

**MIMO HAMMERSTEIN MODEL
IDENTIFICATION BASED ON PSO GSA**

BY

AMRO F. ABDOUN

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

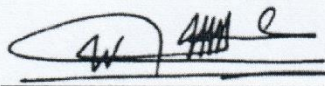
MAY 2021

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **AMRO FAISAL ABDOUN** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**.



Dr. Abdallah S. Al-Ahmari

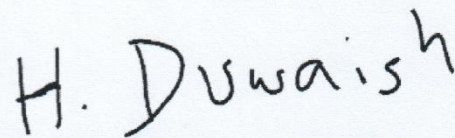
Department Chairman



Dr. Suliman Saleh Al-Homidan

Dean of Graduate Studies

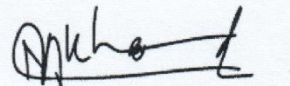
Date



Dr. Hussain N. Al-Duwaish
(Advisor)



Dr. Salim Ibrir
(Member)



Dr. Muhammad Khalid
(Member)



© Amro F. Abdoun

2021

[This thesis is dedicated to my Mother and Father]

ACKNOWLEDGMENTS

First and foremost I would like to express my sincere gratitude and thanks to my thesis advisor Dr. Hussain N. Al-Duwaish for his invaluable advice, guidance and patience throughout this study.

Furthermore, I would like to extend my sincere thanks to the thesis committee members Prof. Salim Ibrir and Dr. Muhammad Khalid for their cooperation, support and understanding.

Many thanks also to the Electrical Engineering department family for their help and support.

My special thanks to my Mother, Father, family and friends for their unconditional support and encouragement.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	V
TABLE OF CONTENTS	VI
LIST OF FIGURES	IX
LIST OF ABBREVIATIONS	XIII
ABSTRACT	XIV
ملخص الرسالة	XV
1 CHAPTER 1 INTRODUCTION	1
1.1 Thesis Overview	1
1.2 Thesis Objective	2
1.3 Thesis Contribution	3
1.4 Thesis Organization	4
2 CHAPTER 2 LITERATURE REVIEW	6
2.1 System Identification	6
2.2 Block-Oriented Nonlinear Systems	10
2.3 Hammerstein Model	13
3 CHAPTER 3 THE HAMMERSTEIN SYSTEM STRUCTURE	25
3.1 The Hammerstein Model Representation	25
3.2 Static Nonlinearity and Neural Network	26
3.2.1 The Perceptron	27

3.2.2 Multilayer Perceptron (MLP)	28
3.2.3 Training of Artificial Neural Networks	29
3.3 Linear Dynamic and State Space	31
3.4 Controllable Canonical Form	33
3.4.1 SISO Controllable Canonical Form	33
3.4.2 MIMO Controllable Canonical Form	34
4 CHAPTER 4 PROPOSED IDENTIFICATION ALGORITHM	36
4.1 Heuristic Optimization Algorithms	36
4.2 Particle Swarm Optimization (PSO)	38
4.3 Gravitational Search Algorithm (GSA)	42
4.4 A hybrid algorithm between PSO and GSA (PSOGSA)	46
4.5 Fitness Function	49
5 CHAPTER 5 SIMULATION RESULTS	51
5.1 Simulation results for SISO models	51
5.1.1 Case study 1: saturation nonlinearity	51
5.1.2 Case 2: Coupled electrical drives	57
5.2 Simulation results for MIMO models:	65
5.2.1 Case study 3: Dead zone and Relay nonlinearities	65
5.2.2 Case study 4: Investigate the effect of noise	73
5.2.3 Case 5: MIMO Industrial dryer	80
6 CHAPTER 6 CONCLUSION AND FUTURE WORK	97
6.1 Conclusion	97

6.2 Future Work	99
REFERENCES.....	100
VITAE	115

LIST OF FIGURES

Figure 2- 1 Modeling and system identification flowchart.....	8
Figure 2- 2 Hammerstein model.	12
Figure 2- 3 Wiener model.	13
Figure 2- 4 Wiener-Hammerstein model.	13
Figure 3- 1 Proposed Structure of Hammerstein model.	26
Figure 3- 2 Schematic of the perceptron.....	25
Figure 3- 3 Multilayer perceptron with one hidden layer.	29
Figure 4- 1 Flowchart of the PSO algorithm.	41
Figure 4- 2 Flowchart of the GSA algorithm.....	45
Figure 4- 3 Flowchart of the PSOGSA algorithm.	48
Figure 5- 1 Saturation Nonlinearity with identified nonlinearity Using PSO.	53
Figure 5- 2 Convergence Curve of PSO algorithm.....	54
Figure 5- 3 Saturation Nonlinearity with identified nonlinearity Using PSOGSA.	55
Figure 5- 4 Convergence Curve of the PSOGSA algorithm.....	56
Figure 5- 5 Schematic for coupled electric drive.....	58
Figure 5- 6 Uniformly distributed input signal	59
Figure 5- 7 Measured output signal	59
Figure 5- 8 Convergence curve of the PSO algorithm.....	61
Figure 5- 9 Actual output versus estimated output based on PSO algorithm.	61
Figure 5- 10 Learnt nonlinearity of the neural network based on PSO.	62
Figure 5- 11 Convergence curve of the PSOGSA algorithm.....	63

Figure 5- 12 Actual output versus estimated output based on PSOGSA algorithm.	64
Figure 5- 13 Learnt nonlinearity of the neural network based on PSOGSA.	64
Figure 5- 14 Convergence Curve of the PSO algorithm.....	68
Figure 5- 15 Dead zone Nonlinearity with identified nonlinearity Using PSO.....	68
Figure 5- 16 Comparison between Actual and estimated identified nonlinear function... Using PSO.....	69
Figure 5- 17 Convergence Curve of the PSOGSA algorithm.....	71
Figure 5- 18 Dead zone Nonlinearity with identified nonlinearity Using PSOGSA.....	71
Figure 5- 19 Comparison between Actual and estimated identified nonlinear function ... Using PSOGSA.....	72
Figure 5- 20 Convergence Curve of the PSOGSA algorithm with SNR=30.....	75
Figure 5- 21 Dead zone Nonlinearity with identified nonlinearity Using PSOGSA with... SNR=30.	75
Figure 5- 22 Comparison between Actual and estimated identified nonlinear function... using PSOGSA SNR=30.	76
Figure 5- 23 Convergence Curve of the PSOGSA algorithm with SNR=20.....	78
Figure 5-24 Dead zone Nonlinearity with identified nonlinearity Using PSOGSA... with SNR=20.	78
Figure 5- 25 Comparison between Actual and estimated identified nonlinear function... Using PSOGSA SNR=20.....	79
Figure 5- 26 Industrial dryer process.	82
Figure 5- 27 Inputs of the Industrial Dryer.....	83

Figure 5- 28 Outputs of the Industrial Dryer.	84
Figure 5- 29 Comparison between estimated and actual output (y1) of the industrial dryer... identified by PSO.	87
Figure 5- 30 Captured nonlinearity for output (y1) of the industrial dryer identified by... PSO.	87
Figure 5- 31 Comparison between estimated and actual output (y2) of the industrial dryer... identified by PSO.	88
Figure 5- 32 Captured nonlinearity for output (y2) of the industrial dryer identified by... PSO.	88
Figure 5- 33 Comparison between estimated and actual output (y3) of the industrial dryer... identified by PSO.	89
Figure 5- 34 Captured nonlinearity for output (y3) of the industrial dryer identified by... PSO.	89
Figure 5- 35 Convergence curve of the Industrial Dryer model identified based on PSO.	90
Figure 5- 36 Comparison between estimated and actual output (y1) of the industrial dryer... identified by the PSOGSA algorithm.	93
Figure 5- 37 Captured nonlinearity for output (y1) of the industrial dryer identified by the... PSOGSA algorithm.	93
Figure 5- 38 Comparison between estimated and actual output (y2) of the industrial dryer... identified by the PSOGSA algorithm.	94

Figure 5- 39 Captured nonlinearity for output (y2) of the industrial dryer identified by the... PSOGSA algorithm.....	94
Figure 5- 40 Comparison between estimated and actual output (y3) of the industrial dryer... identified by the PSOGSA algorithm.	95
Figure 5- 41 Captured nonlinearity for output (y3) of the industrial dryer identified by the... PSOGSA algorithm.....	95
Figure 5- 42 Convergence curve of the Industrial Dryer model identified based on the... PSOGSA algorithm.....	96

LIST OF ABBREVIATIONS

SISO	:	Single-Input Single-Output
MIMO	:	Multi-Input Multi-Output
PSO	:	Particle Swarm Optimization
GSA	:	Gravitational Search Algorithm
PSOGSA	:	A hybrid algorithm between PSO and GSA
MSE	:	Mean Squared Error
ARX	:	Autoregressive with eXogenous Input
SNR	:	Signal to Noise Ratio
MLP	:	Multi-Layer Perceptron
GA	:	Genetic Algorithm
DE	:	Differential Evolution

ABSTRACT

Full Name : Amro Faisal Abdoun Mohammed

Thesis Title MIMO Hammerstein model identification based on PSOGSA

Major Field : Electrical Engineering

Date of Degree : April 2021

The Hammerstein model is among one of the block-oriented models used for the purpose of system identification. Unique to this type of model is its construction of cascading a static nonlinear component with a dynamic linear component. This study aims to introduce a new and nascent method for MIMO Hammerstein model identification. A multilayer feedforward neural network represents the static nonlinear part of the Hammerstein model – and for the dynamic linear part, state-space in the controllable canonical form is used.

In order to sufficiently estimate the parameters of both the static nonlinear part and the dynamic linear part, heuristic optimization algorithms (PSO and PSOGSA) are applied. Notably, a variety of simulation examples have been performed to substantiate the efficiency and robustness of the proposed algorithm.

The influence and impact of white noise on the proposed identification model is also studied. In this particular case, white noise (comprised of Gaussian noise) with varying signal to noise ratios is applied for the proposed model. Moreover, practical SISO and MIMO systems are identified using the proposed Hammerstein model. These practical simulation examples prove the applicability of the introduced algorithms in solving real-life problems.

ملخص الرسالة

الاسم الكامل: عمرو فيصل عبدون محمد

عنوان الرسالة: تعريف نموذج هامرشتاين ذو النظام النموذج متعدد المدخلات والمخرجات بالاعتماد على الخوارزميات الإرشادية.

التخصص: الهندسة الكهربائية.

تاريخ الدرجة العلمية: ابريل 2021 م

يعد نموذج هامرشتاين (Hammerstein model) أحد النماذج المعتمدة على الكتل (Block-oriented) المستخدمة لغرض تحديد النظام. ينفرد هذا النوع من النماذج ببنائه المتتالي الذي يتكون من جزء ثابت غير خطي متصل بجزء خطي ديناميكي. تهدف هذه الدراسة الى تقديم طريقة جديدة لتعريف نموذج هامرشتاين ذو النظام متعدد المدخلات والمخرجات. تم اختيار الشبكة العصبية متعددة الطبقات ذات التغذية او التوصيل الامامي لتمثيل الجزء الثابت غير الخطي من نموذج هامرشتاين - وبالنسبة للجزء الخطي الديناميكي (المتحرك) ، تم استخدام نموذج الحالة والفراغ (state-space) في شكل القابل للتحكم (controllable canonical form) .

من أجل تقدير عوامل كل من الجزء غير الخطي الثابت والجزء الخطي الديناميكي بشكل مرضي، تم تطبيق خوارزميات التحسين الإرشادية (PSO و PSOGSA). والجدير بالذكر أنه تم إجراء مجموعة متنوعة من أمثلة المحاكاة لإثبات كفاءة ومتانة الخوارزمية المقترحة.

كما تمت دراسة تأثير الضوضاء البيضاء على نموذج التعريف المقترح. في هذه الحالة بالذات ، تم تطبيق الضوضاء البيضاء (المكونة من ضوضاء غاوسين) مع نسبة إشارة إلى الضوضاء (signal to noise) متغيرة على النموذج المقترح. علاوة على ذلك ، تم تعريف أنظمة عملية احادية المدخلات و المخرجات (SISO) و متعددة المدخلات و المخرجات (MIMO) باستخدام نموذج هامرشتاين المقترح. تثبت أمثلة المحاكاة العملية هذه قابلية تطبيق الخوارزميات المقدمة في حل مشاكل الحياة الواقعية.

CHAPTER 1

INTRODUCTION

1.1 Thesis Overview

Mathematical models are the cornerstone for control systems, with regard to their triad of design, implementation, and operations. Models are utilized in particular for advanced subjects, such as controllers design, supervision, optimization, and fault detection. These models could be obtained through theoretical modelling if the physical laws are known. However, in cases where some or all these laws are unknown, experimental modelling should be performed, with system identification in particular being the preferable model. System identification is the experimental method used in determining models by fitting experimental data to a model configuration that is suitable for the given data.

Block-oriented models are one of the most well-known methods used for the purpose of system identification. Block-oriented models are defined as the interconnection between linear dynamics and static nonlinearities. The Hammerstein method in particular is the most customary and commonly applied block-oriented model structure.

Constructed by cascading a static nonlinearity to linear dynamics, Hammerstein system is effective and efficient in its ability to represent a broad variety of nonlinear models. In addition to this, the proposed algorithm structure for the Hammerstein model is computationally fast and can be fine-tuned with relative ease to work on a multitude of structures. This thesis, then, is finely focused on the application of Hammerstein system identification.

1.2 Thesis Objective

The thesis' utmost and primary goal is to develop a new method for identifying Hammerstein models as well as demonstrating the efficiency of proposed algorithm. The Hammerstein model is commonly considered a flexible description of the nonlinear systems. Structure of Hammerstein model is represented by the complete separation of nonlinearity of the identified model, from its linear part.

The proposed algorithm will be applied to both the SISO and MIMO systems. Nevertheless, an extension of SISO system to MIMO system is not a straightforward process due to the plethora of parameters and coupled variables on the MIMO system that require studying.

The robustness of the introduced Hammerstein model will be ascertained through studying the effect of the white noise. In addition to this, data from the real system will be applied to measure the efficiency of the proposed algorithm.

1.3 Thesis Contribution

In accordance with the identification of Hammerstein system, several algorithms were employed from the literature. These methods have two main differences, namely the method for determining the linear and nonlinear part's parameters, and the representation of static nonlinear function.

To describe the static nonlinear part of this study, feedforward neural network is utilized – and state-space system is used for dynamic linear part. This will result neatly in both the multilayer perceptron (MLP) weights and biases, as well as state-space model parameters, being identified. Heuristic optimization algorithms, then, are used to solve this identification problem.

For this study, the Hammerstein model identification problem is modelled as an optimization problem, allowing a diverse range of optimization algorithms such as Particle Swarm Optimization (PSO), Gravitational Search Algorithm (GSA), and hybrid algorithms (PSOGSA) to be used.

To estimate weights and biases of the neural network, as well as identify the parameters of the state space matrices, the PSO and Hybrid algorithm (PSOGSA) will be applied.

Core contributions of the thesis include:

- 1- Identification of SISO Hammerstein models using PSO and PSOGSA.
- 2- Identification of MIMO Hammerstein models using PSO and PSOGSA.

- 3- Controllable canonical state-space form, on the linear component of Hammerstein model, used to minimize the number of parameters to be estimated.
- 4- Testing the validity of proposed algorithms on different data.
- 5- Simulation of SISO and MIMO models to known nonlinearity.
- 6- Applying the proposed algorithm of the SISO model to the nonlinear coupled electrical drive dataset.
- 7- Evaluating the robustness of a MIMO system with known nonlinearity against white noise at a various signal to noise ratios (SNR).
- 8- Testing the proposed algorithm of MIMO model on a dataset that extracted from real nonlinear industrial dryer system, with three input and three outputs.
- 9- Drawing a conclusion and recommendation regarding the most effective optimization approach for each testing data that should be used for the identified Hammerstein model.

1.4 Thesis Organization

The following delineates the structure of this thesis: Across a range of strategic pillars in 2, through describing the literature review of the proposed algorithms – general description of system identification and its types, information about the block-oriented system identification, and a survey on Hammerstein model identification methods has been provided.

The Hammerstein model's structure is elaborated upon in Chapter 3. Details about linear and nonlinear components of Hammerstein system is subsequently presented in this chapter.

Chapter 4 entails details about Hammerstein model identification, for which algorithms are given. This chapter necessarily covers the entirety of the algorithm's steps and procedures.

The proposed system's simulation findings are presented in Chapter 5. Examples of SISO and MIMO simulations are expanded upon and compared.

In 6, a detailed conclusion about the research is given, in conjunction with possible proposed improvements to it.

CHAPTER 2

LITERATURE REVIEW

2.1 System Identification

Modelling a real system is beneficial for obtaining a better analysis and thus understanding of the system. Indeed, these models could be useful for a multitude of practical applications, including the prediction and simulation of a system, the design of new processes, the design of controllers, adaptive control algorithms, supervision, fault diagnosis, optimization and other applications [1].

In order to obtain a model or mathematical description for a given system, there are two approaches, namely the theoretical approach and the empirical approach. The theoretical method is fundamentally based on laws of matter, energy or thermodynamics. The empirical method, otherwise known as system identification depends on analysis of the experimental and operating data. System identification approach is widely used due to a combination of its broader practicability and the complexity of most of its processes, which are too complex to comprehend based on physical laws [2].

System identification is widely known as the experimental approach of modelling which is conducted by optimally fitting the experimental data to a suitable model structure [3].

This approach has been initiated from developments in mathematics and probability theory since the eighteenth and nineteenth-century, with examples including the conditional probability (Bayes, 1763), the least square (LS) method (Gauss and Legendre, 1809), Fourier transforms (Fourier, 1822) among others [4]. A pivotal breakthrough in the theory and practice for the field of system identification occurred through the introduction of the prediction error method (Ljung, 1976,1978) due to changes in the problem formulation stemming from the search of a true model, ultimately extending to a wider and more practical search for the best approximate models. With this necessary context, prediction error minimization methods, then, are considered the heart of many system identification algorithms.

The primary objective of system identification is model development. An identification procedure should be adhered to, ensuring the identified model carries the desired degree of both accuracy and precision. Figure 2-1 represents the procedures and algorithms that should be followed for the purposes of modelling and identification.

Modelling engenders an enormous diversity of applications, such as control, signal processing, simulation, prediction and others. Therefore, it is an essential step in determining the purpose of the model at the beginning of the identification. Periodically, physical laws could be used to derive models in cases where some parameters are known and others are unknown, and it is probable to perform model estimation using the known parameters.

This structure is referred to in the nomenclature as grey-box identification, unless part of the system is unknown in which case it is aptly named black-box identification. The next step in identification is data acquisition through experimental design, which when resulted from input and output data, will be tested before proceeding to the model structure selection.

After choosing the structure of the model, estimation of its parameters should be conducted using different methodologies such as prediction-error minimization, instrumental variable, and other methods.

The last step in the cumbersome system identification procedure is model validation. Many model validation and assessment tests are available to define quality of the identified model. Examples of these tests include model fit analysis, analysis of estimation error, analysis of prediction error (residual), cross-validation and other validation tests [2].

System identification modelling techniques are generally divided into two types: linear and nonlinear identification. Indeed, the selection of this technique is regarded as a trade-off

between the precision and complexity of the system. Linear system identification undertakes input-output data as linear equations and expresses the nonlinearities as uncertainties [5],[6].

Irrespective of this, linear identification could be applied when the model remains in proximity with a nominal operating point which keeps linearity valid [7]. Numerous real systems have been identified using linear representation, as demonstrated in [8],[9],[10]. With the inclusion of many operation points, the linear identification will not be accepted, and the nonlinear model should be used for the purpose of handling the nonlinearities of the system.

2.2 Block-Oriented Nonlinear Systems

Nonlinearity comprises the common nature of systems, whereas linearity is treated as an exceptional case. Many nonlinear practical systems are expounded in the relevant literature for instance where aircraft, automotive industry, mechatronics systems, civil engineering, and others are concerned. A survey regarding the behavior of nonlinear systems in real-life applications can be found in the following source [11]. There is yet to exist a single nonlinear system identification technique that possesses the capability to model all systems in across a range of different circumstances. Moreover, many identification algorithms that

have the capability of deal with low dimensional systems, are inapplicable when the factor of dimension increases. According to [12], nonlinear models have been exceedingly utilized due to:

- (1) Progress the nonlinear systems theory, resulting in increasing the methodologies of modelling those systems.
- (2) Improvements in the nonlinear system identification algorithms.
- (3) Developments in the hardware and software of the control.

Many nonlinear systems are introduced in the literature for the sole function of performing the identification of nonlinear models, notable examples include Nonlinear Autoregressive with eXogenous Inputs (NARX) [13], [14], Volterra [15], multi-model representations [16], neuro-fuzzy models [17], block-oriented models [7], [18] among others.

The notion of the block-oriented systems is the connection of the static nonlinearities with the linear dynamics. Where the linear component can be parametric (FIR, IIR, State Space, Transfer Function and other), or nonparametric like impulse and frequency responses. This concept, originally proposed by Billings and Fakhouri in the 1980's [19], was introduced as a viable alternative to the Volterra series. This method of identification has been widely used due to the requirement for much less complex design features while maintaining satisfactory results for real systems approximation.

Perhaps the most well-known block-oriented nonlinear system is the Hammerstein model. This model, first introduced by the mathematician A. Hammerstein in 1930 [20], is

comprised of a static nonlinear element connected in cascade with linear dynamic elements – illustrated in Figure 2-2.

Among the popular structures of the block-oriented systems, include the Wiener system [21], Wiener-Hammerstein system, Hammerstein-Wiener system, additive NARMAX system, arbitrary structure systems and others. The structures of Wiener and Wiener-Hammerstein models are depicted in both Figure 2-3 and Figure 2-4 respectively.

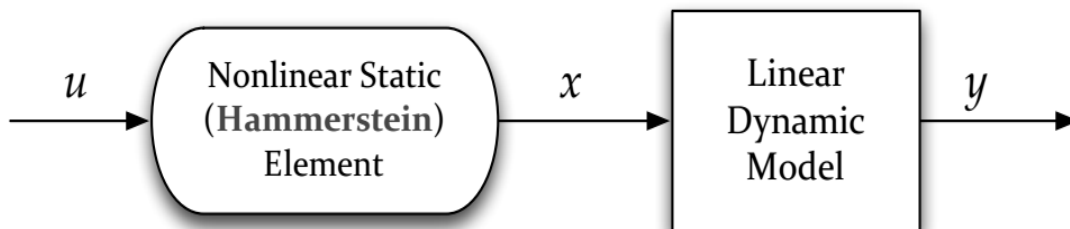


Figure 2- 2 Hammerstein model.

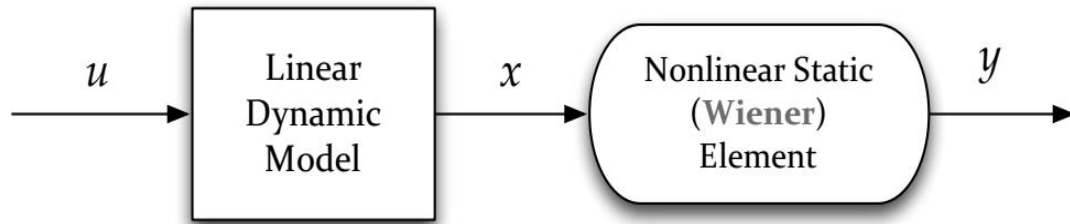


Figure 2- 3 Wiener model.

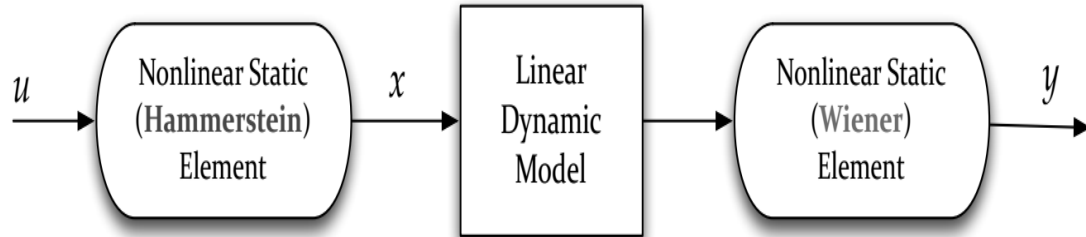


Figure 2- 4 Wiener-Hammerstein model.

2.3 Hammerstein Model

Hammerstein system, represented in Figure 2-2, is constructed by cascading a static nonlinear element in series with a linear dynamic part. This model in particular was applied to a vast number of nonlinear systems such as solid oxide fuel cells [22], heat exchangers [23], valve actuators [24], power amplifiers [25], wind turbines [26], electrical drives [27],

dryer process [28], signal processing [29], [30], distillation column [31], physiological systems [32], continuous stirred tank reactor [33], electrically stimulated muscles [34], financial volatility process [35] and many others.

Hammerstein model identification remains an active research area, in which a range of diverse techniques and algorithms have been tested and implemented over its developmental years. In 1966, an iterative approach for Hammerstein system identification was introduced by Narendra and Gallman [36]. This methodology has since been adopted by another author, Ding et al in [37] who used a stochastic gradient algorithm for Hammerstein model identification. Additionally, J Vörös [38] applied an iterative algorithm for Hammerstein system identification with two-segment nonlinearity.

Moreover, Q Shen and F Ding [39] proposed an iterative algorithm founded on gradient descent for the modelling of the Hammerstein system. Subsequently, Ding et al. used the hierarchical identification algorithm for the gradient as well as Least-Squares-Based iterative algorithm of the Hammerstein model [40].

In order to increase the convergence rate of the stochastic gradient process, a multi-innovation identification algorithm is proposed. Hu et al. [41] tested the multi-innovation stochastic gradient identification technique of the Hammerstein autoregressive model, based on the separation theory of key-term. Moreover, a similar model was applied, but in this instance was established on the filtering technique [42].

Algorithms used in [41], [42] assume that the static nonlinearities exist in a polynomial form. J. Vörös studied a switching function for Hammerstein model identification with the backlash nonlinearities [43] as well as the multi-segment piecewise linear characteristics algorithm [44].

Identification based on the Hammerstein model through the use of multi-layer perceptron was studied in 1992 by Back & Tsoi [45]. Following its successful completion, Al-Duwaish et al. proposed the SISO Hammerstein model identification through the implementation of a feedforward neural network [46]. In addition to this, Al-Duwaish and M. N. Karim extended their algorithm to study the MIMO Hammerstein model identification case in [47].

Hachino, Deguchi and Takata [48] succeeded in their implementation of the radial basis function and genetic algorithm (GA) for Hammerstein system identification.

Han and de Callafon [49] proposed a closed-loop Hammerstein system identification method, based on an iterative instrumental variable where static nonlinearities are explained as piecewise triangular function.

One of the most effective methods for Hammerstein model identification involves over-parametrization approaches. This method is introduced by Bai [50] where static nonlinearities are presented as a linear arrangement of a basis function, and linear dynamics are treated as an FIR (Finite Impulse Response). As an extension of this approach, T Falck et al. [51] applied the over-parametrization approach for the Hammerstein system under

regularization. Moreover, Risuleo et al. [52] utilized a kernel-based technique to implement the parametrization approach for Hammerstein system identification.

A blind identification algorithm has been applied to the Hammerstein model. This principle is commonly employed when the system input is unknown. In [53], blind maximum likelihood approximation is applied for the Hammerstein model identification. Additionally, Bai [54] was able to employ the blind approach for the identification of Hammerstein-Winer model.

The subspace approach was successfully operated for the identification of MIMO Hammerstein systems, for instance in the example of Verhaegen & Westwick [55] and D. and I Goethals et al. [56] where they extended the subspace algorithm by writing oblique projection as a set of regression Least Square support vector machine.

Support vector machine is yet another useful method that has been used for the modelling of the Hammerstein system. I. Goethals and K. Pelckmans succeeded in applying the least square support vector machine for identification of subspace Hammerstein system in addition to the MIMO Hammerstein model as shown in [56], [57] respectively.

Stochastic methods were also applied in the Hammerstein system identification. Krzyżak et al. [58] used a recursive kernel approach for Hammerstein model identification. Other notable stochastic algorithms are represented in [59] and [60].

The frequency-domain method is exceedingly used for Hammerstein model identification. This method relies primarily on both the input and output signals of frequency content. In this method, the spectrum of the output is considered as a function of magnitude frequency of the input, in addition to other parameters which could be estimated by implementing multiple sinusoidal input signal [61]. Many efforts have been expended to model the Hammerstein system using the frequency methods as demonstrated in [62], [63], [64], [65] and others.

It has become quite evident that the kernel-based mixed parametric and nonparametric algorithms for the Hammerstein model identification, have become increasingly popular over the last couple of years. Various viable strategies have been proposed in the literature, contingent on multiple factors including prior knowledge, the capacity of the data and the structure of the system. Estimation is conducted by including the parametric part into separate orthogonal, kernel or correlation nonparametric algorithm [18]. In 2012, a survey by Hasiewicz et al. [66] was constructed based on this algorithm.

In [67], a test signal has been utilized to decouple the linear dynamic element from the static nonlinear element of Hammerstein system.

Hammerstein model identification with FIR (Finite Impulse Response) is considered as the linear component of the system by Risuleo et al. (2015), using a linear combination of a basis-function to represent static nonlinearities. In this paper, this method is restricted to white noise and the static nonlinearity is explained by a well-known basis function that represents the model. Therefore, Risuleo et al. (2017) [61] extended their work to tackle

this limitation, firstly by considering a model with coloured noise, like ARX, ARMAX to represent the linear component. Secondly, they handled the static nonlinearity through the application of the Gaussian process.

Cui et al. [68] developed a new technique for Hammerstein model identification. They used a functional link artificial neural network to define the static nonlinearities and ARMA model, which is used to characterize the distinctly linear dynamics of the model.

Włodzimierz Greblicki and Mirosław Pawlak [69] recovered the nonlinearity of the Hammerstein model by applying the nearest neighbor nonparametric algorithm.

In 2006, W. Lin et al. [70] developed a method for the Hammerstein modelling which used bacterial foraging optimization. Outcomes of this algorithm provided better results than those of the Particle Swarm Optimization (PSO).

Proposed by Jia et al. is correlation analysis [71], [72] for the MIMO Hammerstein model identification. To tackle the MIMO Hammerstein system identification, a particular testing signal with an independent separable signal and randomly generated multiple-step signals are used.

In [73] Feng Li applied the same correlation method, but this time with coloured noise. Identification for static nonlinearities and dynamic linearities is conducted using specific signals like the Gaussian signal, binary signal, sine signal and random multiple-step input signal. Parameters of linear elements are estimated using correlation analysis. Following this analysis, the noise correlation function is determined by embedding a filter for the

purpose of extending the Hammerstein system. The static nonlinearity parameters, then, are computed through the recursive Least Square algorithm.

The Cuckoo search algorithm has been applied by Gotmare et al. [74] for Hammerstein model identification. Unique to this journal, was the authors' use of functional link neural networks in representing the static nonlinearities – and an adaptive Infinite Impulse Response filter was applied for the dynamic linearity. Following this, the model was subsequently trained using the cuckoo search stochastic algorithm. Results of this method were contrasted with other revolutionary computing algorithms like PSO and Differential Evolution (DE). The Cuckoo search technique engendered a superior result in direct comparison with other revolutionary computing algorithms.

P. Mattsson and T. Wigren [75] identified the Hammerstein model through the derivation the recursive prediction error. They analyzed the convergence property of this algorithm, proving that this methodology reaches a stable minimum point of the criteria. Moreover, the algorithm was evaluated on a cement mill model.

Y. Tang, Z.Li and X. Guan [76] developed an extreme learning machine based on neural network for the Hammerstein model identification. In this algorithm, the static nonlinearities are represented by the extreme learning machine algorithm. Lipschitz quotient algorithm was used to calculate the parameters of the extreme learning machine-based Hammerstein system.

In 2016, Wang, Feng Ding [77] identified a multivariable Hammerstein model based on Controlled Auto-Regressive Moving Average (CARMA). They transformed the multivariable Hammerstein CARMA-based model into a model of bilinear parameter vectors. Following this, they applied a hierarchical identification algorithm for the estimation of the two sets of vectors.

Chaudhary and Raja [78] developed an adaptive fractional least mean square algorithm for the model identification of the Hammerstein nonlinear Auto-Regressive Moving Average with eXogenous noise (ARMAX). In accordance with the parameter estimation results of this model, a comparison was drawn with the kernel-based least mean square and Volterra-based least mean square algorithms.

Nanda et al. [79] introduced a hybrid algorithm for Hammerstein system identification. They combined the application of Artificial Immune systems (AIS) and PSO to form Immunized PSO or IPSO, with the aim to improve the performance of the identification. Furthermore, they applied an additional algorithm called Clonal PSO or CPSO to model the Hammerstein system.

Xia Hong and Sheng Chen [80] proposed a new approach for the identification of the Hammerstein system. Where the nonlinear static element of it was concerned, it was modelled using the non-uniform rational B-spline neural network. This approach was applied in two phases: Phase one was performed using PSO algorithm in order to estimate shaping parameters; phase two was to identify the remaining parameters through singular value decomposition method.

In 2011, Al-Duwaish [81] used the PSO algorithm for the Hammerstein model identification with known nonlinearities.

Zhiyong Du and Xianfang Wang [82] presented a new PSO based technique for the Hammerstein system identification, by employing the Quantum Delta potential well based Particle Swarm Optimization abbreviated as QDPSO for modelling of Hammerstein system.

In 2013, Jianliang Sun and Xinggao Liu [83] established a novel approach for Hammerstein identification with the application of coloured noise. In their approach, the PSO algorithm was imposed into the maximum likelihood estimator to form Maximum Likelihood PSO (ML-PSO). Following this, they used the evolution state estimation technique to develop an adaptive algorithm, which, in the end, would form the Maximum Likelihood Adaptive Particle Swarm Optimization (ML-APSO). This ultimately resulted in them utilizing a simulation example to compare ML-PSO and ML-APSO.

In [84], the authors applied the key-term separation algorithm to transfer the identification problem into a nonlinear optimization problem. They subsequently used Improved Particle Swarm Optimization (IPSO) to identify three pivotal components: the linear element, nonlinear element and noise of the proposed Hammerstein model structure.

Jin Q et al. [85] proposed an identification method that would solve the MIMO Hammerstein process under the influence of the heavy-tailed noise problem. They introduced a Gaussian mixture distribution intelligent optimization algorithm (GMDA) and

converted the identification problem into an optimization problem. The nonlinear component of Hammerstein system used for their study was the radial basis function (RBF).

Another MIMO Hammerstein system identification algorithm has been introduced by Ricardo Castro-Garcia et al. [86]. For this, they incorporated the impulse response of the model to Least Square support vector machine (LS-SVM) where the LS-SVM has been applied to the whole model.

Dongqing Wang et al. [87] introduced a prudent methodology for investigating both the parameters and order of Hammerstein system. The orthogonal matching pursuit algorithm is used for the compressive sensor theory. Weighted parameter estimation algorithm of the Hammerstein nonlinear ARX system is introduced in [88]. The multi innovation stochastic gradient algorithm is proposed to enhance the convergence curve of the Hammerstein system identification problem.

In [89], a key term separation technique is applied for the parameter identification of Hammerstein autoregressive model. Hierarchical Least Squares algorithm is used to identify parameters of subsystems, where the system is converted into a combination of subsystems.

Presented in [90] is a novel technique for Hammerstein system identification with the absence of the output data. The expectation-maximization method is used in this paper to estimate systems with multiple vectors of the parameter.

The Hammerstein based output error identification model is applied to deal with load disturbance and input delay that can occur during the nonlinear system identification of industrial applications [91]. In this work, load disturbance response is considered a time-varying parameter – in which case two recursive Least Square algorithms are used to estimate both parameters of the Hammerstein system, as well as the time-varying disturbance. Moreover, the authors introduced two adaptive forgetting factors to recursive Least Square to improve the convergence of identified model.

A stochastic computational algorithm is introduced for the parameter estimation of the Hammerstein system [92]. Three applications across various areas, namely differential evolution, genetic algorithm and pattern search methods are used to tackle the estimation problem.

Jie Hou et al. [93] proposed a subspace identification algorithm to handle a Hammerstein nonlinear identification system that been exposed to a periodic disturbance with unidentified waveforms. Authors used the superposition principle to separate the deterministic system response from the periodic disturbance response. To ensure the consistent estimation of subspace matrices of both the linear subsystem and non-linear function parameters, a bias eliminated non-iterative approach is used. This method could also be applied with the arbitrary waveform of the harmonic or periodic disturbance.

A new method of evolutionary computational heuristics is given in [94]. A backtracking search algorithm, as well as DE and GA algorithms are used for the identification of the nonlinear Hammerstein controlled auto-regressive auto-regressive (NHCARAR) system. Various scenarios based on noise variance and degree of freedom have been applied.

CHAPTER 3

THE HAMMERSTEIN SYSTEM STRUCTURE

3.1 The Hammerstein Model Representation

The Hammerstein model is constructed with a nonlinear static element connected to the linear dynamic element. In the literature, many techniques have been utilized for the identification of this model. These methods have two main differences: The first involves the specific strategy used to estimate parameters of the linear and nonlinear component – and the second is representation method of the static nonlinear function.

In this research, the feedforward neural network is applied to represent the static nonlinear element. For the dynamic linear element, the state-space model is applied. Therefore, the weights and biases of the multilayer perceptron (MLP), as well as parameters of state-space system, require identification. In order to tackle this identification problem, heuristic optimization algorithms are employed. In this thesis, the Hammerstein model identification problem is posed as an optimization problem that enables use across a broad range of optimization algorithms. Figure 3-1 illustrates the proposed configuration of the Hammerstein model.

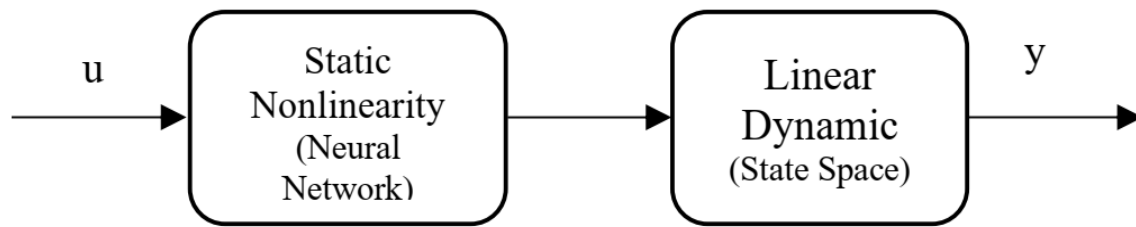


Figure 3- 1 Proposed Structure of the Hammerstein model.

3.2 Static Nonlinearity and Neural Network

Inspired by the nature of our biological nervous systems, artificial neural networks (ANNs) seek to achieve their versatility through learning. WS McCulloch and W Pitts were the first to introduce ANNs as a mathematical model. ANNs are often used in the realms of artificial intelligence (AI), machine learning, and pattern recognition.

ANNs may be conceived of as a set of linked processing units, each of which takes input from external sources and related systems to determine a distributable output.

These processing units reflect the neurons of the human brain which are interconnected by synapses. The ANNs are implemented with the aid of an artificial neuron; a propagation law and an activation function make up this neuron (processing unit).

3.2.1 The Perceptron

The perceptron, comprised of only one completely functioning neuron, functions as the simplest neural network. The structure of the perceptron consists of inputs along with their weights, outputs and step or sigmoid activation function. This structure is represented in Figure 3-2. A single perceptron can solve a whole class of problems, however, the ANNs that consist of a single perceptron cannot be utilized to resolve complex problems. In such cases, a perceptron could be combined to form a multilayer perceptron network (MLP) which could then be used to tackle more complex problems.

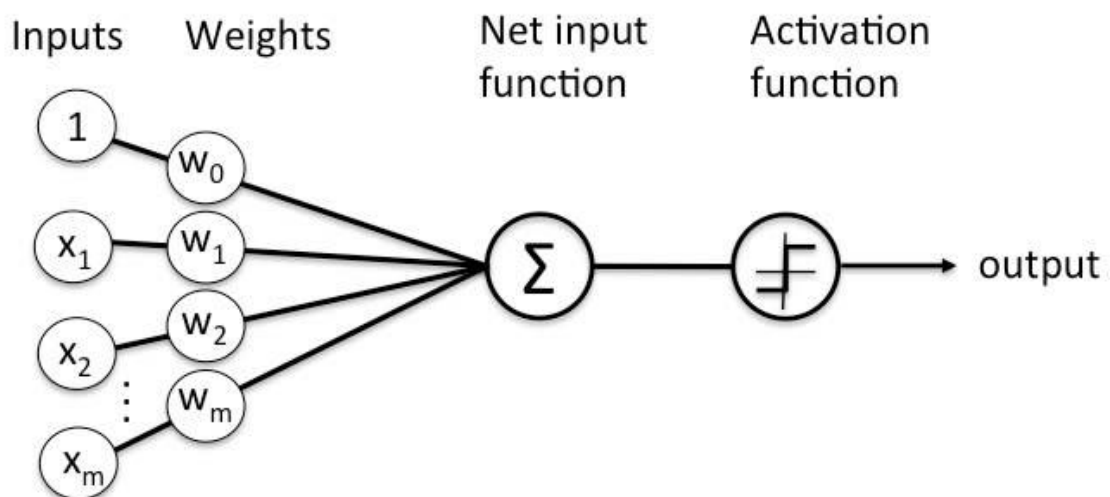


Figure 3- 2 Schematic of the perceptron.

3.2.2 Multilayer Perceptron (MLP)

The Multilayer Perceptron (MLP) is a feed-forward neural network consisting of several layers of neurons, all of which are mutually interconnected. It was designed to solve complex tasks that cannot be solved using only a perceptron.

the two-layer feed-forward neural network has been frequently used in practical implementations, such as classification of nonlinearly separable patterns [95], data prediction [96] and functions approximation [97].

Structurally, the MLPs' layers are stacked on top of each other, with every neuron in one layer attached to every neuron in the next layer. One or more hidden layers, whose computation nodes are known as hidden neurons or hidden units, comprise this kind of network.

The main purpose of hidden neurons is to communicate in some useful way with the external input and network output, as well as obtaining higher-order statistics. A unit step activation function is not suitable for MLP, and the sigmoid activation function is more popularly utilized due to its properties of continuity and flexibility.

Figure 3-3 depicts the construction of a neural network with a two-layer perceptron with C input units, D output units and M hidden units.

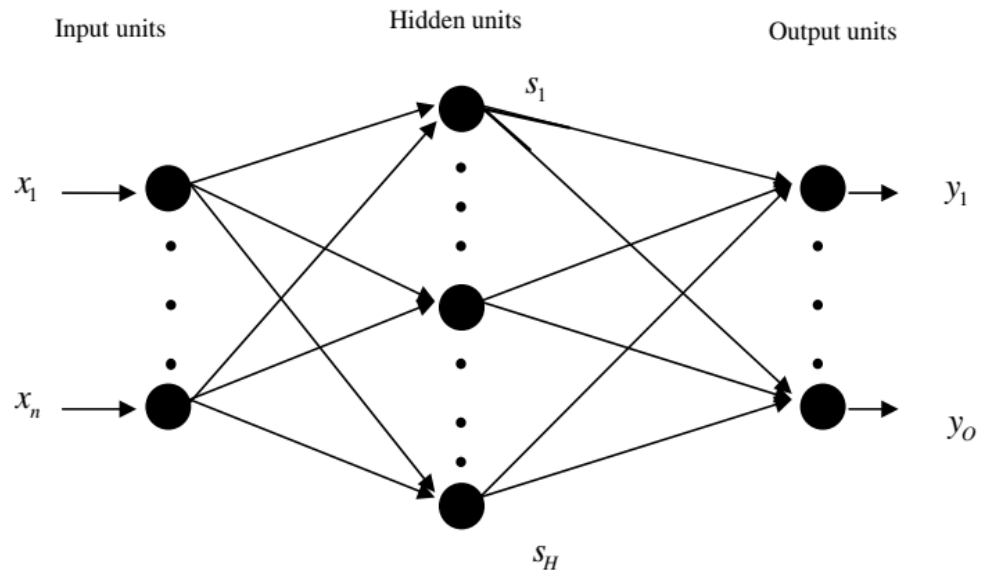


Figure 3- 3 Multilayer perceptron with one hidden layer.

3.2.3 Training of Artificial Neural Networks

The central concept of neural networks lies in its capacity to learn. The objective of the method, then, is to determine the optimal parameters and structure of the network to solve the problem. The network parameters must be initialized before the training phase begins.

Initial values are often arbitrarily selected, although certain heuristics can contribute to a faster parameter adjustment with the optimal values. Learning is then performed on the training set by sending the training data across the network.

This is an iterative process in which the outputs generated on each input of the training set are evaluated, while the network is periodically updated to achieve improved outcomes.

The network is trained to achieve the goal output of the training results. The method of learning from the labeled data set is referred to as supervised learning. If the data labels are unavailable, it is possible to use special algorithms for unsupervised learning.

The ultimate goal of the learning process is to determine the right values for weights and biases to obtain a minimal error. Many MLP implementations use either the traditional back-propagation [99] or the enhanced back-propagation [100][101] for the learning process.

Back-propagation is known as a gradient-based algorithm with some drawbacks, such as local minima trapping, poor convergence and also extreme reliance on initial values of parameters, weights, and biases.

In this work, static nonlinearity will be represented by a multilayer feedforward neural network. Equations (3.1) and (3.2) are the mathematical representation of the multilayer feedforward neural network provided in Figure 3-3:

$$s_j = \sum_{i=1}^n w_{ij} \times x_i - \theta_j \quad (3.1)$$

Where n is the number of input nodes, the hidden layer is given with H nodes and the output layer have O output nodes. $j = 1, 2, \dots, H$, w_{ij} is the weight from i th node of input layer to j th node of the hidden layer, x_i is i th input and θ_j is bias of j th hidden layer.

The output of k th output layer is given by

$$y_k = \sum_{j=1}^H w_{kj} \times f(s_j) - \theta_k \quad (3.2)$$

Where, $k = 1, 2, \dots, O$, w_{kj} is the weight between j th hidden node to k th output node.

f is the activation function and θ_k is bias of the output layer.

3.3 Linear Dynamic and State Space

To describe the linear component of the Hammerstein model, the state-space model is used. The definition of state is the minimum number of state variables required to characterize the system and provide a direct response to any input. Three variables are used for modeling a dynamic system with state-space system: input variables, output variables, and state variables.

The state-space representation for a given system is not unique, and there are no unique state variables that describe any given system, thus, a different set of variables could be chosen to get a system description. It is possible to mathematically transform one set of

state variables to another; the main notion is that any set of state variables must provide a full description of the system under study.

State-space equations are shown in the below equation:

$$\dot{x}(t) = Ax(t) + B v(t) \quad (3.3)$$

$$y(t) = Cx(t) + D v(t) \quad (3.4)$$

Equation (3. 3) represents the state equation, (3. 4) represents the output equation. A is state matrix, B represents the input matrix, C is output matrix and D is the direct transition or feed-through matrix. $v(t)$ represents input of the system, $x(t)$ are the state variables and $y(t)$ give the model output.

In this work, heuristic algorithms will be applied to identify parameters of the state space matrices (A, B, C and D).

3.4 Controllable Canonical Form

State-space could be represented in an infinite number of forms, a limited example being controllable form, observable form, and Jordan forms among others. In this study, a controllable canonical form is utilized to represent the linear element of Hammerstein model identification. Controllable canonical form has been utilized to minimize number of parameters being estimated.

3.4.1 SISO Controllable Canonical Form

If we consider a system defined as

$$\frac{Y(s)}{V(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \dots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} \quad (3.5)$$

Where:

$V(s)$ is the input, $Y(s)$ is the output.

The controllable canonical form for the system given in (3. 5) will be:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \dots & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} v \quad (3.6)$$

$$y(t) = [b_n - a_n b_0 \quad b_{n-1} - a_{n-1} b_0 \quad \dots \quad b_1 - a_1 b_0] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + b_0 v$$

3.4.2 MIMO Controllable Canonical Form

A MIMO controllable canonical form can be defined in different methods. One of these methods for representing a MIMO controllable canonical form is given in [98].

Let assume that we have a system $G(s)$ is a 2×2 matrix as below.

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \quad (3.7)$$

Where the realization of $g_{ij}(s)$ is

$$\dot{x}_{ij} = A_{ij}x_{ij} + b_{ij}v_j \quad , \quad y_{ij} = c_{ij}x_{ij} + d_{ij}v_j$$

The realization of $G(s)$ will be:

$$\begin{bmatrix} \dot{x}_{11} \\ \dot{x}_{12} \\ \dot{x}_{21} \\ \dot{x}_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 & 0 \\ 0 & A_{12} & 0 & 0 \\ 0 & 0 & A_{21} & 0 \\ 0 & 0 & 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & 0 \\ 0 & b_{12} \\ b_{21} & 0 \\ 0 & b_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (3.8)$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & 0 & 0 \\ 0 & 0 & c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

CHAPTER 4

PROPOSED IDENTIFICATION ALGORITHM

4.1 Heuristic Optimization Algorithms

The heuristic describes those computational procedures seeking to find the optimal solution through the improvement of a candidate solution iteratively, under the implication of a given quality measure. Heuristic optimization techniques have become increasingly popular over the last decade in part due to their simplicity, but also because of their cheap computing expense, flexibility, and derivation free mechanism. In addition, heuristic algorithms have been extended to a broad variety of real-life applicable problems.

In the last few years alone, we have witnessed many heuristic algorithms being implemented. These include Particle Swarm Optimization (PSO), Differential Evolution (DE), Ant Colony (AC), Gravitational Search Algorithm (GSA), Genetic Algorithm (GA) among others.

These algorithms are designed solely to deduce the optimal result or global optimum from all possible inputs. Most of these algorithms are population-based, iteratively evaluating and enhancing the sets of the solution to comply with the predefined criteria. To perform

this function, two characteristics should be adopted, namely exploration (diversification) and exploitation (intensification)[99].

The main objective of the exploration phase is to discover promising regions within the search space by randomly changing the candidate solutions to guarantee that they explore different areas of the search landscape.

The exploitation or intensification phase is the potential of the algorithm to converge near the best solution –in other words, the algorithm pushes the candidate solution to search near the best solution founded. This means that in the exploitation phase, the search is conducted locally, near the best solutions founded through exploration.

Perhaps among the most complicated steps in the design of heuristic algorithms, or more precisely population-based algorithms, is to find a compromise between the phases of exploration and exploitation. These two steps necessarily conflict due to their functionality.

Algorithms with only exploration features will result in poor solutions as they will not have the opportunity to improve the accuracy of solutions. On the other hand, a mere exploitation algorithm will lead the outcomes to be trapped in local solutions, because the algorithm does not change the search space randomly.

The ultimate objective of designing heuristic algorithms, then, is to determine the most efficient balance between the exploration phase and exploitation phase to reach a global optimum solution.

In this analysis, various heuristic optimization algorithms will be used to identify the Hammerstein model.

4.2 Particle Swarm Optimization (PSO)

Regarded as one of the most reputable evolutionary computation techniques in the literature of the optimization methods, the particle swarm optimization (PSO) algorithm was introduced by Kennedy and Eberhart [100], [101]. The implementations of this algorithm focus on a broad variety of fields, in which it is used to solve a multitude of problems.

Yet another algorithm founded on the concept of natural biology, the PSO algorithm is modeled from the navigation mechanism of birds and their social behaviour. In PSO, every candidate solution to the problem is considered a particle that has the ability to move in the search area.

The location of each particle is known to be updated by two vectors: the position vector and the velocity vector. The position vector contains values for all variables of the problem under study and is considered as the location of the particle. The velocity vector is used to update position of particles.

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (4.1)$$

t represents the current iteration, v_i^t is velocity of particle i and x_i^t is the position of particle i .

From equation (4.1) it is clear that position updating relies on the velocity vector, which can be described below as:

$$v_i^{t+1} = w \times v_i^t + c_1 \times r_1 \times (Pbest_i - x_i^t) + c_2 \times r_2 \times (Gbest - x_i^t) \quad (4.2)$$

c_1 is the individual coefficient and c_2 signifies the social coefficient where c_1 and c_2 are called the weighting factors, r_1 and r_2 are random numbers between [0,1] and w is the inertia weighting function.

$Pbest_i$ is the personal best of the best solution obtained by particle i until the t iteration.

$Gbest$ is the global best or the best solution founded by all particles until the t iteration.

Optimization using the PSO is commenced by choosing a set of random particles. A position vector and velocity vector should be assigned to each particle. After that, all parameters of PSO including c_1 , c_2 and w should be initialized. Random numbers (r_1 and r_2), weighting function (w), velocity and position vectors are updated repeatedly until the optimization criteria are met.

For each problem, the best solution obtained by the entire swarm of particles is saved as the best estimate of the global optimum (*Gbest*). The increase of convergence probability by applying the PSO algorithm is attributed to the use of best solutions, obtained by each particle *Pbest_i* and the whole swarm *Gbest*, during the position update of the particles.

PSO algorithm maintains the best solutions founded so far and orders all particles in the swarm to look around them. This raises the likelihood of obtaining a better solution. Figure 4-1 represents the flowchart for the steps of the PSO algorithm.

In this study, the PSO algorithm will be used to identify the Hammerstein model. Weights and biases of the feed-forward neural network representing the non-linear portion of Hammerstein model, as well as parameters of state-space model, will be estimated using the PSO algorithm.

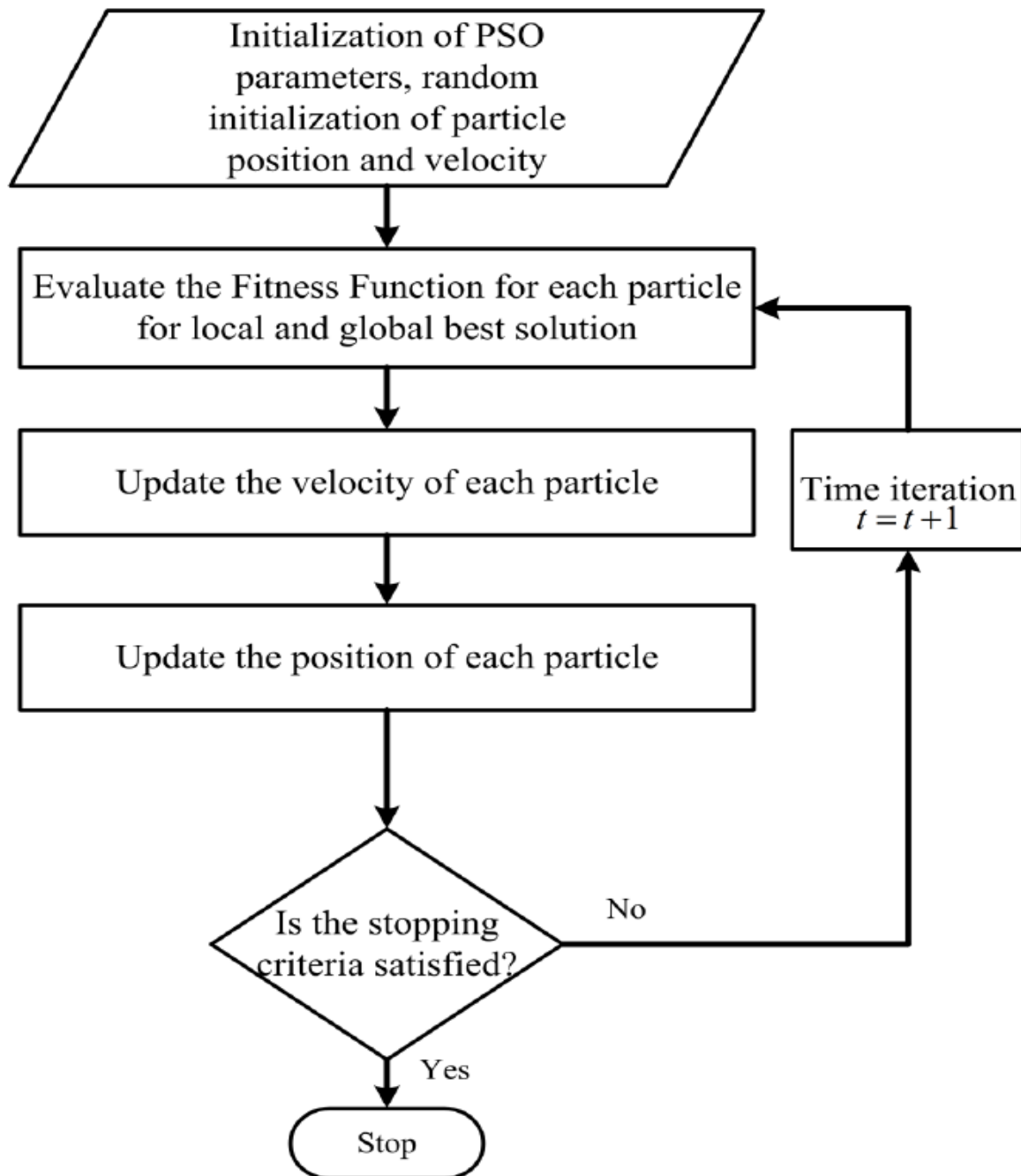


Figure 4- 1 Flowchart of the PSO algorithm.

4.3 Gravitational Search Algorithm (GSA)

Gravitational Search Algorithm (GSA) was initially introduced in 2009 [102] by Rashedi et al. This algorithm has been inspired by physical laws, specifically Newton's theory of gravity.

In this algorithm, agents (solutions) have a mass value that is relative to the fitness function value. On each epoch of the algorithm, agents attract each other, and the heavier mass has a strong force that pulls the other masses to its location. A mathematical model of the forces between solution (j) and solution (k) can be written as follows:

$$f_{jk}^d(t) = G(t) \frac{M_{pj}(t) \times M_{ak}(t)}{R_{jk}(t) + \varepsilon} (x_j^d(t) - x_k^d(t)) \quad (4.3)$$

Where x_j^d & x_k^d are positions of agents (solutions) j and k on d dimension, M_{pj} is agent j passive gravitational mass, and M_{ak} is the active gravitational mass related to solution k .

R_{jk} is Euclidean norm between j and k , ε is a constant, and $G(t)$ is gravitational constant that could be measured as:

$$G(t) = G_0 \times \exp \left(-\alpha \frac{\text{iteration}}{\text{maximum iteration}} \right) \quad (4.4)$$

Where G_0 is initial gravitational constant and α is the descending coefficient.

Total forces that act on the agent (j) in a (d) dimensional problem are given by:

$$f_j^d(t) = \sum_{i=1, i \neq j}^N r_i f_{ji}^d(t) \quad (4.5)$$

r_i is a random number. With the help of the law of motion, acceleration of agent (j) can be written as:

$$C_j^d(t) = \frac{f_j^d(t)}{M_j} \quad (4.6)$$

Where C_j^d is agent (j) acceleration and M_j is mass of agent (j).

Then velocity and position of agent (j) can be given in (4.7) and (4.8) respectively.

Masses of agents are defined using the fitness function, implying that the agent with the heaviest mass will be the ideal agent, as the agent with the heaviest mass will have a higher attraction force and slower movement.

$$v_j^d(t + 1) = r_j \times v_j^d(t) + C_j^d(t) \quad (4.7)$$

$$x_j^d(t + 1) = x_j^d(t) + v_j^d(t + 1) \quad (4.8)$$

Update equations for the masses of all agents are:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (4.9)$$

$fit_i(t)$ is the fitness value of agent i at time t , $best(t)$ is the best agent at time t and $worst(t)$ is the weakest or the worst agent at time t .

Normalization of masses $m_i(t)$ is calculated as below:

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (4.10)$$

By selecting proper values for G_0 and α , exploration and exploitation characteristics could be quarantined. Figure 4-2 shows flowchart of the GSA algorithm.

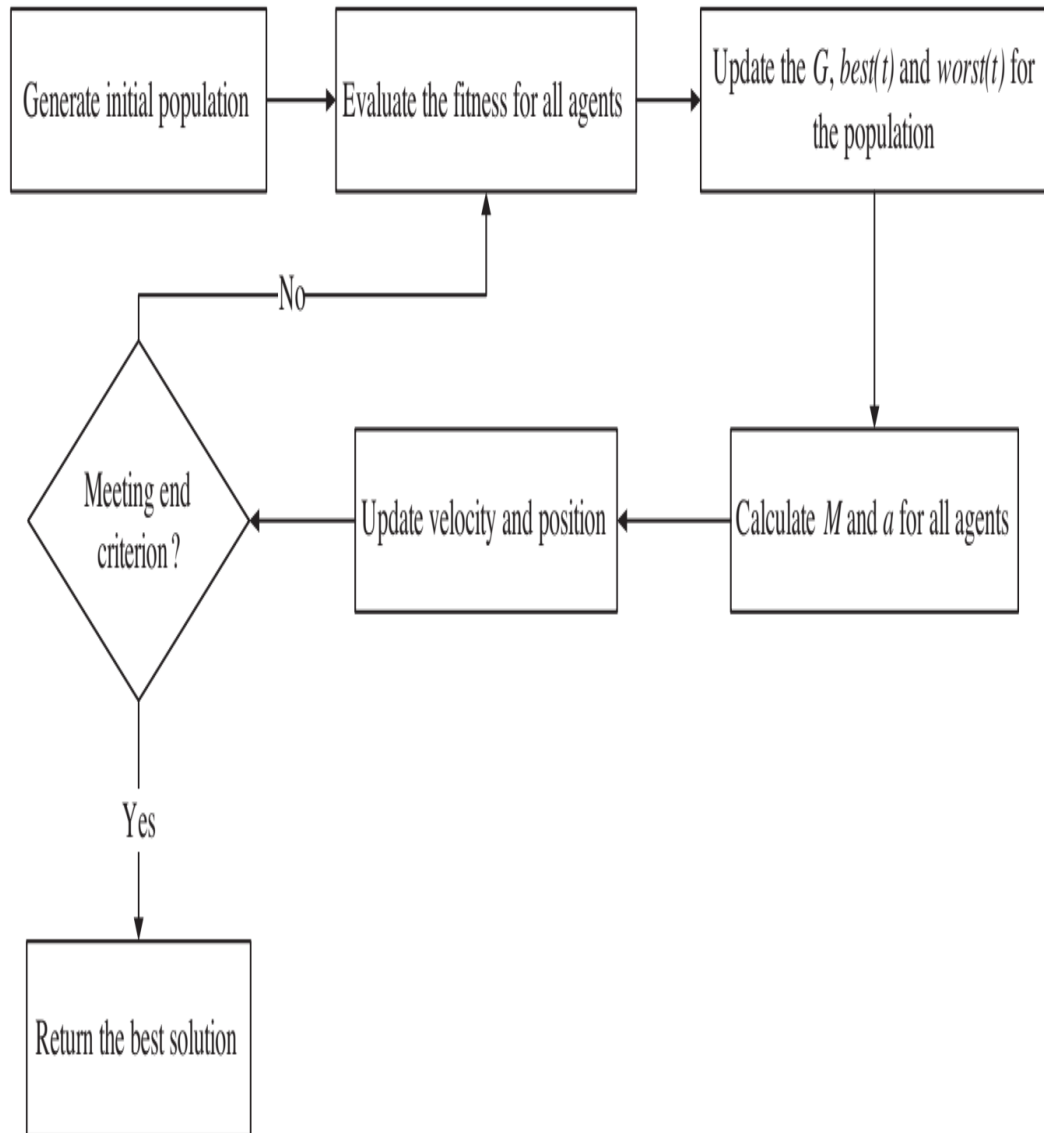


Figure 4- 2 Flowchart of GSA algorithm.

4.4 A hybrid algorithm between PSO and GSA (PSOGSA)

Proposed in 2010 by Mirjalili and SVM Hashim, is a hybrid heuristic algorithm between PSO and GSA named PSOGSA [103]. The main concept of PSOGSA is to integrate the social thinking of PSO with the local search capability of GSA.

Velocity and position of solution (j) using PSOGSA can be given in (4.11) and (4.12) respectively:

$$v_j(t + 1) = w \times v_j(t) + c'_1 \times r_1 \times c_j(t) + c'_2 \times r_2 \times (Gbest - x_j(t)) \quad (4.11)$$

$$x_j(t + 1) = x_j(t) + v_j(t + 1) \quad (4.12)$$

Where t represents current iteration, $v_j(t)$ is the velocity of solution (j), $c_j(t)$ is the acceleration of solution (j), and $Gbest$ represents the best solution of the entire search space.

In the PSOGSA algorithm, agents considered as candidate solutions and are initialize arbitrarily. After algorithm initialization, gravitational force, gravitational constant, and agents' forces are determined utilizing equations (4.3), (4.4) and (4.5) respectively.

Following this step, the acceleration of the particles is measured as follows (4.6). The best solution will be updated for each iteration. Then the position and velocity of all agents will be calculated using equations (4.11) and (4.12) respectively. This function will continue until one of the ending criteria is met.

The efficiency of the PSOGSA algorithm can be highlighted in the points below:

- In PSOGSA, the quality of solutions or fitness is considered in the updating process.
- Agents close to the good options seek to attract other agents exploring new areas of the search field.
- When all agents are close to a good solution, they move slowly. In this situation, the *Gbest* helps to exploit the global best.
- PSOGSA uses a memory (*Gbest*) to save the best solution found to date, making it accessible at any time.
- Each agent has the capability to observe best solutions or (*Gbest*) and lean toward it.
- By modifying c'_1 and c'_2 , global searching and local searching features can be balanced.

Figure 4-3 represents the flowchart of PSOGSA algorithm.

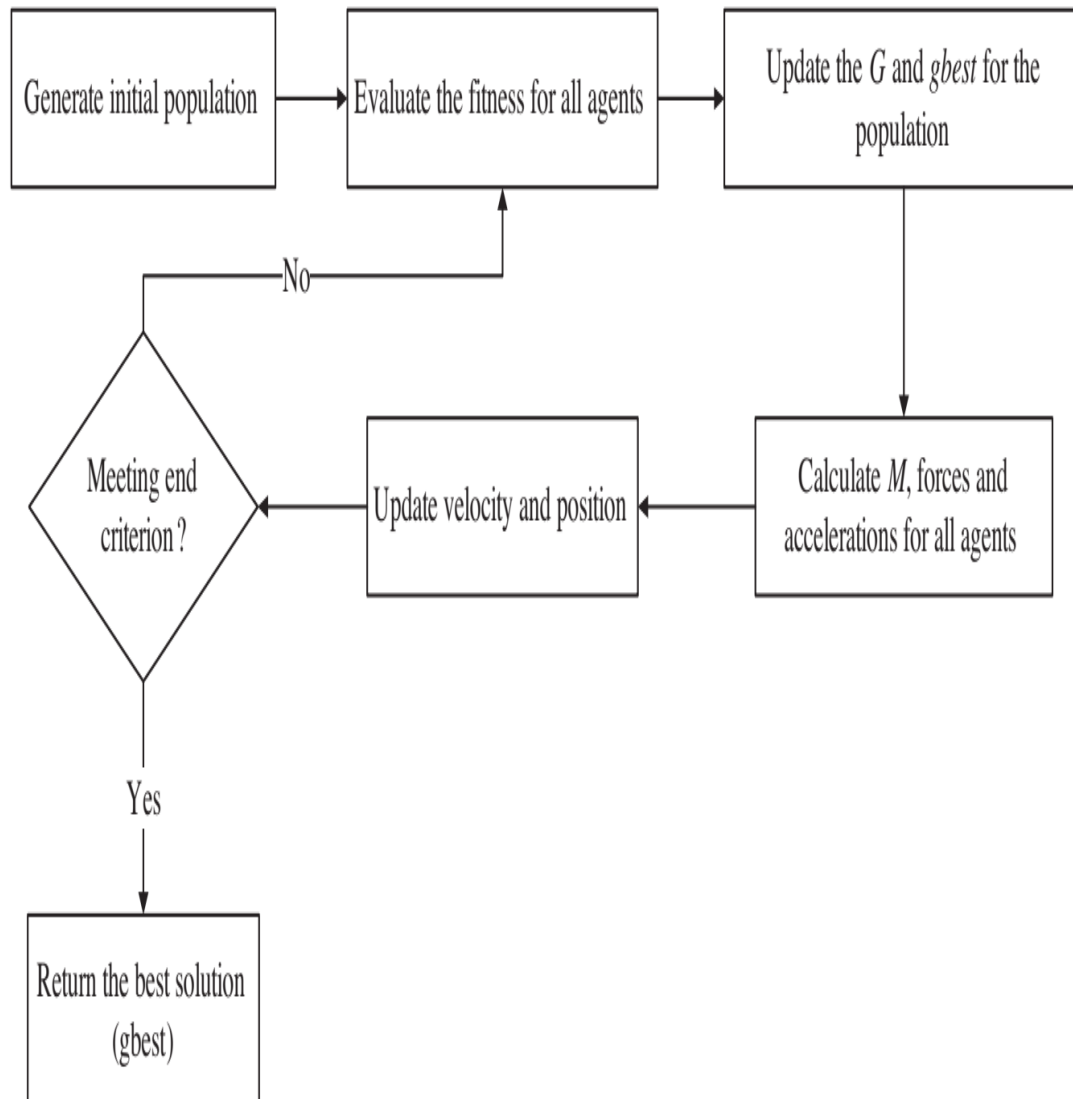


Figure 4- 3 Flowchart of the PSOGSA algorithm.

Considering the aforementioned remarks, the PSOGSA algorithm can be used to solve a vast variety of optimization problems.

4.5 Fitness Function

Heuristic optimization algorithms are utilized to address the Hammerstein model identification problem. This problem is interpreted as one of optimization, that has enabled the use of a broad variety of optimization algorithms like Particle Swarm Optimization (PSO), the hybrid algorithm between Particle Swarm Optimization (PSO) and Gravitational Search Algorithm (GSA) named (PSOGSA).

For this study, PSO and Hybrid Algorithm (PSOGSA) was used to approximate the weights and biases of the feedforward neural network, as well as to determine parameters of the state-space matrices.

Minimization is the objective of using the heuristic algorithm in this work. The error between the actual and estimated models will be minimized using a suitable fitness function.

The Mean Square (MSE) error is the fitness function of the heuristic algorithms used in this research. MSE calculates the difference between measured and identified output as:

$$fitness (MSE) = \frac{1}{N} \sum_{k=1}^N [y(k) - \hat{y}(k)]^2 \quad (4.13)$$

Where

$y(k)$ is the measured output, $\hat{y}(k)$ is the identified output and N is represents training samples.

CHAPTER 5

SIMULATION RESULTS

5.1 Simulation results for SISO models

5.1.1 Case study 1: saturation nonlinearity

Equation (5.1) represents saturation nonlinearity that is used as the nonlinear element of the Hammerstein model. The state-space equation (5.2) is used as the linear element of Hammerstein model.

$$f(t) = \begin{cases} -0.5 & \text{when } -0.5 \geq u(t) \\ u(t) & \text{when } 0.5 > u(t) > -0.5 \\ 0.5 & \text{when } u(t) > 0.5 \end{cases} \quad (5.1)$$

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.12 & -1.72 & -5.63 & -4.4 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v(t) \quad (5.2)$$

$$y(t) = [2 \quad 1 \quad 4 \quad 0] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}$$

PSO Simulation Results:

The input used to generate the data was a random number distributed between -1 and 1. Hyperbolic tangent sigmoid and linear activation functions of the multilayer perceptron (MLP) are used in the hidden layer and output layer respectively.

A comparison between output of the nonlinear part in (5.1) and identified model is depicted in Figure 5-1. The estimated linear element of the Hammerstein model possesses the following parameters:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.135 & -1.83 & -5.42 & -4.56 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v(t) \quad (5.3)$$

$$y(t) = [1.9 \quad 1.05 \quad 4.18 \quad 0] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}$$

The convergence curve of PSO is given in Figure 5-2, with a Mean Square Error (MSE) of 0.062 and 1000 iterations.

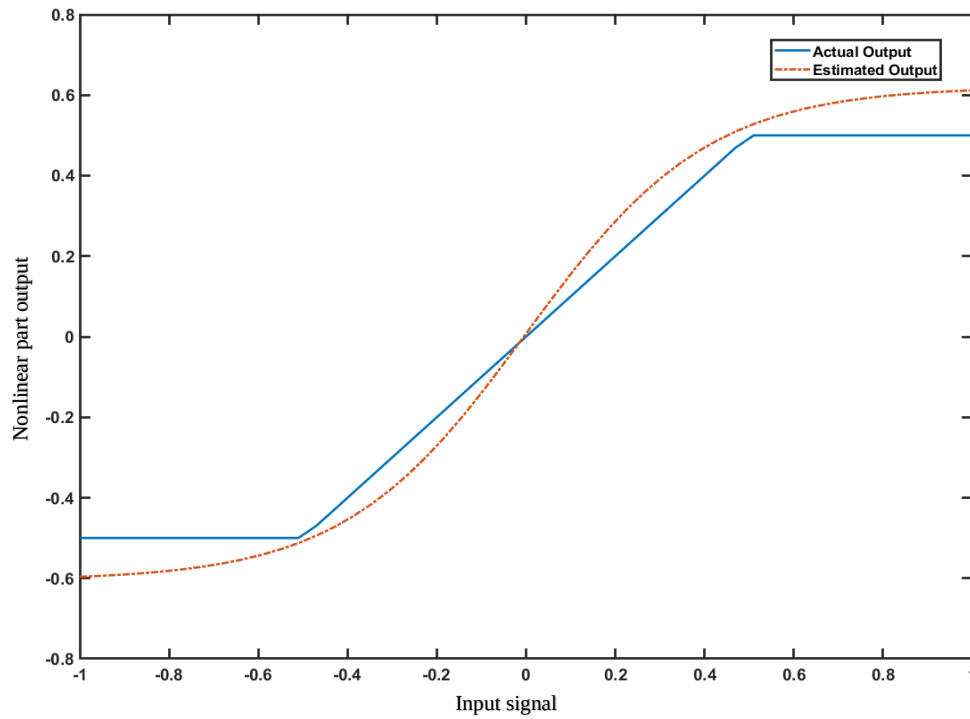


Figure 5- 1 Saturation Nonlinearity with identified nonlinearity Using PSO.

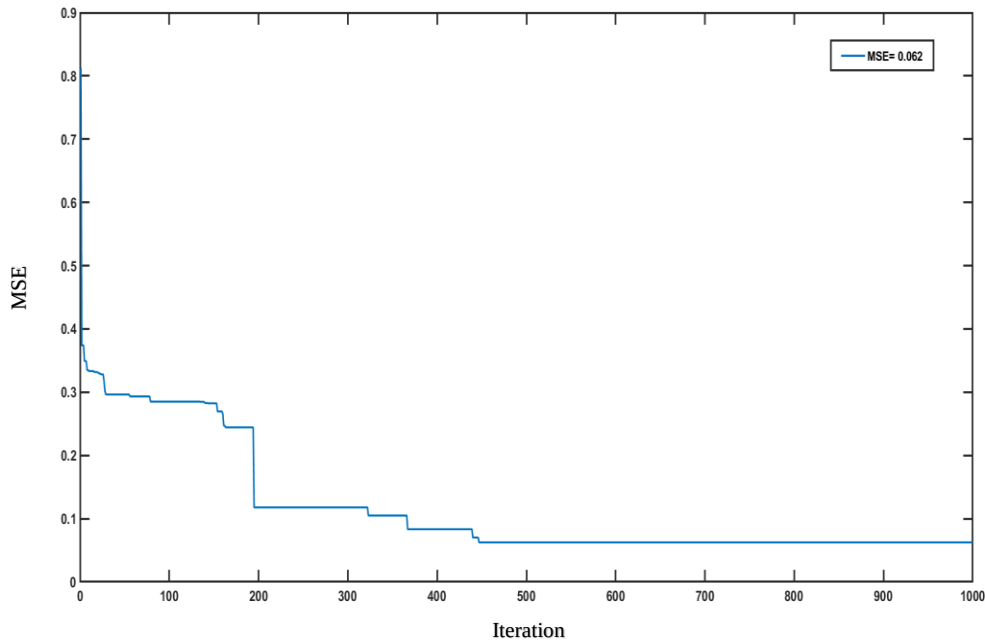


Figure 5- 2 Convergence Curve of the PSO algorithm.

PSOGSA Simulation Results:

To excite the system represented in (5.1), uniform random numbers are used. The hyperbolic tangent sigmoid activation function is applied to the hidden layer and linear activation function to the output layer of the MLP . Figure 5-3 depicts a comparison between the saturation nonlinearity and the identified nonlinear element of the Hammerstein model using PSOGSA.

Parameters and eigenvalues of the identified state-space model are:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.1193 & -1.752 & -5.608 & -4.4 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v(t) \quad (5.4)$$

$$y(t) = [1.97 \quad 1 \quad 4.06 \quad 0] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}$$

Figure 5-4 illustrates the convergence curve of the PSOGSA algorithm where the MSE was 0.0107 within 1000 iterations.

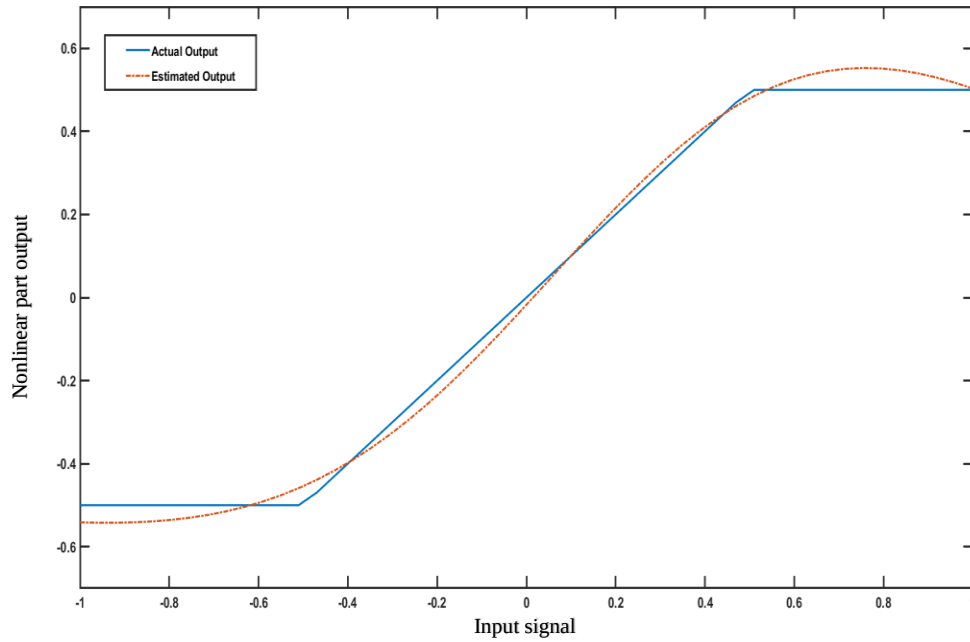


Figure 5- 3 Saturation Nonlinearity with identified nonlinearity Using PSOGSA.

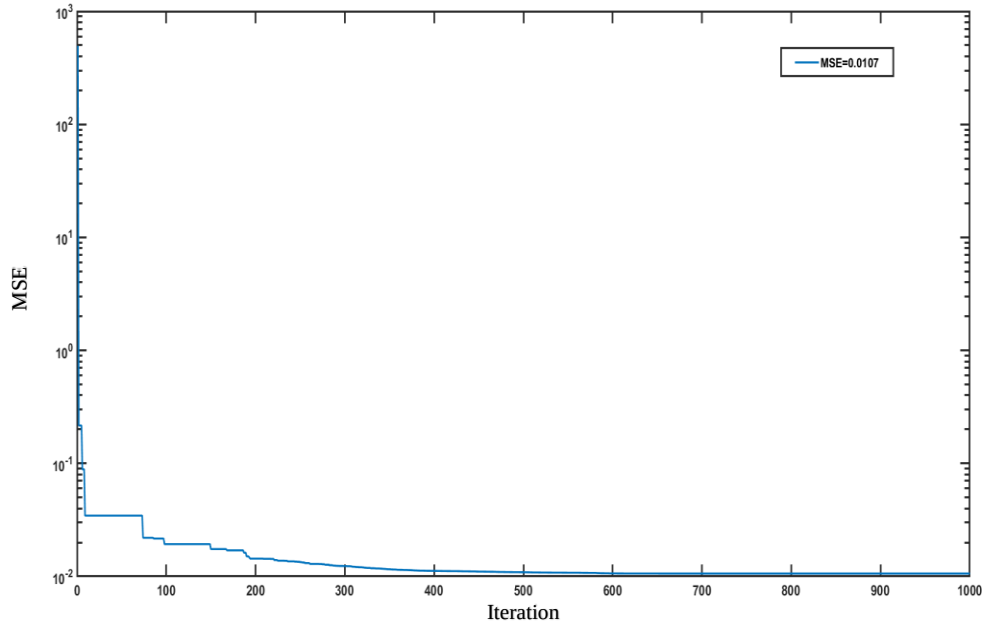


Figure 5- 4 Convergence Curve of PSOGSA algorithm.

Results evaluation:

PSO and PSOGSA heuristic algorithms have been applied in this work to estimate weights and biases of MLP and parameters of state-space matrices. Both algorithms were capable of capturing the nonlinearity of the proposed model, as well as estimating parameters of the state-space system with good accuracy. PSOGSA demonstrates a better performance in terms of MSE.

5.1.2 Case 2: Coupled electrical drives

In this example, the coupled electric drive is modelled with different algorithms, and induced by the need for machines to transfer material from one spool to another, using hoist and two electric drives. This system is one of the most widely applied nonlinear benchmarks in system identification. [104]. Figure 5-5 represents a schematic for the coupled electric drive. In this system, the input is applied to the electric drives that move the belt – after which the flexible belt rotates a hoist which been held by the spring. A pulse transducer is employed to measure angular speed of the hoist, and readings are sent through both a digital anti-aliasing filter and an analogue low pass filter to generate output of the system. This model is considered a speed control system.

The data, generated experimentally, is then used to formulate the physical model. The sum of voltages applied to the electric drives is used as an input signal. The input signal is a uniformly distributed signal, obtained from pseudorandom binary sequence (PRBS) with a clock time of 5 times the sampling period converting from -1 Volt to +3 Volt. This value is multiplied by a uniformly distributed random number between 0 and 1. The data set contains only 500 input-output samples; a challenge for system identification.

Figure 5-6 and figure 5-7 demonstrate the input and output data used for system modelling.

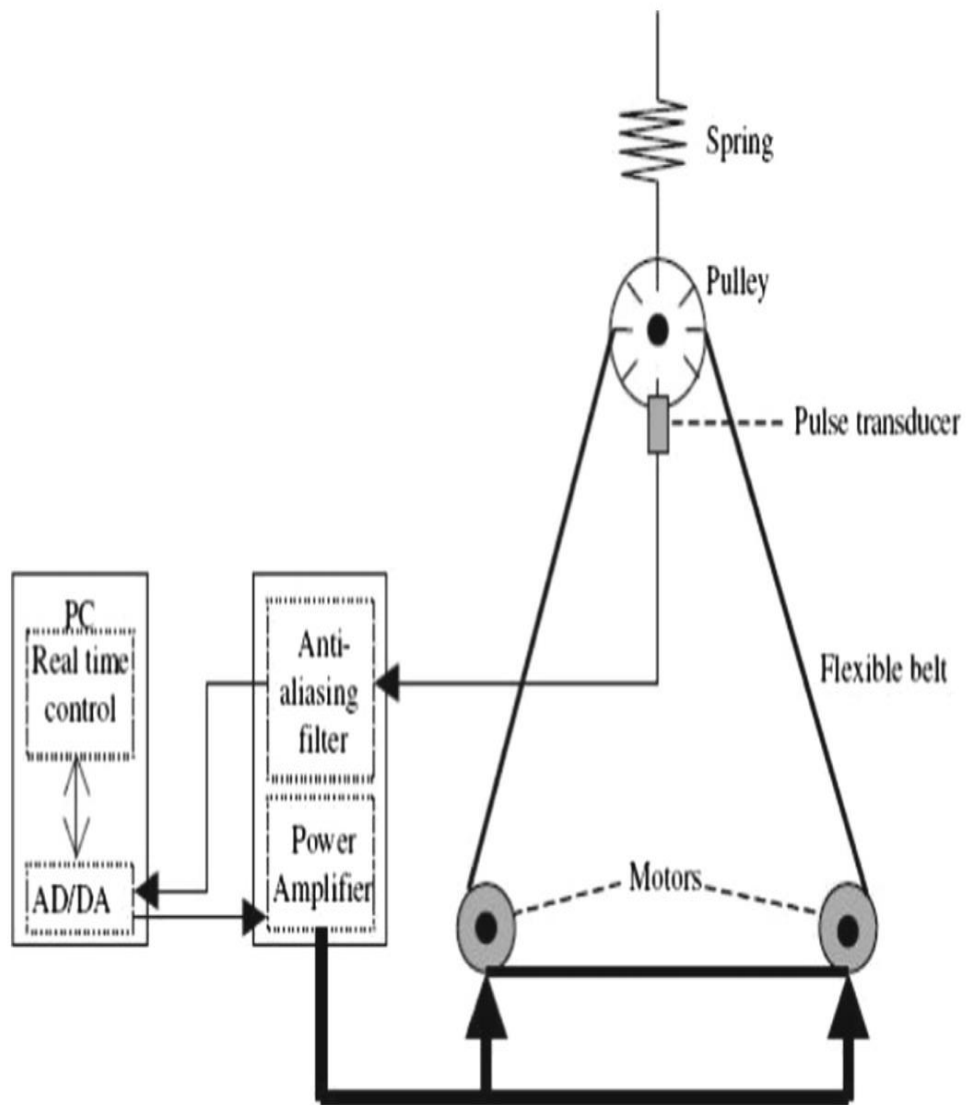


Figure 5- 5 Schematic for coupled electric drive.

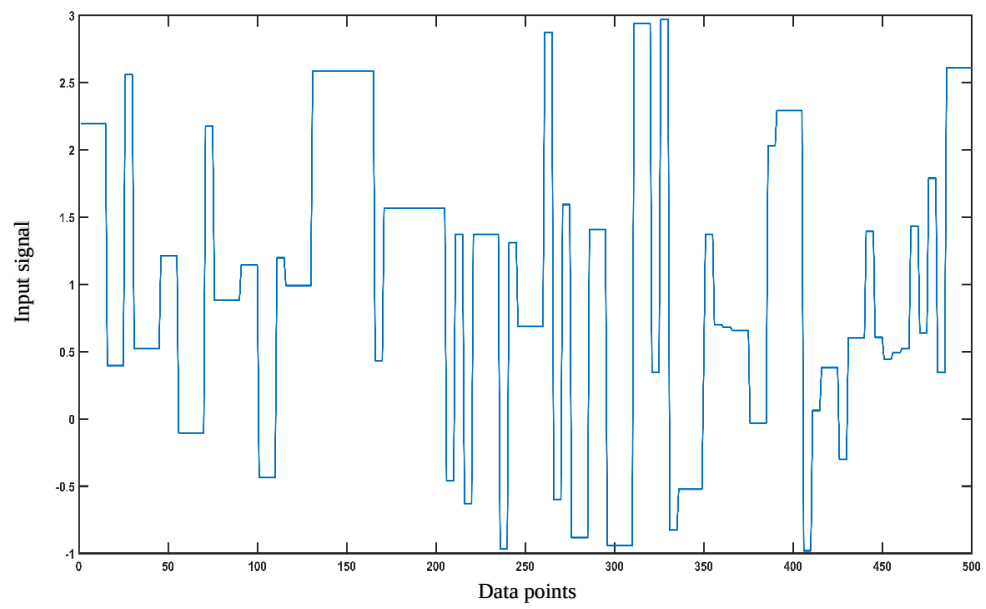


Figure 5- 6 Uniformly distributed input signal.

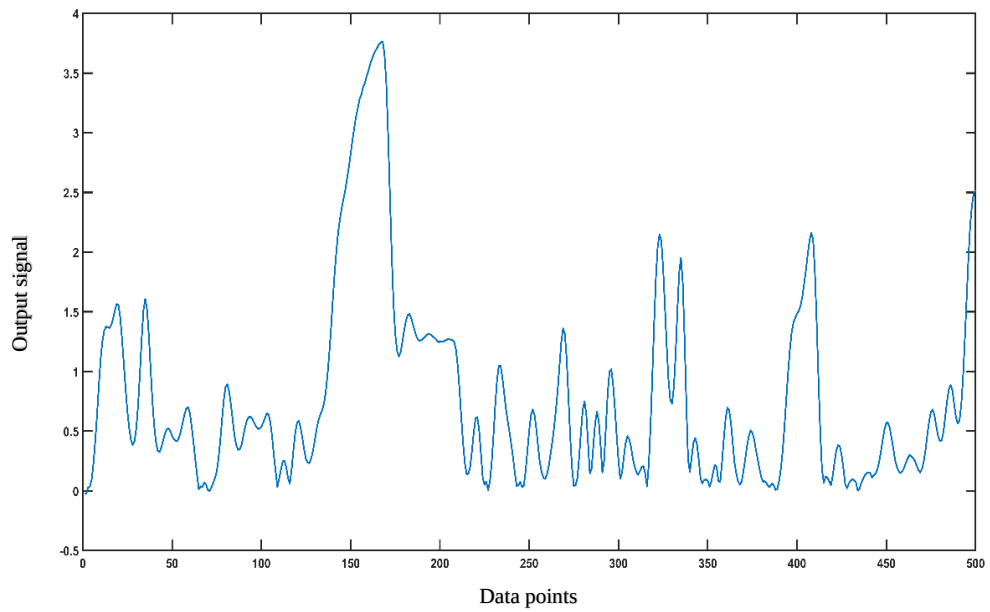


Figure 5- 7 Measured output signal.

The Hammerstein model is used for system identification of the coupled electric drive. The linear component of Hammerstein model is represented by a third order continuous time state-space model. Nonlinearity is represented by the multilayer feedforward neural network. PSO and PSOGSA algorithms are used to estimate the parameters of the state-space model as well as the weights and biases of the neural network.

Modelling using PSO algorithm

Generated inputs and outputs are used for identification. The neural network is used to describe the non-linear portion of the Hammerstein model. The tangent sigmoid activation function is utilized for the hidden layer, with linear activation function used for the output layer. The third-order continuous time state-space system is calculated as shown in (5.5). Figure 5-8 demonstrates the convergence curve of the algorithm.

MSE between the actual output and estimated output is 0.12 within 5000 iterations. Validation of the identified model is given in figure 5-9. Nonlinearity of the neural network is represented in figure 5-10.

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.01 & -0.15 & -0.1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} v(t) \quad (5.5)$$

$$y(t) = [-1.31 \quad -0.05 \quad -0.014] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

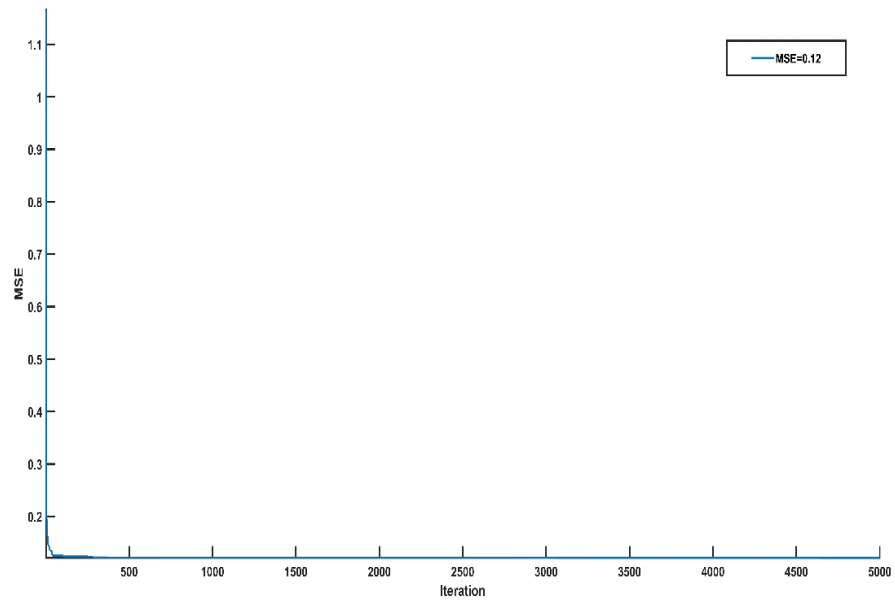


Figure 5- 8 Convergence curve of the PSO algorithm.

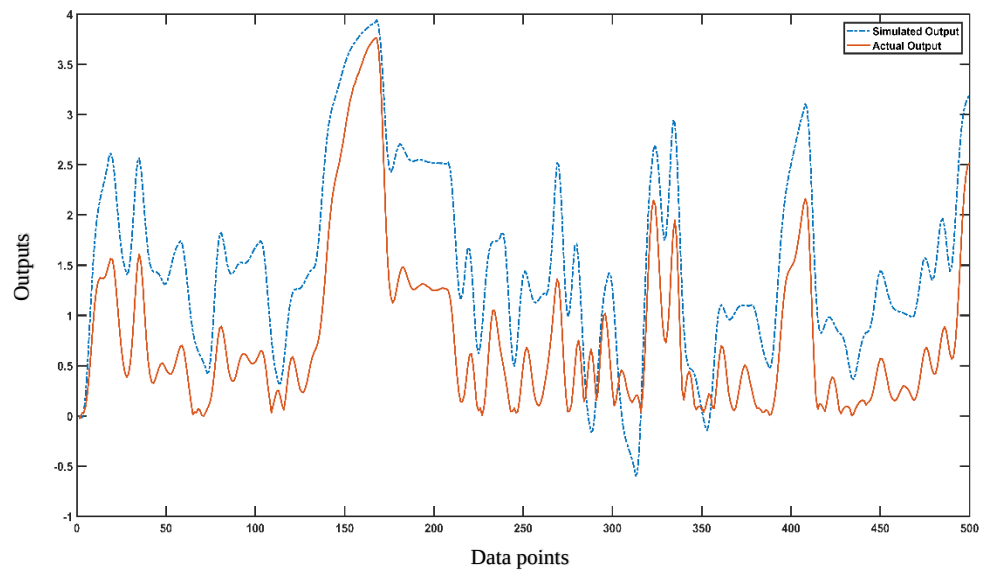


Figure 5- 9 Actual output versus estimated output based on PSO algorithm.

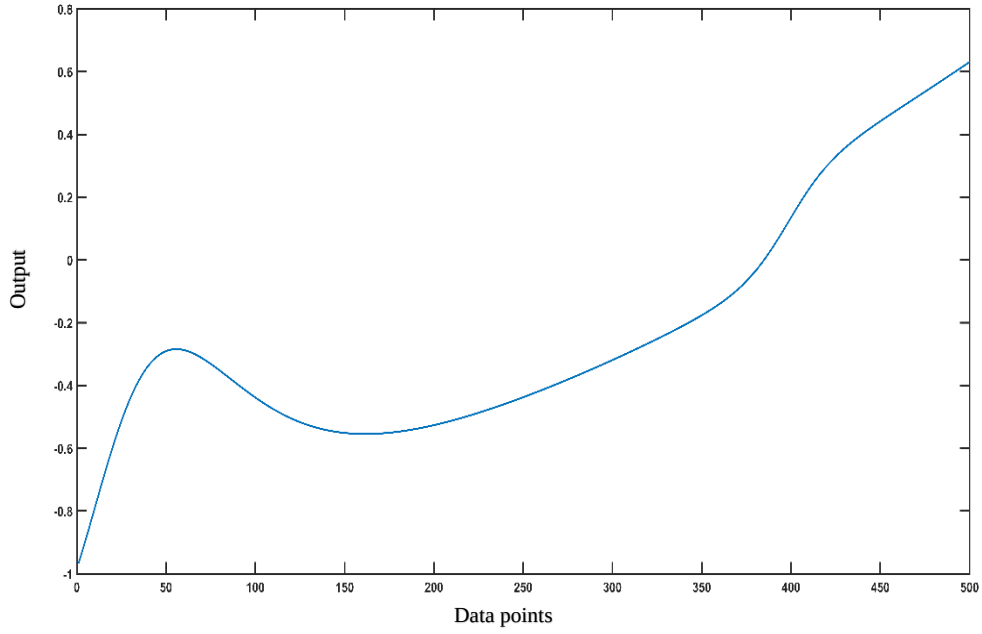


Figure 5- 10 Learnt nonlinearity of the neural network based on PSO.

Modelling using PSOGSA algorithm

The same data set is used to identify the system based on the PSOGSA algorithm. The neural network involves the same form as the PSO-based algorithm. The linear part of the estimated model is represented (5.6).

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.031 & -0.366 & -0.477 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} v(t) \quad (5.6)$$

$$y(t) = [-0.038 \quad 0.06 \quad -0.054] \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

MSE is converged around 0.03 within 3000 iterations, as given in figure 5-11. A comparison between actual and estimated output is depicted in figure 5-12. Detected nonlinearity of the neural network based on the PSOGSA algorithm is given in figure 5-13.

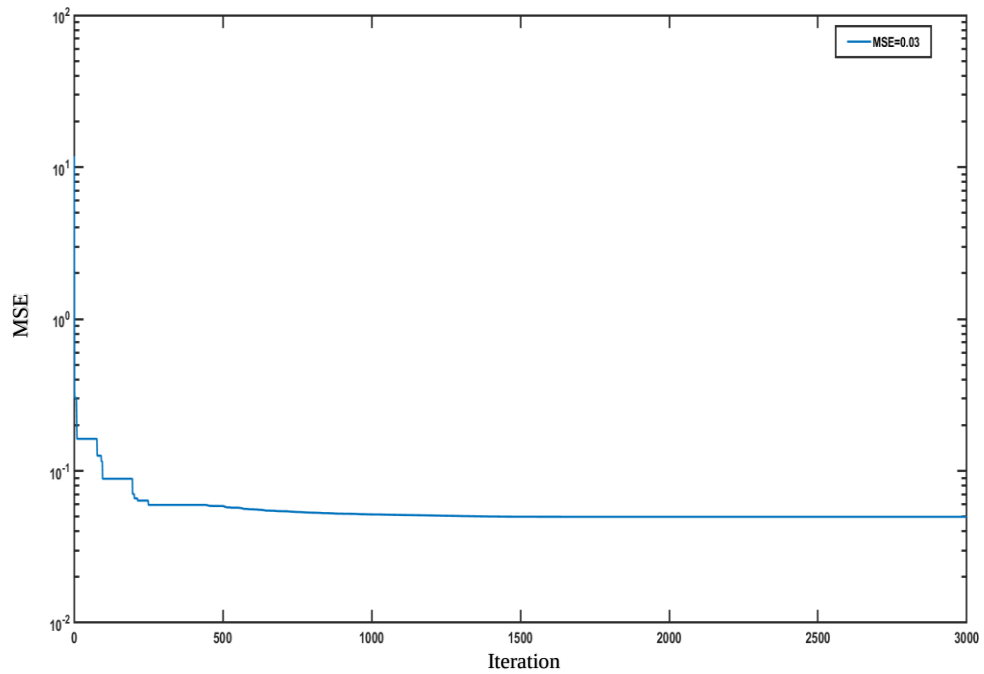


Figure 5- 11 Convergence curve of the PSOGSA algorithm.

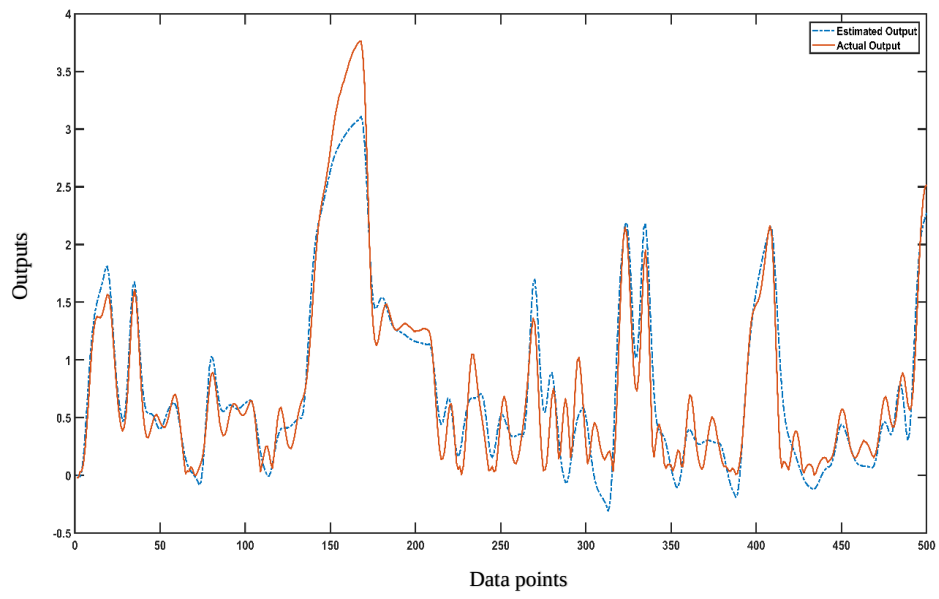


Figure 5- 12 Actual output versus estimated output based on the PSO-GSA algorithm.

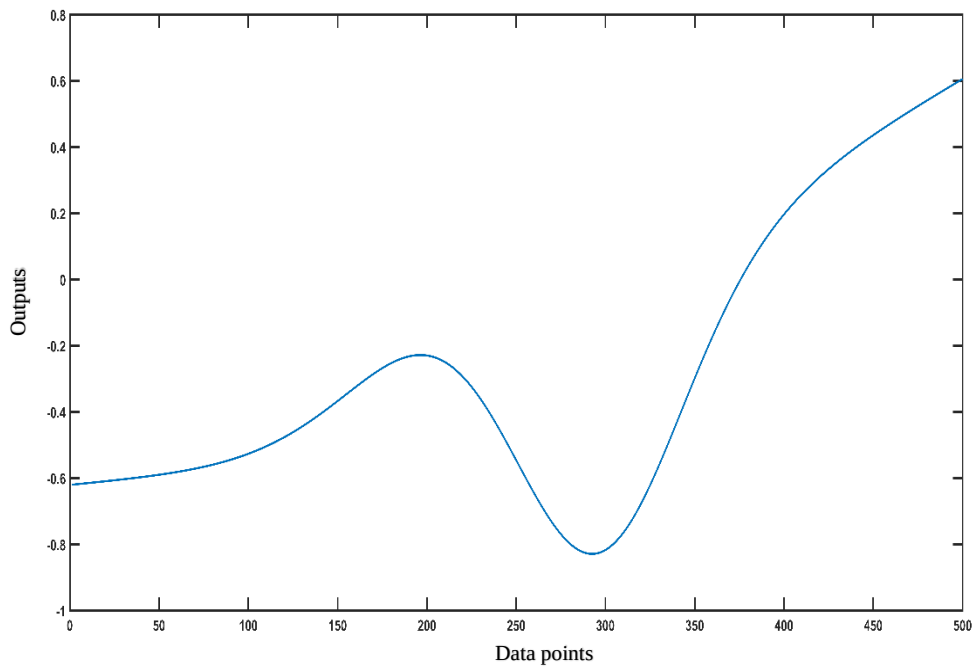


Figure 5- 13 Learnt nonlinearity of the neural network based on the PSO-GSA algorithm.

Results evaluation:

Both algorithms successfully identify the coupled electric drive system. PSOGSA algorithm has faster convergence and a better result in terms of MSE. Results of simulations confirm the excel of the PSOGSA algorithm over the PSO algorithm.

5.2 Simulation results for MIMO models:

5.2.1 Case study 3: Dead zone and Relay nonlinearities

Equations (5. 7) and (5.8) represent dead zone and relay nonlinearities, respectively. These equations are employed to represent the nonlinear element of the Hammerstein model. Uniformly random numbers between -1 and 1 are utilized to excite the nonlinear element of the Hammerstein system. The state-space equation (5.9) explains the linear part of this model.

$$f(t) = \begin{cases} 0 & \text{when } -0.4 \leq u(t) < 0.4 \\ u(t) + 0.4 & \text{when } u(t) < -0.4 \\ u(t) - 0.4 & \text{when } u(t) \geq 0.4 \end{cases} \quad (5.7)$$

$$f(t) = \begin{cases} 1 & \text{when } u(t) > 0 \\ 0 & \text{when } \text{otherwise} \\ -1 & \text{when } u(t) < 0 \end{cases} \quad (5.8)$$

The MIMO state space is given below

$$\begin{aligned} \dot{x}(t) = & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.6 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.5 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -2.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -4 & -4 \end{bmatrix} x(t) \\ & + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} v(t) \end{aligned} \quad (5.9)$$

$$y(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & -6 & 0 & 3 & 0 \\ 0 & 0 & 0.5 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} x(t)$$

PSO Simulation Results:

Used to generate the input signal are the distributed random numbers between -1 and 1. Hyperbolic tangent sigmoid and linear activation functions of the multilayer perceptron (MLP) are applied in the hidden layer and output layer, respectively. The estimated linear element of the MIMO Hammerstein model has the following parameters:

$$\begin{aligned}
 \dot{x}(t) = & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.645 & -0.52 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.43 & -1.92 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.11 & -2.39 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -4.13 & -4.13 \end{bmatrix} x(t) \\
 & + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} v(t) \\
 y(t) = & \begin{bmatrix} 0 & 0 & 0 & 0 & -5.86 & 0 & 2.93 & 0 \\ 0 & 0 & 0.524 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} x(t)
 \end{aligned} \tag{5.10}$$

The convergence curve of the PSO algorithm is given in Figure 5-14, where the MSE is equal to 0.062. Figures 5-15 and 5-16 provide a contrast between the estimated and actual model outputs.

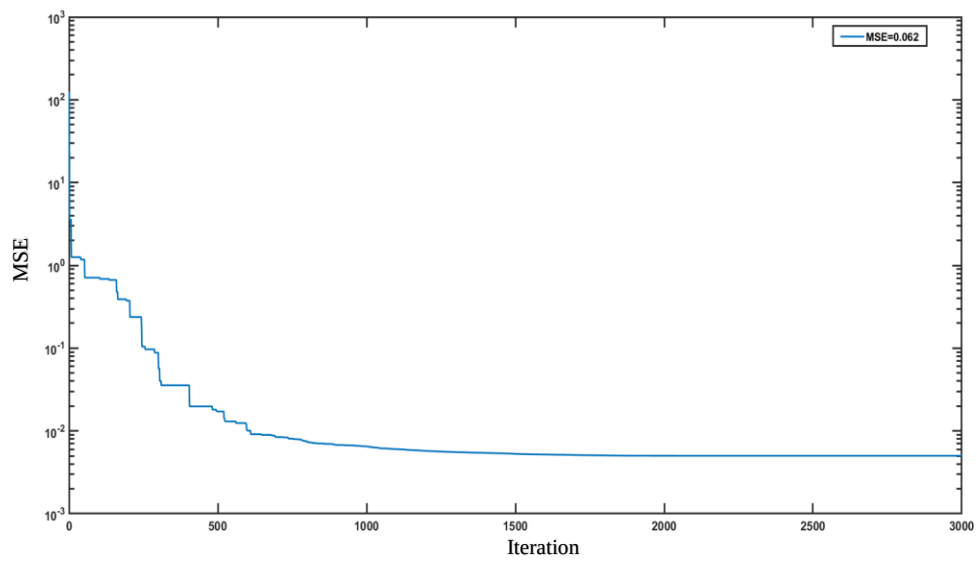


Figure 5- 14 Convergence Curve of the PSO algorithm.

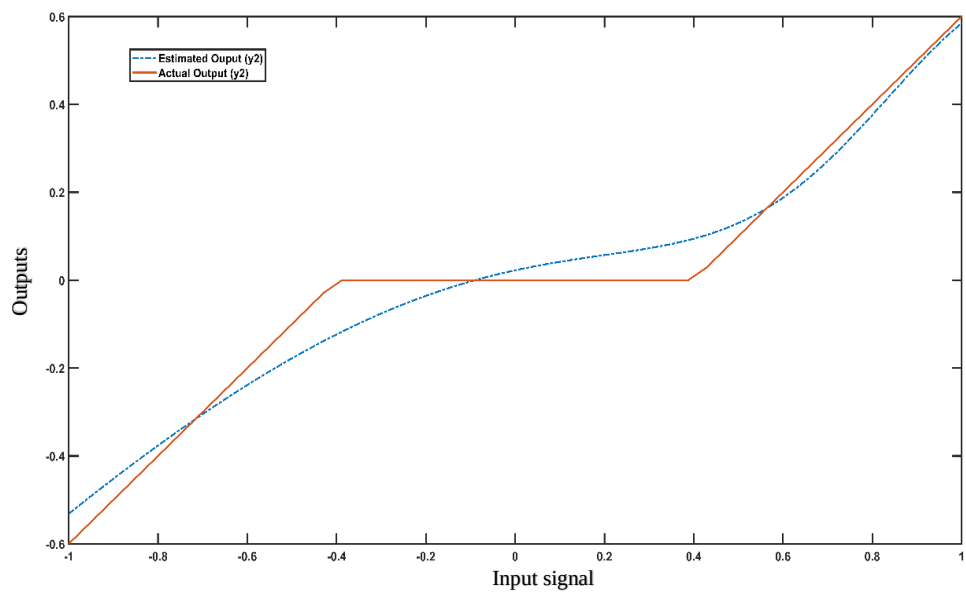


Figure 5- 15 Dead zone Nonlinearity with identified nonlinearity Using PSO.

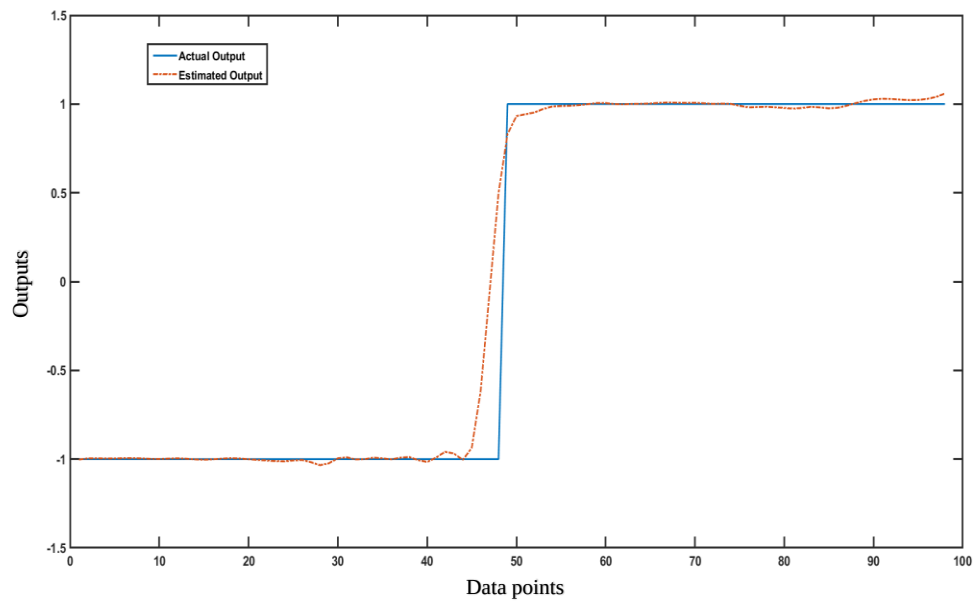


Figure 5- 16 Comparison between Actual and estimated identified nonlinear function Using PSO.

PSOGSA Simulation Results:

The same model structure and input signals used for the application of the PSO algorithm are repeated for the PSOGSA.

Parameters and eigenvalues of the identified state space system are:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.604 & -0.495 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.496 & -2.03 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -2.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -3.98 & -3.98 \end{bmatrix} x(t)$$

$$+ \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} v(t) \quad (5.11)$$

$$y(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & -6.05 & 0 & 3 & 0 \\ 0 & 0 & 0.51 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} x(t)$$

Figure 5-17 illustrates the convergence curve of the PSOGSA algorithm, where the MSE was 0.0036 within 3000 iterations. Figures 5-18 and 5-19 provide a comparison of the nonlinear component of both the Hammerstein model's estimated and actual outputs.

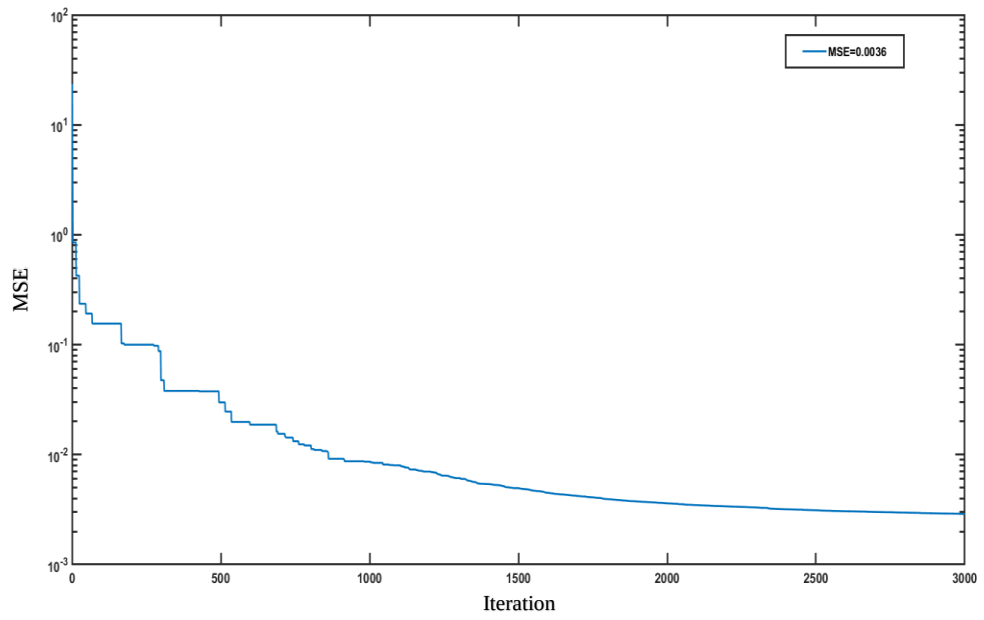


Figure 5- 17 Convergence Curve of the PSOGSA algorithm.

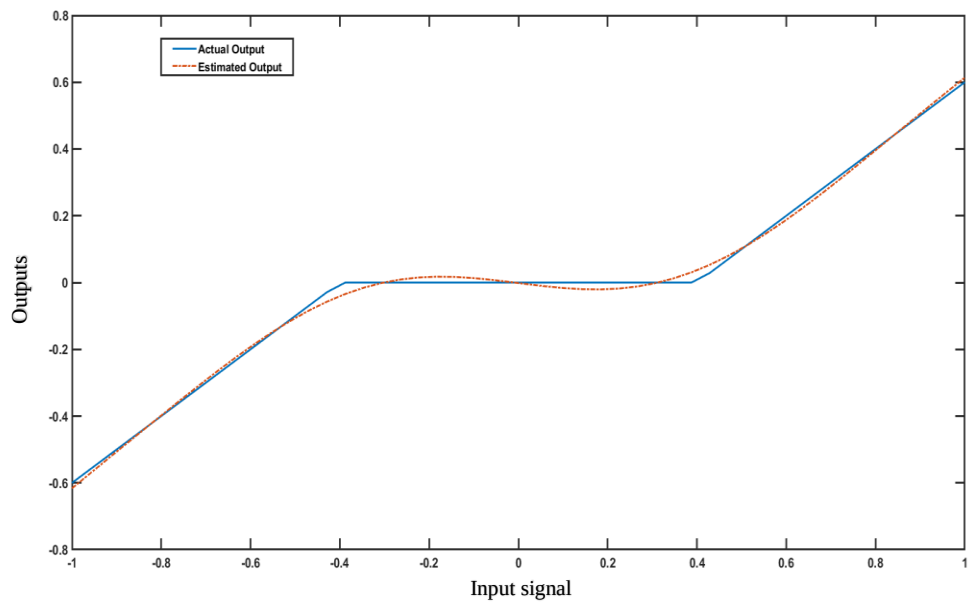


Figure 5- 18 Dead zone Nonlinearity with identified nonlinearity Using the PSOGSA algorithm.

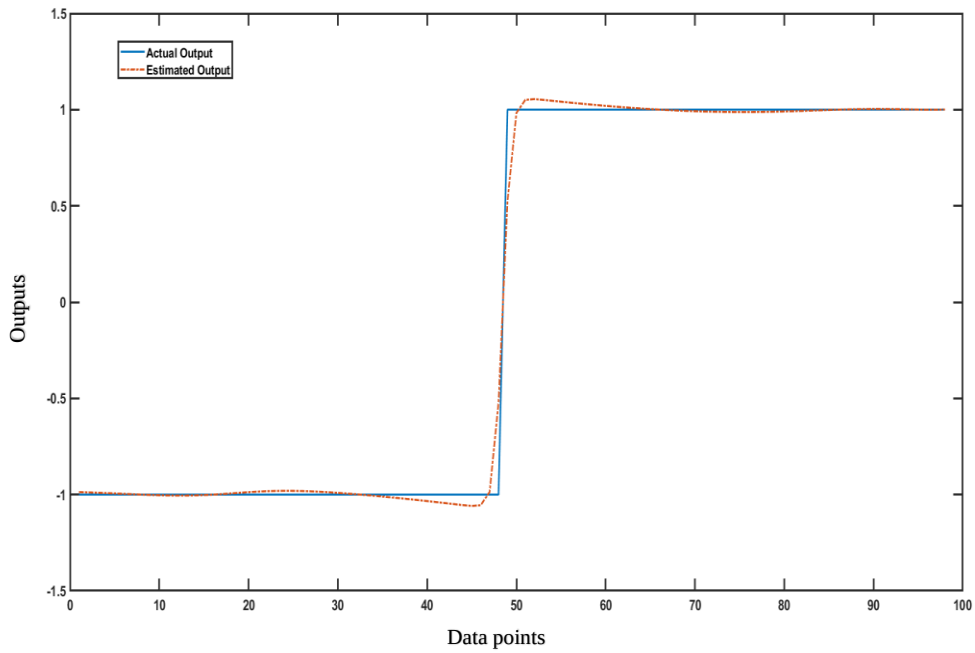


Figure 5- 19 Comparison between Actual and estimated identified nonlinear function Using the PSOGSA algorithm.

Results evaluation:

PSO and PSOGSA algorithms estimated the nonlinearities of the MIMO model with a moderately good accuracy. PSOGSA algorithm represents better results for the linear state-space model than the PSO algorithm.

In addition, MSE results clearly demonstrate that PSOGSA engenders superior estimation results than PSO.

5.2.2 Case study 4: Investigate the effect of noise

The influence of noise on the estimated model will be evaluated in this example. White Gaussian noise will be added to case study 3 to examine the robustness of the proposed system.

White noise has zero mean, so a different signal to noise ratio (SNR) will be achieved by changing the noise variances. White Gaussian noise will be added to the MIMO Hammerstein model estimated in case study 3, based on the PSOGSA algorithm. Two levels of SNR will be applied as below.

- **SNR = 30**

Based on case study 3, the nonlinear and linear parts of the Hammerstein model is given in (5.7), (5.8) and (5.9). The estimated model based on the PSOGSA algorithm with SNR =30 is given below:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.587 & -0.484 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.46 & -1.93 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.03 & -2.56 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -3.88 & -3.88 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} v(t) \quad (5.12)$$

$$y(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & -6.1 & 0 & 3.05 & 0 \\ 0 & 0 & 0.487 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} x(t)$$

The convergence curve of the identified model is given in Figure 5-20. The nonlinear part outputs response, compared to actual outputs, are illustrated in Figure 5-21 and Figure 5-22.

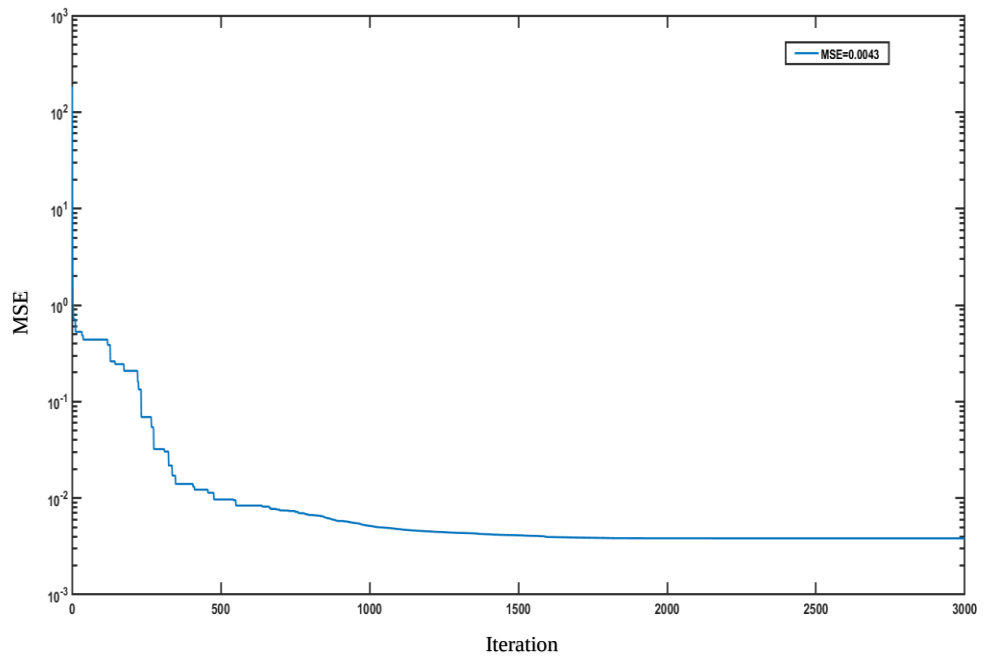


Figure 5- 20 Convergence Curve of the PSOGSA algorithm with SNR=30.

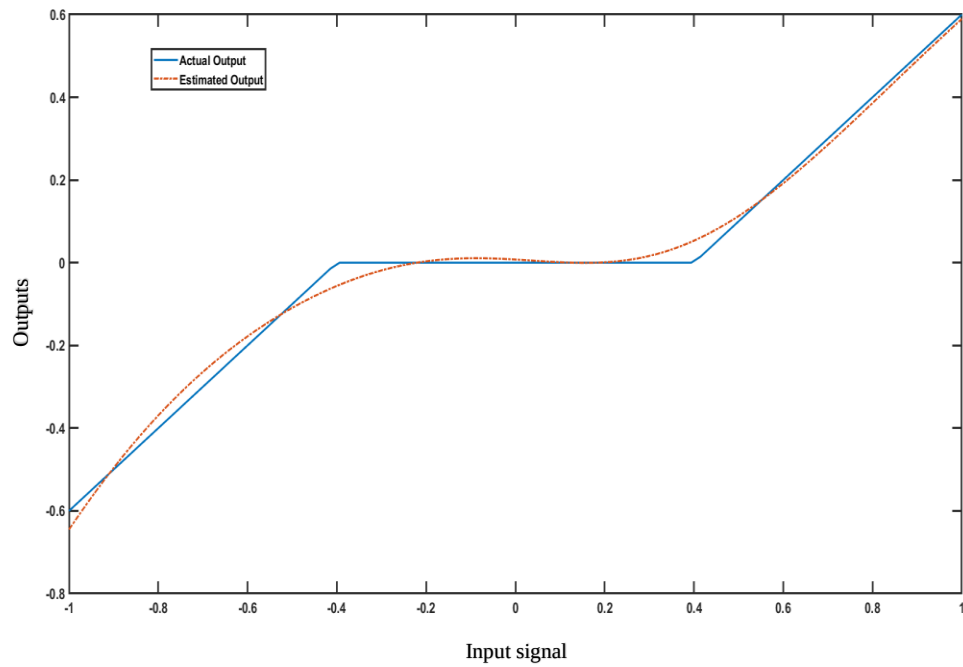


Figure 5- 21 Dead zone Nonlinearity with identified nonlinearity Using the PSOGSA algorithm with SNR=30.

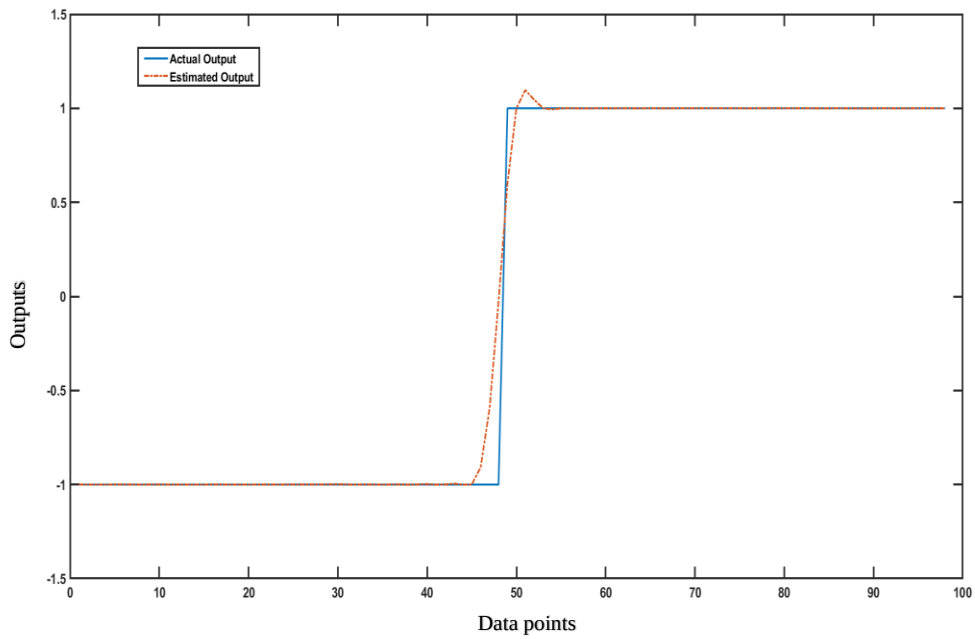


Figure 5- 22 Comparison between Actual and estimated identified nonlinear function Using the PSOGSA algorithm with SNR=30.

- SNR = 20

The estimated parameters of the linear element of the Hammerstein system with the SNR=20 is given below:

$$\begin{aligned}
\dot{x}(t) = & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.568 & -0.475 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.59 & -1.915 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -2.41 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -3.87 & -3.87 \end{bmatrix} x(t) \\
& + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} v(t)
\end{aligned} \tag{5.13}$$

$$y(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & -5.84 & 0 & 3.17 & 0 \\ 0 & 0 & 0.56 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} x(t)$$

The convergence curve of the identification algorithm is shown in 5-23. Figures 5-24 and 5-25 provide a summary of the estimated and actual outputs.

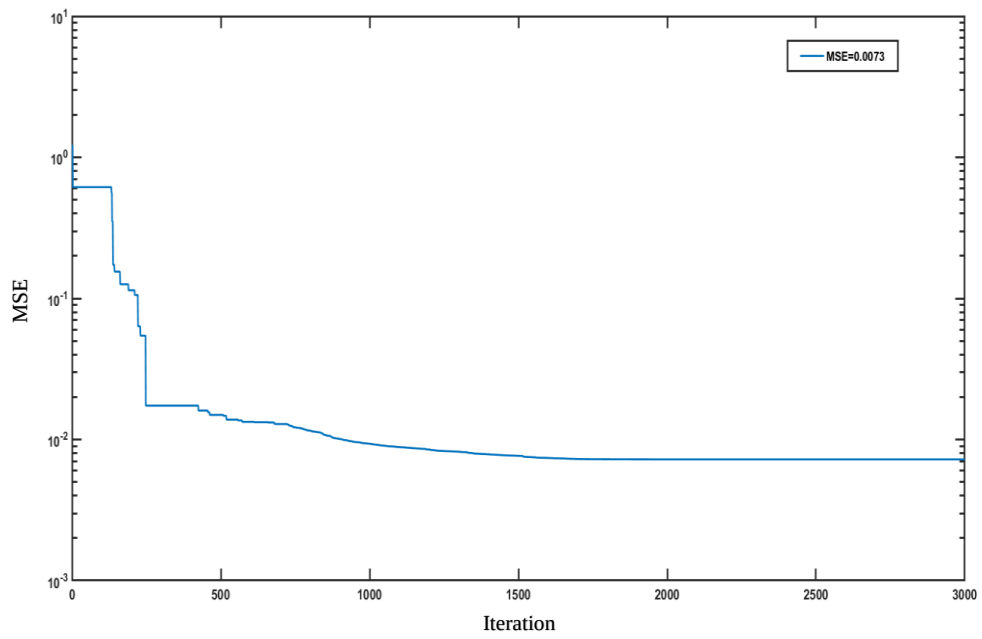


Figure 5- 23 Convergence Curve of the PSOGSA algorithm with SNR=20.

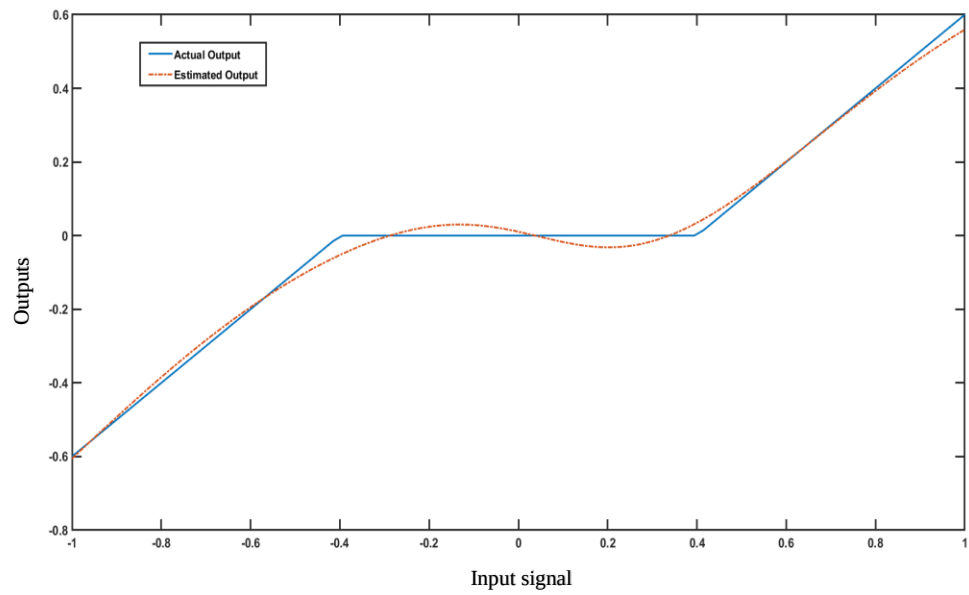


Figure 5- 24 Dead zone Nonlinearity with identified nonlinearity using the PSOGSA algorithm with SNR=20.

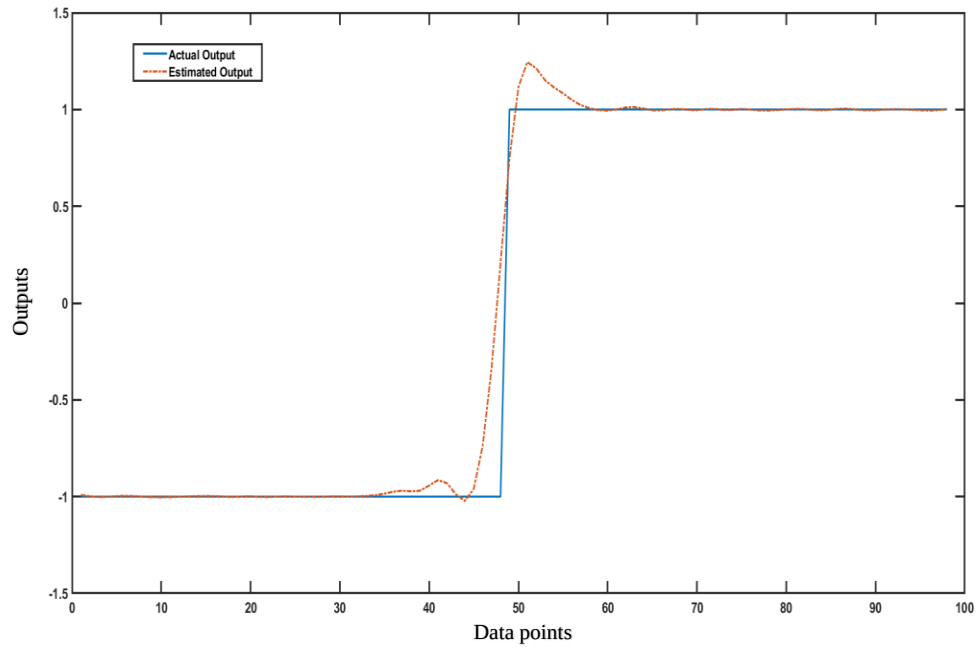


Figure 5- 25 Comparison between Actual and estimated identified nonlinear function Using the PSOGSA algorithm with SNR=20.

Previous results indicate that the PSOGSA algorithm can accurately identify the MIMO Hammerstein model in the presence of noise – suggesting that the proposed algorithm is efficient.

5.2.3 Case 5: MIMO Industrial dryer

For this example, an industrial drying process with three inputs and outputs is adopted. This dryer is used to reduce the moisture of the raw material by bringing it in direct contact with the hot gas. The efficiency of the operation is determined not only by the temperature of the hot gas, but also the rate at which it flows. Wet raw material is continuously fed to the dryer, which is dried by passing hot gas over it.

Details of industrial dryer system can be found in [105]. Figure 5-26 represents the industrial dryer process. The process inputs and outputs are given below.

The process inputs are:

- 1- u_1 : Fuel flow rate
- 2- u_2 : Hot gas exhaust fan speed.
- 3- u_3 : Rate of flow of raw material.

The Process outputs are:

- 1- y_1 : Dry bulb temperature.
- 2- y_2 : Wet-bulb temperature.
- 3- y_3 : Moisture content of the raw material.

This process' data is comprised of measurements obtained from a real industrial dryers. The fuel flow rate and the hot gas exhaust fan speed, are regulated (manipulated) using

pseudo-random binary signals. The third input was a non-manipulable external disturbance that could be calculated, and its real value recorded.

The final data contain 867 samples for each input and output, with a sampling period of 10 seconds, after a certain amount of data preprocessing that includes detrending, prefiltering, and decimation. This data was taken from the Daisy database.

Figure 5-27 and figure 5-28 represent the inputs and outputs of the industrial dryer respectively.

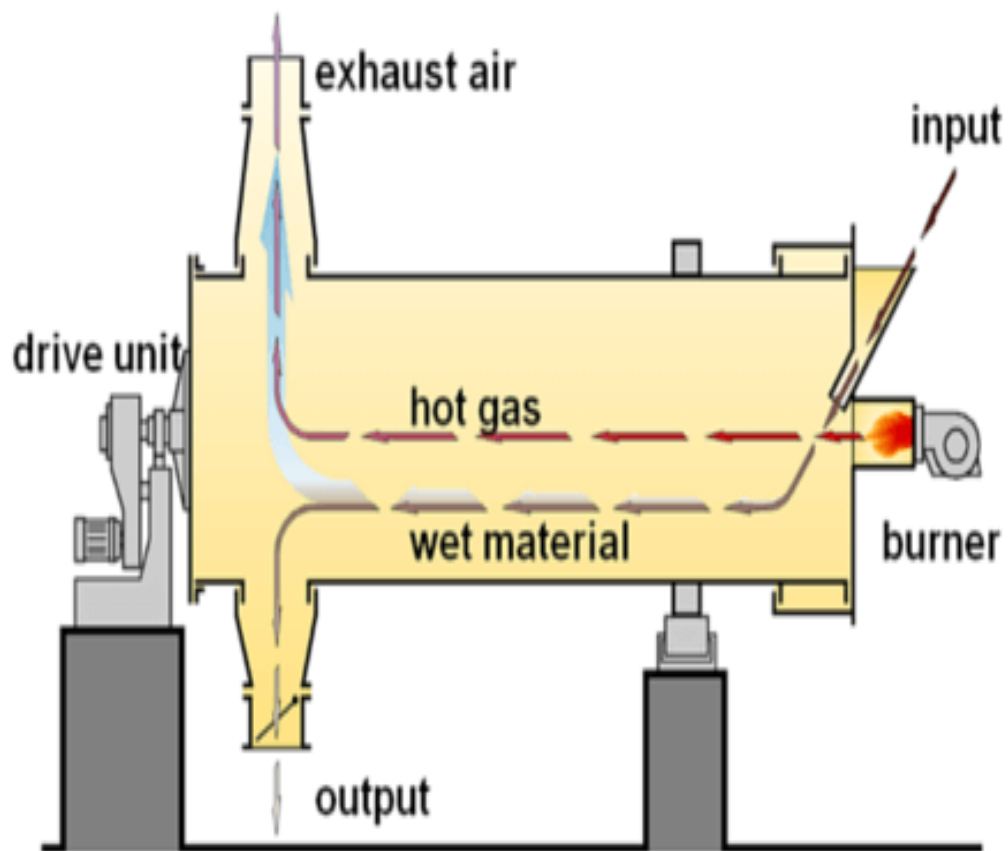


Figure 5- 26 Industrial dryer process.

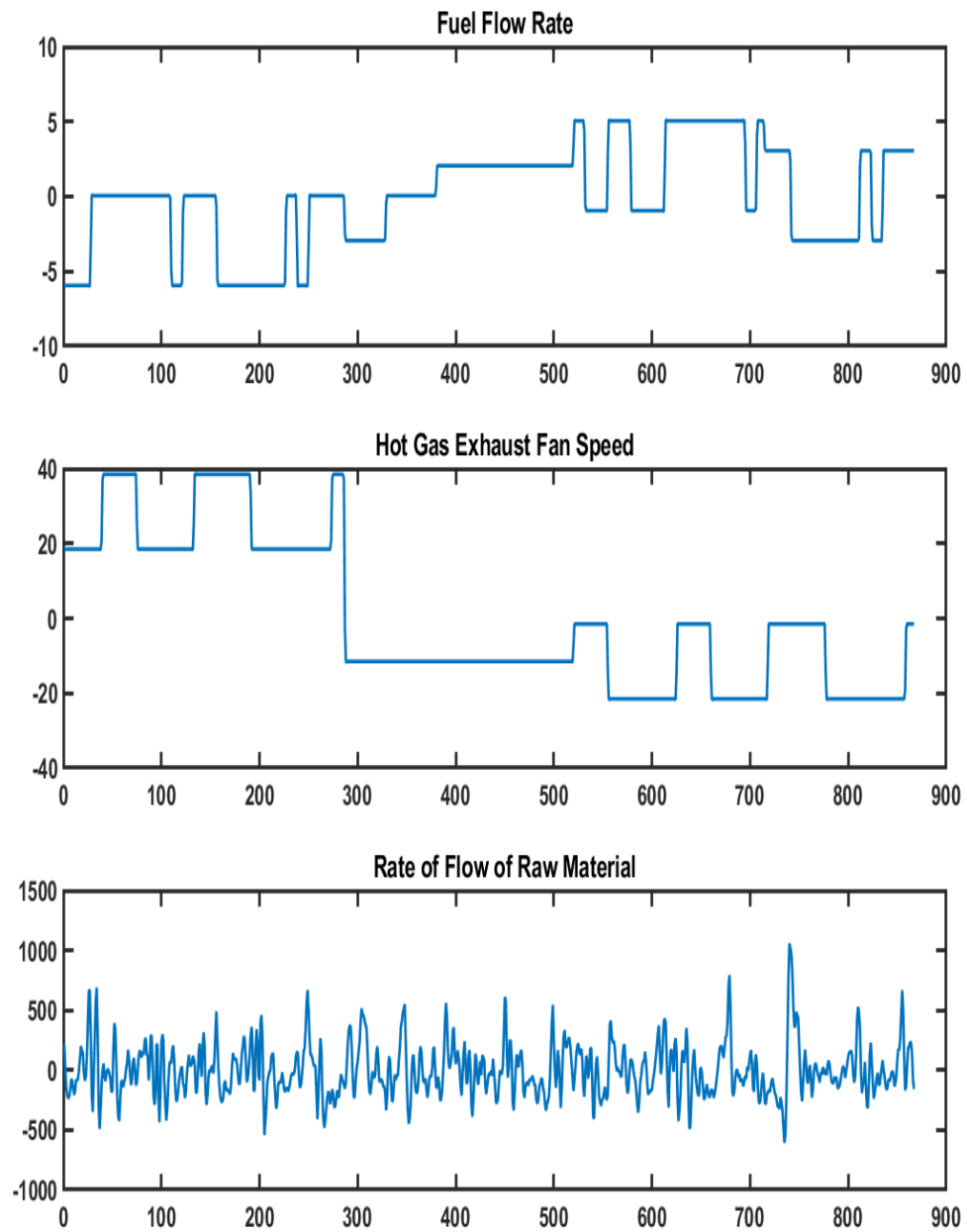


Figure 5- 27 Inputs of the Industrial Dryer.

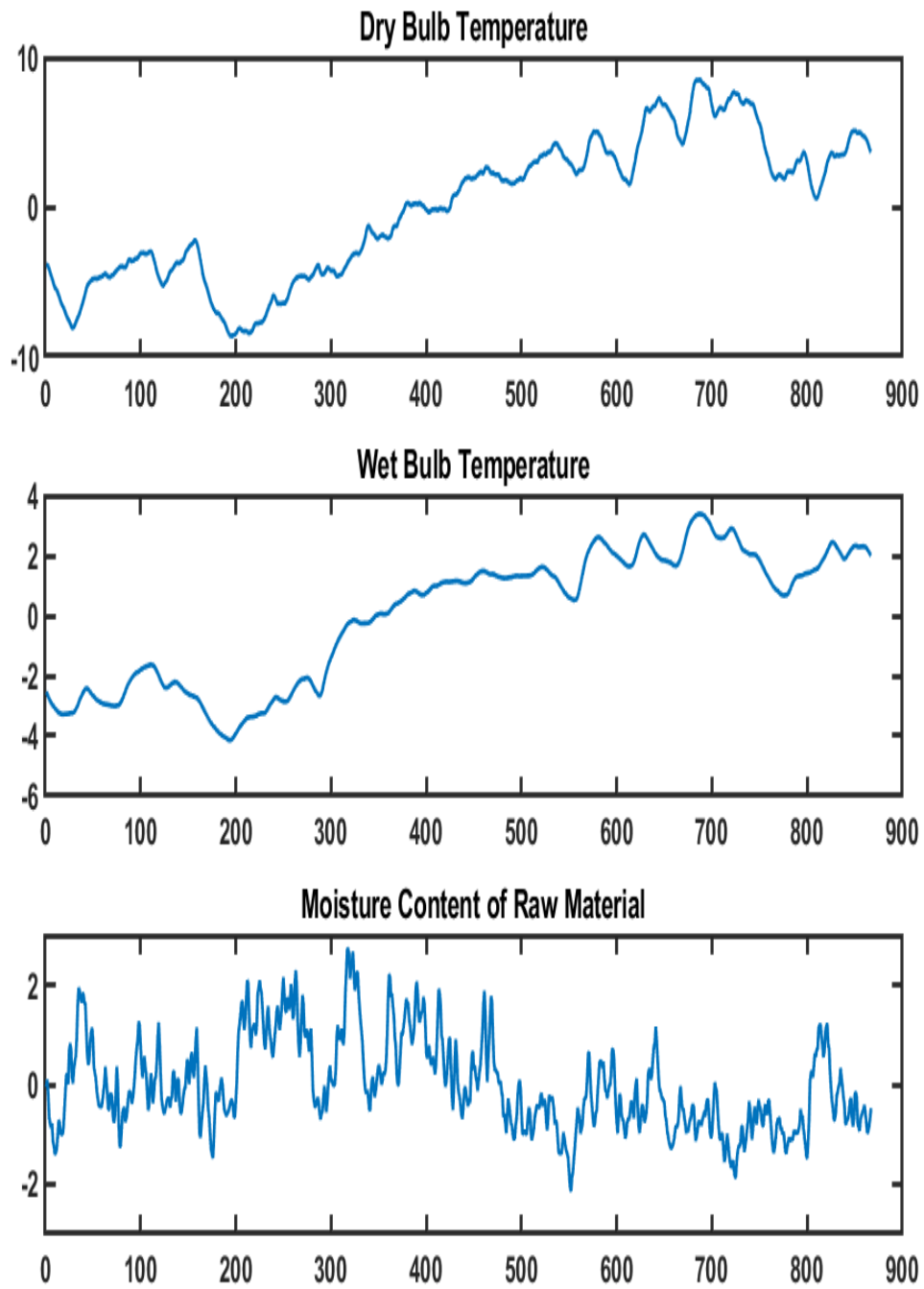


Figure 5- 28 Outputs of the Industrial Dryer.

PSO Simulation Results:

The PSO algorithm will be applied for the identification of industrial dryer. The Hammerstein model structure is applied for this process where the static nonlinearity is interconnected with linear dynamic. Static nonlinearity is represented by a multilayer feed-forward neural network and a state-space system is utilized for its linear dynamic element.

Estimated using the PSO algorithm will be the weights and biases of the feed-forward neural network that represent the nonlinear element of the Hammerstein model, as well as the parameters of the state-space model. State-space will have the seventh order, selected based on algorithm inspection. The result of identifying the state space for the linear component of the proposed Hammerstein model is seen in equation (5.13).

Model estimation with actual output simulation results for , y_1 , y_2 and y_3 are provided in Figures 5-29, 5-31 and 5-33, respectively. Additionally, nonlinearity outputs of the neural network for the estimated model are given in Figures 5-30, 5-32 and 5-34.

The convergence curve of the PSO algorithm is given in Figure 5-35. The MSE of the PSO algorithm is 0.54.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.05 & -2.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.61 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.5 & -0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2.45 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.174 & -0.97 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.29 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2.9 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.2 & -0.4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.36 & -1.4 & 0 \end{bmatrix} x(t)$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V(t) \quad (5.13)$$

$$y(t) = \begin{bmatrix} -0.38 & 0.5 & 0 & 0 & 0 & 0 & -0.96 & -0.03 & 0 & 0 & 0 & 0 & 1.36 & -0.34 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.05 & 0.58 & 0 & 0 & 0 & 0 & 0.2 & -0.75 & 0 & 0 & 0 & 0 & 0.16 & 0.097 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.43 & 0.58 & 0 & 0 & 0 & 0 & 0.85 & -0.4 & 0 & 0 & 0 & 0 & -0.29 & -1.1 \end{bmatrix} x(t)$$

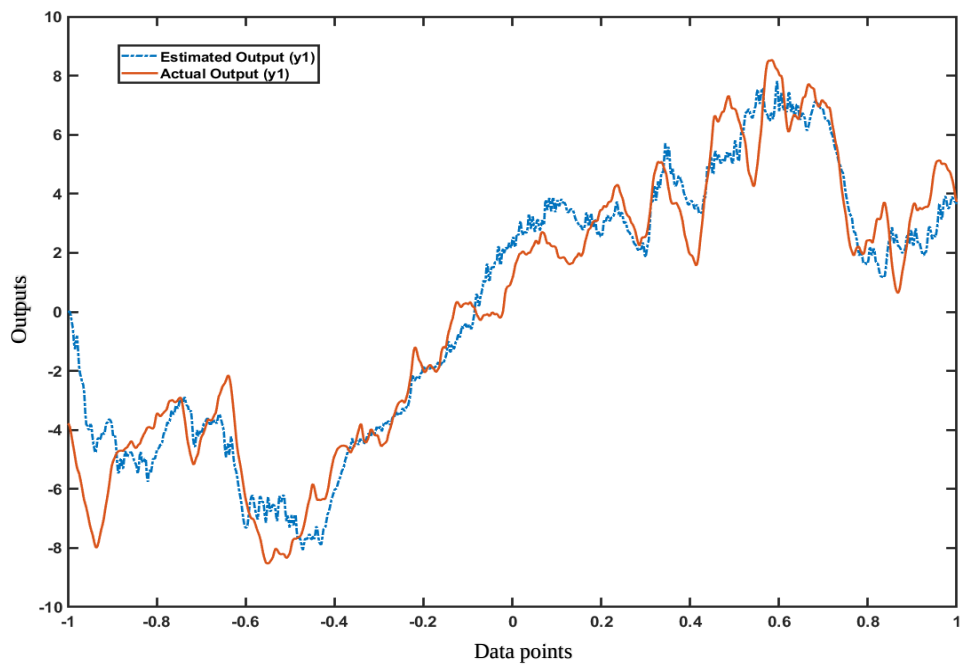


Figure 5- 29 Comparison between estimated and actual output (y_1) of the industrial dryer identified by PSO.

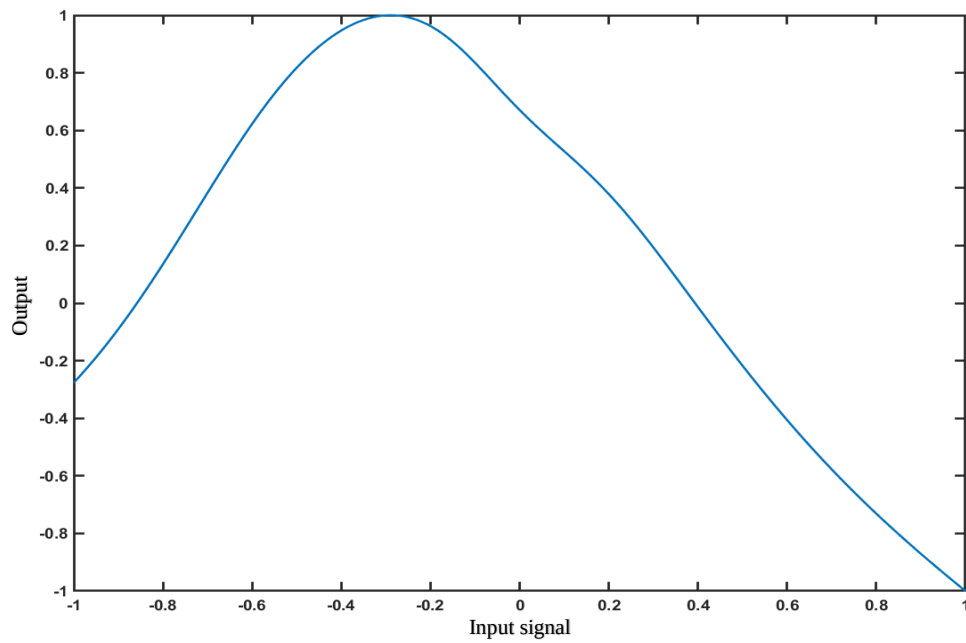


Figure 5- 30 Captured nonlinearity for output (y_1) of the industrial dryer identified by PSO.

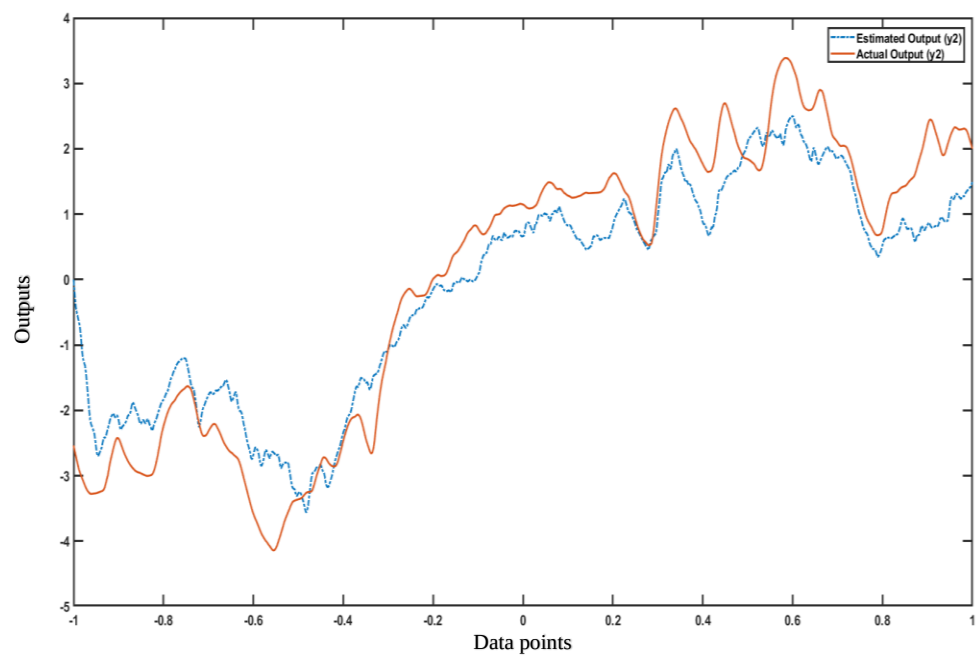


Figure 5- 31 Comparison between estimated and actual output (y_2) of the industrial dryer identified by PSO.

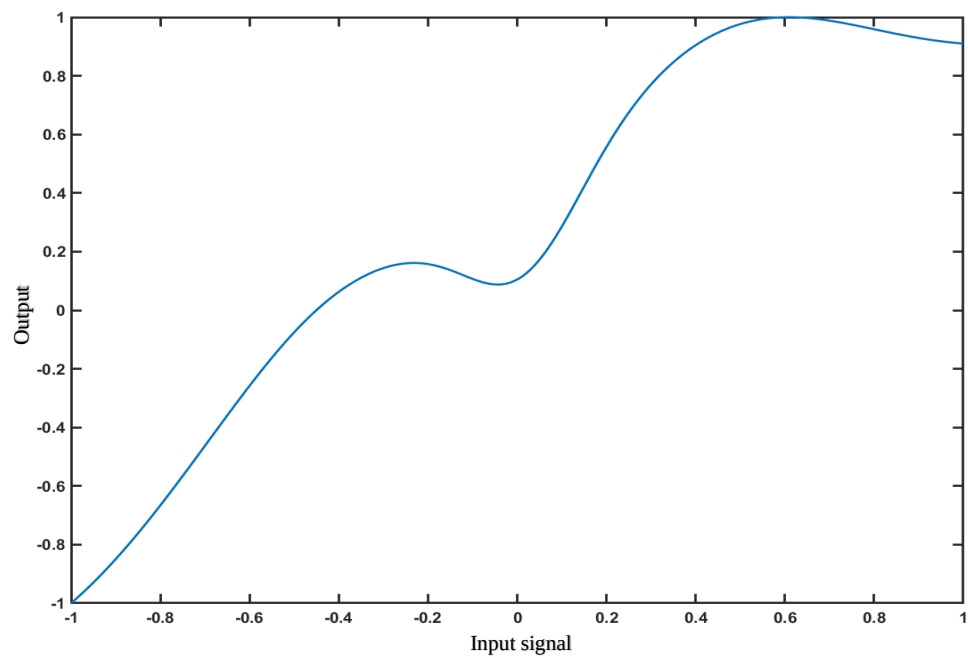


Figure 5- 32 Captured nonlinearity for output (y_2) of the industrial dryer identified by PSO.

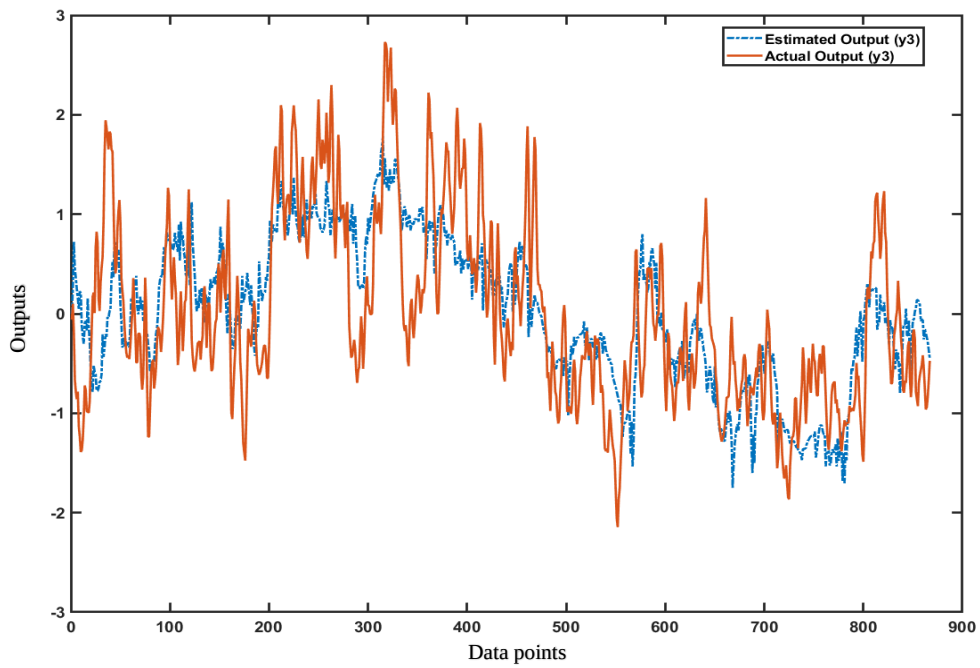


Figure 5- 33 Comparison between estimated and actual output (y_3) of the industrial dryer identified by PSO.

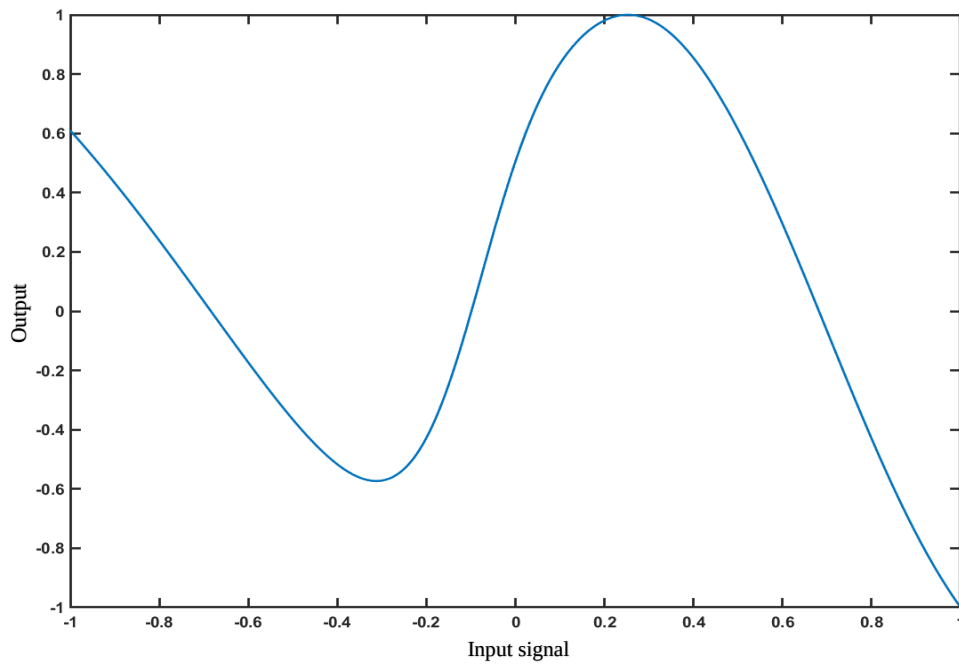


Figure 5- 34 Captured nonlinearity for output (y_3) of the industrial dryer identified by PSO.

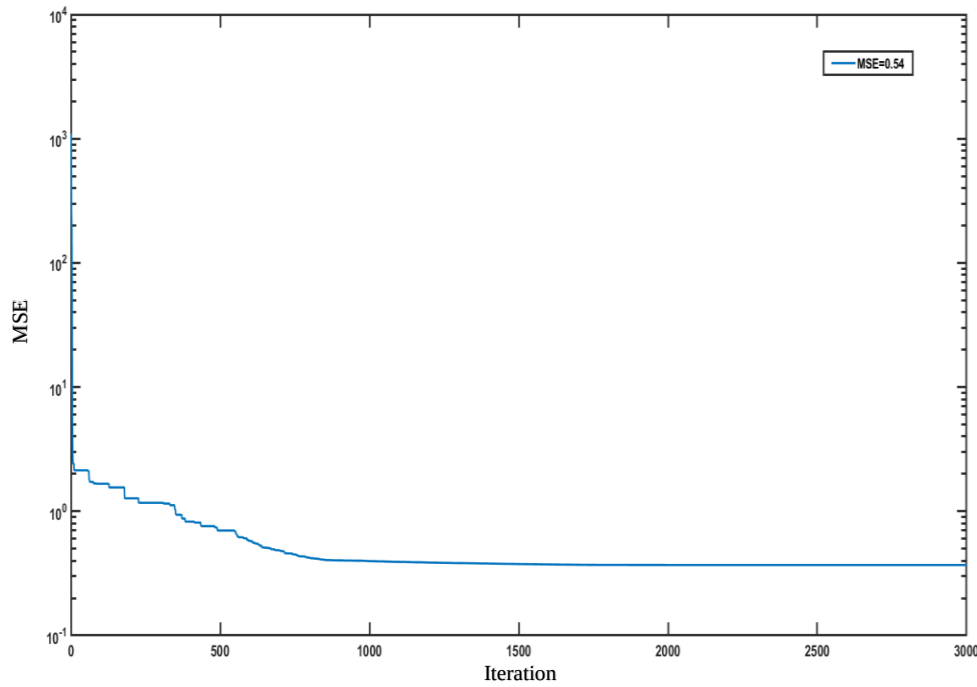


Figure 5- 35 Convergence curve of the Industrial Dryer model identified based on PSO.

PSOGSA Simulation Results:

The PSOGSA algorithm is applied for the industrial dryer identification problem where the Hammerstein system structure is used to represent the process. The Hammerstein system will have the feedforward neural network as the static nonlinear element, and the state-space system as the dynamic linear element.

Parameters of linear and nonlinear parts of the Hammerstein model will be estimated using the PSOGSA algorithm. The state-space of the estimated model has the seventh order,

selected based on algorithm inspection. Results of the estimated state-space are depicted in (5.14).

Figures 5-36, 5-38, and 5-40, representing industrial dryer estimation results based on the PSOGSA algorithm for outputs y_1 , y_2 , and y_3 , respectively. In these Figures, the actual output of the industrial dryer was simulated against the estimated output.

Moreover, the learnt nonlinearity of the neural network for the estimated model is given in Figures 5-37, 5-39, and 5-41.

The convergence curve for the MSE between actual and estimated output is given in Figure 5-42. The final MSE of the PSOGSA algorithm is 0.36.

[illegible]

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} v(t) \quad (5.14)$$

$$y(t) = \begin{bmatrix} 0.03 & 0.89 & 0 & 0 & 0 & 0 & -1.55 & 0.86 & 0 & 0 & 0 & 0 & 0.46 & -2.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.04 & -0.77 & 0 & 0 & 0 & 0 & 1 & -0.3 & 0 & 0 & 0 & 0 & -3.5 & 1.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.55 & -1.87 & 0 & 0 & 0 & 0 & 0.66 & 0.29 & 0 & 0 & 0 & 0 & 0.38 & -0.065 \end{bmatrix} x(t)$$

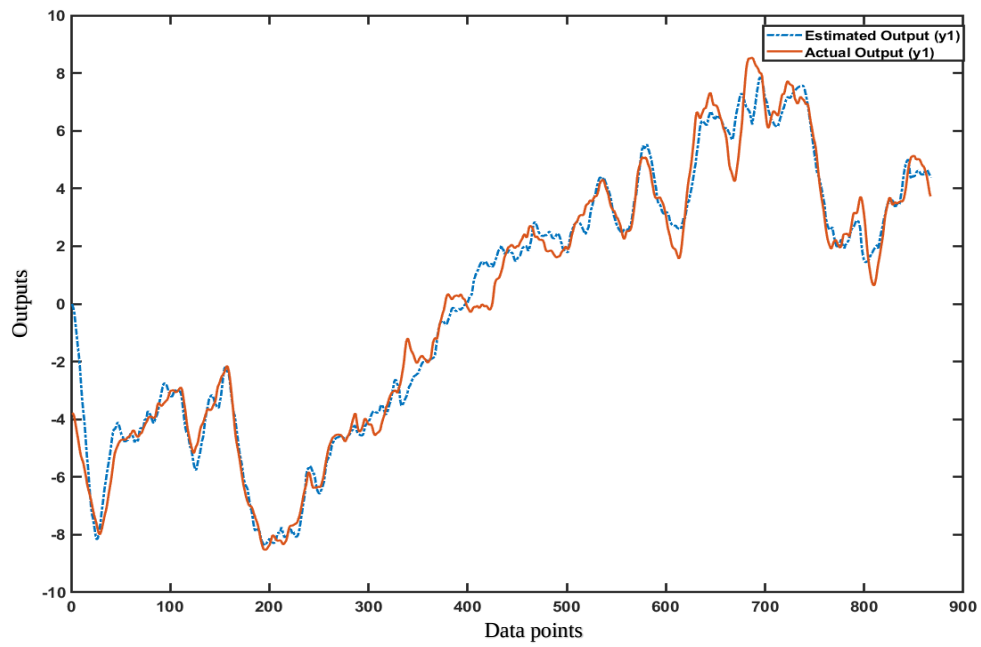


Figure 5- 36 Comparison between estimated and actual output (y_1) of the industrial dryer identified by the PSOGSA algorithm.

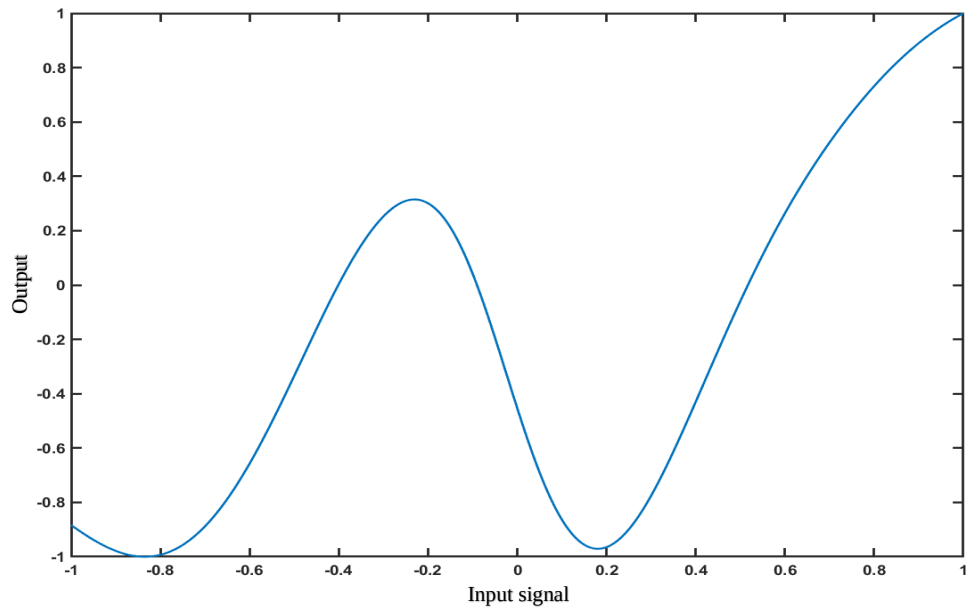


Figure 5- 37 Captured nonlinearity for output (y_1) of the industrial dryer identified by the PSOGSA algorithm.

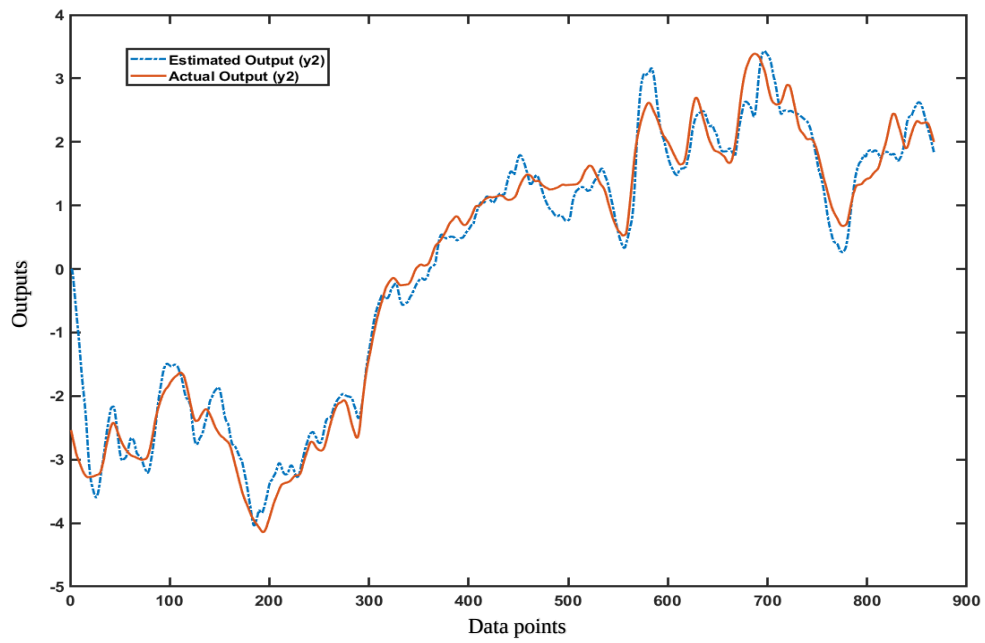


Figure 5- 38 Comparison between estimated and actual output (y_2) of the industrial dryer identified by the PSOGSA algorithm.

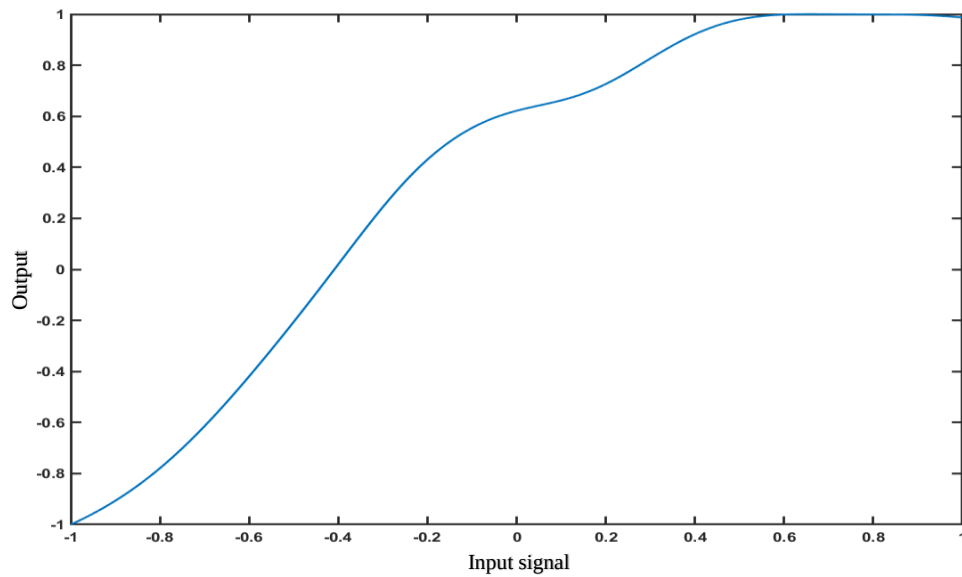


Figure 5- 39 Captured nonlinearity for output (y_2) of the industrial dryer identified by the PSOGSA algorithm.

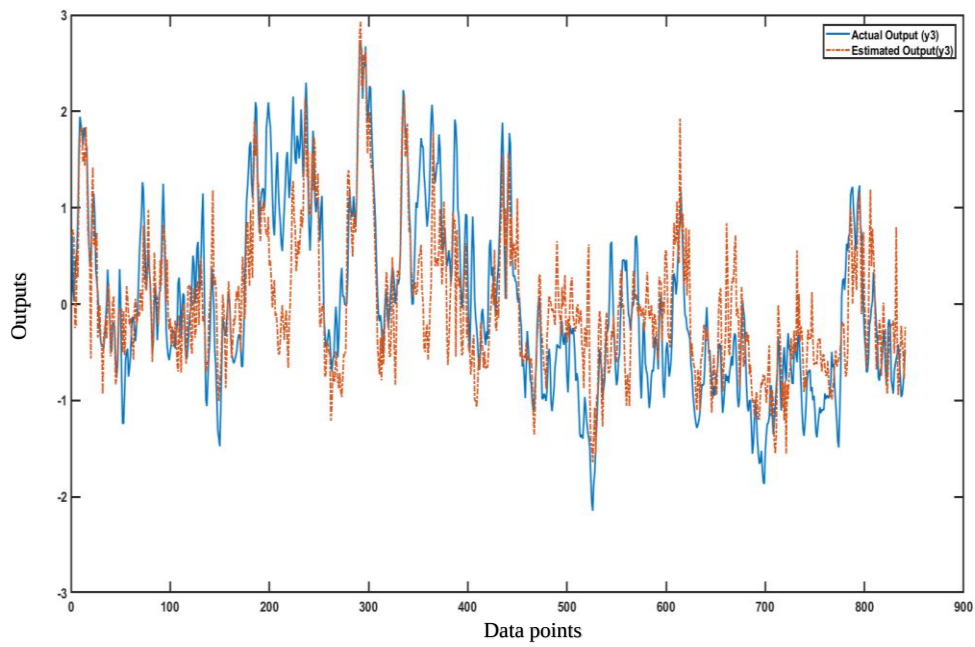


Figure 5- 40 Comparison between estimated and actual output (y_3) of the industrial dryer identified by the PSO-GSA algorithm.

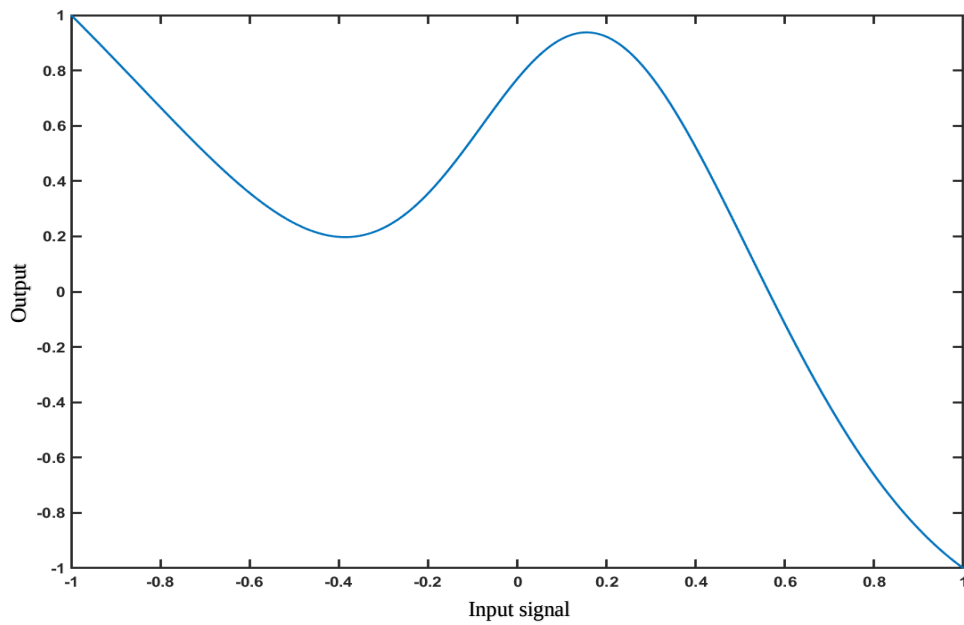


Figure 5- 41 Captured nonlinearity for output (y_3) of the industrial dryer identified by the PSO-GSA algorithm.

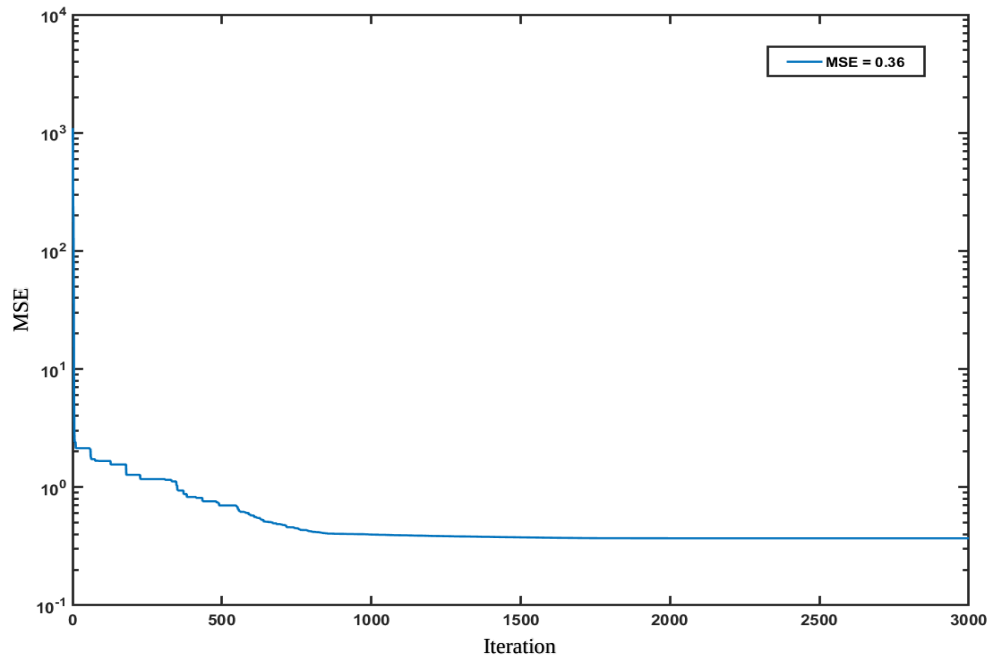


Figure 5- 42 Convergence curve of the Industrial Dryer model identified based on the PSOGSA algorithm.

Results evaluation:

The two algorithms were successful in estimating the industrial dryer model, though PSOGSA algorithm has faster convergence and a superior result in terms of MSE. Moreover, PSOGSA algorithm outperformed PSO algorithm in capturing the nonlinearity of the industrial dryer model. Simulations results confirm the supremacy of the PSOGSA algorithm over the PSO algorithm.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

Introduced within this thesis is a novel algorithm for the identification of the MIMO Hammerstein model. This model is constructed by the interconnection between a static nonlinear component with a linear dynamic component. The static nonlinear component is presented by a multilayer feed-forward neural network and the state-space model is used for the linear dynamic component.

The Hammerstein identification problem was proposed as an optimization problem allowing the use of a variety of optimization algorithms. PSO and PSOGSA algorithms are applied for identification of the SISO and MIMO Hammerstein system – these are applied to estimate the weights and biases of the multilayer feed-forward neural network, as well as parameters of the state space system.

The state space model represented in the controllable canonical form is utilized for the linear component of the Hammerstein model for both SISO and MIMO systems. This reconfiguration is implemented to minimize number of parameters to be estimated.

Various Hammerstein model identification examples are provided in this analysis to ensure the efficacy of the proposed algorithms.

For SISO Hammerstein model identification, known nonlinearity is applied to assess the quality of the proposed algorithms. Subsequently, the coupled electric drive example is used to substantiate its applicability to real-life problems.

Conversely, the MIMO Hammerstein model identification problem has been tackled in this work. The efficiency of the PSOGSA algorithm is investigated for training the neural network and estimating state-space parameters in comparison with the traditional PSO algorithm.

To evaluate the robustness of these algorithms, two examples are used: The first employs known nonlinearities (Dead zone and relay nonlinearities) to gauge the efficiencies of the identification techniques. After estimating the known nonlinearities model, the effect of noise was studied. White noise (Gaussian noise) with different signal to noise ratio (SNR) was added to the data. This example is used to evaluate the robustness of the proposed identification algorithms against the effect of noise.

Furthermore, an actual industrial process example is employed with the MIMO Hammerstein model identification. The industrial dryer system is identified using the proposed algorithm. The industrial dryer process contains data with many disturbances, and both a high nonlinear feature and high dynamic order. The proposed algorithms have good

identification results which prove the effectiveness and efficiency of these algorithms in dealing with practical real-life systems.

6.2Future Work

This research on the Hammerstein model identification can be extended by the following points:

- 1- Proposed algorithms could be applied to different industrial processes and real-life examples.
- 2- Other heuristic optimization algorithms like the grey wolf optimizer, the whale optimization algorithm, the ant lion optimizer and others, could be applied for Hammerstein model identification.
- 3- Linear and nonlinear components of the Hammerstein model could be replaced with different systems, potentially leading to a new identification algorithm of the Hammerstein model.

References

- [1] Nelles, O. (2002). Nonlinear system identification.
- [2] Tangirala, A. K. (2018). Principles of system identification: theory and practice. Crc Press.
- [3] Ljung, L. (1999). System identification.
- [4] Gevers, M. (2006). A personal view of the development of system identification: A 30-year journey through an exciting field. IEEE Control systems magazine, 26(6), 93-105.
- [5] Kolodziej, J. R., & Mook, D. J. (2011). Model determination for nonlinear state-based system identification. Nonlinear Dynamics, 63(4), 735-753.
- [6] Bai, E. W. (2010). Non-parametric nonlinear system identification: An asymptotic minimum mean squared error estimator. IEEE Transactions on automatic control, 55(7), 1615-1626.
- [7] Giri, F., & Bai, E. W. (Eds.). (2010). Block-oriented nonlinear system identification (Vol. 1, pp. 0278-0046). London: springer.
- [8] Lai, C. Y., Xiang, C., & Lee, T. H. (2011). Data-based identification and control of nonlinear systems via piecewise affine approximation. IEEE transactions on neural networks, 22(12), 2189-2200.

- [9] Zhao, Y., Westwick, D. T., & Kearney, R. E. (2010). Subspace methods for identification of human ankle joint stiffness. *IEEE transactions on biomedical engineering*, 58(11), 3039-3048.
- [10] Westwick, D. T., & Perreault, E. J. (2010). Closed-loop identification: application to the estimation of limb impedance in a compliant environment. *IEEE Transactions on Biomedical Engineering*, 58(3), 521-530.
- [11] Kerschen, G., Worden, K., Vakakis, A. F., & Golinval, J. C. (2006). Past, present and future of nonlinear system identification in structural dynamics. *Mechanical systems and signal processing*, 20(3), 505-592.
- [12] Yassin, I. M., Taib, M. N., & Adnan, R. (2013). Recent advancements & methodologies in system identification: A review. *Scientific Research Journal*, 1(1), 14-33.
- [13] Chen, S., Billings, S. A., & Luo, W. (1989). Orthogonal least squares methods and their application to non-linear system identification. *International Journal of control*, 50(5), 1873-1896.
- [14] Piroddi, L., & Spinelli, W. (2003). An identification algorithm for polynomial NARX models based on simulation error minimization. *International Journal of Control*, 76(17), 1767-1781.
- [15] Ogunfunmi, T. (2007). *Adaptive nonlinear system identification: The Volterra and Wiener model approaches*. Springer Science & Business Media.

- [16] Murray-Smith, R., & Johansen, T. (Eds.). (2020). Multiple model approaches to nonlinear modelling and control. CRC press.
- [17] Brown, M., & Harris, C. J. (1994). Neurofuzzy adaptive modelling and control.
- [18] Mzyk, G. (2014). Combined parametric-nonparametric identification of block-oriented systems (Vol. 238). Berlin: Springer.
- [19] Billings, S. A., & Fakhouri, S. Y. (1982). Identification of systems containing linear dynamic and static nonlinear elements. *Automatica*, 18(1), 15-26.
- [20] Hammerstein, A. (1930). Nichtlineare integralgleichungen nebst anwendungen. *Acta mathematica*, 54, 117-176.
- [21] Wiener, N. (1958). *Nonlinear Problems in Random Theory*, Tech. Press MIT and John Willey, New York.
- [22] Jurado, F. (2006). A method for the identification of solid oxide fuel cells using a Hammerstein model. *Journal of Power Sources*, 154(1), 145-152.
- [23] Lakshminarayanan, S., Shah, S. L., & Nandakumar, K. (1995). Identification of Hammerstein models using multivariate statistical tools. *Chemical Engineering Science*, 50(22), 3599-3613.
- [24] Wang, J., & Zhang, Q. (2014). Detection of asymmetric control valve stiction from oscillatory data using an extended Hammerstein system identification method. *Journal of Process Control*, 24(1), 1-12.

- [25] Kim, J., & Konstantinou, K. (2001). Digital predistortion of wideband signals based on power amplifier model with memory. *Electronics Letters*, 37(23), 1417-1418.
- [26] Van der Veen, G., van Wingerden, J. W., & Verhaegen, M. (2012). Global identification of wind turbines using a Hammerstein identification method. *IEEE transactions on control systems technology*, 21(4), 1471-1478.
- [27] Balestrino, A., Landi, A., Ould-Zmirli, M., & Sani, L. (2001). Automatic nonlinear auto-tuning method for Hammerstein modeling of electrical drives. *IEEE Transactions on Industrial Electronics*, 48(3), 645-655.
- [28] Rollins, D. K., Bhandari, N., Bassily, A. M., Colver, G. M., & Chin, S. T. (2003). A continuous-time nonlinear dynamic predictive modeling method for Hammerstein processes. *Industrial & engineering chemistry research*, 42(4), 860-872.
- [29] Stapleton, J., & Bass, S. (1985). Adaptive noise cancellation for a class of nonlinear, dynamic reference channels. *IEEE transactions on circuits and systems*, 32(2), 143-150.
- [30] Mkadem, F., & Boumaiza, S. (2009). Extended Hammerstein behavioral model using artificial neural networks. *IEEE Transactions on Microwave Theory and Techniques*, 57(4), 745-751.
- [31] Biagiola, S. I., & Figueroa, J. L. (2011). Identification of uncertain MIMO Wiener and Hammerstein models. *Computers & chemical engineering*, 35(12), 2867-2875.

- [32] Dempsey, E. J., & Westwick, D. T. (2004). Identification of Hammerstein models with cubic spline nonlinearities. *IEEE Transactions on Biomedical Engineering*, 51(2), 237-245.
- [33] Naeem, O., & Huesman, A. E. M. (2011). Non-linear model approximation and reduction by new input-state Hammerstein block structure. *Computers & chemical engineering*, 35(5), 758-773.
- [34] Hunt, K. J., Munih, M., Donaldson, N. D. N., & Barr, F. M. (1998). Investigation of the Hammerstein hypothesis in the modeling of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, 45(8), 998-1009.
- [35] Capobianco, E. (2002). Hammerstein system representation of financial volatility processes. *The European Physical Journal B-Condensed Matter and Complex Systems*, 27(2), 201-211.
- [36] Narendra, K., & Gallman, P. (1966). An iterative method for the identification of nonlinear systems using a Hammerstein model. *IEEE Transactions on Automatic control*, 11(3), 546-550.
- [37] Ding, F., Liu, X. P., & Liu, G. (2011). Identification methods for Hammerstein nonlinear systems. *Digital Signal Processing*, 21(2), 215-238.
- [38] Voros, J. (1999). Iterative algorithm for parameter identification of Hammerstein systems with two-segment nonlinearities. *IEEE transactions on Automatic Control*, 44(11), 2145-2149.

- [39] Shen, Q., & Ding, F. (2014). Iterative estimation methods for Hammerstein controlled autoregressive moving average systems based on the key-term separation principle. *Nonlinear Dynamics*, 75(4), 709-716.
- [40] Ding, F., Liu, X., & Chu, J. (2013). Gradient-based and least-squares-based iterative algorithms for Hammerstein systems using the hierarchical identification principle. *IET Control Theory & Applications*, 7(2), 176-184.
- [41] Mao, Y., & Ding, F. (2015). Multi-innovation stochastic gradient identification for Hammerstein controlled autoregressive autoregressive systems based on the filtering technique. *Nonlinear Dynamics*, 79(3), 1745-1755.
- [42] Mao, Y., & Ding, F. (2015). Multi-innovation stochastic gradient identification for Hammerstein controlled autoregressive autoregressive systems based on the filtering technique. *Nonlinear Dynamics*, 79(3), 1745-1755.
- [43] Vörös, J. (2011). Identification of nonlinear cascade systems with time-varying backlash. *Journal of Electrical Engineering*, 62(2), 87.
- [44] Voros, J. (2002). Modeling and parameter identification of systems with multisegment piecewise-linear characteristics. *IEEE Transactions on Automatic control*, 47(1), 184-188.
- [45] Back, A. D., & Tsoi, A. C. (1992). Nonlinear system identification using multilayer perceptrons with locally recurrent synaptic structure. In *Proceedings of 1992 IEEE-SP Workshop on Neural Networks for Signal Processing II* (pp. 444-453).

- [46] H. AL-DUWAISH, M. NAZMUL KARIM, and V. CHANDRASEKAR , (Jan. 1997) “Hammerstein model identification by multilayer feedforward neural networks,” *Int. J. Syst. Sci.*, vol. 28, no. 1, pp. 49–54.
- [47] H. Alduwaish and M. N. Karim, (1997) “A new method for the identification of hammerstein model,” *Automatica*, vol. 33, no. 10, pp. 1871–1875.
- [48] Hachino, T., Deguchi, K., & Takata, H. (2004, July). Identification of Hammerstein model using radial basis function networks and genetic algorithm. In 2004 5th Asian Control Conference (IEEE Cat. No. 04EX904) (Vol. 1, pp. 124-129). IEEE.
- [49] Han, Y., & de Callafon, R. A. (2011). Closed-loop identification of Hammerstein systems using iterative instrumental variables. *IFAC Proceedings Volumes*, 44(1), 13930-13935.
- [50] Bai, E. W. (1998). An optimal two-stage identification algorithm for Hammerstein–Wiener nonlinear systems. *Automatica*, 34(3), 333-338.
- [51] Falck, T., Suykens, J. A., Schoukens, J., & De Moor, B. (2010, December). Nuclear norm regularization for overparametrized Hammerstein systems. In 49th IEEE Conference on Decision and Control (CDC) (pp. 7202-7207). IEEE.
- [52] Risuleo, R. S., Bottegal, G., & Hjalmarsson, H. (2015, December). A new kernel-based approach to overparameterized Hammerstein system identification. In 2015 54th IEEE Conference on Decision and Control (CDC) (pp. 115-120). IEEE.

- [53] Vanbeylen, L., Pintelon, R., & Schoukens, J. (2008). Blind maximum likelihood identification of Hammerstein systems. *Automatica*, 44(12), 3139-3146.
- [54] Bai, E. W. (2002). A blind approach to the Hammerstein–Wiener model identification. *Automatica*, 38(6), 967-979.
- [55] Verhaegen, M., & Westwick, D. (1996). Identifying MIMO Hammerstein systems in the context of subspace model identification methods. *IFAC Proceedings Volumes*, 29(1), 4092-4097.
- [56] Goethals, I., Pelckmans, K., Suykens, J. A., & De Moor, B. (2005). Subspace identification of Hammerstein systems using least squares support vector machines. *IEEE Transactions on Automatic Control*, 50(10), 1509-1519.
- [57] Goethals, I., Pelckmans, K., Suykens, J. A., & De Moor, B. (2005). Identification of MIMO Hammerstein models using least squares support vector machines. *Automatica*, 41(7), 1263-1272.
- [58] Krzyżak, A., & Partyka, M. A. (2005). Global identification of nonlinear Hammerstein systems by recursive kernel approach. *Nonlinear Analysis: Theory, Methods & Applications*, 63(5-7), e1263-e1272.
- [59] Greblicki, W. (2000). Continuous-time Hammerstein system identification. *IEEE Transactions on Automatic Control*, 45(6), 1232-1236.
- [60] Billings, S. A., & Fakhouri, S. Y. (1978, July). Identification of a class of nonlinear systems using correlation analysis. In *Proceedings of the Institution of Electrical*

Engineers (Vol. 125, No. 7, pp. 691-697). IET Digital Library.

- [61] Risuleo, R. S., Bottegal, G., & Hjalmarsson, H. (2017). A nonparametric kernel-based approach to Hammerstein system identification. *Automatica*, 85, 234-247.
- [62] Lauwers, L., Schoukens, J., Pintelon, R., & Enqvist, M. (2008). A nonlinear block structure identification procedure using frequency response function measurements. *IEEE Transactions on Instrumentation and Measurement*, 57(10), 2257-2264.”
- [63] Giri, F., Ahmed-Ali, T., Brouri, A., Chaoui, F. Z., & Radouane, A. (2015). Frequency identification of a feedback Hammerstein system. *IFAC-PapersOnLine*, 48(28), 945-950.
- [64] Pintelon, R., & Schoukens, J. (2012). *System identification: a frequency domain approach*. John Wiley & Sons.
- [65] Crama, P., Schoukens, J., & Pintelon, R. (2004). Generation of enhanced initial estimates for Hammerstein systems. *Automatica*, 40(7), 1269-1273.
- [66] Hasiewicz, Z., Mzyk, G., Śliwiński, P., & Wachel, P. (2012). Mixed parametric-nonparametric identification of Hammerstein and Wiener systems-a survey. *IFAC Proceedings Volumes*, 45(16), 464-469.
- [67] Sung, S. W. (2002). System identification method for Hammerstein processes. *Industrial & engineering chemistry research*, 41(17), 4295-4302.

- [68] Cui, M., Liu, H., Li, Z., Tang, Y., & Guan, X. (2014). Identification of Hammerstein model using functional link artificial neural network. *Neurocomputing*, 142, 419-428.
- [69] Greblicki, W., & Pawlak, M. (2017). Hammerstein system identification with the nearest neighbor algorithm. *IEEE Transactions on Information Theory*, 63(8), 4746-4757.
- [70] Lin, W., & Liu, P. X. (2006). Hammerstein model identification based on bacterial foraging. *Electronics Letters*, 42(23), 1332-1333.
- [71] Jia, L., Li, X., & Chiu, M. S. (2016). Correlation analysis based MIMO neuro-fuzzy Hammerstein model with noises. *Journal of Process Control*, 41, 76-91.
- [72] Jia, L., Li, X., & Chiu, M. S. (2016). The identification of neuro-fuzzy based MIMO Hammerstein model with separable input signals. *Neurocomputing*, 174, 530-541.
- [73] Li, F., Jia, L., Peng, D., & Han, C. (2017). Neuro-fuzzy based identification method for Hammerstein output error model with colored noise. *Neurocomputing*, 244, 90-101.
- [74] Gotmare, A., Patidar, R., & George, N. V. (2015). Nonlinear system identification using a cuckoo search optimized adaptive Hammerstein model. *Expert systems with applications*, 42(5), 2538-2546.
- [75] Mattsson, P., & Wigren, T. (2016). Convergence analysis for recursive Hammerstein identification. *Automatica*, 71, 179-186.

- [76] Tang, Y., Li, Z., & Guan, X. (2014). Identification of nonlinear system using extreme learning machine based Hammerstein model. *Communications in Nonlinear Science and Numerical Simulation*, 19(9), 3171-3183.
- [77] Wang, D., & Ding, F. (2016). Parameter estimation algorithms for multivariable Hammerstein CARMA systems. *Information Sciences*, 355, 237-248.
- [78] Chaudhary, N. I., & Raja, M. A. Z. (2015). Identification of Hammerstein nonlinear ARMAX systems using nonlinear adaptive algorithms. *Nonlinear Dynamics*, 79(2), 1385-1397.
- [79] Nanda, S. J., Panda, G., & Majhi, B. (2010). Improved identification of Hammerstein plants using new CPSO and IPSO algorithms. *Expert systems with applications*, 37(10), 6818-6831.
- [80] Hong, X., & Chen, S. (2012). The system identification and control of Hammerstein system using non-uniform rational B-spline neural network and particle swarm optimization. *Neurocomputing*, 82, 216-223.
- [81] Al-Duwaish, H. N. (2011). Identification of Hammerstein models with known nonlinearity structure using particle swarm optimization. *Arabian Journal for Science and Engineering*, 36(7), 1269-1276.
- [82] Du, Z., & Wang, X. (2011). Hammerstein model identification using quantum delta-potential-well-based particle swarm optimisation. *International Journal of Modelling, Identification and Control*, 12(4), 421-427.

- [83] Sun, J., & Liu, X. (2013). A novel APSO-aided maximum likelihood identification method for Hammerstein systems. *Nonlinear Dynamics*, 73(1), 449-462.
- [84] Xu, X., Wang, F., Liu, G., & Qian, F. (2013). Identification of Hammerstein systems using key-term separation principle, auxiliary model and improved particle swarm optimisation algorithm. *IET Signal Processing*, 7(8), 766-773.
- [85] Jin, Q., Wang, H., Su, Q., Jiang, B., & Liu, Q. (2018). A novel optimization algorithm for MIMO Hammerstein model identification under heavy-tailed noise. *ISA transactions*, 72, 77-91.
- [86] Castro-Garcia, R., Agudelo, O. M., & Suykens, J. A. (2019). Impulse response constrained LS-SVM modelling for MIMO Hammerstein system identification. *International Journal of Control*, 92(4), 908-925.
- [87] Wang, D., Li, L., Ji, Y., & Yan, Y. (2018). Model recovery for Hammerstein systems using the auxiliary model based orthogonal matching pursuit method. *Applied Mathematical Modelling*, 54, 537-550.
- [88] Ding, J., Cao, Z., Chen, J., & Jiang, G. (2020). Weighted parameter estimation for Hammerstein nonlinear ARX systems. *Circuits, Systems, and Signal Processing*, 39(4), 2178-2192.
- [89] Ding, F., Chen, H., Xu, L., Dai, J., Li, Q., & Hayat, T. (2018). A hierarchical least squares identification algorithm for Hammerstein nonlinear systems using the key term separation. *Journal of the Franklin Institute*, 355(8), 3737-3752.

- [90] Wang, D., Zhang, S., Gan, M., & Qiu, J. (2019). A novel EM identification method for Hammerstein systems with missing output data. *IEEE Transactions on Industrial Informatics*, 16(4), 2500-2508.
- [91] Dong, S., Liu, T., & Wang, Q. G. (2018). Identification of Hammerstein systems with time delay under load disturbance. *IET Control Theory & Applications*, 12(7), 942-952.
- [92] Mehmood, A., Aslam, M. S., Chaudhary, N. I., Zameer, A., & Raja, M. A. Z. (2018). Parameter estimation for Hammerstein control autoregressive systems using differential evolution. *Signal, Image and Video Processing*, 12(8), 1603-1610.
- [93] Hou, J., Liu, T., & Wang, Q. G. (2019). Subspace identification of Hammerstein-type nonlinear systems subject to unknown periodic disturbance. *International Journal of Control*, 1-11.
- [94] Mehmood, A., Chaudhary, N. I., Zameer, A., & Raja, M. A. Z. (2019). Backtracking search optimization heuristics for nonlinear Hammerstein controlled auto regressive auto regressive systems. *ISA transactions*, 91, 99-113.
- [95] Isa, N. A. M., & Mamat, W. M. F. W. (2011). Clustered-hybrid multilayer perceptron network for pattern recognition application. *Applied Soft Computing*, 11(1), 1457-1466.
- [96] Guo, Z. X., Wong, W. K., & Li, M. (2012). Sparsely connected neural network-based time series forecasting. *Information Sciences*, 193, 54-71.

- [97] Malakooti, B., & Zhou, Y. (1998). Approximating polynomial functions by feedforward artificial neural networks: capacity analysis and design. *Applied Mathematics and Computation*, 90(1), 27-51.
- [98] Chen, C. T. (1984). *Linear System Theory and Design*, Holt, Winehart and Winston. Inc., New York, NY..
- [99] S. M.-S. in C. Intelligence a 2019, “Evolutionary algorithms and neural networks,” Springer.
- [100] Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). IEEE.
- [101] Shi, Y., & Eberhart, R. (1998, May). A modified particle swarm optimizer. In *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence* (Cat. No. 98TH8360) (pp. 69-73). IEEE.
- [102] Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information sciences*, 179(13), 2232-2248.
- [103] S. Mirjalili, S. H.-2010 international conference on, 2010, “A new hybrid PSOGSA algorithm for function optimization,”.
- [104] Wigren, T., & Schoukens, M. (2017). Coupled electric drives data set and reference

models. Department of Information Technology, Uppsala Universitet.

- [105] Chou, C. T., & Maciejowski, J. M. (1997). System identification using balanced parametrizations. *IEEE transactions on automatic control*, 42(7), 956-974.

|

Vitae

Name :Amro Faisal Abdoun Mohamed |

Nationality :Sudanese |

Date of Birth :6/25/1992|

Email :amori1899@gmail.com|

Address :Dammam, Saudi Arabia|

Academic Background :Received degree of Bachelor of Engineering (Honors) in Electrical Engineering (control) from Sudan University of Science and Technology in 2014. Started working with Zamil Plastic Industries Ltd. as maintenance engineer in 2015. From 2019 till now working as electrical supervisor at Zamil Plastic Industries Ltd. |