and the experiments were run systematically. This is to avoid wasting time with untargeted experiments and to have an idea of the behaviour of the optimisation algorithms for different settings, and based on these results to perform a fine tune. The trends for the values of parameters were defined to determine whether the best values are within the range or there is a need to further experiments. The parameter values were varied several times until a sensible solution was obtained. However, most often only good settings are obtained, perhaps, not the “best” settings.

### 9.3 Simulation of the developed WP models

The proposed modelling schemes are expected to create models for  $T_1$  and  $T_3$  of the WP with explainable structures that link inputs and outputs with key parameters selection. The leading objective is to originate  $\hat{T}_1(t)$  and  $\hat{T}_3(t)$  models that describe the aspects of the WP that often resemble the actual  $T_1$  and  $T_3$ , respectively. Three models were developed using ANN-PSO, ANN-GOA and ANN-GWO for  $T_1$  and  $T_3$ . Each developed model was trained on a data set of 1250 samples of each input and output parameters and evaluated on a separate data set of 1250 previously unseen samples. The simulation results are shown below during training and verification stages to illustrate the capability of the modelling schemes.

**Table 2** Values of the controlling parameters of the tuned algorithms

Algorithm	Parameter	Value
PSO	Populations	200
	Generations	1200
	Cognitive and social constants	1.5 and 2.5
	Inertia weights	Linearly decrease from 0.9 to 0.3
GOA	Search agents	200
	Generations	1200
	$c_{max}$	1
	$c_{min}$	0.00004
GWO	Search agents	200
	Generations	1200
	The vector $\mathbf{a}$	Linearly decreases from 2.0 to 0.0

### 9.3.1 Simulation result-based regression

Multiple nonlinear regression was used to build a regression model for the WP. Least square estimation was used to estimate the model parameters. The produced models for the web tension models,  $T_1$  and  $T_3$ , are given in Eqs. (28) and (29), respectively.

$$T_1 = 0.04365 - 0.00217 S_1 + 0.00356 S_2 - 0.00852 S_3 - 0.00742 I_1 + 0.00169 I_2 \quad (28)$$

$$T_3 = 0.04158 - 0.00019 S_1 - 0.00219 S_2 - 0.00050 S_3 - 0.00322 I_1 + 0.00729 I_2 \quad (29)$$

The actual and predicted web tension values,  $T_1$  and  $T_3$ , obtained based on MNLR in both the training and testing cases are shown in Figs. 5 and 6 for both  $T_1$  and  $T_3$ , respectively.

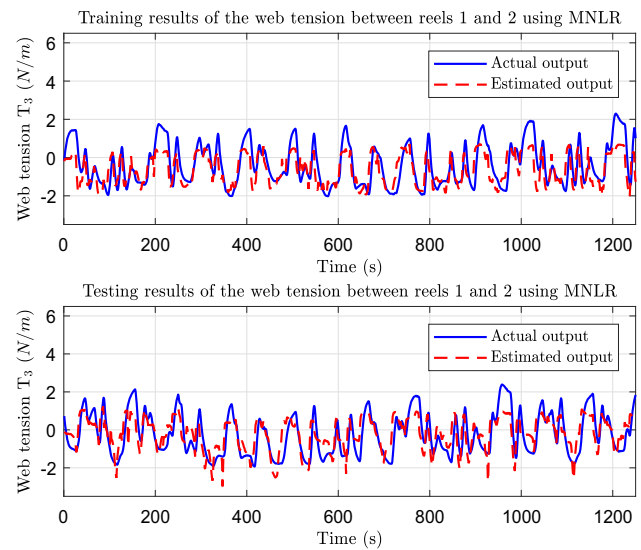
The study of residuals is essential to MNLR modelling process and deciding the performance of the proposed models. The computed correlation coefficients ( $R$ ) values over training and testing data for  $T_1$  and  $T_3$  are shown in Figs. 7 and 8, respectively.

### 9.3.2 Simulation result-based ANN-PSO model

The actual and predicted web tension between reels 1 and 2 and web tension between reels 2 and 3 in both training and testing cases are shown in Figs. 9 and 10, respectively, where the models were created based on ANN-PSO.

The obtained correlation coefficients ( $R$ ) values based on ANN-PSO models for both the training and testing data for  $T_1$  and  $T_3$  are shown in Figs. 11 and 12, respectively.

It is observed from the modelling and identification results shown in Figs. 9 and 10 that the predicted data output identified by red plots fit competently to the actual data output as identified by blue plots for each subsystem of the winding process reactor, as judged visually. These results appear



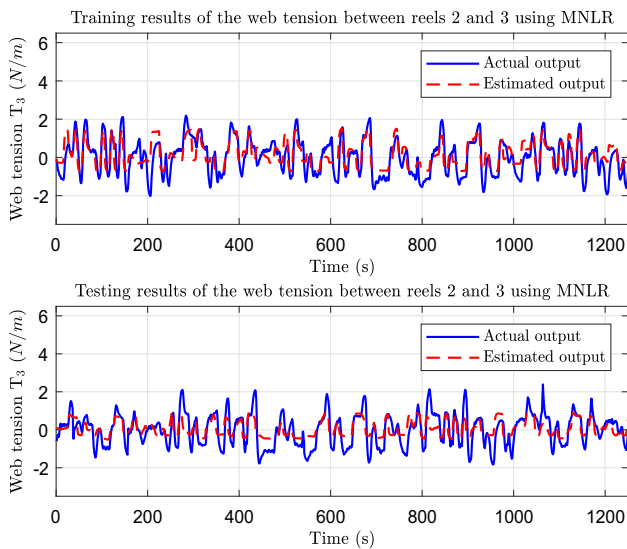
**Fig. 5** Actual and predicted  $T_1$  in training and testing cases based on multiple nonlinear regression

particularly prominent given a model with a relatively large training dataset. These results further underline the validity and feasibility of such ANN-PSO model in describing the dynamic behaviour of the winding process, enabling it to model any industrial process. This in turn means that refinement and data normalisation procedures are valuable to retain the significant aspects of the datasets.

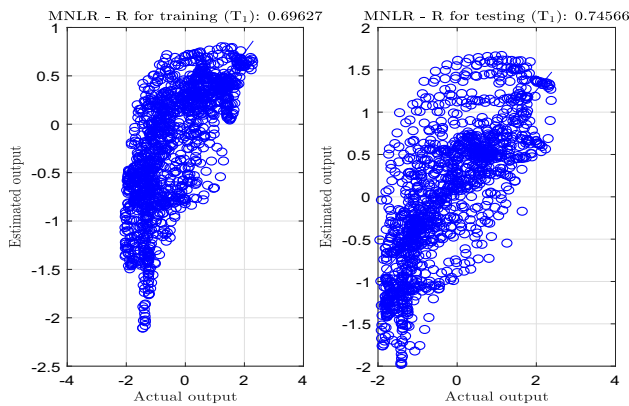
### 9.3.3 Simulation result-based ANN-GOA model

The actual and predicted web tension between reels 1 and 2 and between reels 2 and 3 in both training and testing processes based on the final developed ANN-GOA models for  $T_1$  and  $T_3$  of the winding process is shown in Figs. 13 and 14, respectively. The top part of the figure gives the training results, while the test results are given in the bottom part. The

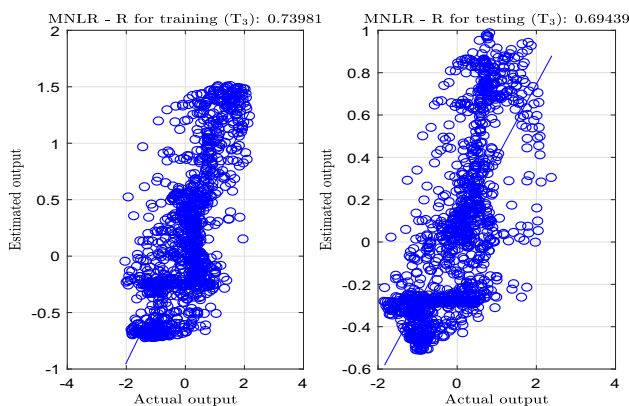




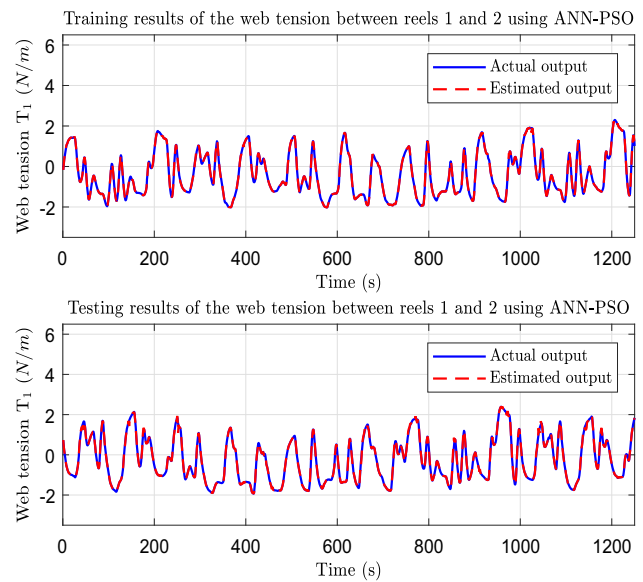
**Fig. 6** Actual and predicted  $T_3$  in training and testing cases based on multiple nonlinear regression



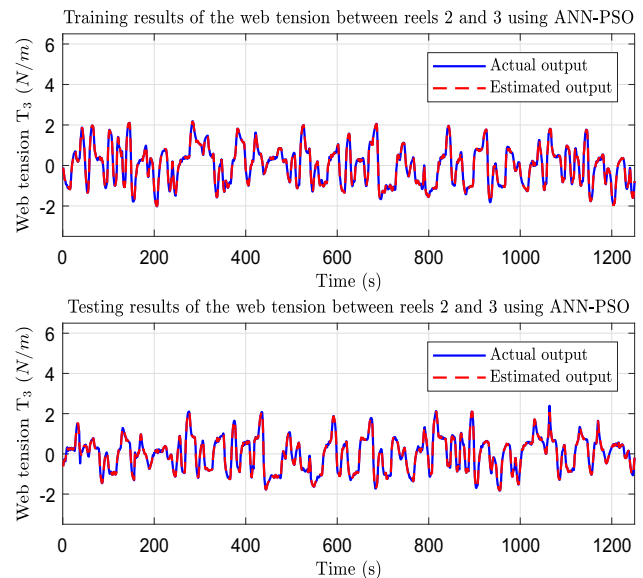
**Fig. 7** Multiple nonlinear regression: computed  $R$  over training and testing data of  $T_1$



**Fig. 8** Multiple nonlinear regression: computed  $R$  over training and testing data of  $T_3$



**Fig. 9** Actual and predicted results of  $T_1$  in training and testing cases (color figure online)



**Fig. 10** Actual and predicted results of  $T_3$  in training and testing cases based on ANN-PSO (color figure online)

actual web tension values are shown in red, and the predicted web tension values are shown by a blue line.

The computed correlation coefficients ( $R$ ) values based on ANN-GOA models for training and testing cases for both  $T_1$  and  $T_3$  are shown in Figs. 15 and 16, respectively.

Figure 13 shows that in the case of  $T_1$ , the simulation results visually are a little better than the  $T_3$  case, as shown in Fig. 14 for training and testing data (shown in red and blue, respectively), and both results are reasonable. However, a modest error can be observed between the target and the predicted output in the testing cases, but it is non-significant.

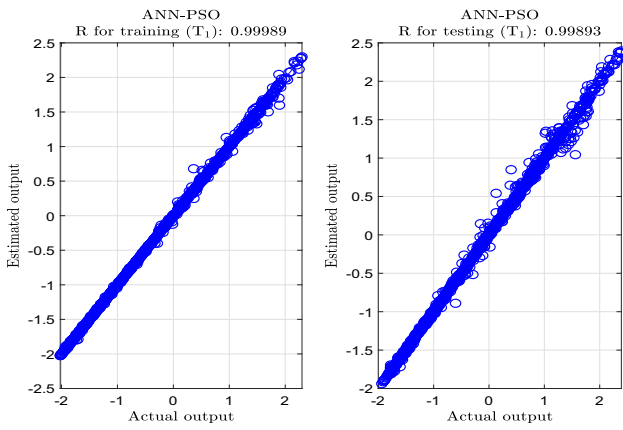


Fig. 11 Computed  $R$  over training and testing data of  $T_1$  using ANN-PSO

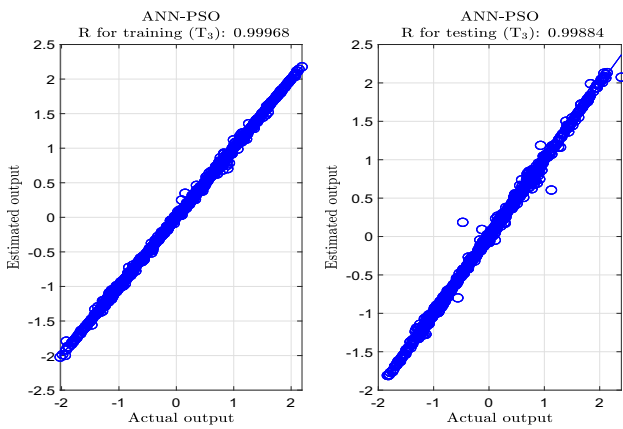


Fig. 12 Computed  $R$  over training and testing data of  $T_3$  using ANN-PSO

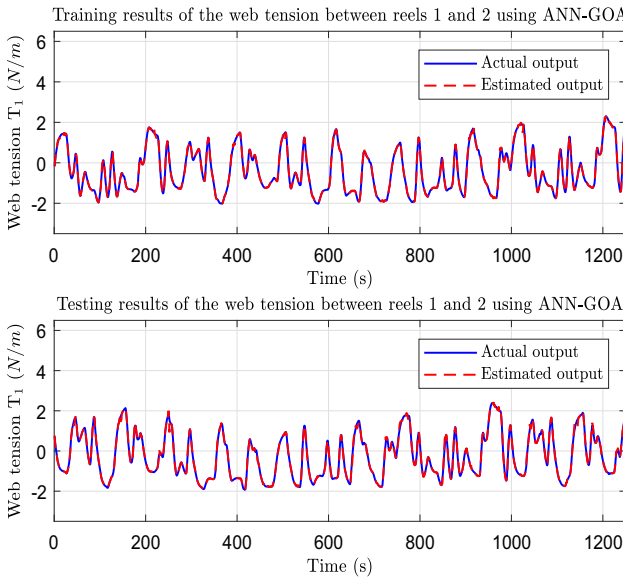


Fig. 13 Actual and predicted results of  $T_1$  in training and testing cases based on ANN-GOA (color figure online)

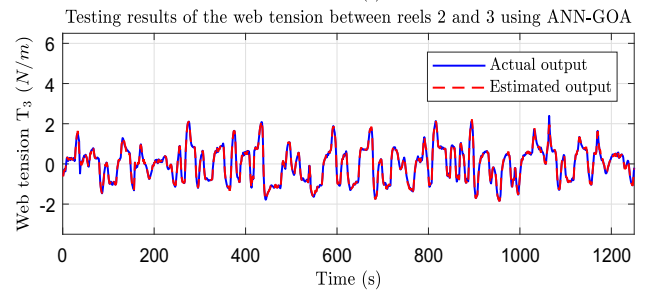
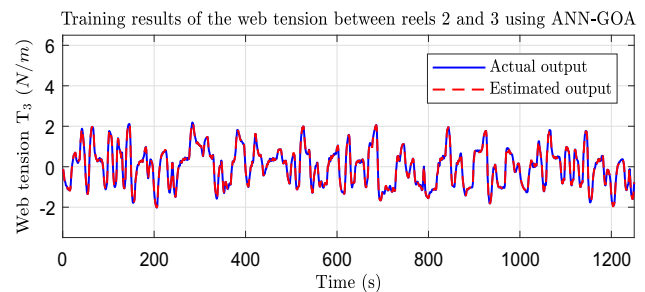


Fig. 14 Actual and predicted results of  $T_3$  in training and testing cases based on ANN-GOA (color figure online)

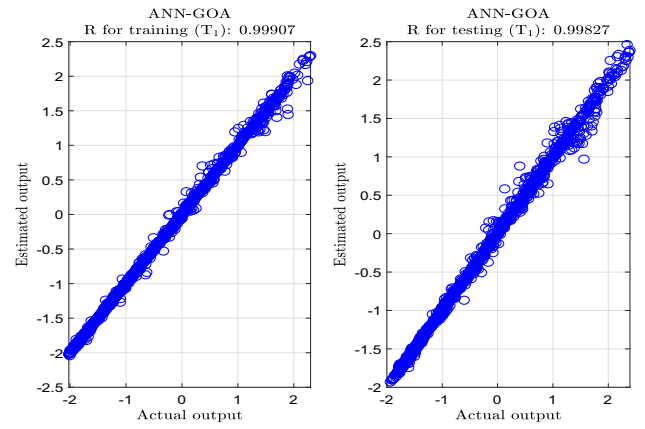


Fig. 15 Computed  $R$  over training and testing data of  $T_1$  using ANN-GOA

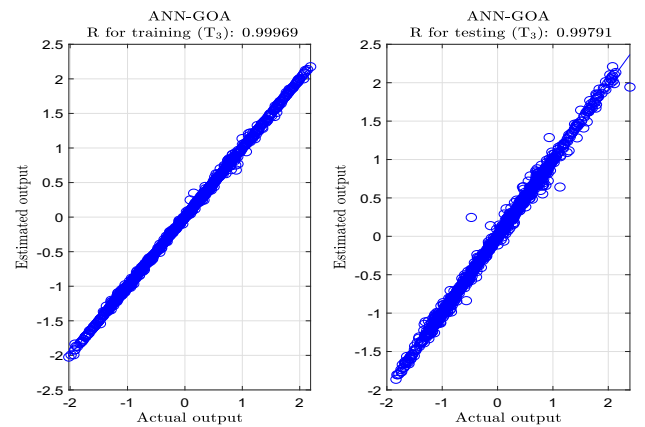
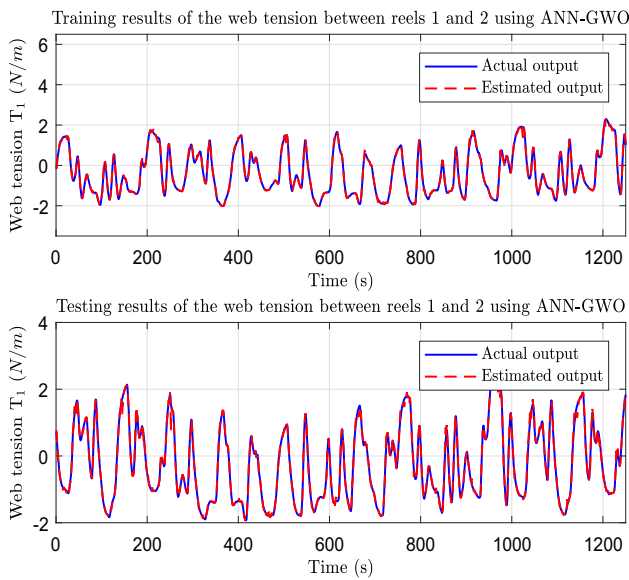
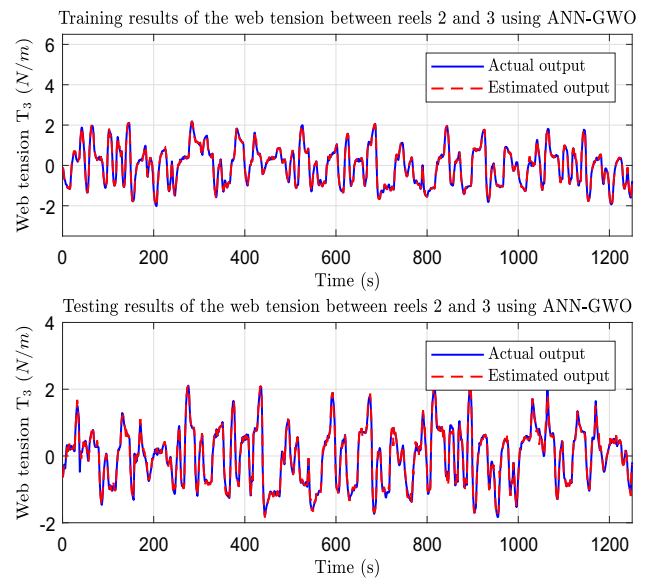


Fig. 16 Computed  $R$  over training and testing data of  $T_3$  using ANN-GOA



**Fig. 17** Actual and predicted results of  $T_1$  in training and testing cases using ANN-GWO (color figure online)



**Fig. 18** Actual and predicted results of  $T_3$  in training and testing cases using ANN-GWO (color figure online)

These simulation results affirm the effectiveness of the ANN-GOA-based approach for creating a model-based system. The nature of the errors in Figs. 13 and 14 was associated with the fact that the predicted web tensions between reels 1 and 2 and between reels 2 and 3 are not following exactly the corresponding actual web tensions and that the target values were not detected well.

### 9.3.4 Simulation result-based ANN-GWO model

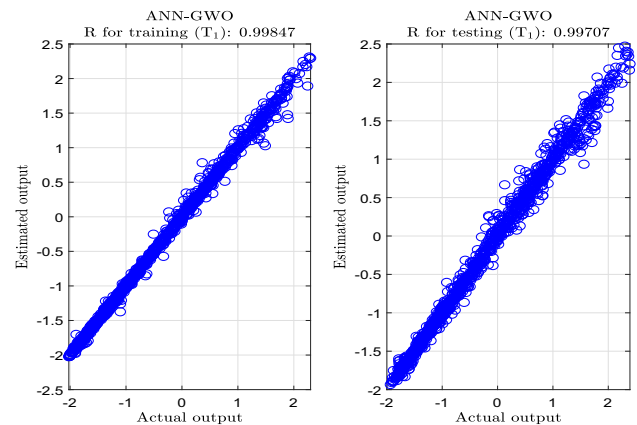
The actual and the predicted  $T_1$  and  $T_3$  values based on the developed ANN-GWO models are appeared in Figs. 17 and 18 for both training and testing cases. The actual values are identified by blue and the predicted values by a red line.

The correlation coefficients ( $R$ ) values obtained based on ANN-GWO models for training and testing cases for both  $T_1$  and  $T_3$  are shown in Figs. 19 and 20, respectively.

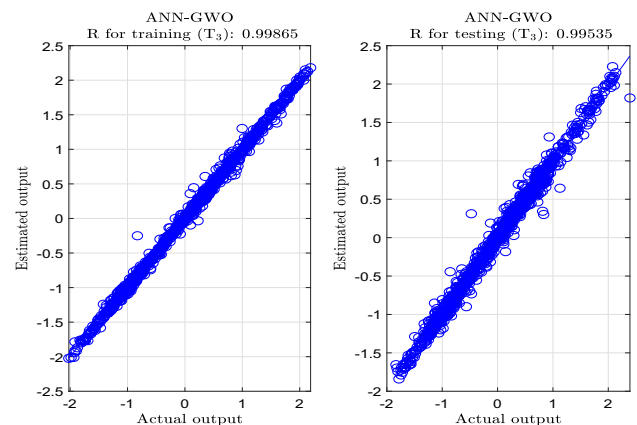
The simulation results shown in Figs. 17 and 18 confirm the validity and appropriateness of the ANN-GWO-based model for modelling the winding process in both training and testing cases. It is observed that the ANN-GWO-based modelling approach showed better results for  $T_1$  than the  $T_3$  as visually observed.

In short, the correlation coefficients ( $R$ ) results obtained-based ANN-PSO, ANN-GWO and ANN-GOA modelling approaches for both training and testing cases demonstrate that a high degree of performance and consistent results were repeatedly obtained over the entire period of the sampling time.

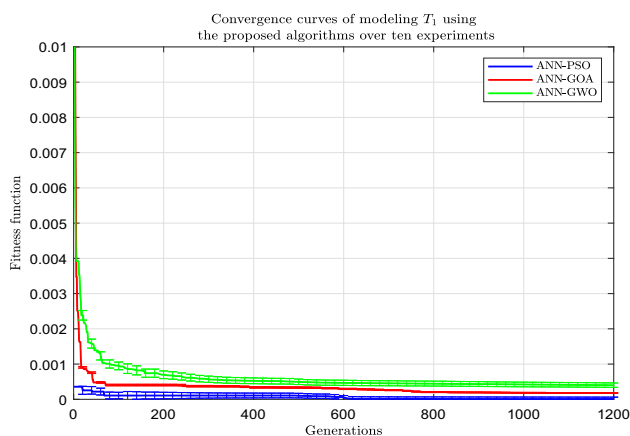
The results illustrated in Figs. 9, 10, 11, 12, 14, 14, 15, 16, 17 and 18 show the ability and reliability of the proposed



**Fig. 19** Computed  $R$  over training and testing data of  $T_1$  using ANN-GWO



**Fig. 20** Computed  $R$  over training and testing data of  $T_3$  using ANN-GWO



**Fig. 21** Best convergence curves over ten experiments for the ANN-PSO, ANN-GOA and ANN-GWO models for the web tension  $T_1$  at different MSE values

modelling approaches in the modelling of the web tensions of the WP. The plot results of the models for the actual and predicted web tension between reels 1 and 2 and between reels 2 and 3 in the training and testing cases are almost replica to each other, meaning that there is no much difference with a deviation in the data in the corresponding points of the plots, as visually shown. So, these obtained results prove a highly convincing level of modelling and identification for training and testing processes for both web tension cases. It is also observed from Figs. 9, 10, 11, 12, 14, 14, 15, 16, 17 and 18 that the presented ANN-PSO-, ANN-GOA- and ANN-GWO-based data modelling is at a high degree of performance, such that the estimated  $T_1$  and  $T_3$  outputs highly resemble the actual  $T_1$  and  $T_3$  outputs, respectively.

We can observe from all these figures that the estimated winding process output is almost identical to the actual winding process output. This demonstrates the efficacy of each of ANN-PSO, ANN-GOA and ANN-GWO in training ANNs. These results further show superior performance of the proposed modelling methods compared to previous methods in the literature such as MNL method.

#### 9.4 Convergence

The convergence curves that identify the performance of the ANN-PSO, ANN-GOA and ANN-GWO models, for modelling the web tension  $T_1$  of the WP, are presented for up to 1200 iterations in Fig. 21.

The data of the plots in Fig. 21 represent the mean of the sum of the squared errors between the estimated data values created by the presented modelling schemes and the corresponding real data measurements. The vertical error bars that appear on the convergence curves represent one standard error of the mean.

The standard error of the mean was used to determine whether the differences between the means of the data measurements of the models are statistically significant. The convergence curves show that the ANN model-based PSO, GOA and GWO converge rapidly to the desired global minimum error. There is a consistent difference between ANN-PSO and ANN-GOA as well as the ANN-GWO results. The error bars of the curves generated by ANN-GOA and ANN-GWO overlap at many points along the curves. The curve plots of ANN-PSO and ANN-GOA in addition to ANN-GWO models are obviously separated and the error bars do not interfere along the curves. In sum, the convergence curves in Fig. 21 confirm that the ANN-PSO, ANN-GOA and ANN-GWO models have reached a high degree of performance in modelling  $T_1$ .

#### 9.5 Evaluation results

To obtain a quantitative assessment of the performance of the proposed models: ANN-PSO, ANN-GOA and ANN-GWO, a set of measures were calculated to identify the degree of similarity between the actual output data and the corresponding estimated data. The performance results for modelling the web tensions  $T_1$  and  $T_3$  are shown in Tables 3 and 4, respectively.

Tables 3 and 4 show that the modelling and evaluation results are reasonable in both training and testing. There is a relatively high correlation between the predicted and true web tensions datasets as observed in Tables 3 and 4, whereas a minimum value of 99.707 was reported for the correlation  $R$  at testing the ANN-GWO model. Interestingly, the ANN-PSO models reported better results than the ANN-GOA and ANN-GWO models. The ANN-GOA models reported correlation rates in training and testing for both  $T_1$  and  $T_3$  comparable to the correlation rates reported by ANN-PSO model. The ANN-GWO model did not perform as the other models on both  $T_1$  and  $T_3$  datasets. However, the difference margin to the corresponding data of ANN-GOA results is relatively small.

The results in terms of VAF measure for a series of experiments to evaluate  $T_1$  and  $T_3$  models for all proposed models, with a different number of hidden neurons in each experiment, are shown in Tables and, respectively.

It is observed from Tables 5 and 6 that there is a gradual improvement in the VAF rate as the number of hidden neurons increases. Further, there is a very small increase in the VAF rate when the number of neurons increases from 18 to 24. The difference is less than 0.02 on average which is a small and not statistically significant. This finding shows that 18 or 24 hidden neurons in the models are not critical to performance. The application of a smaller number of neurons will be more rapid. There is a factor of approximately 0.04 on average in VAF rate between the smallest and biggest num-



**Table 3** Training and testing results of web tension  $T_1$  arrived at by the proposed models

Model	Training $T_1$ data			Testing $T_1$ data		
	Evaluation criteria			Evaluation criteria		
	MAPE	$R(\%)$	$R^2(\%)$	MAPE	$R(\%)$	$R^2(\%)$
ANN-PSO	0.0727	99.988	99.966	0.0966	99.893	99.786
ANN-GOA	0.101	99.969	99.937	0.155	99.827	99.653
ANN-GWO	0.153	99.865	99.731	0.194	99.707	99.414

**Table 4** Modelling and evaluation results of web tension  $T_3$  based on the proposed models

Model	Training $T_3$ data			Testing $T_3$ data		
	Evaluation criteria			Evaluation criteria		
	MAPE	$R(\%)$	$R^2(\%)$	MAPE	$R(\%)$	$R^2(\%)$
ANN-PSO	0.0931	99.980	99.960	0.105	99.884	99.768
ANN-GOA	0.143	99.907	99.814	0.190	99.791	99.581
ANN-GWO	0.192	99.847	99.694	0.232	99.535	99.072

**Table 5** Web tension  $T_1$ : Evaluation VAF results of the proposed models

Number of hidden neurons	Training case		
	ANN-PSO	ANN-GOA	ANN-GWO
6	99.7252	98.9331	97.8358
12	99.7592	98.9973	97.8588
18	99.7768	99.0158	97.8769
24	99.7852	99.0244	97.8846
	Testing case		
6	98.7218	97.6211	96.1001
12	99.0237	98.163	96.9871
18	99.0237	98.1806	97.0047
24	99.0332	98.2182	97.0143

**Table 6** Web tension  $T_3$ : Evaluation results based on the proposed models

Number of hidden neurons	Training case		
	ANN-PSO	ANN-GOA	ANN-GWO
6	99.6215	98.7143	96.9842
12	99.6842	98.7762	97.0928
18	99.7008	98.7961	97.1095
24	99.7196	98.8133	97.1362
	Testing case		
6	98.2276	97.8533	96.7754
12	98.8143	98.002	96.8221
18	98.9063	98.0236	96.8577
24	98.9239	98.0412	96.8753

ber of neurons. This is a major difference in computational burden.

To sum up, the evaluation results in Tables 5 and 6 divulge that the ANN-PSO approach showed a large level of accuracy in terms of training and testing in comparison with ANN-GOA and ANN-GWO; also, there is no significant difference between ANN-GOA and ANN-GWO in this study. Further,

the results showed that ANN-PSO, ANN-GOA and ANN-GWO models have a significant level of modelling ability, where the ANN-PSO model has better accuracy.

The applications of the models proposed to the winding process presented very satisfactory results, where there are always gains when adding more hidden neurons to the ANN, systematically increasing fit performance. In addition,

**Table 7** A comparison in terms of the MSE between the proposed modelling approaches and other reported modelling approaches for  $T_1$  and  $T_3$  of the WP for a set of ten experiments

Model	Number of neurons		Training data		Testing data	
	$T_1$	$T_3$	MSE		MSE	
			$T_1$	$T_3$	$T_1$	$T_3$
MNLR	–	–	0.3925	0.4024	0.5912	0.6346
LSE	–	–	0.1247	0.2547	0.1359	0.2430
MLP	12	5	8.93e-04	9.27e-04	8.61e-04	1.10e-03
RBF	10	10	8.76e-04	8.37e-04	8.54e-04	2.20e-03
LLNF	20	25	1.44e-04	8.86e-04	7.72e-04	1.10e-03
ANN-PSO	18	18	3.17e-06	3.54e-06	4.22e-06	4.91e-06
ANN-GOA	18	18	1.22e-04	1.58e-04	3.17e-04	4.13e-04
ANN-GWO	18	18	4.91e-05	5.08e-05	5.21e-05	5.95e-05

it is possible to reach levels close to unit in performance, indicating that it is possible to reach desired levels of fit performance, even when the target of performance is unit. This ratifies that the proposed models achieved a high level of modelling performance in the modelling of the winding process.

While the effectiveness of the proposed models is highly reliable in modelling the winding process, there is a need in some evaluation cases to increase the number of hidden neurons to augment the performance level of the developed models. Another limitation of the proposed models is related to the relatively high computational cost. The computational cost of the proposed modelling method depends on the parameters of both the ANN model and the optimisation algorithms used to optimise the ANN weights, as the optimisation algorithms require a number of search agents and generations that are sufficiently adequate to obtain a high level of performance.

## 9.6 A comparison with other reported models

The aforementioned rendering criteria are used to obtain a quantitative evaluation of the performance obtained through the modelling schemes proposed. The models were trained ten times on a data set of 1250 samples for each input variable of the WP. The models were then tested on a different data set of 1250 samples for each input variable of the WP. The performance of the proposed ANN-PSO, ANN-GOA and ANN-GWO models is compared to conventional and state-of-the-art approaches reported in the literature, in which they are modelled the same process. The conventional approaches include MNLR and LSE. The state-of-the-art approaches are the ANN-based RBF and MLP as well as the RLLNF reviewed in the literature. The performances of the developed models for the web tensions between reels 1 and 2 ( $T_1$ ) and between reels 2 and 3 ( $T_3$ ) in terms of the MSE are displayed in Table 7 for both training and testing data sets. In Table 7, the number of neurons of the neural network is given.

**Table 8** Variance account for values for  $T_1$  and  $T_3$  of the WP using the developed models over ten experiments

Model	Training data		Testing data	
	VAF		VAF	
	$T_1$	$T_3$	$T_1$	$T_3$
MNLR	55.5995	47.3457	54.6961	44.3654
LSE	87.3062	75.2313	86.7406	73.9855
MLP	98.4746	93.5317	98.6752	89.9052
RBF	98.5999	94.4855	98.6899	86.7895
LLNF	99.6915	94.0865	98.8234	92.2721
ANN-PSO	99.7768	99.7008	99.0237	98.9063
ANN-GOA	99.0158	98.7961	98.1806	98.0236
ANN-GWO	97.8769	97.1095	97.0047	96.8577

The evaluation results presented in Table 7 show the appropriateness of the proposed model-based methods for modelling  $T_1$  and  $T_3$ . It is clear that a high level of performance was achieved using the proposed ANN-PSO model in both training and testing. Besides, the ANN-GOA-based approach achieved a faintly better degree of performance for  $T_1$  and  $T_3$  than the ANN-GWO model-based RMLP approach. As Table 7 demonstrates, the proposed models presented superior performance compared to the other model-based traditional and state-of-the-art methods.

The evaluation training and testing results for the presented modelling systems in modelling the  $T_1$  and  $T_3$  are summarised in terms of the VAF criterion in Table 8.

Table 8 compares the evaluation strategy and the performance accuracy of the models presented in this paper to four modelling methods for the same industrial process. It is clear that a high level of performance is obtained using the proposed models. Moreover, the VAF results of the proposed models are significantly much better than the conventional models and slightly better than a state-of-the-art model Nozari et al. 2012. This illustrates that the presented models were managed to be a well representation of the WP

**Table 9** Average rankings of the presented models (Friedman)

Model	Ranking
LSE	7.0
MLP	5.0
RBF	4.5
LLNF	3.25
ANN-PSO	1.0
ANN-GOA	3.0
ANN-GWO	4.25
MNLR	8.0

web tensions. The results in Tables 7 and 8 reveal that PSO, GOA and GWO algorithms are efficacious in adjusting the weights of ANNs and reaching optimal MSE and VAF values. The overall results prove that the proposed modelling techniques have produced valuable and adaptable models in representing the behaviour of the WP web tensions and further are capable of obtaining good mathematical models for other complex industrial and control problems.

## 9.7 Statistical analysis

A statistical analysis was performed using Friedman statistical test Friedman 1937, at a significance level of  $\alpha = 0.05$ , to verify the efficiency of each model presented and to rope the statistical preciseness of the proposed models with the accuracy of conventional and state-of-the-art models. This is to rank the presented models alongside conventional and state-of-the-art models. It is also able to identify the best model that yielded the best performance that can be defined as the control model. Holm's procedure Holm 1979 was then performed in accordance to the Friedman test results to identify the degree of significance of the differences between the control model and the other models. In addition, Holm's procedure was conducted to obtain the adjusted  $p$  values for each comparison addressed between the control model and the other models, as well as rejecting the null hypothesis of comparable performance between the models. Table 9 shows the ranking produced by Friedman test for the presented model.

The ranking results in Table 9 show that ANN-PSO model is ranked first, followed in order by ANN-GOA, LLNF, ANN-GWO, RBF, MLP and LSE models. The  $p$ -value of Holm's procedure in Table 10 was adjusted using the ANN-PSO model because it was identified as the control model. The  $p$  value computed by Friedman test is 0.00651, which is less than  $\alpha = 0.05$ . This means that there is a large difference between the results of the evaluated models. Thus, Holm's procedure was applied to demonstrate whether this difference is statistically significant between the control model and the other models.

**Table 10** Adjusted  $p$  value and the null hypothesis based on the Holm's procedure with  $\alpha = 0.05$  (Friedman)

$i$	Model	$p$ value	$\alpha \div i$	Null hypothesis
7	MNLR	5.3121E-5	0.0071	Rejected
6	LSE	5.3200E-4	0.0083	Rejected
5	MLP	0.0209	0.01	Not rejected
4	RBF	0.0433	0.0125	Not rejected
3	ANN-GWO	0.0606	0.0166	Not rejected
2	LLNF	0.1939	0.025	Not rejected
1	ANN-GOA	0.2482	0.05	Not rejected

Holm's procedure rejects those hypotheses that have a  $p$  value  $\leq 0.01$ . The Holm's results, presented in Table 10, show that ANN-PSO model, statistically, reported results significantly better than the classical and state-of-the-art models for the winding process data. Further, the results shown in Table 10 confirm that ANN-GOA and ANN-GWO models are, statistically, exhibited a considerable level of performance.

To recapitulate, the sensible performance of the ANN-based PSO, GOA and GWO models demonstrates the generalisation capabilities and substantiates the modal basis for these models in modelling the case study.

The developed models have learned to precisely characterise the behaviour of the winding process due to the finding of the optimum weights for the proposed ANN model on the basis of the algorithms PSO, GOA and GWO. As a result, the developed models have achieved high accuracy values and very low error rates. This high degree of performance is considered a big advantage of the proposed modelling method. However, due to the use of PSO, GOA and GWO in training the ANN, the models ANN-PSO, ANN-GOA and ANN-GWO are subject to a relatively high computational burden. Additionally, increasing the number of neurons in the ANN model will raise the cost.

The nonlinear simulator models, ANN-PSO, ANN-GOA and ANN-GWO, presented in this work, have the ability to simulate winding plant behaviour afterward training without requiring real winding plant output data, but simply by providing the models with the inputs. This feature allows the proposed models to design and simulate model-based controllers over the full operating ranges of a nonlinear winding system. In short, the ANN-PSO, ANN-GOA and ANN-GWO models are highly efficient in the modelling of the winding process which can further be utilised to model any other complex industrial process.

## 10 Conclusion and future work

The proposed work has demonstrated the use of three bio-inspired algorithms (BIAs) to train artificial neural networks (ANNs) to derive three intelligent models for an industrial winding process (WP). These BIAs include particle swarm optimisation (PSO), grasshopper optimisation algorithm (GOA) and grey wolf optimiser (GWO). These models are referred, respectively, to as ANN-PSO, ANN-GOA and ANN-GWO. In the context of creating an optimal model structure for the WP, the mean square error (MSE) was used as a fitness function to evaluate each solution. The performance of the proposed models was assessed through conducting several experiments using the MSE, Variance-Accounted-For (VAF) and the correlation criteria. All evaluation measures reported a high level of agreement between modelled and observed WP output data, suggesting that the presented models are highly accurate. On all criteria, a very high degree of performance was realised by the proposed intelligent models with a correlation coefficient of more than 99% in all training and testing processes. In the simulation results, the proposed ANN-PSO, ANN-GOA and ANN-GWO models have created a faithful replica for their corresponding WP output data. These findings clearly evidenced the potential of PSO, GOA and GWO as promising learning algorithms for the optimisation of ANN weights. The results of the proposed models favourably outperformed state-of-the-art models and adduced superior performance rates considerably better than the conventional Least Square Error (LSE) method. Consequently, the proposed modelling method can be used as a base system upon to model any nonlinear industrial system and can offer a way to compare improvements between convincing modelling methods. The use of a new model-based identification system to model the WP is a useful direction and an interesting trend for future research and experiments. This model-based simulator approach could be further used to design a model predictive controller system or other winding systems with a set of variable parameters. Further work is demanded to assess the aptness of the proposed modelling approach to other real industrial benchmark datasets.

## Compliance with ethical standards

**Conflict of interest** The authors declare that there is no conflict of interest regarding the publication of this paper.

**Human and animal rights statement** This article does not contain any studies with human participants or animals performed by any of the authors.

## References

- Al-Azzeh J, Alqadi Z, Abuzalata M (2019) Performance analysis of artificial neural networks used for color image recognition and retrieving. *Int J Comput Sci Mob Comput* 8(2):20–33
- Ansari A, Gupta NK (2011) Automated diagnosis of coronary heart disease using neuro-fuzzy integrated system. In: *World congress on information and communication technologies*. IEEE, vol 2011, pp 1379–1384
- Arunkumar N, Mohammed MA, Mostafa SA, Ibrahim DA, Rodrigues JJ, de Albuquerque VHC (2020) Fully automatic model-based segmentation and classification approach for MRI brain tumor using artificial neural networks. *Concurr Comput Pract Exp* 32(1):e4962
- Asadi Asad Abad MR, Borghei AM, Ahmadi H, Minaei S, Beheshtivoli B (2015) Fuzzy logic based classification of faults in mechanical differential. *J Vibroeng* 17(7):3635–3649
- Ayough A, Khorshidvand B (2019) Designing a manufacturing cell system by assigning workforce. *J Ind Eng Manag* 12(1):13–26
- Azizivahed A, Narimani H, Fathi M, Naderi E, Safarpour HR, Narimani MR (2018) Multi-objective dynamic distribution feeder reconfiguration in automated distribution systems. *Energy* 147:896–914
- Babuška R, Verbruggen H (2003) Neuro-fuzzy methods for nonlinear system identification. *Annu Rev Control* 27(1):73–85
- 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 RD, Ogunnaike BA, Featherstone AP (1996) Identification, estimation, and control of sheet and film processes. *IFAC Proc Vol* 29(1):6638–6643
- Braik M, Sheta A, Arieqat A (2008) A comparison between GAs and PSO in training ANN to model the TE chemical process reactor. In: *Proceedings of the AISB 2008 convention in communication, interaction and social intelligence*, vol 1, p 24
- Chang Y-W, Hsieh C-J, Chang K-W, Ringgaard M, Lin C-J (2010) Training and testing low-degree polynomial data mappings via linear SVM. *J Mach Learn Res* 11(Apr):1471–1490
- Chang P-C, Wu J-L, Xu Y, Zhang M, Lu X-Y (2019) Bike sharing demand prediction using artificial immune system and artificial neural network. *Soft Comput* 23(2):613–626
- Crone SF, Kourentzes N (2009) Input-variable specification for neural networks-an analysis of forecasting low and high time series frequency. In: *International joint conference on neural networks, IJCNN 2009*. IEEE, pp 619–626
- Dao SD, Abhary K, Marian R (2017) Optimisation of assembly scheduling in VCIM systems using genetic algorithm. *J Ind Eng Int* 13(3):275–296
- Dixit SR, Das SR, Dhupal D (2019) Parametric optimization of Nd: Yag laser microgrooving on aluminum oxide using integrated RSM-ANN-GA approach. *J Ind Eng Int* 15(2):333–349
- El-Thalji I, Jantunen E (2015) A summary of fault modelling and predictive health monitoring of rolling element bearings. *Mech Syst Signal Process* 60:252–272
- Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J Am Stat Assoc* 32(200):675–701
- Gupta S, Deep K (2018) A novel random walk grey wolf optimizer. *Swarm Evol Comput* 44:101–112
- Haddad M, Guillaumat L, Terekhina S, Crozatier M (2017) Analytical and numerical study based on experimental investigation of different curved sandwich composites manufactured by filament winding process. *J Compos Mater* 2017:0021998317714858
- Holm S (1979) A simple sequentially rejective multiple test procedure. *Scand J Stat* 6:65–70
- Hussein E, Sheta A, El Wahab A (2001) Modeling of a winding machine using non-parametric neural networks. In: *WSEAS international*

- conference on scientific computation and soft computing, pp 528–533
- Hussian A, Sheta A, Kamel M, Telbaney M, Abdelwahab A (2000) Modeling of a winding machine using genetic programming. In: Proceedings of the 2000 congress on evolutionary computation, IEEE, vol 1, pp 398–402
- Jain M, Narayan S, Balaji P, Bhowmick A, Muthu RK et al (2020) Speech emotion recognition using support vector machine. [arXiv:2002.07590](https://arxiv.org/abs/2002.07590)
- Kennedy J (2011) Particle swarm optimization. In: Encyclopedia of machine learning. Springer, pp 760–766
- Khosravi A, Koury R, Machado L, Pabon J (2018) Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. *Sustain Energy Technol Assess* 25:146–160
- Li J, Cheng J-H, Shi J-Y, Huang F (2012) Brief introduction of back propagation (BP) neural network algorithm and its improvement. In: Advances in computer science and information engineering. Springer, pp 553–558
- Liu J, Shao Y (2017) Dynamic modeling for rigid rotor bearing systems with a localized defect considering additional deformations at the sharp edges. *J Sound Vib* 398:84–102
- Liu J, Shao Y (2018) Overview of dynamic modelling and analysis of rolling element bearings with localized and distributed faults. *Nonlinear Dyn* 93:1765–1798
- Liu J, Shao Y, Lim TC (2014) Impulse vibration transmissibility characteristics in the presence of localized surface defects in deep groove ball bearing systems. *Proc Inst Mech Eng Part K J Multi-body Dyn* 228(1):62–81
- Ljung L (1987) Theory for the user. Prentice Hall, Upper Saddle River
- Masadeh R, Alzaqebah A, Hudaib A, Rahman AA (2018) Grey wolf algorithm for requirements prioritization. *Mod Appl Sci* 12(2):54
- Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. *Adv Eng Softw* 69:46–61
- Moslemipour G (2018) A hybrid CS-SA intelligent approach to solve uncertain dynamic facility layout problems considering dependency of demands. *J Ind Eng Int* 14(2):429–442
- Mousavi SH, Nazemi A, Hafezalkotob A (2015) Using and comparing metaheuristic algorithms for optimizing bidding strategy viewpoint of profit maximization of generators. *J Ind Eng Int* 11(1):59–72
- Naderpour H, Mirrashid M (2019) Shear failure capacity prediction of concrete beam–column joints in terms of ANFIS and GMDH. *Pract Period Struct Des Constr* 24(2):04019006
- Nikabadi M, Naderi R (2016) A hybrid algorithm for unrelated parallel machines scheduling. *Int J Ind Eng Comput* 7(4):681–702
- Noura H, Theilliol D, Ponsart J-C, Chamseddine A (2009) Fault-tolerant control systems: design and practical applications. Springer, Berlin
- Nozari HA, Banadaki HD, Mokhtare M, Vahed SH (2012) Intelligent non-linear modelling of an industrial winding process using recurrent local linear neuro-fuzzy networks. *J Zhejiang Univ Sci C* 13(6):403–412
- Omotosho A, Oluwatobi AE, Oluwaseun OR, Chukwuka AE, Adekanmi A (2018) A neuro-fuzzy based system for the classification of cells as cancerous or non-cancerous. *Int J Med Res Health Sci* 7(5):155–166
- Parant F, Coeffier C, lung C (1992) Modeling of web tension in a continuous annealing line. *Iron Steel Eng (USA)* 69(11):46–49
- Pirdashti M, Curteanu S, Kamangar MH, Hassim MH, Khatami MA (2013) Artificial neural networks: applications in chemical engineering. *Rev Chem Eng* 29(4):205–239
- Rajasekaran MP, Sri Meena R (2012) Application of adaptive neuro-fuzzy inference systems for MR image classification and tumour detection. *Int J Biomed Eng Technol* 9(2):133–146
- Ranganathan A (2004) The Levenberg–Marquardt algorithm. *Tutor LM Algorithm* 11(1):101–110
- Sadati N, Chinnam RB, Nezhad MZ (2018) Observational data-driven modeling and optimization of manufacturing processes. *Expert Syst Appl* 93:456–464
- Santillan JH, Tapucar S, Manliguez C, Calag V (2018) Cuckoo search via Lévy flights for the capacitated vehicle routing problem. *J Ind Eng Int* 14(2):293–304
- Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw* 105:30–47
- Saritas MM, Yasar A (2019) Performance analysis of ANN and Naive Bayes classification algorithm for data classification. *Int J Intell Syst Appl Eng* 7(2):88–91
- Schlei-Peters I, Wichmann MG, Matthes I-G, Gundlach F-W, Spengler TS (2018) Integrated material flow analysis and process modeling to increase energy and water efficiency of industrial cooling water systems. *J Ind Ecol* 22(1):41–54
- Sheta AF, Braik M, Al-Hiary H (2009) Identification and model predictive controller design of the Tennessee Eastman Chemical Process using ANN. In: Proceedings of the international conference on artificial intelligence (ICAI'09), July 13–16, USA, vol 1, pp 25–31
- Sheta AF, Braik M, Öznergiz E, Ayeshe A, Masud, M (2013) Design and automation for manufacturing processes: an intelligent business modeling using adaptive neuro-fuzzy inference systems. In: Business intelligence and performance management. Springer, pp 191–208
- Sheta AF, Öznergiz E, Abdelrahman M, Babuska R (2009) Modeling of hot rolling industrial process using fuzzy logic. In: CAINE, pp 81–86
- Sheta A, Braik M, Al-Hiary H (2019) Modeling the Tennessee Eastman chemical process reactor using bio-inspired feedforward neural network (bi-ff-nn). *Int J Adv Manuf Technol* 103(1–4):1359–1380
- Sievers L, Balas MJ, von Flotow A (1988) Modeling of web conveyance systems for multivariable control. *IEEE Trans Autom Control* 33(6):524–531
- Torres PJR, Mercado ES, Rifón LA (2018) Probabilistic Boolean network modeling of an industrial machine. *J Intell Manuf* 29(4):875–890
- Wang Y-R, Yu C-Y, Chan H-H (2012) Predicting construction cost and schedule success using artificial neural networks ensemble and support vector machines classification models. *Int J Project Manag* 30(4):470–478
- Wang G, Tang W, Xia J, Chu J, Noorman H, Gulik WM (2015) Integration of microbial kinetics and fluid dynamics toward model-driven scale-up of industrial bioprocesses. *Eng Life Sci* 15(1):20–29
- Wang Y, Li H, Qi C (2020) An adaptive mode convolutional neural network based on bar-shaped structures and its operation modeling to complex industrial processes. *Chemom Intell Lab Syst* 2020:103932
- Yıldız AR (2008) Hybrid Taguchi-Harmony search algorithm for solving engineering optimization problems. *Int J Ind Eng* 15(3):286–293
- Zhang X, Han Q, Peng Z, Chu F (2016) A comprehensive dynamic model to investigate the stability problems of the rotor-bearing system due to multiple excitations. *Mech Syst Signal Process* 70:1171–1192
- Zingg DW, Nemeč M, Pulliam TH (2008) A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization. *Eur J Comput Mech* 17(1–2):103–126