



Estimating ARMA Model Parameters of an Industrial Process Using Meta-Heuristic Search Algorithms

Alaa F. Sheta^{1*}, Hossam Faris² and Ibrahim Aljarah²

¹*Department of Computing Sciences, Texas A&M University-Corpus Christi, Corpus Christi, TX 78412, alaa.sheta@tamucc.edu*

²*King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan, hossam.faris@ju.edu.jo, i.aljarah@ju.edu.jo*

Abstract

This paper addresses the parameter estimation problem for a manufacturing process based on the Auto-Regressive Moving Average (ARMA) model. The accurate estimation of the ARMA model's parameter helps to reduce the production costs, provide better product quality, increase productivity and profit. Meta-heuristic algorithms are among these approximate techniques which have been successfully used to search for an optimal solution in complex search space. Meta-heuristic algorithms can converge to an optimal global solution despite traditional parameter estimation techniques which stuck by local optimal. A comparison between Meta-heuristic algorithms: Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Accelerated PSO, Cuckoo Search, Krill Herd and Firefly algorithm is provided to handle the parameter estimation problem for a Winding process in the industry. The developed ARMA-meta-heuristics models for a winding machine are evaluated based on different evaluation metrics. The results reveal that meta-heuristics can provide an outstanding modeling performance.

Keywords: *Manufacture Process; Meta-Heuristic Search Algorithms; Parameter Estimation*

1. Introduction

Currently, physical process modeling has become one of the challenging problems in the application of control system design [1, 2]. Traditional modeling techniques are complex, time-consuming and unsatisfactory especially when there is a lack of exact knowledge about the system or present noise. However, the development of an appropriate model structure for a winding process in the industry appears to be essential for model-based control. The goal of modeling is always to develop a correct and reliable model of a real manufacture process.

With the growth of manufacture process complexity and the need for more advanced software simulation tools for process monitoring and control; a need for the development of advanced methods for the analysis and design of nonlinear industrial processes was created [3, 4]. The goal of the new process models is to meet the required high process safety standard and top product quality. Automation of manufacture processes involves the development of several precise mathematical models to enhance the potentiality of simulating and quantifying complex industrial operation.

Linear models have been proposed as a solution for the system identification process. These models are simple, like the Moving Average (MA) model. Thus, we may use a simple algorithm to estimate the model parameters like Least-Square Estimation (LSE) [5-7]. LSE has been used as a useful tool

in solving the parameter estimation problem for a known system structure. This structure can be either linear or nonlinear. Although, most nonlinear systems can be linearized by representation with differential or difference equations, modeling nonlinear systems using linear models involves many approximations. These approximations are sometimes not sufficient to reflect the real behavior of the nonlinear systems. Thus, to get a suitable model structure which reflects real system information, there is usually an increasing cost. This cost is due to the need for more advanced algorithms which can handle complex model structures.

To solve the model identification problem many models were introduced [8-11]. A few of these models are the Hammerstein, Winer, Winer-Hammerstein and the Volterra-series. Indeed, for many dynamic systems, the use of nonlinear models can characterize physical processes [12]. Unfortunately, most of these models have some difficulties. For example, a large number of parameters that need to be identified, i.e., when using the Volterra series [13, 14].

Modeling and control of various winding process were explored in the literature [1, 2, 15, 16]. The winding process at the Alum factory at Naga Hammadi in Egypt was studied in many publications. A Genetic Programming (GP) model was considered to model the dynamic of a winding process. GP was able to explore the space of possible tree structure and find a simple mathematical model the provided better results than the traditional linear model [17]. A Feedforward



Neural Network with BackPropagation (FFBP) was also proposed to develop a nonlinear model for the winding process [18]. Recently, authors in [19] provided a new method based Bi-directional Echo State Reservoir Networks (Bi-ESNs) to model the winding system.

In this paper, a number of nature-inspired meta-heuristic algorithms for parameter tuning of an ARMA model to better model the dynamics of a manufacturing process in a real winding machine. The meta-heuristic algorithms applied in this work include common algorithms such as GA, PSO, Accelerated PSO (APSO), Cuckoo Search (CS), Krill Herd (KH) and Firefly algorithm (FA). The developed and optimized models are evaluated and compared based on different evaluation metrics.

This paper is organized as follows. In Section 2, background on various metaheuristic algorithms and how they work in practice is presented. The description includes Genetic Algorithms, Particle Swarm Optimization and the Accelerated PSO, Cuckoo Search, Krill Herd and Firefly algorithm. Various model structure for modeling dynamic system is presented in Section 3. We give an introduction to the winding process in the industry in Section 4. In Section 5, the formulation of the parameter estimation problem-based ARMA model is presented. The evaluation metrics adopted to test the performance of the developed estimate based on various metaheuristic algorithms are introduced in Section 6. In Section 7, we provide the setup parameters for tuning each algorithm adopted in this study and the experiments and results.

2. Nature-inspired Metaheuristics

Various Meta-Heuristic algorithms were explored to solve different types of problems such as global function optimization [20], optimizing neural networks [21–24], software effort estimation [25], and parameter estimation problem for manufacturing processes. Due to space constraints, we focus only on closely related work of based estimation problem for manufacturing processes that used nature-inspired algorithms. System identification of single-input-multiple-output (SIMO) systems was studied in [26]. In [27], GAs was used to solve the parameters of an induction motor from machine test data. GAs was also used to estimate the longitudinal aerodynamic parameters of an airplane in [28]. Another trend for parameter optimization PSO was introduced to solve many parameter estimation problems in the industry. In [29], PSO algorithm was used to estimate the parameters of nonlinear chaotic systems. PSO successfully estimated the model parameters in the presence of noise.

Metaheuristics are global search algorithms, and their goal is to find an acceptable solution within a reasonable timeframe when the problem is very complex, and the search space is extremely large. In their essence, metaheuristics incorporate randomness and a local search in their process. These features support metaheuristics to find a suboptimal solution when applying traditional algorithms for evaluating every possible solution is impossible. However, this doesn't mean grantee that metaheuristics will always find the optimal solution neither that they will work. There are two main components of the metaheuristic algorithms: exploration and exploitation. In the exploration component, the algorithm tries to explore and test different areas in the search space, while on the other hand, in the exploitation component, the algorithm tries to focus the search around some suboptimal found solutions [30]. Nature-inspired algorithms are popular types of metaheuristics. As their name implies, the idea and the design of these

algorithms are inspired by phenomena that occur in nature. Most of the nature-inspired algorithms are population-based algorithms where they start by randomly generating a pre-determined number of candidate solutions (also called individuals) then they start to iteratively update the generated solutions using a specifically designed mechanism. In every iteration, the algorithm evaluates all individuals using a fitness function to assess their quality considering them as possible solutions for the targeted problem. In some metaheuristics, fitness values affect the search direction of the algorithm. Evolutionary Algorithms were introduced to solve many problems in manufacturing and industry [31–33, 33]. Evolutionary algorithms are mainly inspired by the Darwinian theory of evolution and natural selection. The best example of this type is the well-regarded Genetic Algorithm (GA). GA was first proposed and designed in the works of John Holland [34, 35]. GA is distinguished by its reproduction operators namely; the crossover and mutation operators. On the other side, most of the Swarm Intelligence algorithms are inspired by the movement or interaction of some families of birds, fish or animals in nature. A well-regarded example of this category is the Particle Swarm Optimization (PSO). PSO was first introduced by Kennedy and Eberhart in 1995 [36]. In PSO, individuals (or particles) are updated based on the best found solution for all individuals and the best solution found by the updated individual itself. Other remarkable and more recent nature-inspired metaheuristic algorithms are: Krill Herd (KH) [37], Accelerated PSO (APSO) [30], Cuckoo Search (CS) [38, 39] and Firefly Algorithm (FA) [30, 40].

3. ARMA Model

One of the most significant step of modeling a system is the choice of a suitable model structure. The determination of model order help to fit the model with the collected measurements via the input output process. This step if implemented correctly lead to a better model performance [41]. An example of an input-output model is given in Equation 1.

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-n), u(t-1), u(t-2), \dots, u(t-m)) \quad (1)$$

where $u(t), y(t)$ represents the system input-output at time t . n and m represent the number of past outputs and inputs, respectively. f can be a static or dynamic function which provides the input-output relationship. If a system under study is linear, f is a linear function and Equation 1 can be revised to become:

$$y(t) = a_1y(t-1) + a_2y(t-2) + \dots + a_ny(t-n) + b_1u(t-1) + b_2u(t-2) + \dots + b_mu(t-m) \quad (2)$$

ARMA model represent a form of linear time series models which are widely used to solve many modeling problems in engineering [17, 18]. The equation that formulate a ARMA model with two inputs $u_1(t)$ and $u_2(t)$ and a single output $y(t)$ can be described in Equation 3.

$$y(t) = \sum_{i=1}^n \alpha_i u_1(t-\tau) + \sum_{i=1}^n \beta_i u_2(t-\tau) + \sum_{i=1}^n \gamma_i y(t-\tau) \quad (3)$$

$y(t-\tau)$ represents system output response where τ ($\tau \in 1, 2, \dots, m$). m is the model order. i is the time samples that takes values from $i = 1, \dots, n$.

4. Winding Machine in Industry

The winding model plant presented, in this study, characterizes a subsystem frequently used in numerous manufacture processes such as in paper industry where rolling mills and web conveyance are used [15]. Its main function is to control the linear speed, the thickness or/and the tension of a strip. Tension is defined as the longitudinal force applied to process material, or in another word, how tight the material is pulled. Assume that the line speed of a process is kept constant with a pinch roll. Thus, the winder motor is going to be controlled by a motor controller which will regulate a fixed motor current. A cross-section of the Winding Machine in the industry is shown in Figure 1. The PLC is used to collect data which helps in the development of a winding machine process model.

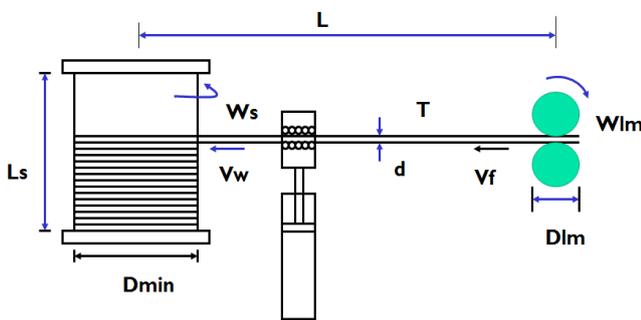


Figure 1: Cross section of the Winding Machine

4.1. Winding Process Model

The winding process can be described by physical laws based Hooke's Equation 4 with given parameters in Table 1.

$$\frac{dT}{dt} = \frac{EA}{l} [V_w - V_f] \quad (4)$$

Table 1: Definition of system parameters

T	Wire tension	(N)
V_w	linear speed of winding spool	(m/s)
V_f	linear feeding speed	(m/s)
E	The wire Young's modulus	(N/mm ²)
l	feeding and winding section's distance	(m)
A	Wire cross section	(mm ²)

In our winding system V_w and V_f are defined as in Equation 5 with the equation's parameters are defined in Table 2. A simple diagram that shows the wire path inside the wire feed component is shown in Figure 2 [8]. The radius variations results in a corresponding variations in the moment of inertia of the winding spool.

$$\begin{aligned} V_w &= \frac{2\pi w_{sm}}{60n_s} R_i \\ V_f &= \frac{2\pi w_{fm}}{60n_f} \frac{D_{ml}}{2} \\ R_i &= \left[\frac{D_{min}}{2} + id \right] \end{aligned} \quad (5)$$

Table 2: Definition of system parameters

w_{fm}	Angular speed of the feeding motor	(rad/min)
w_{sm}	Angular speed of the spool motor	(rad/min)
D_{ml}	Rolling mill diameter	(m)
R_i	Radius of spool at later i	(m)
n_f	Gear box ratio of the feeding motor	
n_s	Gear box ratio of the spool motor	

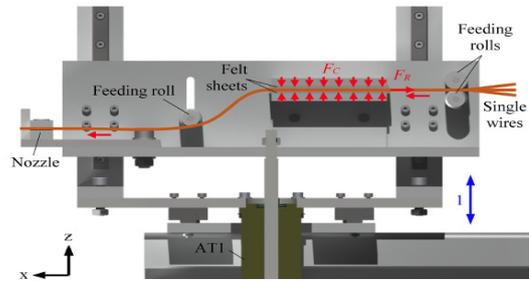


Figure 2: Sectional view of the wire feed [8]

5. Problem Formulation

To see how we can use the metaheuristic algorithms to solve the parameter tuning problem for the winding process, we assume we can formulate the problem to a minimization problem. This problem should have a function f to be minimized with a set of parameters n as follows:

$$\text{Min. } f(\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3, \gamma_1, \gamma_2, \gamma_3)$$

Each parameters is defined in a certain domain $D = [d_{min}, d_{max}]$, and $f(\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3, \gamma_1, \gamma_2, \gamma_3)$ is a positive function for all values of $\alpha_i, \beta_i, \gamma_i \in D$. Candidate solutions are defined as n -dimensional vectors of parameters of the form:

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \beta_1 & \beta_2 & \beta_3 & \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}$$

The parameters are presented in a vector notation format. For each vector of parameter, there is an associated function value serves as its fitness, with lower values preferred for minimization problems.

6. Evaluation Metrics

To see how the estimated parameters using metaheuristic search algorithms perform well in the modeling of the manufacture process, we adopted number of evaluation criteria such as VAF, MAE, MMRE, RMSE and R^2 . The performance of the optimized ARMA based model is evaluated using the following evaluation criteria:

$$VAF = \left[1 - \frac{\text{var}(y - \hat{y})}{\text{var}(y)} \right] \times 100\% \quad (6)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (7)$$

$$MMRE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} \quad (8)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2} \quad (9)$$

$$R^2 = \frac{\sum_{i=1}^n (y_i - \bar{Y}_i)^2 - \sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (10)$$

7. Experimental Setup and Results

7.1. Tuning Parameters

Tuning parameters of the metaheuristic algorithms have a high influence on their performance and the quality of the generated solutions. Therefore, having more parameters makes the tuning problem itself more difficult. On the other, having fewer parameters are considered as an advantage for the algorithm. Among all the algorithms investigated in this study, Krill Herd and Cuckoo search have the smallest number of parameters.

Krill Herd depends only on the number of Krills and a scale factor for the speed vector while Cuckoo search depends only the P_a fraction on besides the number of Cuckoos. P_a fraction is set to 0.25. Speed vector factor of the Krill Herd algorithm depends on the search space and can be empirically set to a value in $[0,2]$ [37]. In our experiments, we set this value to 0.25.

Population size, number of particles, flies, and Krills are unified and set to 50 in all the algorithms. The number of iteration is also unified and set to 500. The tuning parameters of GA, PSO, APSO and FA are presented in Tables 4, 5, 6 and 3, respectively.

The domain of the search space is also unified for all algorithms and set to $[-5,5]$. The parameters of APSO algorithm are tuned empirically as recommended by [42]. We used the Root Mean Squared Error (RMSE) as an objective evaluation criterion to evaluate the developed models. RMSE criteria is shown in Equation 9. All metaheuristic algorithms have experimented 25 independent times then the average of the evaluation measures was calculated.

Table 3: Firefly algorithm parameters settings

Parameter	Value
Alpha	0.4
Betamin	1
Gamma	0.4

Table 4: GA parameters settings

Parameter	Value
Selection method	Tournament selection
Crossover probability	80%
Mutation probability	5%

Table 5: PSO parameters settings

Parameter	Value
Acceleration constant	$[2.1,2.1]$
Inertia Weight	$[0.9,0.6]$
Maximum Velocity	100

7.2. Results

In this section, we apply a number of metaheuristic search algorithms to estimate the parameters of the winding machine described in Section 4. Six metaheuristic algorithms:

Table 6: APSO parameters settings

Parameter	Value
Randomness amplitude of particles	0.2
Speed of convergence	0.5
Randomness reduction control	0.95

GA, PSO, APCO, KH, CS, and FA were used. To develop our experiments we used 1000 measurements that were split into two sets each with 500 measurements for training and testing. This winding process inputs are presented in Figure 3. The goal of the implemented 25 experiments is to find the best parameter model that can minimize the error difference between the actual and the estimated response of the winding model.

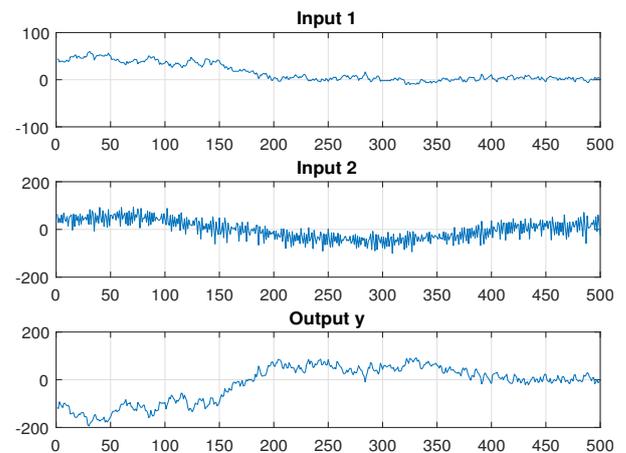


Figure 3: Inputs and output for the winding process

Carrying out the experiments described earlier, we obtain the average convergence and the best-so-far curves for the metaheuristic algorithms as shown in Figure 4, Figure 5 and Figure 6 for the Cuckoo, Firefly, GA, PSO, Krill Herd, and APSO, respectively. It can be noticed that GA is the fastest to converge while Firefly is the slowest to converge. The evaluation results for training and testing cases are shown in Tables 7 and 8, respectively. Based on these results, it can be noticed that Cuckoo search algorithm outperforms the other metaheuristic algorithms in optimizing the ARMA model of the winding machine considering all the evaluation metrics. On the other side, Firefly algorithm showed the worst results with a relatively big difference from the second worst algorithm which is APSO. The algorithms from the best to worst are ordered as follows: Cuckoo, Firefly, GA, PSO, Krill Herd, and APSO.

For further analysis, we show the boxplots in Figure 7 for the results of various metaheuristic algorithms calculated based on the RMSE results. As it can be seen in the figure, Cuckoo search algorithm shows the minimum median value and the most compact box which indicates the stability of the results of this algorithm. On the contrary, Firefly algorithm has a broader box which means that the results are more spread from the median.

In summary, Cuckoo search as a metaheuristic optimization algorithm over-performs the other algorithms regarding higher estimation accuracy for the ARMA based models of the winding machine studied in work. Moreover, Cuckoo search has only one parameter which is the P_a fraction that needs to be tuned. Thus, Cuckoo search has an advantage over other the algorithms which are highly influenced by the tuning settings of their parameters.

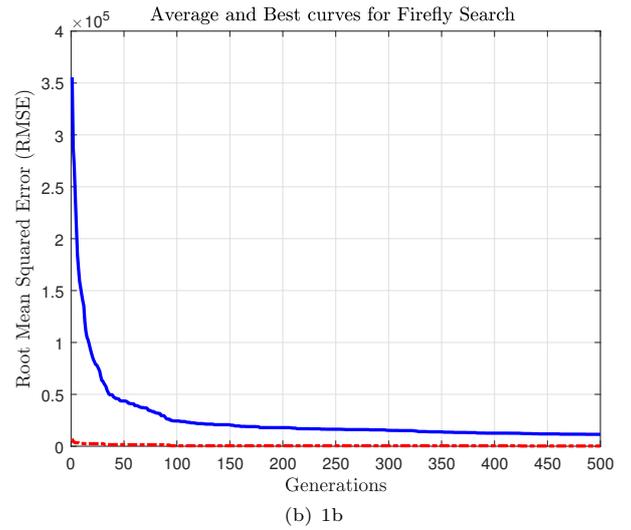
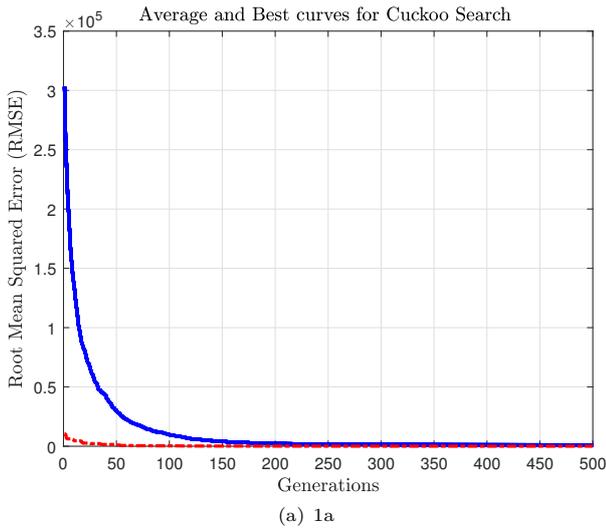


Figure 4: Convergence curves (a) Cuckoo (b) Firefly

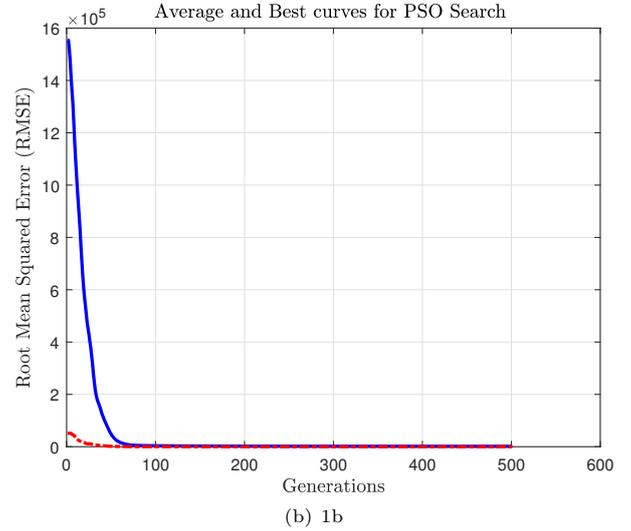
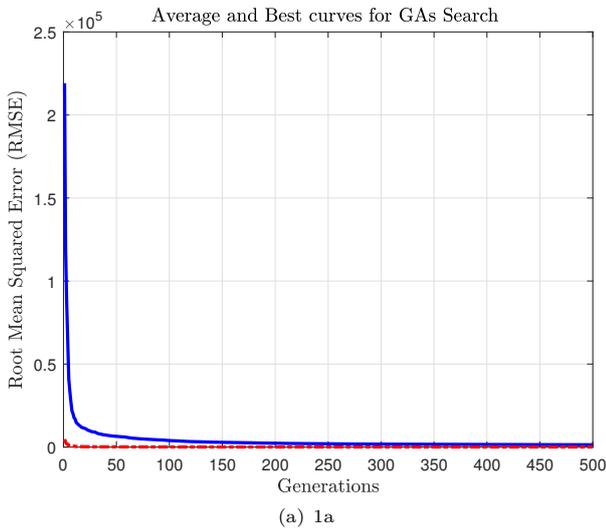


Figure 5: Convergence curves (a) GAs (b) PSO

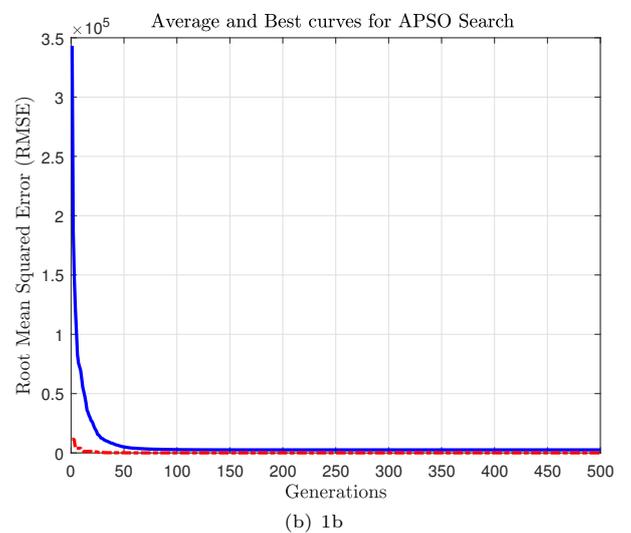
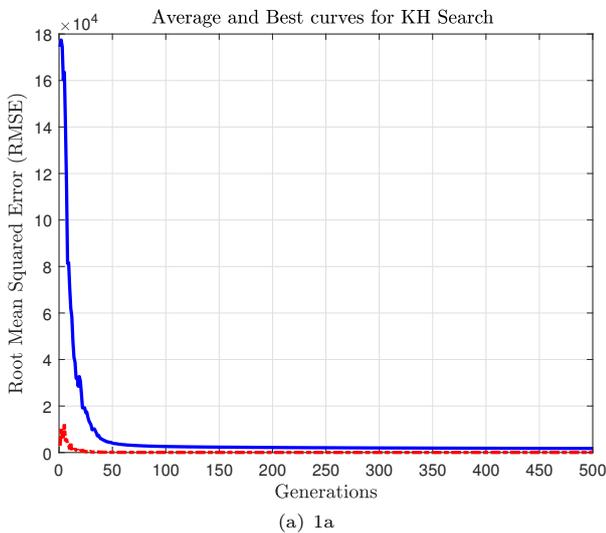


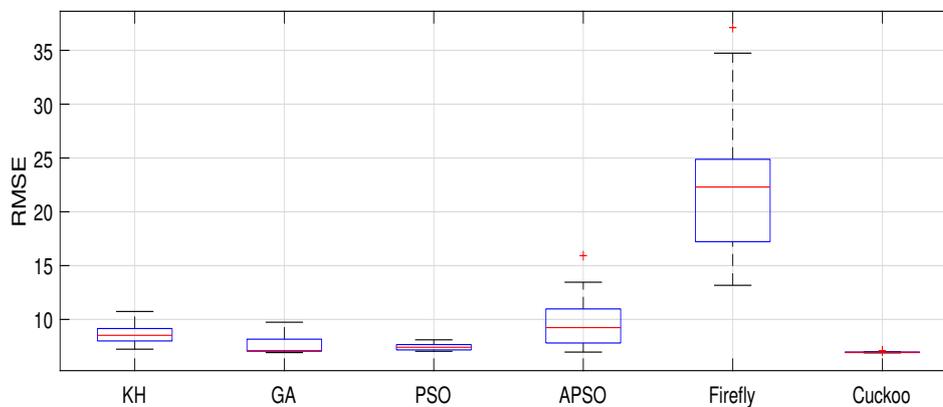
Figure 6: Convergence curves (a) KH (b) APSO

Table 7: Evaluation Metrics for Training Case

	VAF	MAE	MMRE	RMSE	R^2	Rank
KH	98.77%	6.45	0.030	8.30	0.9876	4
GA	99.04%	5.68	0.006	7.31	0.9904	3
PSO	99.08%	5.64	0.013	7.20	0.9908	2
APSO	98.24%	7.28	0.029	9.67	0.9824	5
Firefly	92.88%	16.14	0.007	20.59	0.9219	6
Cuckoo	99.20%	5.22	0.008	6.72	0.9920	1

Table 8: Evaluation Metrics for Testing Case

	VAF	MAE	MMRE	RMSE	R^2	Rank
KH	92.84%	6.85	0.125	8.68	0.9280	4
GA	94.53%	5.77	0.117	7.57	0.9452	3
PSO	94.74%	5.71	0.113	7.47	0.9472	2
APSO	90.88%	7.50	0.131	9.58	0.9086	5
Firefly	52.47%	18.05	0.405	22.30	0.4941	6
Cuckoo	95.43%	5.24	0.104	6.95	0.9543	1

**Figure 7:** Boxplots for the results of the metaheuristic algorithms based on the RMSE results

8. Conclusions

In this paper, we present the initial idea of tuning the parameter of an ARMA model using different nature-inspired meta-heuristic algorithms for an industrial winding machine. Data sets generated from an actual industrial process was used throughout the experiments. Experimental results are presented and discussed. The results show that some meta-heuristic optimization algorithms like Cuckoo search and Particle Swarm Optimization can successfully be used to tune the ARMA model parameters with minimum modeling error. However, unlike the other algorithms, Cuckoo search has the advantage of being a non-parametric algorithm which doesn't need any additional effort for parameters tuning.

Acknowledgement

The dataset explored in this article was composed as part of a project managed by Dr. M. Kamel at the Aluminum factory in Egypt. This data was analyzed by Dr. Aboabbas Hussein while doing his M.Sc. thesis.

References

- [1] M. H. Ali, N. H. El-Amary, H. E. A. Ibrahim, and R. I. Mashaly, "An upgraded soft winding machines based on advanced yarn tension modelling for using cotton," in *Proceedings of the 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, Feb 2016, pp. 17–21.
- [2] Y. Eddoukali, E. E. Adel, A. Benzaouia, and M. Ouladsine, "Fault detection for hybrid dynamical system application to winding machine," in *Proceedings of the 2016 5th International Conference on Systems and Control (ICSC)*, May 2016, pp. 394–399.
- [3] W. Mo, M. Wang, J.-S. Lin, H. Zan, and G. Sun, "Control system based on plc for winding machine," in *Proceedings of the 2014 International Symposium on Computer, Consumer and Control (IS3C)*, June 2014, pp. 74–77.
- [4] W. Jian, Z. Xu, W. Minghong, and D. Dehong, "Research on spindle drive control system of high speed winding machine," in *Proceedings of the 8th International Conference on Electronic Measurement and Instruments (ICEMI'07)*, Aug 2007, pp. 473–477.
- [5] H. Abbasi Nozari, H. Dehghan Banadaki, M. Mokhtare, and S. Hekmati Vahed, "Intelligent non-linear modelling of an industrial winding process using recurrent local linear neuro-fuzzy networks," *Journal of Zhejiang University - Science C*, vol. 13, no. 6, pp. 403–412, Jun. 2012.
- [6] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification*. Cambridge MA: MIT Press, 1983.
- [7] L. Ljung, *System Identification Theory for the User*. Prentice Hall, 1987.
- [8] N. Borchardt, R. Hinzelmann, D. S. Pucula, W. Heinemann, and R. Kasper, "Winding machine for automated production of an innovative air-gap winding for lightweight electric machines," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1509–1517, June 2016.
- [9] R. Johansson, *System Modeling and Identification*. NJ: Prentice Hall, 1993.
- [10] A. Isidori, *Nonlinear Control Systems*. Germany: Springer-Verlage, 1995.
- [11] W. M. Aly and A. Sheta, "Parameter estimation of nonlinear systems using levy flight cuckoo search," in *Research and Development in Intelligent Systems XXX*, M. Bramer and M. Petridis, Eds. Cham: Springer International Publishing, 2013, pp. 443–449.
- [12] A. Gelb, *Applied Optimal Estimation*. Cambridge: MIT Press, 1974.
- [13] M. Tummala, "Efficient iterative methods for FIR least squares identification," *IEEE Transaction Acoust., Speech, Signal Processing*, vol. 38, no. 5, pp. 887–890, 1990.
- [14] S. Nam, S. Kim, and E. Powers, "On the identification of a third-order Volterra nonlinear system using a frequency domain block RLS algorithm," in *Proceedings of the IEEE Transaction Acoust., Speech, Signal Processing*, 1990, pp. 2407–2410.
- [15] H. Hailiang, W. Zhong, N. Xiaohong, and S. Jing, "Robust decentralized control of web-winding systems without tension sensor," in *Proceedings of the 2015 34th Chinese Control Conference (CCC)*, July 2015, pp. 8850–8854.
- [16] M. H. Ali, H. E. A. Ibrahim, N. H. El-Amary, and R. I. Mashaly, "An upgraded soft winding machines based on advanced yarn tension modelling," in *Proceedings of the 2015 IEEE International Conference on Applied Superconductivity and Electromagnetic Devices (ASEMD)*, Nov 2015, pp. 571–572.
- [17] A. Hussian, A. Sheta, M. Kamel, M. Telbany, and A. Abdelwahab, "Modeling of a winding machine using genetic programming," in *Proceedings of the Congress on Evolutionary Computation (CEC2000)*, vol. 82, 2000, pp. 398–402.
- [18] A. Hussian, A. Sheta, and A. Abdelwahab, "Modeling of a winding machine using non-parametric neural network model," in *WSEAS International Conference on Scientific Computation and Soft Computing*, Athens, Greece, 2001, pp. 528–553.
- [19] A. Rodan, A. F. Sheta, and H. Faris, "Bidirectional reservoir networks trained using SVM+ privileged information for manufacturing process modeling," *Soft Computing*, pp. 1–14, 2016.
- [20] H. Faris, I. Aljarah, and S. Mirjalili, "Improved monarch butterfly optimization for unconstrained global search and neural network training," *Applied Intelligence*, vol. 48, no. 2, pp. 445–464, 2018.
- [21] I. Aljarah, H. Faris, S. Mirjalili, and N. Al-Madi, "Training radial basis function networks using biogeography-based optimizer," *Neural Computing and Applications*, vol. 29, no. 7, pp. 529–553, 2018.
- [22] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Computing*, vol. 22, no. 1, pp. 1–15, 2018.
- [23] H. Faris, I. Aljarah, and S. Mirjalili, "Evolving radial basis function networks using moth-flame optimizer," in *Handbook of Neural Computation*, 2017, pp. 537–550.
- [24] H. Faris, I. Aljarah, N. Al-Madi, and S. Mirjalili, "Optimizing the learning process of feedforward neural networks using lightning search algorithm," *International Journal on Artificial Intelligence Tools*, vol. 25, no. 06, p. 1650033, 2016.
- [25] N. Ghatasheh, H. Faris, I. Aljarah, and R. M. Al-Sayyed, "Optimizing software effort estimation models using firefly algorithm," *Journal of Software Engineering and Applications*, vol. 8, no. 03, p. 133, 2015.
- [26] B. L. Andersen, W. C. Page, and J. R. McDonnell, "Multi-output system identification using evolutionary programming," in *Proceedings of the 1991 Conference Record of the Twenty-Fifth Asilomar Conference on Signals, Systems and Computers*, Nov 1991, pp. 546–550.
- [27] R. R. Bishop and G. G. Richards, "Identifying induction machine parameters using a genetic optimization algorithm," in *Proceedings of the Southeastcon*, vol. 2, Apr 1990, pp. 476–479.
- [28] Y. Shi, W. Qian, Q. Wang, and K. He, "Aerodynamic parameter estimation using genetic algorithms," in *Proceedings of the 2006 IEEE International Conference on Evolutionary Computation (CECO)*, July 2006, pp. 629–633.
- [29] W. D. Chang, J. P. Cheng, M. C. Hsu, and L. C. Tsai, "Parameter identification of nonlinear systems using a particle swarm optimization approach," in *Proceedings of the 2012 Third International Conference on Networking and Computing (ICNC)*, Dec 2012, pp. 113–117.
- [30] Y. X. S., *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, USA, 2008.
- [31] L. Costa and P. Olivera, "Evolutionary algorithms approach to the solution of mixed integer non-linear programming problems," *Computers and Chemical Engineering*, vol. 25, pp. 257–266, 2001.
- [32] H. Faris and A. F. Sheta, "A comparison between parametric and non-parametric soft computing approaches to model the temperature of a metal cutting tool," *International Journal of Computer Integrated Manufacturing*, vol. 29, no. 1, pp. 64–75, 2016.
- [33] H. Faris, A. Sheta, and E. Öznergiz, "Modelling hot rolling manufacturing process using soft computing techniques," *International Journal of Computer Integrated Manufacturing*, vol. 26, no. 8, pp. 762–771, 2013.
- [34] J. Holland, "Genetic algorithms," *Scientific American*, pp. 66–72, 1992.
- [35] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA, USA: MIT Press, 1992.
- [36] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*. Piscataway, NJ, USA, 1995, pp. 1942–1948.

- [37] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Communications in Non-linear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831 – 4845, 2012.
- [38] X.-S. Yang and S. Deb, "Cuckoo search via lévy flights," in *NaBIC*, 2009, pp. 210–214.
- [39] X. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010.
- [40] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Proceedings of the 5th International Conference on Stochastic Algorithms: Foundations and Applications*, ser. SAGA'09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 169–178.
- [41] R. Isermann, K. Lachmann, and D. Matko, *Adaptive Control Systems*. New York: Prentice Hall, 1992.
- [42] X.-S. Yang, S. Deb, and S. Fong, "Accelerated particle swarm optimization and support vector machine for business optimization and applications," in *Networked Digital Technologies*, ser. Communications in Computer and Information Science, S. Fong, Ed. Springer Berlin Heidelberg, 2011, vol. 136, pp. 53–66.