IDENTIFICATION OF CONTINOUOUS-TIME MODEL OF HAMMERSTEIN SYSTEM USING ARCHIMEDES OPTIMIZATION ALGORITHM

CHO BO WEN

B. ENG(HONS.) ELECTRICAL ENGINEERING

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name   : <u>CHO BO WEN</u>

Date of Birth   : <u>12/3/1998</u>

Title   : <u>IDENTIFICATION OF CONTINOUOUS-TIME MODEL OF</u>
<u>HAMMERSTEIN SYSTEM USING ARCHIMEDES</u>
<u>OPTIMIZATION ALGORITHM</u>

Academic Session   : <u>SEM II 2021/2022</u>

I declare that this thesis is classified as:

☐   CONFIDENTIAL   (Contains confidential information under the Official
                                       Secret Act 1997)*

☑   RESTRICTED   (Contains restricted information as specified by the
                                        organization where research was done)*

☐   OPEN ACCESS   I agree that my thesis to be published as online open access
                                        (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.
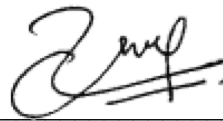
Certified by:

_____   _____
(Student's Signature)   (Supervisor's Signature)

CHO BO WEN   ASSOC PROF. DR MOHD.
                                       ASHRAF BIN AHMAD
_____
New IC/Passport Number   _____
Date:20/6/2022   Name of Supervisor
                                      Date: 20/6/2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Engineering in Electrical engineering (electronics)

(Supervisor's Signature)

Full Name   : DR MOHD. ASHRAF BIN AHMAD
Position   : ASSOCIATE PROFESSOR
Date   : 20/6/2022

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

_____

(Student's Signature)

Full Name   : CHO BO WEN

ID Number   : EA18092

Date   : 20/6/2022

IDENTIFICATION OF CONTINOUOUS-TIME MODEL OF HAMMERSTEIN
SYSTEM USING ARCHIMEDES OPTIMIZATION ALGORITHM

CHO BO WEN

Thesis submitted in fulfillment of the requirements
for the award of the
Bachelor's Degree in electrical engineering (Electronics) With Honours

College of Engineering

UNIVERSITI MALAYSIA PAHANG

JUNE 2022

ACKNOWLEDGEMENTS

Throughout this difficult journey, many people have greatly supported in the completion of this thesis, and I would like to express my thankfulness to every single one of them.

Firstly, I would like to thank my supervisor, Associate Professor, TS.Dr. Mohd. Ashraf Bin Ahmad. Throughout this difficult endeavour, he never gave up on me and continue to offer guidance and support till the very end of it. His vast knowledge in this field and great understanding skills shows how much he cared for not only me but all his students. I have learned a great deal from him and hope one day to be able to as great of a person as him.

Secondly, I would like to thank all the staffs and lectures of Universiti Malaysia Pahang that taught me and helped me throughout these four years of study. They have provided me a good and nice environment for me to smoothly complete my studies. Special thanks to my academic advisor Associate Professor Ir. Dr. Nurul Hazlina Binti Noordin who guided me with great thoughtfulness in the completion of my study.

Finally, I am eternally grateful to my family and friends, for their continuous support throughout the completion of this thesis. Without their love and guidance, I would surely lose all motivations in completing this thesis. One day, I would hope to repay all the sacrificed they made for me to reach this point in my life.

ABSTRAK

Tesis ini mencadangkan kaedah pengenalan baru yang dikenali sebagai improved archimedes optimization algorithm (IAOA) untuk mengenal pasti model Hammerstein yang berterusan. Dua pengubahsuaian telah digunakan untuk menyelesaikan beberapa kelemahan archimedes optimization algorithm (AOA) asal. Seterusnya, kaedah yang dicadangkan digunakan dalam mengenal pasti pembolehubah subsistem linear dan bukan linear dalam model Hammerstein yang berterusan menggunakan data input dan output yang diberikan. Untuk mengesahkan kecekapan kaedah yang dicadangkan, satu contoh berangka dan dua eksperimen dunia sebenar, iaitu Twin-rotor system (TRS) dan electro-mechanical positioning system telah dijalankan. Keputusan dianalisis dari segi keluk penumpuan fungsi kecergasan, pembolehubah de indeks viation, domain masa dan respons domain frekuensi bagi model yang dikenal pasti, dan ujian jumlah pangkat Wilcoxon. Keputusan yang diperolehi menunjukkan bahawa kaedah yang dicadangkan, menghasilkan penyelesaian dengan ketepatan dan ketekalan yang lebih baik jika dibandingkan dengan kaedah metaeuristik terkenal yang lain seperti Particle Swarm Optimizer, Grey Wolf Optimizer, Multi-Verse Optimizer, Archimedes Optimization Algorithm dan kaedah hybrid dinamakan Average Multi-Verse Optimizer and Sine Cosine Algorithm.

ABSTRACT

This thesis proposed a novel identification method known as the improved archimedes optimization algorithm (IAOA) for identifying the continuous-time Hammerstein model. Two modifications were employed to solve several demerits of the original archimedes optimization algorithm (AOA). The first modification was an alteration of the density decreasing factor to solve the imbalance of the exploration and exploitation phases. The second one was the introduction of safe updating mechanism to solve the local optima issue. Next, the proposed method was utilized in identifying the variables of the linear and nonlinear subsystems in a continuous-time Hammerstein model using the given input and output data. To verify the efficiency of the proposed method, a numerical example and two real-world experiments, namely the twin-rotor system and the electro-mechanical positioning system were carried out. The results were analysed in terms of the convergence curve of the fitness function, the variable deviation index, time-domain and frequency-domain responses of the identified model, and the Wilcoxon's rank-sum test. The obtained results showed that the proposed method, yields solutions with better accuracy and consistency when compared with other well-known metaheuristics methods such as the Particle Swarm Optimizer, Grey Wolf Optimizer, Multi-Verse Optimizer, Archimedes Optimization Algorithm and a hybrid method named the Average Multi-Verse Optimizer and Sine Cosine Algorithm.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

$u(t)$	Input of the nonlinear function
$\beta(u(t))$	Output of the nonlinear function
$\bar{\beta}(u(t))$	Estimated output of the nonlinear function
$\mathbb{R}$	Real numbers
$\bar{Y}(w)$	Estimated linear dynamic subsystem
Z_b	Coefficient of the polynomial function
X_e	Coefficient of the polynomial function
ζ_i	Coefficient of the polynomial function
$v(t)$	Noise signal
$P(\bar{Y}, \bar{\beta})$	Quadratic output estimation error
$o(t)$	Output of the linear subsystem
$\bar{o}(t)$	Estimated output of the linear subsystem
$\acute{o}(t)$	Hammerstein model output
$P(\vartheta)$	Fitness function
N	Total number of samples
t_s	Sampling time
ub_i	Upper boundary
lb_i	Lower boundary
ϑ_i	i^{th} element of the design variable
δ	Variable deviation index
k	Current iteration
k_{max}	Maximum iteration
$rand$	Random vector
x_i	Position of object
x_{best}	Best position of object
acc	Acceleration of object
vol	Volume of object
den	Density of object
max	Maximum
min	Minimum
F	Flag
TF	Transfer operator
d	Density decreasing factor

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimizer
AMVO-SCA	Average Multi-Verse Optimizer and Sine Cosine Algorithm
AOA	Archimedes Optimization Algorithm
BFO	Bacterial Foraging Optimization
BSO	Brainstorm Optimization
CBO	Colliding Bodies Optimization
COVID-19	Coronavirus
CPSO	Clonal Particle Swarm Optimizer
DEA	Differential Evolution Algorithm
EO	Equilibrium Optimizer
GA	Genetic Algorithm
GA-PSO	Genetic Algorithm-Particle Swarm Optimizer
GOA	Grasshopper Optimization Algorithm
GSA	Gravitational Search Algorithm
GSO	Glow-worm Swarm Optimization
GWO	Grey Wolf Optimizer
GWO-CFA	Grey wolf optimizer based chaotic firefly algorithm
IAOA	Improved Archimedes Optimization Algorithm
LMS	Least Means Square
MPPT	Maximum Power Point Tracker
MVO	Multi-Verse Optimizer
MIMO	Multi Input Multi Output
NARMAX	Nonlinear Auto Regressive Moving Average with exogenous input
NFL	No Free Lunch
PSO	Particle Swarm Optimizer
PSO-CS	Particle Swarm Optimization-Cuckoo Search
SCA	Sine Cosine Algorithm

CHAPTER 1

INTRODUCTION

1.1   Research Background

Mathematical modelling is the process of creating a mathematical representation of a system that can accurately predict the output from the given input. One such method of creating a mathematical model is through the process of system identification which utilizes the observed input and output data. Most of the actual dynamic systems cannot be fully described with a linear model as they contain nonlinear characteristics (Eskinat et al., 1991). For example, in ground vibration testing of airplanes, nonlinear stiffness and nonlinear damping results in resonance frequencies and damping effects that differs with different excitation levels. In addition, in biological systems like our eyes, nose, tongue, and ears that governs our senses through stimuli follows a type of nonlinear compression called the Weber-Fechner laws (Dehaene, 2003). As such, the use of nonlinear system identification is the preferred method in forming mathematical models. Nonlinear system identification can be broadly categorized into five models which are the Volterra series, block-structured, neural network, Nonlinear Auto Regressive Moving Average with exogenous input (NARMAX) and state-space models.

Among the models, the block-structured model is popular in nonlinear modelling applications as the structure is straightforward to comprehend, easy to apply as the linear time invariant dynamics and the static nonlinearities are separated (Eskinat et al., 1991). In block structured models, there exist three different types, which are the Hammerstein model where the nonlinear block comes before the linear block, the Wiener model where the linear block comes before the nonlinear block, and the Hammerstein-Wiener model in which the linear block is located between two nonlinear blocks (Schoukens & Ljung, 2019). The Hammerstein model are widely used by researchers to model nonlinear systems due to its versatility in the identification of real experimental plants and processes

(Jui et al., 2022) like the as the pH neutralization process (Zou et al., 2013), the dynamics of electrically stimulated muscles (Farahat & Herr, 2005), solid oxide fuel cells (Jurado et al., 2006), wind speed forecasting (Ait Maatallah et al., 2015), and detecting valve stiction (Yan et al., 2017). Furthermore, the Hammerstein model can represent many kinds of nonlinearities of a process plant without much difficulty (Manenti, 2011). Hence, it is worth to choose the Hammerstein model for nonlinear system identification of actual process plants.

In previous literature, deterministic methods for instance the Least Mean Square (LMS) (Chaudhary & Raja, 2015) and the Gradient Search Algorithm (F. Ding et al., 2011) were used to solve the identification of Hammerstein models. This type of method usually produces quality results but that's not always the case as they are not adaptive and they might produce high computational burden for large-scale and complex problems (Madić et al., 2013).

In recent years, metaheuristic methods are widely popular due to their adaptivity and ability in finding more accurate solutions (Yang, 2010). At-present, many different kinds of metaheuristics methods utilized in the identification of the Hammerstein model, such as Genetic Algorithm (GA) (Akramizadeh et al., 2002), Particle Swarm Optimizer (PSO) (Nanda et al., 2010), Clonal PSO (CPSO) and Immunized PSO (IPSO) (Nanda et al., 2010), Gravitational Search Algorithm (GSA) (Cuevas et al., 2018)., differential evolution algorithm (DEA) (Mete et al., 2016), and Colling Bodies Optimization (CBO) (Panda & Pani, 2014). In addition to that hybrid metaheuristic methods have also been applied to identify the Hammerstein models such as Grey wolf optimizer based chaotic firefly algorithm (GWO-CFA) (Ganguli et al., 2019), Particle Swarm Optimization-Cuckoo Search (PSO-CS) (J. Ding et al., 2019) and Genetic Algorithm-Particle Swarm Optimizer (GA-PSO) (Hachino et al., 2009).

One of the alternative algorithm that could identify the Hammerstein model is called the Archimedes Optimization Algorithm (AOA) proposed by Hashim et al. (Hashim et al., 2022). This metaheuristic method is based on the Archimedes' principle which explains the law of buoyancy, particularly the relation between an object immersed in a fluid and the buoyant force applied on it. The AOA method has been successfully shown to outperform other popular metaheuristic methods such as PSO, GA, sine-cosine algorithm (SCA) and Equilibrium Optimizer (EO) in multiple benchmark test. The merits

of AOA came from its simple structure and a smaller number of coefficients, while it has been utilized in solving a broad range of optimization problems for instance identifying the optimal coefficients of different fuel cells (Houssein et al., 2021), tuning a maximum power point tracker (MPPT) controller for wind energy generation system (Fathy et al., 2022), optimizing an intelligent control of power system stabilizer (Aribowo et al., 2021), diagnosing Coronavirus (COVID-19) from X-Ray images on chest of the human body (Chen & Rezaei, 2021), and designing a microstrip patch antenna (Singh & Kaur, 2022). As a result, AOA is an excellent method to identify a continuous-time Hammerstein model.

1.2 Motivation and Problem Statement

In this research, the main focus is to develop an improved version of the Archimedes optimization algorithm for identifying the unknown variables of the continuous-time Hammerstein model. The modifications done to the AOA is to improve the effectiveness of the method. Two demerits had been identified in the original AOA method.

Firstly, AOA is vulnerable to uneven exploration and exploitation phases, which result in a decrease in its searching capabilities. Exploration and exploitation are two conflicting mechanisms in theory, where favouring one phase will result in the deterioration of the other phase. Specifically, excessive exploration makes it impossible for the algorithm to find an accurate global optimum value. In contrast, excessive exploitation, on the other hand, slowed the convergence process toward the global optimum solution. Thus, an adequate balance between the exploration and exploitation phases is necessary in the original AOA. Secondly, since the updated object position is exclusively dependent on its prior or best object position, there is a high considerable risk of AOA becoming stuck in the local optima. If the best object location is entrapped in the local optimal zone, it will attract other objects into the same zone and be trapped together. Similarly, if the current object location is entrapped in the local optimal zone, escaping the zone is difficult since the object lacks information of the best object position. Hence, it is worthwhile to thoroughly analyse and pursue these difficulties in order to deliver a better version of the AOA.

The demerits above, of the AOA would reduce the variable identification accuracy of the continuous-time Hammerstein model. Particularly, if the optimization method lacked the capability to escape from the local optima and an appropriate balance of both the exploration and exploitation phase. The resulting identification accuracy would be lower. In addition, according to the theorem of no free lunch (NFL), there is no equally successful global optimizer that can provide answers to all optimization issues. Therefore, it is paramount that we have a wide range of optimization algorithms, to solve various kinds of optimization issues. Hence, the original AOA method would need to be improved upon to achieve a higher identification accuracy of the continuous-time Hammerstein model.

1.3   Research Objectives

The main goal of this study is to develop an improved archimedes optimization algorithm for identifying a continuous-time Hammerstein model, and the proposed approach is produced by modifying the original archimedes optimization algorithm. The optimal outcomes of the suggested algorithms are then compared to popular metaheuristics methods. The following are the precise objectives envisioned for this work:

1. To propose an improved archimedes optimization algorithm (IAOA) for solving the local optima demerit and the imbalance in exploration and exploitation phases of the original archimedes optimization algorithm (AOA).

2. To establish a structure for the identification of continuous-time Hammerstein model based on the Improved Archimedes optimization algorithm (IAOA)

3. To investigate the efficacy of the IAOA for continuous-time Hammerstein model identification in terms of the quadric output estimation error and variable deviation index

1.4   Scope of Research

This study provides an improved archimedes optimization algorithm (IAOA) for the identification for the continuous-time Hammerstein model. Two modifications are introduced to the original AOA. Firstly, is the modification to the density decreasing

factor equation. Secondly, the safe updating mechanism is introduced to the original AOA method.

To establish the structure for this identification for a Hammerstein model in continuous time which contains nonlinear function which comes before the linear subsystem, the nonlinear function is assumed to be unknown. Contrarily, the linear subsystem is known. Then, the proposed improved archimedes optimization algorithm (IAOA) will be utilized to identify the variables of the linear and nonlinear subsystems from the given input and output data.

To validate the proposed IAOA method, three experiments were conducted. This includes one numerical example and two real world experiments. The two experiments are Electro-Mechanical Positioning System (EMPS) and the Twin-Rotor System (TRS). In all three experiments the results are assessed through five criteria, which are the convergence curve of the fitness function, variable deviation index, time-domain and frequency-domain responses, and Wilcoxon's rank-sum test. All findings acquired from the proposed IAOA method will be then compared with AOA (Hashim et al., 2022), Average Multi-Verse Optimizer and Sine Cosine Algorithm (AMVO-SCA) (Jui & Ahmad, 2021), PSO (Kennedy & Eberhart, 1995), GWO (Mirjalili, Mirjalili, et al., 2014), and Multi-verse Optimizer (MVO) (Mirjalili et al., 2016).

Only a few studies had been published up to this point that discussed the identification of continuous-time nonlinear models. As a result, the efficacy of the proposed IAOA method for continuous-time nonlinear system identification problems is worth investigating.

1.5   Overview of the Thesis

Overall, this thesis contains five chapters which includes the current chapter. In chapter one the problem statement, motivation, objectives and scope of the thesis is described. The following is how the remaining chapters are organised:

In chapter 2, an overview of the nonlinear system identification models are presented. Various kinds of metaheuristic method utilized in the identifications of Hammerstein model is then discussed. Next, a review on the Archimedes Optimization Algorithm (AOA) and some of its variants are discussed. Moreover, the justification for

adopting the Hammerstein model for nonlinear system identification, as well as the MVO algorithm for identifying the parameters of the linear and nonlinear subsystems of the Hammerstein system, will be discussed.

In chapter 3, discusses about the methodology of the thesis and proposed a novel optimization method called the improved archimedes optimization algorithm (IAOA). Firstly, is the formulation of the identification problem of the Continuous-time Hammerstein model. Then, an overview of the original archimedes optimization algorithm (AOA) was presented. Next, the proposed improved archimedes optimization algorithm (IAOA) is discussed. Finally, the continuous-time Hammerstein model identification method is presented.

For Chapter 4, it presents the results and discussions of the thesis, in which the efficacy of the newly proposed IAOA is compared with other methods. Firstly, the IAOA efficacy will be validated by a numerical method and two real-world experiments, which are Electro-Mechanical Positioning System (EMPS) and the Twin-Rotor System (TRS in terms of the convergence curve of the fitness function, variable deviation index, time-domain and frequency-domain responses, and Wilcoxon's rank-sum test. All findings acquired from the proposed IAOA method will be then compared with AOA (Hashim et al., 2022), Average Multi-Verse Optimizer and Sine Cosine Algorithm (AMVO-SCA) (Jui & Ahmad, 2021), PSO (Kennedy & Eberhart, 1995), GWO (Mirjalili, Mirjalili, et al., 2014), and Multi-verse Optimizer (MVO) (Mirjalili et al., 2016).

The thesis' conclusion and the research's future recommendation are presented in Chapter 5.

CHAPTER 2

Literature Review

2.1   Introduction

This chapter presents the literature on nonlinear system identification models and the various types of metaheuristics methods utilized in identification of such nonlinear models. Firstly, an overview of the nonlinear system identification models is presented in Section 2.2. Secondly, Section 2.3 different metaheuristic methods that was utilized in the identification of nonlinear Hammerstein models were discussed. Thirdly, in Section 2.4 the original AOA and its variants were presented. Finally, the research gap in section 2.5 was discussed.

2.2   Overview of Nonlinear System Identification

System identification is the process of which a mathematical model is modelled using the given input and output signals of a system. This methodology could be applied in both linear and nonlinear models. The real world is nonlinear and time-varying, therefore, linear models which does not consider the nonlinearities of the system would often yield imprecise models which does not accurately reflects the behaviour of the system . The demand for nonlinear system identification extends well beyond the control field. Nonlinear models are useful for gaining a fundamental understanding of a variety of issues, such as brain activity modelling and chemical interactions. This demonstrates that there are numerous reasons to switch from linear to nonlinear models. However, rising demands for increased performance and efficiency force systems towards nonlinear functioning, necessitating the use of nonlinear models in their design and control.

System identification is a critical component of any control system design, yet identifying nonlinear systems is difficult at the moment. In recent years, due to their

simple block-oriented structure nonlinear block-oriented systems are a popular field of research in nonlinear system modelling . There are four main types of nonlinear block-oriented structure models which are the Hammerstein, Wiener, Hammerstein-Wiener, and Wiener-Hammerstein models. The general block diagrams of the models are shown in Figure 2.1. Here, the nonlinear subsystems are represented by $v(\cdot)$ and $w(\cdot)$, and the liner subsystems are represented by $R(s)$ and $W(s)$. In the Hammerstein model the nonlinear block comes before the linear block, as illustrated in Figure 2.1(a). For the Wiener model the nonlinear block comes after the linear block, as illustrated in Figure 2.1(b). Because system outputs are dependent on inputs in a nonlinear fashion, the correlation between input and output can be decomposed into numerous connected components. The relevance of the linear transfer function is reflected in this process. The Hammerstein-Wiener model achieves this arrangement through the coupling of the dynamic linear and static nonlinear blocks. The Hammerstein-Wiener model structure block diagram is shown in Figure 2.1(c). For the Wiener-Hammerstein model which is shown in Figure 2.1(d), it consists of a nonlinear subsystem which is sandwiched between two linear subsystems.
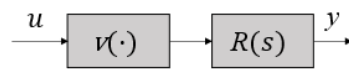
(a) Hammerstein model

(b) Wiener model
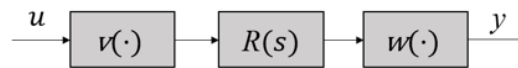
(c) Hammerstein-Wiener model
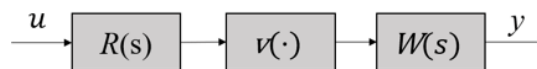
(d) Wiener-Hammerstein model

Figure 2.1   Nonlinear Block-oriented structure models

8

2.3   Metaheuristics Methods for Nonlinear System Identification

Metaheuristic methods for nonlinear system identification are widely popular due to their adaptivity and ability in finding more accurate solutions (Yang, 2010). Metaheuristic methods can be divided into two main class, which single-agent-based or multi-agent-based. The search procedure in the single-agent-based class begins with the randomization of a single solution. Then, as the iterations progress, it will get better. The objective searching process in the latter class, on the other hand, will begin with a collection of random initial solutions, which will be improved throughout the course of iterations.

In addition, depending on the source of a method's inspiration, the multi-agent-based class can be divided into two categories, which are swarm based or evolution based. In swarm-based strategies, which replicate the social behaviour of groups of animals. In comparison to evolution-based methods, swarm-based methods have a few advantages. Swarm-based methods, for example, keep the search space information over iterations, whereas evolution-based methods destroy any knowledge as soon as a new population forms. Particle Swarm Optimization is the most prominent swarm-based method, which was inspired by the social behaviour of flocking birds. To identify the optimal solution, it employs a number of particles that fly about the search space. In the meantime, they're all tracing their pathways to find the optimal location.

Furthermore, the laws of natural evolution are the inspiration for evolution-based methods. The search begins with a population that was produced at random and has evolved over generations. The best individuals are always united to generate the next generation of individuals, which is a strength of these systems. Over numerous generations, the population can be optimised in this way. Genetic Algorithms (GA), which replicates Darwinian evolution, is the most prominent evolution-inspired methods. In addition to GA some other widely utilized metaheuristic methods includes, Genetic Programming (GP), Probability-Based Incremental Learning (PBIL), Asexual Reproduction Optimization (ARO), Differential Evaluation (DE), and Evolution Strategy (ES). In Figure 2.2 a bar chart from the study (Jui et al., 2022) was shown. It presents the number of times each metaheuristic methods were utilized in the identification of nonlinear system using various kinds of block-orientated models from 2011 to 2021. The list of metaheuristic methods utilized includes Glow-worm Swarm Optimization (GSO),

Differential Evolution Algorithm (DEA), Bacterial Foraging Optimization (BFO), Genetic Algorithm (GA), Brainstorm Optimization (BSO). Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO). From this we can clearly see that PSO is the most utilized in the identification of nonlinear system followed with GA and then DE. While the rest of the metaheuristic methods were rarely applied in the identification of nonlinear system.
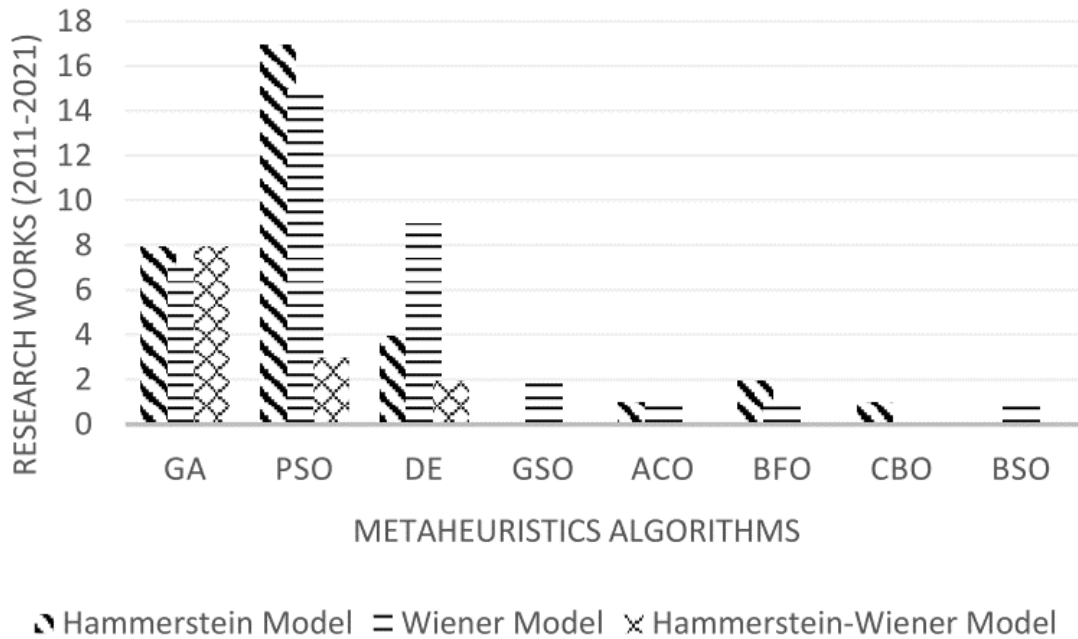
Figure 2.2   Bar chart of popularity of metaheurtsic methods in identificaiton of nonlinear systems based on block-orientated models.
Source: (Jui et al., 2022)

2.3.1   Hammerstein Model Identification

This section discusses the various metaheuristic methods utilized in the identification of Hammerstein model. One of such metaheuristic methods that have been applied to identify the Hammerstein model is the Genetic Algorithm (GA) (Akramizadeh et al., 2002). Here, they have used the GA-based method to tune the parameters in hyperbolic function and discrete-time linear system simultaneously such that mean square error between the actual and approximated outputs is minimized. Particle Swarm Optimizer (PSO) is another popular metaheuristic method that was also used in the identification of Hammerstein models (Nanda et al., 2010). In their approach, they modified the PSO method into Clonal PSO and Immunized PSO to identify the nonlinear

subsystem that was represented by a single layer Functional Link Artificial Neural Network (FLANN) and the linear subsystem that was represented in discrete-time transfer function (Nanda et al., 2010). Through this approach, they manage to achieve better results than GA in terms of identification accuracy. Furthermore, the Gravitational Search Algorithm (GSA) was also used in identification of the Hammerstein model and the structure of the model was based on the Adaptive Network-based Fuzzy interference System (ANFIS) (Cuevas et al., 2018). In the ANFIS structure, the nonlinear subsystem is consisted of the antecedent of fuzzy inference system, while the linear subsystem is represented by the discrete-time transfer functions. Nevertheless, the proposed ANFIS structure may produce high computational burden due to redundancy gain between nonlinear and linear subsystems.

In addition to that, modern metaheuristics algorithm was also used in the identification of Hammerstein models which is the differential evolution algorithm (DEA) (Mete et al., 2016). Here, the structure of the Hammerstein model is based on a nonlinear second order Volterra model (SOV) and a finite impulse response (FIR) for both nonlinear and linear subsystems, respectively. Similar to (Mete et al., 2016), the structure of SOV-FIR still consists of redundant gains between linear and nonlinear subsystems, which may increase the computational burden. Besides that, the Colling Bodies Optimization (CBO) method has also been used in the identification of Hammerstein model where the nonlinear subsystem is based on *t-th* order polynomial function and the linear subsystem is based on discrete-time function or polynomials of delay operators (Panda & Pani, 2014). The results demonstrate superior performance of the CBO with reasonable convergence speed as compared to Bacterial Foraging Optimization and Adaptive PSO methods.

Furthermore, due to limitation of the standalone metaheuristic methods, some studies also utilized hybrid metaheuristic methods where they combined two or more different methods to create a powerful approach in solving the identification of Hammerstein models. One such hybrid metaheuristic method is the Grey wolf optimizer based chaotic firefly algorithm (GWO-CFA) (Ganguli et al., 2019). Here, the GWO was utilized to perform the global search (exploration) whereas CFA was used to execute the local search (exploitation) such that a proper exploration and exploitation phases balance is obtained. By using this hybrid approach, they achieved results that has better accuracy

compared to popular metaheuristic methods like PSO, GWO, and Ant Lion optimization method (ALO) in the identification of Hammerstein models. Next, the Particle Swarm Optimization-Cuckoo Search PSO-CS) is also another hybrid metaheuristic method that was utilized in the identification of Hammerstein models (J. Ding et al., 2019). Through this approach, they produced a Hammerstein model with a higher accuracy compared to standalone metaheuristic. Moreover, the Genetic Algorithm-Particle Swarm Optimizer (GA-PSO) was also utilized in the identification of Hammerstein models (Hachino et al., 2009). In this approach, GA was used during the global optimization phase while PSO was used during the local optimization phase utilizing the strengths of each method. It was demonstrated that the hybrid GA-PSO method can produce better performances in terms of Akaike informatic criterion and means of the output errors as compared to its standalone methods. Even so, all these hybrid approaches impose higher complexity with larger number of coefficients compared to standalone metaheuristics methods (Ting et al., 2015). The increment of complexity will lead to higher computational load and the users would require more effort in tuning the higher number of coefficients.

2.4   Archimedes optimization algorithm (AOA) and it's variants

Archimedes optimization algorithm was introduced in 2020 by Hashim et al. (Hashim et al., 2022). This method was utilized to solve various kinds of optimization issues, but it is not without its demerits. In this section the original AOA is discussed along with its variants where modifications are made to AOA to improves its performance.

2.4.1   AOA

Ahmed Fathy and Abdullah G. Alharbi applied AOA in simulating a maximum power point tracker (MPPT) (Fathy et al., 2022). The metaheuristics method is used in their study to adjust the duty cycle of the converter MOSFET in order to maximise the output power from the wind energy generation system. The results obtained from the AOA method will then be compared to different metaheuristic methods such as electric charged particle optimization (ECPO), cuckoo search (CS) and grasshopper optimization algorithm (GOA). In their findings they found that the AOA-MPPT controller outperformed all considered metaheuristic methods. In the fixed wind-speed setting, the AOA-MPPT controller extracted maximum power of 102.2039 W at a duty cycle of

0.0911, whereas the GOA-MPPT controller extracted maximum power of 101.1967 W. The AOA-MPPT controller extracts 102.136 W for operation at variable wind-speed, while the GOA-MPPT controller extracts 101.7 W. The AOA-MPPT controller retrieved 12.7988 W for in the practical case study, while ECPO-MPPT obtained 9.8654 W. They concluded that AOA be used to tackle numerous optimization problems for other renewable energy systems and the smart grid due to it AOA's robustness and performance.

In another study conducted by in the study (Chen & Rezaei, 2021) they applied AOA as a new optimal diagnosis system for COVID-19 based on X-ray images of patients chest. In the study, the optimization of feature selection and the multilayer perceptron model (MLP)was based on AOA. This optimized MLP model based on AOA was used to determine the suspected cases of the COVID-19. The results obtained were compared with four state-of-the-art methods to examine the model efficiency, which are a neural network named CheXNet (Mangal et al., 2020), a deep learning method in (Ismael & Şengür, 2021), a DeTraC deep convolutional neural network in (Abbas et al., 2021) and a convolutional neural network developed using machine learning in (Abbas et al., 2021). In their findings they found that AOA method outperforms all the other methods in diagnosis of COVID-19 in terms of accuracy, specificity, precision, recall and F1-score.

Furthermore, in (Li et al., 2021), AOA was utilize to determine the optimal configuration of a distributed generation (DG) connection on the power loss and voltage profile. The study deals with the subject of DG optimal configuration while simultaneously dealing with DG locations and capacity. After optimization, the configuration methods ensure that the system operates in an optimal condition while satisfying the goal of minimising power loss. The performance of the configuration produced by AOA method is then compared with other metaheuristic methos such as Particle Swarm Optimization (PSO) well as the Improved Genetic Algorithm (IGA). In their findings, AOA has a faster convergence time and more stable outcomes. Meanwhile, while dealing with DG comprehensive optimization problems, AOA has a faster running time and a better global search capability than the other two methods. Moreover, AOA also yields configurations which minimize power loss and better voltage profiles when compared to the other two methods.

2.4.2 Variants of AOA

In some optimization problems the original AOA was not able to produce excellent results. Therefore, some modifications were applied to AOA to improve its performance. Firstly, in (Desuky et al., 2021) few modifications were made to the original AOA to enhance its classification performance for the issue of feature selection. They called this the Enhance Archimedes Optimization Algorithm (EAOA). In their modification, a new coefficient was added to the original AOA to enhance its balance between exploration and exploitation phase. This coefficient is dependent on the step size of every object's position while revising the object's position over the course of iterations. Furthermore, 23 benchmark functions are used to compare the EAOA method's ability to solve optimization issues to other metaheuristics methods, which are the original AOA, PSO, ALO, Whale Optimization Algorithm (WOA) and Giza Pyramids Construction method (GPC). In their findings, they discovered that the EAOA method has higher exploitation capability for unimodal functions and greater exploration capability for multimodal and fixed-dimensional multimodal functions. In addition, when compared with other methods EAOA is able to escape the entrapment of local optima and have a higher rate of convergence. Moreover, in terms of the feature selection issue EAOA also yields shorter time for classification, while having the best performance when compared to the original AOA and other compared methods.

In the following study (Akdag, 2022), a new and improved AOA is introduced to solve the optimal power flow optimization issue. The first modification made to the original AOA is the introduction of an additional step in selecting the best object's position in each iteration and updating the current object's position for the following iteration. The goal of this modification is to increase population variety in AOA and improve the balance between exploitation and exploration phases. Furthermore, this strategy employs a distinct method for constructing a neighbourhood for each object in which neighbour data can be shared. Secondly, is the usage of a different dimension learning-based strategy to build a neighbourhood for each object in which neighbour data can be transferred between objects. In this modification, it can improve the balance between local and global search of the original AOA. In their findings, for the optima power flow issue found in three different power systems their proposed method yields results that outperforms the original and other compared metaheuristics methods. Where

the proposed method managed to obtain minimized fuel cost and fuel-based emission minimization and improvement in voltage profiles when compared to other methods.

An enhanced Archimedes optimization algorithm based on local escaping operator and orthogonal learning (I-AOA) is presented in (Houssein et al., 2021). This I-AOA was utilized in identification of parameters of a polymer electrolyte membrane (PEM) fuel cell. Two strategies were introduced in the I-AOA method, which are the local escaping operator (LEO) and orthogonal learning (OL). In LEO it improves the quality of solutions by repositioning the individuals in the population them when certain criteria are met. This allows the method from being entrapped in the local optima and also improves the method's convergence characteristics. Next, OL is one of the most popular strategies for optimising the process of finding the best agents in order to get the best global solution. As a result, by introducing OL it would improve the balance of the exploration and exploitation phase of the original AOA. Here, they evaluated the performance of the I-AOA by utilizing the CEC'2020 test suite and three engineering issues , which are rolling element bearing, pressure vessel, and tension compression spring design issues. The results obtained from I-AOA is then compared with other metaheuristic methods such as the original AOA, WOA, SCA, PSO, Moth-Flame Optimization Algorithm (MFO), Harris hawk's optimization (HHO), and Tunicate Swarm Algorithm (TSA). In their findings, they found that for most of the benchmark functions I-AOA yields the best results, and in all three engineering issues I-AOA also yields the lowest fitness function values when compared with all the other methods. Moreover, to verify the performance of I-AOA, the proposed method was utilized in identifying the optimal parameters of the PEM fuel cell. They found that, I-AOA yields the lowest fitness function value when compared with other methods.

2.5 Research Gap

In the studies mentioned in Section 2.4. It was found that they picked AOA as the metaheuristic methods of choice to solve the optimization issues due to its, less control coefficients, simplicity and it most cases it outperforms other metaheuristic methods such as GA, GOA,WOA, SCA, PSO and even some methods utilizing neural networks and deep learning. Nevertheless, in some studies it was shown that AOA still have some

demerits and modifications is needed to improved its performance. Firstly, AOA suffers from the problem of uneven exploration and exploitation phase, which decreases its searching capabilities in some optimization issues. This limits its ability in solving a wide range of optimization issues. Secondly, there is a high considerable risk that AOA would be entrap in the local optima. This is because of how AOA updates its object's position where it is dependent on its prior or best object position. Hence, it is worthwhile to thoroughly analyse and pursue these difficulties in order to deliver a better version of the AOA.

Based on the above literatures in Section 2.3, there are three notable issues can be highlighted. Firstly, in majority of past studies, the models of linear subsystems were described in discrete-time, while parameters of actual dynamic systems can be directly expressed from continuous-time domain. Secondly, the issue of identified parameters redundancy between linear and nonlinear subsystems still occurs in their Hammerstein model structure which could result to high computational burden thus degrade the estimation accuracy. Thirdly, most of the metaheuristic methods used in the identification of Hammerstein models are too complex and also utilized large number of pre-defined coefficients, which may increase the computational time and effort to determine the optimum setting of coefficients. With these three issues in mind, it is worth it to explore an enhanced version of the metaheuristic method as a tool to identify a more proper and practical structure of the Hammerstein model.

2.6   Summary

In the real world, nonlinearities are found in most systems. Building realistic models for such systems and processes remains a difficult task. Because of its simple structure and effective representation of nonlinear dynamic systems, the block-oriented Hammerstein model has been embraced by many researchers to simulate practical systems. Furthermore, the treatment of an optimization problem is a problem that is widely studied and debated by academics from a broad spectrum of domains. As the complexity of genuine nonlinear plants and processes grows, so does the need for novel metaheuristics methods to model the nonlinear system. Even though present optimization methods may successfully and efficiently identify nonlinear systems, the so-called no-free-lunch (NFL) theorem permits more academics to create and propose new optimization algorithms that will be more powerful in the future. As a result, the NFL

pushes us to create an algorithm for identifying the Hammerstein system that will outperform current optimization methods. Undoubtedly, certain metaheuristic methods for the identification of Hammerstein method are yet to be implemented, such as the AOA method. As a result, the fundamental justification for using the metaheuristics method to identify the Hammerstein model is that it successfully solves the nonlinear system identification problem.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter the new Improved Archimedes Algorithm (IAOA) is proposed for the identification of a continuous-time Hammerstein model. Firstly, the Hammerstein model identification problem is formulated in Section 3.2. Then, in Sections 3.3 the original Archimedes Optimization Algorithm (AOA) is briefly discussed. After that, the Improved Archimedes Algorithm (IAOA) is discussed in Sections 3.4. The Improved Archimedes Algorithm (IAOA) is proposed in this chapter for identifying a continuous-time Hammerstein model. In Section 3.2, the Hammerstein model identification problem is formulated. The original Archimedes Optimization Algorithm (AOA) is briefly discussed in Sections 3.3. After that, the Improved Archimedes Algorithm (IAOA) is discussed in Sections 3.4. Here, two modifications are proposed to the original AOA. In our first modification, a new design variable is introduced in the density decreasing factor equation from the original AOA is discussed in the subsection 3.4.1. The second modification is the addition of a safe updating mechanism to the original AOA is discussed in the subsection 3.4.2. Finally, the identification method for the continuous-time Hammerstein systems is stated in Section 3.5.

3.2 Problem Formulation

A nonlinear function β and a linear dynamic system Y with differential operator $w(:= \frac{d}{dt})$ make up the Hammerstein model. The output signal is denoted by the symbol $o(t)$, whereas the output signal disrupted by noise signal $v(t)$ is denoted by the notation $\acute{o}(t)$. The output function of $\acute{o}(t)$ can be represented as follows:

$$\acute{o}(t) = Y(w)\beta(u(t)) + v(t), \qquad\qquad 3.1$$

where

$$Y(w) = \frac{Z_b w^b + Z_{b-1} w^{b-1} + \cdots + Z_0}{w^e + X_{e-1} w^{e-1} + \cdots + X_0}, \qquad 3.2$$

and the nonlinear function's output is as follows:

$$\beta\big(u(t)\big) = \sum_{i=1}^{L} \zeta_i \omega_i(u(t)), \qquad\qquad 3.3$$

where $\omega(.)$ could be a polynomial or non-polynomial function. Several assumptions are made in this identification:

i. $b,\ e$ and i are known,
ii. $X_i (i = 1, 2, \dots, e-1)$, $Z_i (i = 1, 2, \dots, b-1)$ and $\zeta_i (i = 1, 2, \dots, L)$ are real numbers,
iii. $Z_b = 1,$ to acquire $Y(w)$ and $\beta\big(u(t)\big)$ uniquely,
iv. $\beta(0) = 0.$

The identified model is evaluated using the fitness function below, which is based on a quadratic output estimation error,

$$P(\bar{Y}, \bar{\beta}) = \sum_{\eta=0}^{N} (\acute{o}(\eta t_s) - \bar{o}(\eta t_s))^2, \qquad\qquad 3.4$$

where the symbol $\bar{Y}$ is the identified linear dynamic system of Y and the symbol $\bar{\beta}$ is the identified nonlinear function of β and $\bar{o}(t) = \bar{Y}(w)\bar{\beta}(u(t))$. Furthermore, the sampling time is denoted by the symbol t_s for $(u(t), \acute{o}(t))(t = 0, t_s, 2t_s, \dots\dots, Nt_s)$ and $\eta = 0, 1, \dots, N.$

3.3   Archimedes Optimization Algorithm (AOA)

Archimedes optimization algorithm (AOA) which is introduced in (Hashim et al., 2022) is inspired based on the law of physics known as Archimedes' principle. The Archimedes' principle states that when an object is completely or partial immersed in a fluid, the fluid exerts an upward force on the object equal to the weight of the fluid displaced by the object. AOA is a population-based method, where the individuals in the

population are immersed objects. These objects have varied density, volume and acceleration which affects the buoyancy of the objects. The idea of AOA is to reach a point where all objects is in equilibrium state with the fluid. This is reached when the net force of the fluid is equals to zero. The step-by-step procedure of the original AOA is given below:

Step 1-Initialization: The positions of all objects are initialized using Eq. (3.5):

$$x_i = lb_i + rand \times (ub_i - lb_i), \qquad\qquad 3.5$$

for i =1, 2, ..., n, where x_i is the ith object's position in a population with n number of objects. The upper and lower bounds of the search space are denoted by ub_i and lb_i, respectively. Next, the density (den), volume (vol), and acceleration (acc) of the ith objects are initialized using Eq. (3.6):

$$den_i = rand,$$
$$vol_i = rand, \qquad\qquad 3.6$$
$$acc_i = lb_i + rand \times (ub_i - lb_i),$$

where $rand$ is random vector in which its element is independently generated between [0, 1].

Step 2-Update density and volume: The updated density and volume are given by Eq. (3.7):

$$den_i^{k+1} = den_i^k + rand \times (den_{best} - den_i^k), \qquad\qquad 3.7$$
$$vol_i^{k+1} = vol_i^k + rand \times (vol_{best} - vol_i^k),$$

where k is the iteration number, den_{best} and vol_{best} are the best object's density and volume found so far, and $rand$ is another random vector where its element is independently generated between [0, 1].

Step 3-Transfer operator and density factor: Initially, the objects collide. Then, the objects will try to reach the equilibrium state. This is implanted in AOA by using the transfer operator TF which converts the search from exploration phase to exploitation phase. This phenomenon is expressed in Eq. (3.8):

$$TF = exp\left(\frac{k - k_{max}}{k_{max}}\right), \qquad\qquad 3.8$$

where k_{max} is the maximum number of iterations. Here, TF is designed to be gradually increased over iteration until it reaches 1. Moreover, the density decreasing factor, d, helps AOA on local search mechanism, which is expressed using Eq. (3.9):

$$d = exp\left(\frac{k - k_{max}}{k_{max}}\right) - \left(\frac{k}{k_{max}}\right). \qquad\qquad 3.9$$

Note that the decreasing factor d is introduced to give AOA the ability to converge in previously identified promising region. It's worth noting that careful control of TF and d will provide a balance in the exploration phase and exploitation phase in AOA.

Step 4.1- Exploration phase: If $TF \leq 0.5$, collision between objects occurs. Then, a random material (mr) is selected and the object's acceleration is updated for iteration $k+1$ using Eq. (3.10):

$$acc_i^{k+1} = \frac{den_{mr} + vol_{mr} \times acc_{mr}}{den_i^{k+1} \times vol_i^{k+1}}, \qquad\qquad 3.10$$

where den_{mr}, vol_{mr}, and acc_{mr} are the density, volume, and acceleration of a random material.

Step 4.2-Exploitation phase: On the other hand, if $TF > 0.5$, there is no collision between objects. Then, object's acceleration is updated for iteration $k + 1$ using Eq. (3.11):

$$acc_i^{k+1} = \frac{den_{mr} + vol_{mr} \times acc_{best}}{den_i^{k+1} \times vol_i^{k+1}}, \qquad\qquad 3.11$$

where acc_{best} is the acceleration of the best object.

Step 4.3- Normalize acceleration: The normalization of acceleration is performed using Eq. (3.12) to calculate the percentage of change:

$$acc_{i-norm}^{k+1} = u \times \frac{acc_i^{k+1} - min\,(acc)}{max(acc) - min\,(acc)} + l, \qquad\qquad 3.12$$

where u and l are range of normalization, which is set to 0.9 and 0.1, respectively. The acc_{i-norm}^{k+1} determines the percentage change of each object's step. If the object is close to the global optimum, the acceleration value will be low which means that the object

21

will be in the exploitation phase. Otherwise, it is in the exploitation phase. This shows how the search agent change from exploration to exploitation phase.

Step5-Update position: If $TF \leq 0.5$, the ith object's position is updated using Eq. (3.13):

$$x_i^{k+1} = x_i^k + C_1 \times rand \times acc_{i-norm}^{k+1} \times (x_{rand} - x_i^k) \times d, \qquad 3.13$$

where C_1 is a constant that equals to 2, x_{rand} is a random object's position and $rand$ is random vector in which its element is independently generated between [0, 1]. If $TF > 0.5$, the object's position is instead updated using Eq. (3.14):

$$x_i^{k+1} = x_{best} \times F \times C_2 \times rand \times acc_{i-norm}^{k+1} \times (T \times x_{best} - x_i^k) \times d, \qquad 3.14$$

where x_{best} is the best object's position and C_2 is a constant that equals to 6. Here, the variable T is defined by $T = C_3 \times T$, where C_3 is a constant that equals to 2. Note that T increases with each iteration and has a range of $[C_3 \times 0.3, 1]$. In Eq. (3.14):, F is the flag which changes the direction of motion expressed using Eq. (3.15):

$$F = \begin{cases} +1 \; if \; Pr \leq 0.5, \\ -1 \; if \; Pr > 0.5, \end{cases} \qquad 3.15$$

for $Pr = 2 \times rand - C_4$, where $rand$ is random vector where its element is independently generated in the range of [0, 1], and C_4 is a constant that equals to 0.5.

Step 6- Evaluation: Finally, evaluate each object's position using fitness function f and saves the best solutions found so far that are corresponded to the best solution x_{best}, den_{best}, vol_{best}, and acc_{best}.

The AOA flowchart is shown in Figure 3.1.

Begin

Determine the population size n, maximum iterations k_{max}, C_1, C_2, C_3, and C_4

Initialize objects population with random positions, densities, volumes, and acceleration using Eq. (1) and Eq. (2)

Evaluate initial population and select the one with the best fitness value.

Update density and volume of each object using Eq. (3), as well as update transfer and density decreasing factors TF and d using Eq. (4) and Eq. (5), respectively

$TF \leq 0.5$?

No

Yes

Update acceleration using Eq. (7) and normalize acceleration using Eq. (8)

Update acceleration, normalize acceleration, and object's position using Eq. (6), Eq. (8) and Eq. (9), respectively

$Pr > 0.5$?

No

Yes

$F = 1$

$F = -1$

Update object's position using Eq. (10)

Evaluate each object and select the one with the best fitness value and. Then set $k = k + 1$

$k \leq k_{max}$?

No

Yes

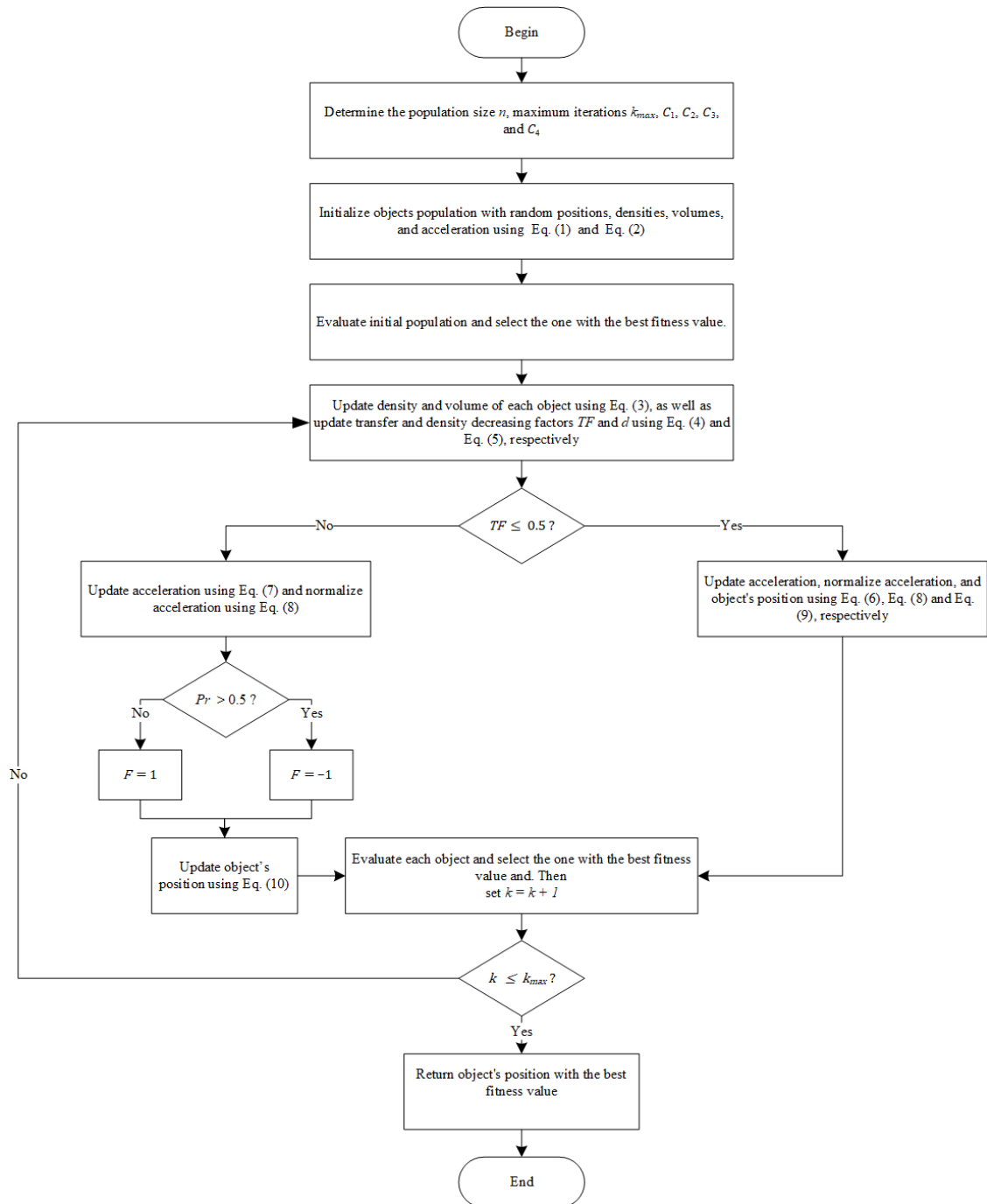Return object's position with the best fitness value

End

Figure 3.1   Flowchart of AOA

3.4 Improved Archimedes Optimization Algorithm (IAOA)

The original AOA is quite versatile and was utilized to solve a wide range of optimization problems, but it still has several demerits. Firstly, the imbalance in the exploration and exploitation phases, which will result in a decrease in its searching capabilities. Secondly, its ability to escape from local optima is hampered due to the way AOA update its object's positions which is by relying exclusively on its prior or best object position. Based on the demerits of AOA, the following changes are proposed to solve those issues

3.4.1 Modified Density Decreasing Factor

In the original AOA, the density decreasing factor, d in Eq. (3.9) decreases nonlinearly until it reaches 0, which is important to provide a proper balance between the exploration and exploitation phases. Nevertheless, the original density decreasing factor is too restrictive as well as the existing exploration and exploitation phases are limited to certain optimization problems only. In other words, the user doesn't have flexibility to control the exploration and exploitation phases according to the handled optimization problem. To address this demerit, a tuneable coefficient α is introduced in the density decreasing factor equation. The modified density decreasing factor equation is shown in Eq. (3.16):

$$\tilde{d} = exp\left(\frac{k - k_{max}}{k_{max}}\right)^{\alpha} - \left(\frac{k}{k_{max}}\right). \qquad 3.16$$

This modification allows the alteration of the ratio of exploration and exploitation phases throughout preliminary tuning of the coefficient α. Precisely, to increase the exploration phase, α can be set to be greater than 1 ($\alpha > 1.0$), which will increase the value of $\tilde{d}$ and thus provide large perturbation step to the object's position. Nevertheless, this setting will reduce the duration for exploitation phase. Conversely, by setting $\alpha < 1.0$, the exploitation phase is increased, where small value of $\tilde{d}$ is generated that will produce only small perturbation to the object's position. As a consequence, this setting might reduce the convergence speed since the duration of exploitation phase is much larger than exploration phase. Therefore, an optimal selection of α can provide a proper balance between the exploration and exploitation phases, thus provides more flexibility for the algorithm in handling various types of optimization problems. The response of $\tilde{d}$ for the

whole iterations corresponds to different values of α is clearly illustrated in Error! Reference source not found. below. As can be noticed, in comparison to its original AOA ($\alpha = 1$), this modification allows the tuning of ratio between the exploration and exploitation phases, which allows the method to be more compatible with a wider range of real-world applications.
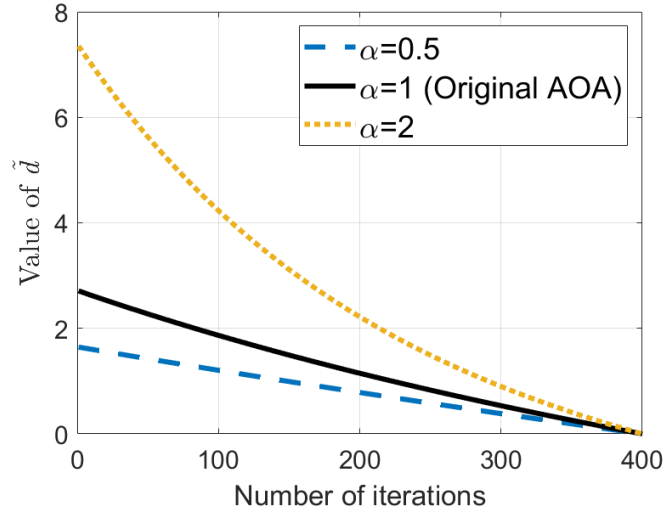
Figure 3.2   Value of $\tilde{d}$ when varying α

3.4.2   Safe updating mechanism

To solve the demerits of being trapped in a local optima found in the original AOA method, a safe updating mechanism is introduced. Specifically, some of the elements in the updated object's position vector after Eq. (3.13) and Eq. (3.14) will maintain its best object's position randomly according to a pre-defined probability. The safe updating mechanism is shown in Eq. (3.17) below:

$$x_{i,j}^{k+1} = \begin{cases} x_{best,j} & \text{if } rand > E, \\ x_{i,j}^{k+1} & \text{otherwise,} \end{cases} \qquad 3.17$$

where E is a probability constant. Note that this kind of updated mechanism is inspired from a safe experimentation dynamics algorithm in (Ghazali et al., 2019) which is in the class of game theoretic method. Here, a safe updating mechanism is guaranteed since only parts of the object's position is perturbed according to Eq. (3.13) and Eq. (3.14), which is when $rand \leq E$. Otherwise, it will maintain the element of the best object's

position vector, x_{best}. Further explanation is given with the aid of the illustration in Figure 3.3, where X and Y axes reflect the independent design variables/elements inside a previously created contour plot. Consider object x_i labelled by a red rectangle has been trapped in the local optimal region. Based on the original AOA method, this object x_i is most likely to remain trapped in that region due to weakness of its updating mechanism, which is solely guided by an object's current position x_i or object's best position x_{best}. Even so, this problem can be lessened thru the safe updating mechanism in which some elements of x_i will be changed according to the element in best object's position x_{best}. In the example given, the second element of the object x_i which is represented by the x-axis has been randomly changed according to the second element in x_{best}. Hence, it allows x_i to exit from the local optimal region and be perturbed to a new position (see the green rectangle). It's worth to mention that this example uses only two elements for ease of understanding. The same concept can be extended for a larger number of elements. Finally, the proposed hybrid technique which combining the good features from both modified density decreasing factor and safe updating mechanism methods is revealed by the pseudocode in **Algorithm 1.**
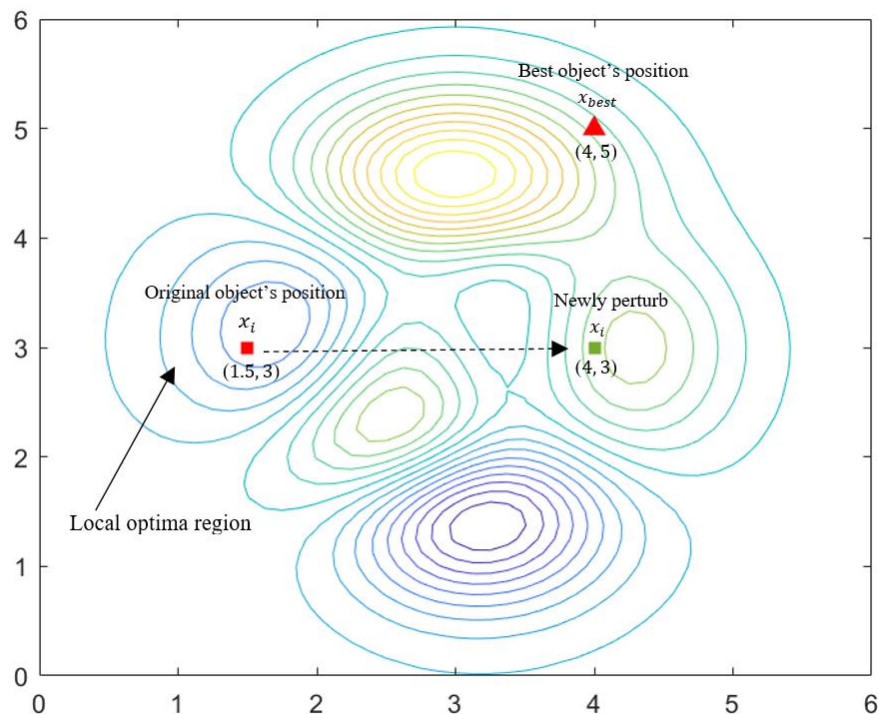

Figure 3.3   Graphical representation of IAOA with safe updating mechanism

Algorithm 1 Pseudo code of IAOA
procedure IAOA (population size n, maximum iterations, k_{max}, $C_1, C_2, C_3, C_4, \alpha$ and E)
Initialize objects population with random positions, densities, volumes, and acceleration using Eq. (3.5), Eq. (3.6), respectively.
Evaluate initial population and select the one with the best fitness value.
Set iteration counter $k = 1$
While $k \leq k_{max}$ **do**
for each object i **do**
Update density and volume of each object using Eq. (3.7), as well as update transfer and density decreasing factors TF and $\tilde{d}$ using Eq. (3.8) and Eq. (3.16), respectively.
If $TF \leq 0.5$ **then** $\rightarrow$ Exploration phase
Update acceleration, normalize acceleration and object's position using Eq. (3.10), Eq. (3.12) and Eq. (3.13), respectively
else $\rightarrow$ Exploitation phase
Update acceleration, normalize acceleration, direction flag F and object's position using Eq. (3.11), Eq. (3.12), Eq. (3.15) and Eq. (3.14)
end if
Update each object's position using Eq. (3.17)

3.5 Application of IAOA for Identifying the Hammerstein Model

This section discusses how IAOA method is utilized in the identification of continuous-time Hammerstein model. Specifically, the identification of the problem is identified by expressing the fitness function as:

$$P(\vartheta) = \sum_{\eta=0}^{N} (\acute{o}(\eta t_s) - \bar{o}(\eta t_s))^2, \qquad 3.18$$

for the design variable

$$\vartheta = [\bar{Z}_0, \bar{Z}_1, \ldots, \bar{Z}_{b-1}, \bar{X}_0, \bar{X}_1, \ldots, \bar{X}_{e-1}, \bar{\zeta}_1, \bar{\zeta}_2, \ldots, \bar{\zeta}_L] \in \mathbb{R}^m, \qquad 3.19$$

where the variable $m = b + e + L$. When ϑ is fixed, the value of $P(\vartheta)$ is calculated by running $\acute{o}(\eta t_s)$ and $\bar{o}(\eta t_s)$, which performs as follows. Firstly, the continuous-time input signal of $u(t)(t = 0, t_s, 2t_s, \ldots, Nt_s)$ is executed. Then, the continuous-time signal $\bar{o}(t)$ is computed using:

$$\bar{o}(t) = \frac{Z_b w^b + Z_{b-1} w^{b-1} + \cdots + Z_0}{w^e + X_{e-1} w^{e-1} + \cdots + X_0} \bar{\beta}(u(t)). \qquad 3.20$$

Then, with a fixed sampling time of $\eta = 0, 1, \ldots, N$, signal $\bar{o}(t)$ is sampled to $\bar{o}(\eta t_s)$. Finally, the continuous-time Hammerstein model is identified using the IAOA method. The following is a summary of the step-by-step procedure:

Step 1: Identify the design variable ϑ in Eq. (3.19).

Step 2: Execute the IAOA method in the pseudocode of **Algorithm 1**by setting $\vartheta := x_i$ and $P(\bar{Y}, \bar{\beta}) = f_i$.

Step 3: When k_{max} is reached, we obtained the best object's position x_{best}. Then, the best design variables, $\vartheta := x_{best}$ is the solution for the continuous-time Hammerstein model.

The continuous-time Hammerstein model identification flow using the proposed IAOA method is depicted in Figure 3.4. The identification of the continuous-time Hammerstein block and the proposed IAOA method block are the two key blocks in this diagram. In the former block, the goal is to obtain the identified model by using the given input $u(t)$ and output $ó(t)$ data. The fitness function is calculated in Eq. (3.4), which is corresponded as the outcome from this block. Then, the proposed IAOA method block uses this fitness function by defining $P(\bar{Y}, \bar{\beta})$ as the fitness value f_i. In the proposed IAOA method block, the pseudocode in **Algorithm 1** is employed to acquire the updated design variable of each object's position x_i. The updated design variable is then utilized in the identified model in the identification of the continuous-time Hammerstein model block by denotating $\vartheta := x_i$. This two-way flow between these two blocks is repeated until the maximum number of iterations are reached to determine the optimal design variables of the identified model.
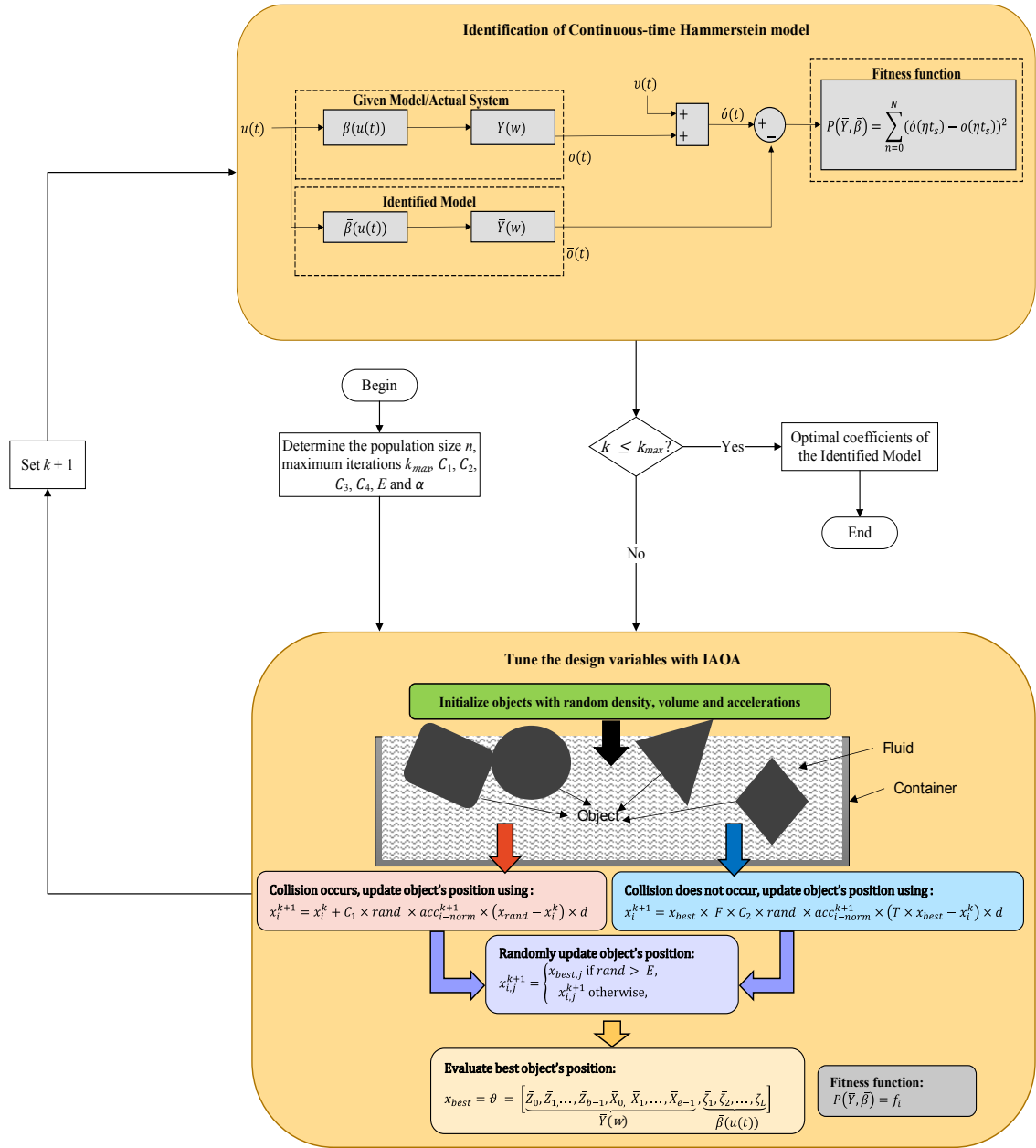
Given Model/Actual System

$u(t)$

$\beta(u(t))$ → $Y(w)$

$o(t)$

$v(t)$

$+$
$+$

$\acute{o}(t)$

$+$
$-$

Fitness function

$$P(\bar{Y},\bar{\beta}) = \sum_{n=0}^{N}(\acute{o}(\eta t_s) - \bar{o}(\eta t_s))^2$$

Identified Model

$\bar{\beta}(u(t))$ → $\bar{Y}(w)$

$\bar{o}(t)$

Begin

Determine the population size n, maximum iterations k_{max}, C_1, C_2, C_3, C_4, E and α

Set $k + 1$

$k \leq k_{max}$? — Yes → Optimal coefficients of the Identified Model → End

No

Tune the design variables with IAOA

Initialize objects with random density, volume and accelerations

Fluid

Container

Object

Collision occurs, update object's position using:
$$x_i^{k+1} = x_i^k + C_1 \times rand \times acc_{i-norm}^{k+1} \times (x_{rand} - x_i^k) \times d$$

Collision does not occur, update object's position using:
$$x_i^{k+1} = x_{best} \times F \times C_2 \times rand \times acc_{i-norm}^{k+1} \times (T \times x_{best} - x_i^k) \times d$$

Randomly update object's position:
$$x_{i,j}^{k+1} = \begin{cases} x_{best,j} \text{ if } rand > E, \\ x_{i,j}^{k+1} \text{ otherwise,} \end{cases}$$

Evaluate best object's position:
$$x_{best} = \vartheta = \left[\underbrace{\bar{Z}_0, \bar{Z}_1, \ldots, \bar{Z}_{b-1}, \bar{X}_0, \bar{X}_1, \ldots, \bar{X}_{e-1}}_{\bar{Y}(w)}, \underbrace{\bar{\zeta}_1, \bar{\zeta}_2, \ldots, \zeta_L}_{\bar{\beta}(u(t))}\right]$$

Fitness function:
$$P(\bar{Y},\bar{\beta}) = f_i$$

Figure 3.4   Block diagram of IAOA implementation for continuous-time Hammerstein Model identification

3.6   Summary

This chapter discusses about a general framework of the continuous-time Hammerstein model based on the improved archimedes optimization algorithm. Firstly, the proposed method for the identification of the continuous-time Hammerstein model problem was formulated where the Hammerstein model input and output data was given, and the linear and nonlinear subsystems needed to be identified so that the fitness function can be minimized. Next, the original archimedes optimization algorithm was briefly

discussed. Next, the modifications of the proposed IAOA method are discussed where two modifications are proposed. In the first modification, the searching capability of the AOA is improved by introducing the coefficient α to the density decreasing factor equation, which allows the alteration of the ratio of the exploration and exploitation phases. In the second modification, the local optima problem of the original AOA was solved by introducing the safe searching mechanism which allows the algorithm to escape from being trapped in a local optima. Finally, the application of the proposed IAOA method on the identification of continuous-time model of Hammerstein model is discussed, where the proposed method is applied on the Hammerstein model identification.

CHAPTER 4

RESULTS AND DISCUSSION

4.1　Introduction

This chapter presents the results of a performance investigation of the continuous-time Hammerstein model using the proposed IAOA method. The effectiveness of the proposed IAOA method is demonstrated here with one numerical example and two real world experiments, which are the Twin-Rotor System (TRS) and the Electro-Mechanical Positioning System (EMPS). The proposed IAOA method is then compared to the original AOA, AMVO-SCA, PSO, GWO, and MVO methods to evaluate its efficiency.

4.2　Performance Criteria

The following performance criteria are taken into account in this study:

1. The convergence curve of the best fitness function from 25 independent runs, variable deviation index δ, identified linear dynamic system $\bar{Y}(w)$ bode plot, plot of the identified nonlinear function $\bar{\beta}(u(t))$, responses of the identified TRS and EMPS model in time and frequency domain. The variable deviation index δ is calculated as follows:

$$\delta = \left\| \left[\frac{\vartheta_1 - \hat{\vartheta}_1}{\hat{\vartheta}_1}, \dots, \frac{\vartheta_m - \hat{\vartheta}_m}{\hat{\vartheta}_m} \right]^T \right\|_2, \qquad 4.1$$

where $\hat{\vartheta} \epsilon \mathbb{R}^m$ is the actual design variable vector. Furthermore, ϑ_i is the ith component of the identified design variable ϑ and $\hat{\vartheta}_i$ is the ith component of the design variable vector $\hat{\vartheta}$.

2. The evaluation of statistical fitness function performance values and variable deviation index values from 25 independent runs based on the mean, best, worst, and standard deviation (Std.) in comparison with IAOA, AOA, AMVO-SCA, PSO, GWO, and MVO methods for varied noise levels.

3. The statistical difference between the Algorithms is assessed using a non-parametric statistical test called Wilcoxon's rank test (García et al., 2009) at a significance level of 5%. The Wilcoxon's test is performed here using the fitness value across 25 independent runs. This statistical test uses two separate experimental results methods to calculate the p and h values, which will be used as significance level indicators. The experimental results of both methods are regarded statistically different when $p < 0.05$ or $h = 1$. We may also say that the efficiency of the two methods is substantial. Nevertheless, when $p > 0.05$ or $h = 0$, the experimental results of both techniques are deemed similar.

4.3   Numerical Example for identifying the Hammerstein Model

The linear and nonlinear subsystems of the numerical example , which were taken from (Jui & Ahmad, 2021), are given as follow:

$$Y(w) = \frac{Z(w)}{X(w)},$$

$$X(w) = w^6 + 10.0000w^5 + 54.7700w^4 + 156.8000w^3 + 87.0843w^2 + 25.2810w + 4.0197, \qquad 4.2$$
$$Z(w) = w^3,$$

$$\beta(u(t)) = 125(u(t) + 0.5u^2(t) + 0.25u^3(t)). \qquad 4.3$$

For this numerical example, $u(t)$ is a variable amplitude of Pseudo Random Binary Sequence (PBRS) signal, with amplitude ranging from $[-1,1]$. Meanwhile, in this study different levels of white noise signals $v(t)$ with zero mean are utilized. Specifically, we established three levels of noise variances, which are $\sigma_v^2 = 0.01$, $\sigma_v^2 = 0.25$ and $\sigma_v^2 = 1.0$. The sampling rate is $t_s = 1 \times 10^{-3}$ for signal $ó(t)$ with $N = 24000$. This numerical example optimization settings are $lb_i = 1$ for every i, $ub_i = 160$ for every i, $k_{max} = 400$, and $n = 25$, while the coefficients of IAOA are $C_1 = 2$, $C_2 = 6$, $C_3 = 2$, $C_4 = 0.5$, $E = 0.93$ and $\alpha = 2$. We conduct 25 independent runs using the same

coefficients to assess the efficiency of the proposed IAOA method against the randomization effect. Meanwhile, Table 4.1 Coefficients of AOA, AMVO-SCA, PSO, GWO, and MVO. Table 4.1 shows the coefficients of the AOA (Hashim et al., 2022), AMVO-SCA (Jui & Ahmad, 2021), PSO (Rauf et al., 2020), GWO (Mirjalili, Mohammad, et al., 2014) and MVO (Mirjalili et al., 2016) methods. The number of agents, maximum iterations, upper and lower bounds of all the compared methods are set to be the same as IAOA method for a fair comparison assessment.

Table 4.1　　　　Coefficients of AOA, AMVO-SCA, PSO, GWO, and MVO

Algorithms	AOA	AMVO-SCA	PSO	GWO	MVO
Coefficients	$C_1 = 2$ $C_2 = 6$ $C_3 = 2$ $C_4 = 0.5$	$w_{max} = 1$ $w_{min} = 0.2$ $p = 4$	$w_{max} = 0.9$ $w_{min} = 0.4$ $c_1 = 1.45$ $c_2 = 1.45$	$a = [2,0]$	$w_{max} = 1$ $w_{min} = 0.2$ $p = 6$
Reference	(Hashim et al., 2022)	(Jui & Ahmad, 2021)	(Rauf et al., 2020)	(Mirjalili, Mohammad, et al., 2014)	(Mirjalili et al., 2016)

The convergence curve of mean fitness function (from 25 independent runs) for the original AOA and proposed IAOA methods with noise variance of $\sigma_v^2 = 0.01$, $\sigma_v^2 = 0.25$ and $\sigma_v^2 = 1.0$ are shown in Figure 4.1(a), Figure 4.1(b) and Figure 4.1(c), respectively. The blue line represents the response of the proposed IAOA method while the dotted red line represents the response of the original AOA. The convergence curves of all levels of noise indicate that the proposed IAOA method is able to successfully minimize the specified fitness function in Eq. (3.18) and obtain better optimal solution as compared to the original AOA method. This can be clearly seen in the plots of Figure 4.1. The best-identified design variable values and their corresponding variable deviation index δ for multiple noise variances out of 25 independent runs are tabulated in Table 4.2. The tabulated values demonstrates that by using the proposed IAOA method, the identified design variable vector ϑ is near to the actual design variable vector $\hat{\vartheta}$, particularly when noise variance is small.

(a) Convergence curve with $\sigma_v^2 = 0.01$ (b) Convergence curve with $\sigma_v^2 = 0.25$

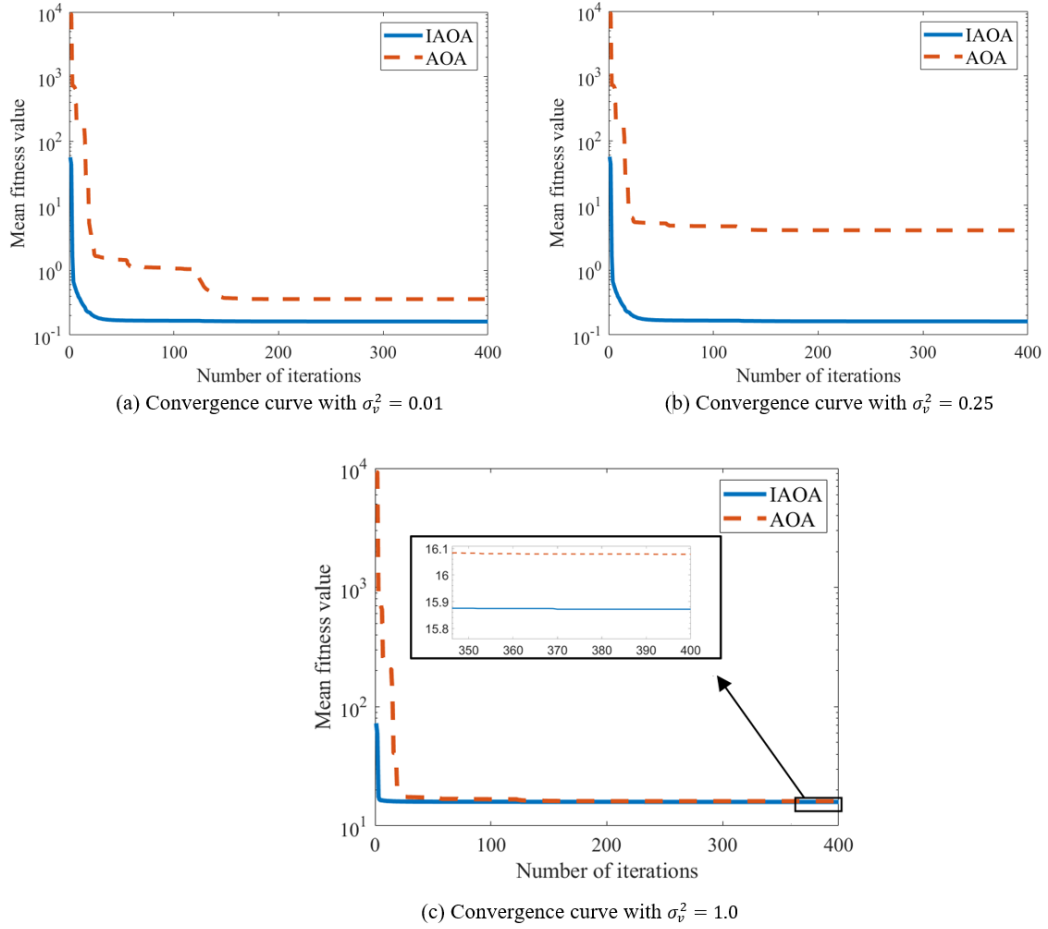(c) Convergence curve with $\sigma_v^2 = 1.0$

Figure 4.1   Convergence curve of the mean fitness function from 25 trials with several noise variances

Table 4.2   The best-identified design variables and its corresponding variable deviation index with several levels noise variances

ϑ^*	Design variables	ϑ			$\hat{\vartheta}$
		$\sigma_v^2 = 0.01$	$\sigma_v^2 = 0.25$	$\sigma_v^2 = 1.0$	
ϑ_1^*	$\bar{X}_0$	3.8087	3.5173	1.8540	4.0197
ϑ_2^*	$\bar{X}_1$	25.9138	21.8049	38.1805	25.2810
ϑ_3^*	$\bar{X}_2$	89.3507	103.7365	102.5055	87.0843
ϑ_4^*	$\bar{X}_3$	159.9760	155.3825	160.0000	156.8000
ϑ_5^*	$\bar{X}_4$	56.3632	59.1699	54.4352	54.7700
ϑ_6^*	$\bar{X}_5$	10.1784	10.6524	10.1838	10.0000
ϑ_7^*	$\bar{\zeta}_1$	130.9863	123.2149	95.7953	125.0000
ϑ_8^*	$\bar{\zeta}_2$	64.5855	67.0685	63.7901	62.5000
ϑ_9^*	$\bar{\zeta}_3$	30.5725	36.9534	45.5866	31.2500
δ		0.09752	0.3475	0.9210	-

Meanwhile, the results of the nonlinear function $\bar{\beta}(u(t))$ plot and the bode plot of the linear dynamic system $\bar{Y}(w)$ for the proposed IAOA method can be seen clearly in Figure 4.2 and Figure 4.3, respectively. Both plots in Figure 4.2 and Figure 4.3 are plotted using the best design variable (out of 25 independent runs). The thick solid-black colour line represents the actual response for both figures, while the identified response for the noise variances $\sigma_v^2 = 0.01$, $\sigma_v^2 = 0.25$ and $\sigma_v^2 = 1.0$ are represented by blue, red and yellow colour lines, respectively. In the nonlinear function plot responses in Figure 4.2, it is demonstrated that the proposed IAOA method can almost perfectly approximate the actual plot response of the nonlinear function $\bar{\beta}(u(t))$. Nonetheless, the difference between the actual plot and the identified plot widens as the noise variance increases, particularly for the noise variance of $\sigma_v^2 = 1.0$. Furthermore, the bode plot responses obtained by the proposed IAOA method are also almost identical to the actual bode plot response. According to the findings, it can be justified that the proposed IAOA method can closely predict the linear and nonlinear subsystems of the Hammerstein model, especially for low level of noise variance.
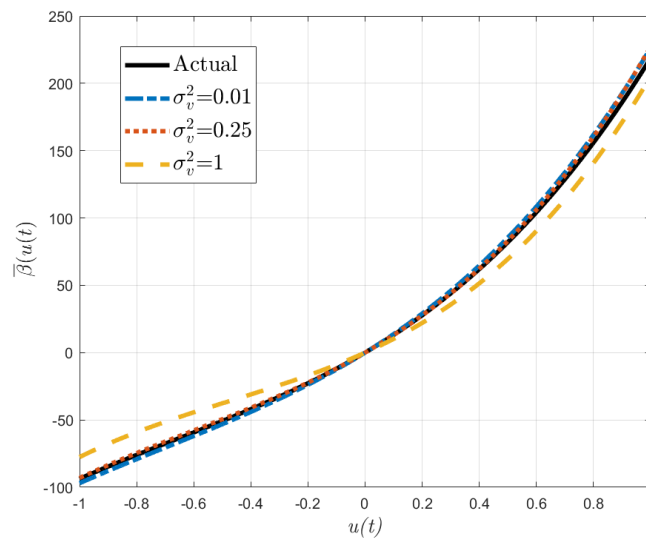

Figure 4.2   Identified nonlinear function $\bar{\beta}(u(t))$ with different noise variances
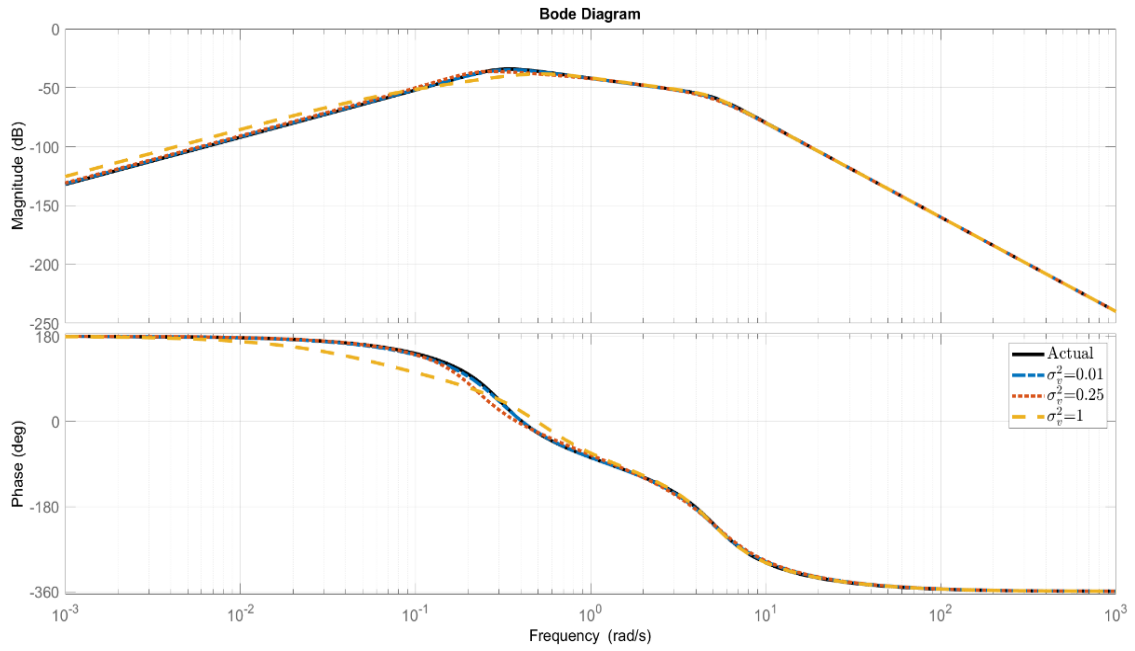
Figure 4.3   Identified linear dynamic system $\overline{Y}(w)$ with different noise variances

Moreover, the analysis of the statistical performance value of the fitness function and the variable deviation index with different noise variances, which are 0.01, 0.25 and 1.0 are shown in Table 4.3. Based on the 25 independent runs with different levels of noise variances, the mean, best, worst, and standard deviation (Std.) values of the fitness function and the variable deviation index are observed. The statistical results of our proposed method are also compared to the original AOA, AMVO-SCA, PSO, GWO, and MVO to determine the efficiency of the proposed method. Most of the findings in Table 4.3 demonstrate that a high degree of noise leads to a higher fitness function value, as can be observed from the mean, best, worst, and standard deviation values. In terms of the variable deviation index, however, the increment in mean, best, worst, and standard deviation values is rather minimal with the increment of noise. Among all the six methods, the proposed IAOA method produces a smaller best value of the fitness function compared to other methods for all levels of noise variances. Furthermore, the proposed IAOA method also yields smaller worst and standard deviation of the fitness function at noise variance level of for almost all noise variances compared to all other methods. Meanwhile, in terms of mean values, the AMVO-SCA method produces slightly smaller value than the proposed IAOA method. However, the mean values produced by the IAOA method is still very competitive with the AMVO-SCA method. In terms of variable deviation index, the proposed IAOA method is shown to be superior method in yielding

smaller best, worst and standard deviation values as compared to all other methods for almost all variances. These findings indicate the ability of the proposed IAOA method in producing precise and consistence results in most of the runs. As a result, the effectiveness of the proposed IAOA method in producing an accurate Hammerstein model is quite good compared to most of the other methods even with different noise levels.

Table 4.3   The analysis of statistical performance value of the fitness function and variable deviation index with several noise variances

Algorithm		IAOA			AOA			AMVO-SCA		
σ_v^2		0.01	0.25	1.0	0.01	0.25	1.0	0.01	0.25	1.0
$P(\vartheta)$	Mean	0.1619	3.9717	15.8717	0.3577	4.1272	16.0745	**0.1608**	**3.9707**	**15.8616**
	Best	**0.1581**	**3.9504**	**15.7960**	0.1668	3.9653	15.8464	0.1585	3.9584	15.8262
	Worst	**0.1695**	**3.9920**	15.9624	0.9114	4.6583	16.4773	0.1949	4.0157	**15.8930**
	Std.	**0.0029**	**0.0090**	0.0350	0.2125	0.1847	0.1841	0.0071	0.0118	**0.0172**
δ	Mean	1.1308	1.7703	1.9436	4.5862	4.4022	4.9474	**0.7551**	1.3573	**1.4814**
	Best	**0.0975**	**0.3475**	0.9210	0.8448	1.3468	1.5729	0.0983	0.4197	**0.4254**
	Worst	**2.5228**	**2.8617**	3.8011	10.0472	9.8189	10.3983	4.2554	3.0844	**3.3482**
	Std.	**0.6033**	**0.6164**	**0.6276**	2.7161	2.8120	2.7882	0.7806	0.7517	0.7622
Algorithm		PSO			GWO			MVO		
σ_v^2		0.01	0.25	1.0	0.01	0.25	1.0	0.01	0.25	1.0
$P(\vartheta)$	Mean	0.5562	4.3377	16.282	0.1629	3.9739	15.8753	0.1620	3.9771	15.8706
	Best	0.1667	3.9898	15.8659	0.1586	3.9615	15.8265	0.1594	3.9602	15.8261
	Worst	2.4391	5.8778	17.7504	0.1992	4.0056	15.9519	0.1767	4.0154	15.9067
	Std.	0.5946	0.4354	0.4336	0.0104	0.0092	0.0323	0.0034	0.0131	0.0229
δ	Mean	5.7767	6.0149	6.3934	1.0048	**1.311**	1.8739	1.0628	2.0303	2.3397
	Best	1.0336	1.2778	1.5427	0.2296	0.5524	0.545	0.3161	0.6013	0.3823
	Worst	17.995	18.2957	20.1424	4.3726	4.4314	7.7378	3.5464	6.229	9.3233
	Std.	4.3665	4.2566	4.8494	1.0429	1.0521	1.622	0.6694	1.135	1.7663

Furthermore, Table 4.4 tabulates the Wilcoxon's rank test findings for the fitness function between IAOA and AOA methods, with a significance level of 5%. It is shown that the *p*-values for the proposed IAOA method compared to the original AOA method are less than 0.05, as well as the *h*-values are equal to 1 for all levels of noise variances. This suggests that the obtained fitness function for the proposed IAOA method is statistically significant than its original version for all level of noises. It also indicates that when compared to the original AOA method, the proposed IAOA method performs statistically better in the pair-wise Wilcoxon rank test.

Table 4.4   Wilcoxon's rank test of the fitness function for IAOA and AOA methods

IAOA vs	Noise Variances					
	$\sigma_v^2 = 0.01$		$\sigma_v^2 = 0.25$		$\sigma_v^2 = 1.0$	
	p-value	*h*-value	*p*-value	*h*-value	*p*-value	*h*-value
AOA	2.5742e-09	1	1.3090e-07	1	9.5133e-08	1

4.4 Twin-Rotor System (TRS)

This section demonstrates the usefulness of the proposed IAOA method in identifying a real Twin-rotor System based on the continuous-time Hammerstein model. The TRS is an experimental system that contains the most important aspects of a helicopter, such as couplings and strong nonlinearities, and may be thought of as an unorthodox and sophisticated 'air vehicle.' These system properties provide substantial issues in modelling, control design, and implementation. The characteristics of a helicopter alter in a real flying environment due to changes in flight conditions. A system identification must be performed in various flying scenarios to upgrade the aircraft model. The TRS is made up of a main rotor and a tail rotor that are hinged on the structure's base at both ends of the horizontal beam and rotate freely in both horizontal and vertical planes. Furthermore, the horizontal beam may be adjusted to regulate the rotors' rotating speed by modifying the input voltage. The horizontal beam may rotate and move its ends on the spherical surfaces because of the joint. To balance the angular momentum, the system has a counterbalance pendulum suspended from the beam. In Figure 4.4, it depicts the schematic diagram of the TRS system (Toha et al., 2012). The main rotor blade revolves around the yaw axis, and the tail rotor blade rotates around the pitch axis, respectively, to shift the system in vertical and horizontal directions. However, due to the imbalanced mass distribution, the system's flexible rotation creates vibrations during operation.

Due to the vertical channel's input, vibrational motion occurs mostly at the pitch angle in the twin-rotor system. As a result, the vertical channel is picked as the TRS output. A random signal of 1 volt with a sample time of 0.1 s was employed as an input into the system's vertical channel. A total of 600 s of real input–output data was obtained to determine the TRS's vertical channel. The system was modelled using the same duration data. Figure 4.5 shows the block diagram of the continuous-time Hammerstein model that was used to verify the twin-rotor system's estimated model. In Figure 4.5, the symbol $\bar{o}(t)$ represents the estimated output, whereas $ó(t)$ and $u(t)$ represent the experimental system's recorded output and input, respectively. Figure 4.6(a) and Figure 4.6(b) illustrate the input signal $u(t)$ and the vertical channel output $ó(t)$, respectively. It is assumed that the measurement noise is included in the vertical channel output at this point.
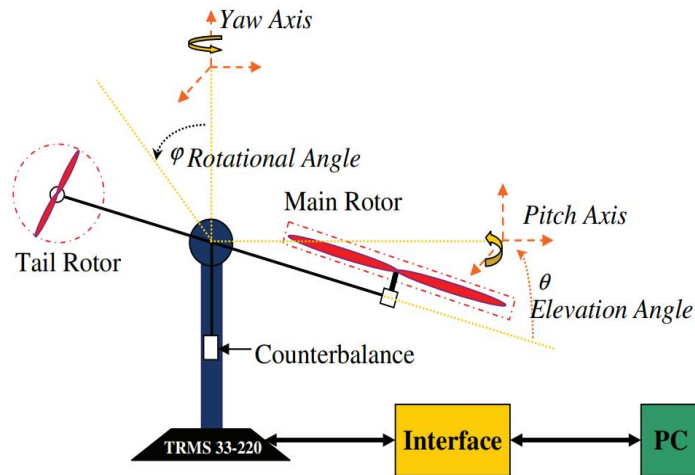
Figure 4.4   Twin-rotor system schematic diagram
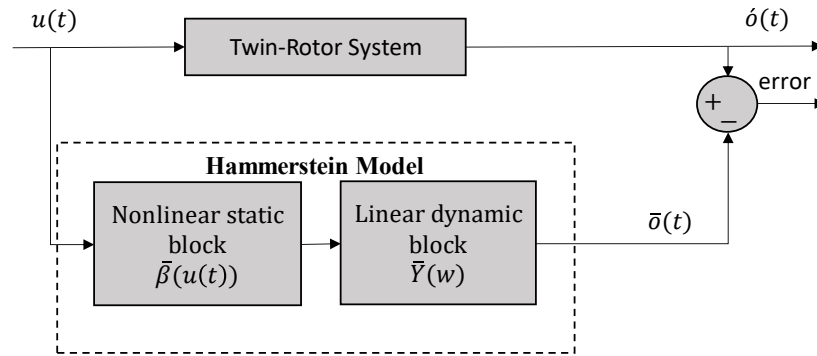Source: (Toha et al., 2012)

Figure 4.5   Block diagram of the Hammerstein model to identify the TRS.

In this experiment, the structure of nonlinear and linear subsystems is taken from (Toha et al., 2012), which are based on a tangent hyperbolic and a second-order continuous-time transfer function, respectively. Note that the nonlinear subsystem is responsible for detecting nonlinear friction at the rotor shaft in order to identify an accurate response of vertical motion. The linear and nonlinear subsystems are both represented by:

$$\bar{Y}(w) = \frac{w + \bar{Z}_0}{w^2 + \bar{X}_1 w + X_0}, \qquad\qquad 4.4$$

$$\bar{\beta}(u(t)) = \bar{\zeta}_1 \tanh(\bar{\zeta}_2 u(t)). \qquad\qquad 4.5$$

Based on the nonlinear model structure in Eq. (4.4) and Eq. (4.5), there are five unknown design variables that will be optimized by the IAOA method. The optimization settings for this TRS experiment are $lb_i = -10$ for every i, $ub_i = 10$ for every i, $k_{max} = 100$, and $n = 50$, while the coefficients of IAOA are $C_1 = 2, C_2 = 6, C_3 = 2, C_4 = 0.5, E = 0.92$ and $\alpha = 1.5$. The proposed IAOA method is compared against original AOA, AMVO-SCA, PSO, GWO, and MVO methods utilizing the identical optimization settings in this experiment. The original AOA, AMVO-SCA, PSO, GWO, and MVO coefficients are set to be the same values as in Table 4.1. Moreover, 25 independent runs were conducted to evaluate the statistical performance of the IAOA method in comparison to other methods.Figure 4.7 shows the convergence curve of the mean fitness functions from 25 independent runs for the proposed IAOA method and the original AOA method. These curves show that the proposed IAOA method, when compared to the original AOA method yields a better fitness function at the end of iterations. This finding confirms the effectiveness of the proposed modifications in the proposed IAOA method in achieving better Hammerstein model accuracy than its original AOA method. The best-identified design variable values for all the methods from 25 independent runs are tabulated in Table 4.5.

(a) Input Signal
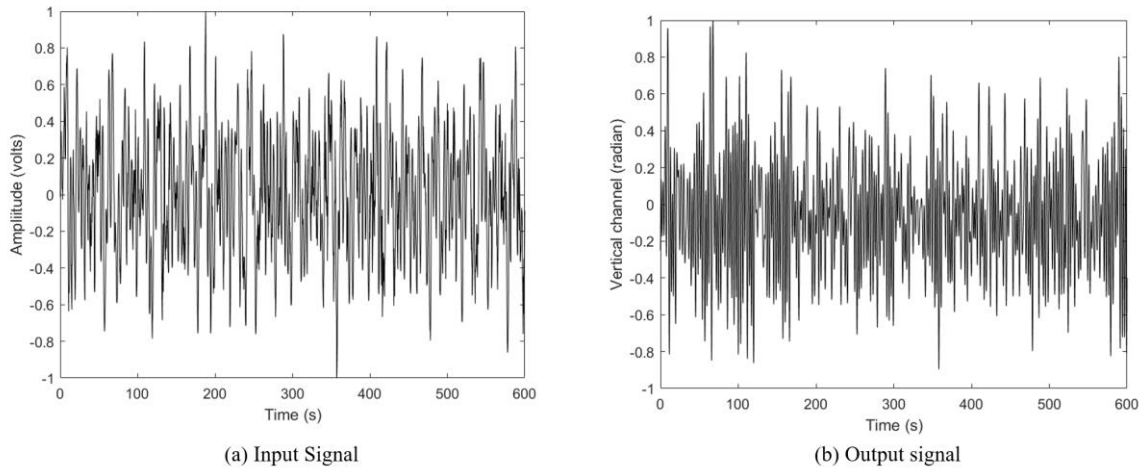
(b) Output signal

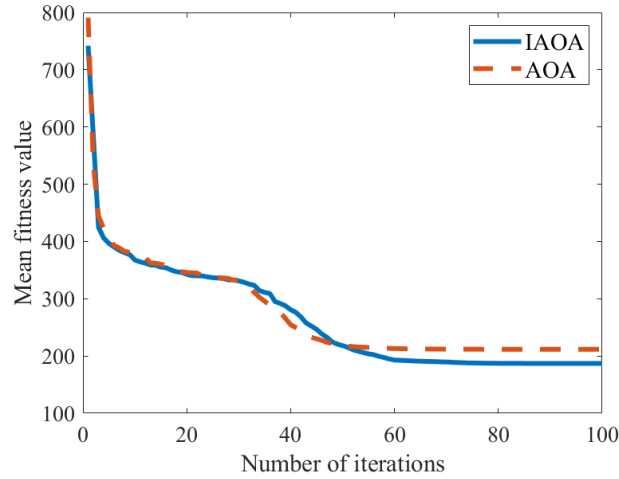Figure 4.6　　Twin-rotor system's random input and vertical channel output

Figure 4.7   Convergence curve of the mean fitness function from 25 independent runs

Table 4.5   The best-identified design variables of IAOA and other methods

ϑ^*	Design variables	IAOA	AOA	AMVO-SCA	PSO	GWO	MVO
ϑ_1^*	$\bar{Z}_0$	−1.8807	−1.7525	−1.9078	−1.8089	−1.8505	−2.9032
ϑ_2^*	$\bar{X}_0$	4.7637	4.7711	4.7607	4.7614	4.7654	4.7548
ϑ_3^*	$\bar{X}_1$	0.1077	0.1109	0.1085	0.1140	0.1076	0.1059
ϑ_4^*	$\bar{\zeta}_1$	3.2848	−2.9407	−9.8045	1.7972	2.2563	−2.3819
ϑ_5^*	$\bar{\zeta}_2$	−0.1712	0.2026	0.0574	−0.3283	−0.2512	0.2211

Moreover, Figure 4.8, Figure 4.9 and Figure 4.10 show the time domain vertical channel response, error response, and power spectrum density for the TRS vertical channel, respectively. Here, the black line indicates the actual response, while the green and red lines indicate the AOA and the IAOA methods' responses, respectively. Meanwhile, the zoomed in views of the output responses and error responses are illustrated in Figure 4.8(b) and Figure 4.9(b), respectively, to clearly visualize the response differences between the proposed IAOA method and the original AOA method. Figure 4.8(b) also shows that the IAOA method was able to produce a vertical channel response that almost mimic the actual signal from the twin-rotor's hardware system. Likewise, the error produced by the proposed IAOA method is slightly smaller compared to the original AOA, as shown in Figure 4.9(b). Furthermore, in Figure 4.10, the proposed IAOA method is also able to successfully identify one dominant resonance mode at 0.35 Hz in the actual power-spectral density (PSD) plot.
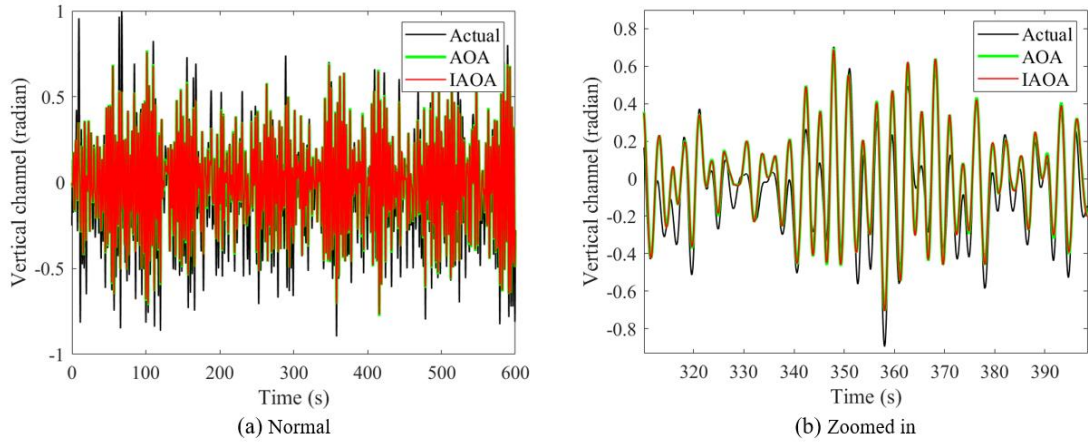
(a) Normal   (b) Zoomed in

Figure 4.8   Identified output responses of the vertical channel of the twin-rotor system

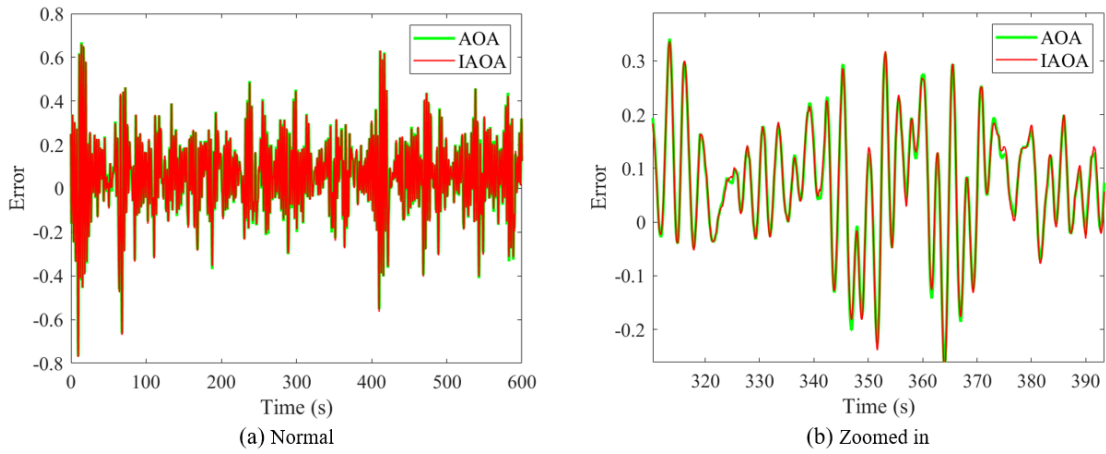(a) Normal   (b) Zoomed in

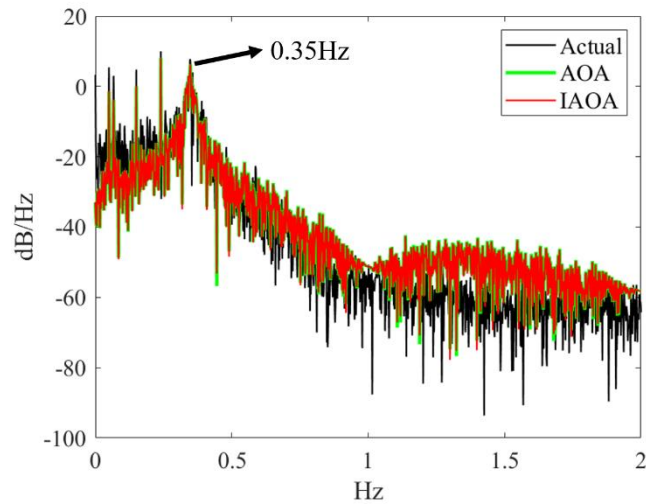Figure 4.9   Error produced by the identified continuous-time Hammerstein Models

Figure 4.10   Power spectrum density of the vertical channel

42

In addition, Table 4.6 demonstrates the analysis of the fitness function's statistical performance value for all methods. The mean, best, worst, and standard deviation of the fitness function are calculated from the 25 independent runs. The tabular results demonstrate that the proposed IAOA method produces smaller mean, worst and standard deviation values than the other five methods. Nevertheless, the best fitness function values for all methods are quite competitive where the AMVO-SCA method yields the smallest value amongst other methods. In overall, it can be justified that for real experiment of nonlinear system identification, the proposed IAOA method still can produce an accurate continuous-time Hammerstein model with higher consistency in most of the runs when compared to other methods.

Table 4.6   The analysis of the fitness function's statistical performance value for all methods.

Algorithm	IAOA	AOA	AMVO-SCA	PSO	GWO	MVO
Mean	187.0144	211.6344	218.5541	293.1188	282.1162	237.1052
Best	178.4015	178.8350	177.9563	180.1552	179.0303	179.6864
Worst	223.4002	347.8677	339.1072	387.7925	339.1171	366.4712
Std.	12.0419	58.3369	69.0269	76.741	77.1528	59.6191

Meanwhile, by performing the Wilcoxon's rank test of the fitness function for the proposed IAOA method with the original AOA method, the p-value and h-value obtained is 0.0283 and 1, respectively. As a result of the p-value obtained is less than 0.05 and the h-value obtained is equal to 1, this supports the fact that the obtained fitness function for the proposed IAOA method is statistically significant than its original AOA method. It also indicates that the proposed IAOA method outperforms the original AOA method in the pair-wise Wilcoxon rank test statistically.

4.5   Electro-mechanical Positioning System

This section demonstrates the performance of the proposed IAOA method in identifying an Electro-Mechanical Positioning System (EMPS) based on the continuous-time Hammerstein model. Figure 4.11 from the study (Brunot, 2019) shows an EMPS arrangement with a conventional drive system configuration that is routinely used in both prismatic joint robotics and industrial applications. A controller has been preliminarily attached to the system alongside a DC motor equipped with a 12500 counts per resolution encoder, as seen from the left side of the exhibited structure. The positioning device is moved by a DC motor that is attached to a star high-precision low-friction ball screw. A second encoder is attached to the ball screw's edge to measure a different location. The object for 45 positional measurement is a load in translation situated in the centre of the

EMPS with an associated accelerometer. It should be noted that the arithmetic data obtained from the encoder at the ball screw's edge and the accelerometer is inapplicable for this study. Instead, the input and output data's trustworthiness were secured by their direct adoption from (Brunot, 2019), with the quality and periodicity of the data being further assured by the use of the dSPACE digital control system. Both the input and output data were defined at a 12-second period with a 0.001-second sampling rate. Because processing was not accounted for, the data involved might be raw. As a result, both the input and output data for the investigated EMPS are based on the force indicated in Figure 4.12(a) and the position shown in Figure 4.12(b), respectively. Figure 4.13 also shows the block diagram of the continuous-time Hammerstein model, which is utilized as a framework to evaluate EMPS's estimated model.
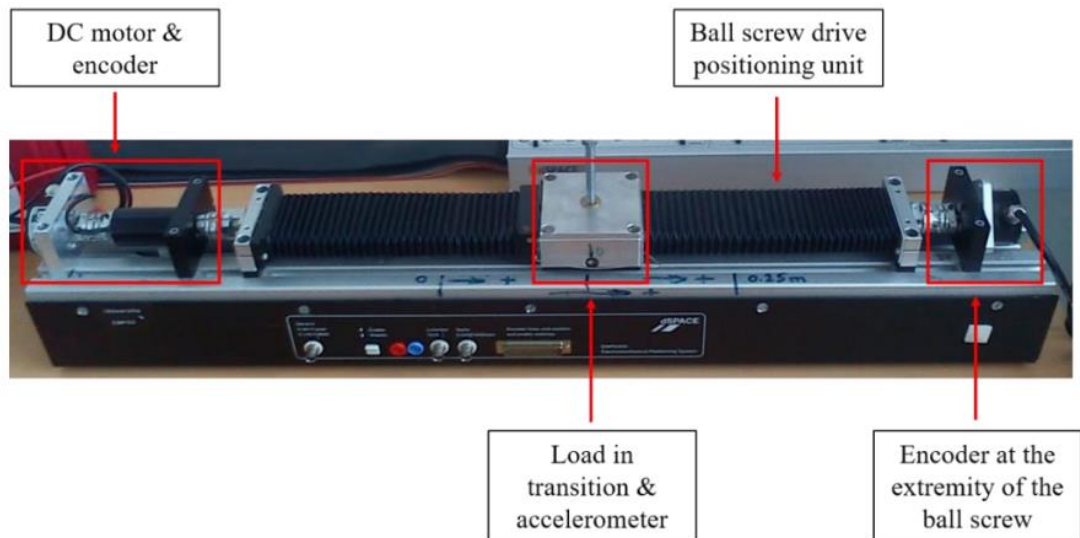

Figure 4.11   Prototype of the EMPS

Source: (Brunot, 2019)
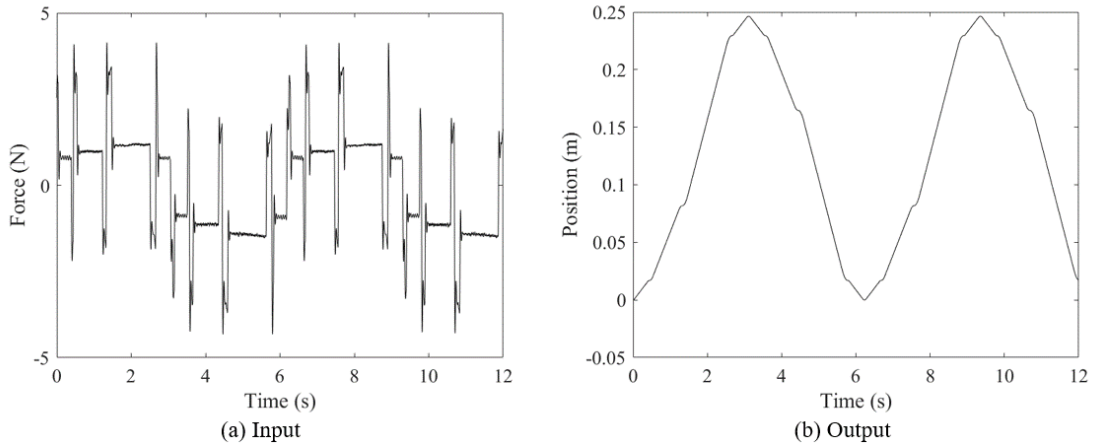
(a) Input   (b) Output

Figure 4.12   Input and output signals of the EMPS
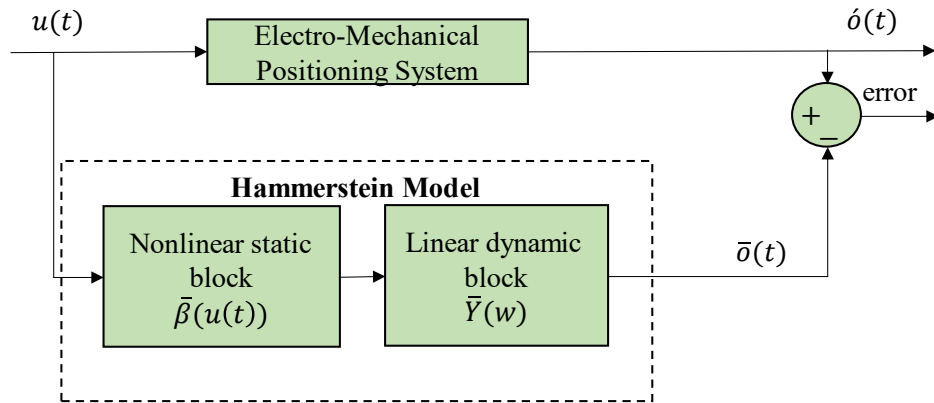

Figure 4.13   Block diagram of the Hammerstein model to identify the EMPS

The linear and nonlinear subsystems of the EMPS are directly adopted from (Brunot, 2019). The linear and nonlinear subsystems are expressed as follows:

$$\bar{Y}(w) = \frac{1}{w^2 + \bar{X}_0 w},$$ 4.6

$$\bar{\beta}(u(t)) = \bar{\zeta}_1 u(t) + \bar{\zeta}_2 \text{sign}(u(t)) + \bar{\zeta}_3.$$ 4.7

The nonlinear subsystem in Eq. (4.6) account for both the Coulomb friction and offset effects, while the linear subsystem in Eq. (4.7) is a second-order transfer function. From the nonlinear model in Eq. (4.6) and Eq. (4.7), there is a total of four unknown design variables to be optimized by the proposed IAOA method. The optimization settings for this EMPS experiment are $lb_i = -1$ for every i, $ub_i = 100$ for every i, $k_{max} = 100$,

45

and $n = 50$, while the coefficients of IAOA are $C_1 = 2, C_2 = 6, C_3 = 2, C_4 = 0.5, E = 0.97$ and $\alpha = 2$. Similar to the TRS experiment, the proposed IAOA method is compared against the same five other methods with identical optimization settings. The coefficients of all other methods are set to the same values as in Table 4.1. Furthermore, to evaluate the statistical performance of the IAOA method in comparison to all other methods, 25 independent runs were conducted.

The convergence curve of the mean fitness functions from 25 independent runs for the IAOA method and the original AOA method can be observed in Figure 4.14. From this figure, we can conclude that the IAOA method yields a better mean fitness function at the end of the iterations when compared to the original AOA method. This outcome confirms the ability of the proposed IAOA method in yielding a continuous-time Hammerstein model with better accuracy for most of the runs than the original AOA method. The best-identified design variable values for all the methods from 25 independent runs are tabulated in Table 4.7.
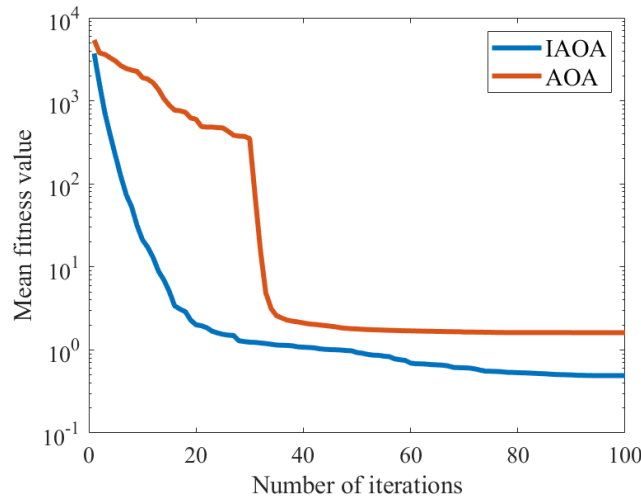

Figure 4.14   Convergence curve of the mean fitness function from 25 independent runs.

Table 4.7   The best-identified design variables of IAOA and other methods

ϑ^*	Design variables	IAOA	AOA	AMVO-SCA	PSO	GWO	MVO
ϑ_2^*	$\bar{X}_0$	3.0981	3.1866	3.1626	3.0283	3.1493	3.0610
ϑ_3^*	$\bar{\zeta}_1$	0.5519	0.4797	0.5485	0.5561	0.5408	0.5619
ϑ_4^*	$\bar{\zeta}_2$	-0.4127	-0.3036	-0.4019	-0.4254	-0.3925	-0.4313
ϑ_5^*	$\bar{\zeta}_3$	0.0457	0.0390	0.0457	0.0460	0.0448	0.0465

The time domain position response, the zoomed in time domain position response and the error response of the EMPS experimental results are shown in Figure 4.15(a), Figure 4.15(b) and Figure 4.15(c) respectively. Here, the lines are labelled in a similar way to the TRS experiment. It is shown that the position output response obtained by the IAOA method closely resembles the actual position recorded from the real EMPS hardware, which could be clearly observed in Figure 4.15(b). This is correlate with the error response produced by the IAOA method that is smaller when compared with the original AOA method.

(a) Time domain position response (normal)   (b) Time domain position response (Zoomed in)
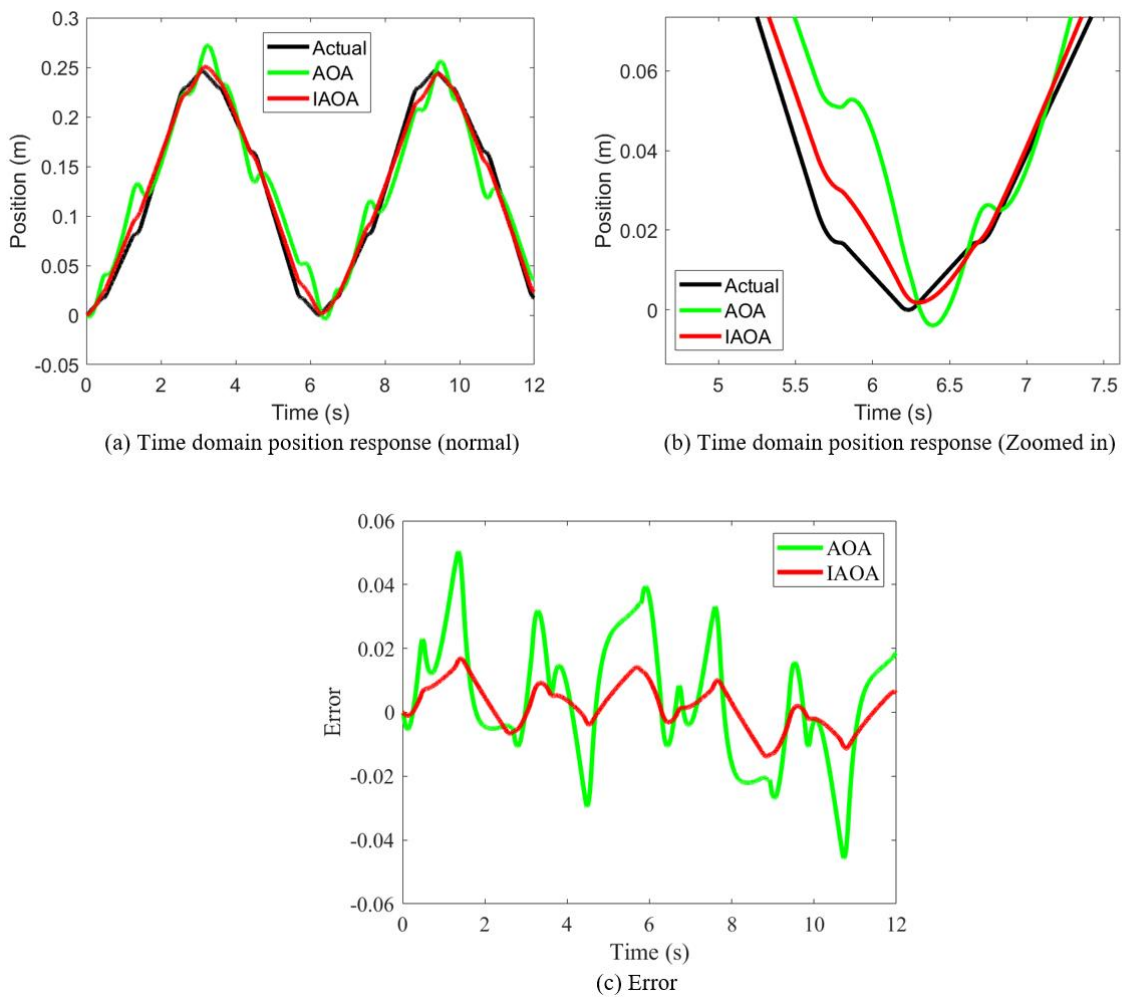
(c) Error

Figure 4.15   EMPS experimental results

Moreover, the analysis of the fitness function's statistical performance values for all methods are tabulated in Error! Reference source not found.. The fitness function's mean, best, worst, and standard deviation are calculated from the 25 independent runs. From these tabulated results, the proposed IAOA method produces a smaller mean, worst,

and standard deviation fitness function values than all the other methods, while the best fitness function value is competitive with the GWO method. Altogether, the results obtained is similar to that of from the TRS experiment, where the proposed IAOA method in this experiment when compared to other methods is able to produce an accurate continuous-time Hammerstein model with small deviations in most of the runs.

Table 4.8   The analysis of the fitness function's statistical performance value for all methods

Algorithm	IAOA	AOA	AMVO-SCA	PSO	GWO	MVO
Mean	**0.4904**	1.6148	0.7151	1.3149	0.7491	1.4104
Best	0.2093	0.2226	0.2103	0.2140	**0.2087**	0.2157
Worst	**0.9241**	8.2644	1.6604	4.6765	1.3752	4.6393
Std.	**0.2388**	1.5740	0.4704	1.3997	0.4303	1.0990

Meanwhile, the Wilcoxon's rank test of the fitness function for the proposed IAOA method with the original AOA method was conducted. Similar to the TRS experiment, the obtained p-value is 0.0209, which is less than 0.005, while the obtained h-value equal to 1. Hence, the obtained fitness function results for the proposed IAOA method is statistically significant compared to the original AOA method.

4.6   Summary

In this chapter, the results and discussion of the proposed IAOA method on the three experiments was discussed. Firstly, the efficacy of the proposed method is validated by a numerical example and two real-world experiments. The two experiments are the electro-mechanical positioning system and the twin-rotor system. The efficacy of the proposed method is analyse based on the convergence curve of the mean fitness function from 25 independent runs, evaluation of variable deviation index, bode plot and nonlinear function plot of identified linear and nonlinear subsystem of the numerical example, time-domain and frequency-domain responses of the identified electro-mechanical positioning system and twin-rotor system. In addition, the statistical performance values of the fitness function (mean, best, worst, and standard deviation) and the variable deviation index from 25 independent runs of IAOA was compared with five other methods, which includes AOA, AMVO-SCA, GWO, and MVO. From the findings of the statistical data for the numerical example it was shown that the proposed IAOA method is able to yield a Hammerstein model which is more accurate, with lower variable deviation index when

compared with the other methods at various levels of noise variations. This can be further justified from the findings of both real-world experiments, where the proposed IAOA method is able to effectively model both the twin-rotor and electro-mechanical positioning systems. Lastly, based on the findings of the Wilcoxon's rank-sum test in all three experiments, it was found that the proposed IAOA method yields statistically significant results when compared with the original AOA method.

CHAPTER 5

CONCLUSION

5.1   Concluding Remarks

In this study, a novel optimization metaheuristic method for the identification of continuous-time Hammerstein models based on the Improved Archimedes optimization algorithm (IAOA) is proposed. The numerical example results suggest that the proposed IAOA methods have the ability to identify continuous-time Hammerstein systems, even with a wide range of noise variations. Precisely, the proposed method has appear to be effective in identifying both the linear and nonlinear subsystems of a Hammerstein model as both the quadratic output estimation error in the fitness function and the variable deviation index is quite small. More notably, the proposed method was also able to adequately model two real world experiments based on the continuous-time Hammerstein model, this includes the electro-mechanical positioning system and the twin-rotor system and. In the numerical example and both real world experiments, the modelling results are statistically validated by the Wilcoxon rank test analysis. Through this analysis, it has shown that the fitness function accuracy produce by the proposed method is statistically significant compared to the original AOA method, which is congruence by the lower error between the actual and estimated output responses when compared to the original AOA method. Hence, the proposed modifications to solve the imbalance exploration and exploitations as well as the local optima issues, are efficient towards improving the original AOA in producing an accurate nonlinear model. Furthermore, the proposed IAOA method also outperforms conventional optimization metaheuristic methods in majority of the test experiments in terms of results accuracy and consistency.

5.2   Research contribution

(i)   The modification of the density decreasing factor is made to the original AOA to regulate the imbalance exploration and exploitation phases, which will improve the searching capabilities of AOA.

(ii)   A safe updating mechanism is introduced in the proposed IAOA method, where it will aid the AOA to flee from the local optima zone. This mechanism allows the updated object's position to be included in the design variable of the latest best object's position, which is chosen at random using a pre-defined probability.

(iii)   Many academics had proposed Hammerstein models in discrete time. Nonetheless, in the proposed Hammerstein model, a continuous-time domain will be established because it directly reflects parameters of real plants. As a result, this study will develop a new structure for continuous-time identification of the Hammerstein model based on IAOA method.

(iv)   The proposed Hammerstein identification structure avoids the problem of gain redundancy between nonlinear and linear subsystem, which lead to high computational burden. Thus, as compared to other existing approaches, our proposed framework may require less computing effort.

(v)   In comparison to other existing methods, the proposed IAOA method requires a reduced number of coefficients. which may diminish the preparatory work required to discover the optimal coefficients for the setting of the method.

5.3   Recommendations and Future Research Works

For future research direction, the current modifications could also be applied to other optimization methods which have similar limitations to the original AOA method. Next, the proposed IAOA method could also be applied to various types of nonlinear systems for instance the Wiener and the Hammerstein-Wiener models, multi-input multi output (MIMO) or even discrete optimization problems. In addition to those, the IAOA method can be further utilized in solving different types real world engineering problems.

REFERENCES

Abbas, A., Abdelsamea, M. M., & Gaber, M. M. (2021). Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. *Applied Intelligence*, *51*(2), 854–864. https://doi.org/10.1007/s10489-020-01829-7

Ait Maatallah, O., Achuthan, A., Janoyan, K., & Marzocca, P. (2015). Recursive wind speed forecasting based on Hammerstein Auto-Regressive model. *Applied Energy*, *145*, 191–197. https://doi.org/10.1016/j.apenergy.2015.02.032

Akdag, O. (2022). A Improved Archimedes Optimization Algorithm for multi/single-objective Optimal Power Flow. *Electric Power Systems Research*, *206*(January), 107796. https://doi.org/10.1016/j.epsr.2022.107796

Akramizadeh, A., Farjami, A. A., & Khaloozadeh, H. (2002). Nonlinear Hammerstein model identification using genetic algorithm. *Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*, 351–356. https://doi.org/10.1109/ICAIS.2002.1048126

Aribowo, W., Muslim, S., Suprianto, B., Haryudo, S. I., & Hermawan, A. C. (2021). Intelligent Control of Power System Stabilizer Based on Archimedes Optimization Algorithm – Feed Forward Neural Network. *International Journal of Intelligent Engineering and Systems*, *14*(3), 43–53. https://doi.org/10.22266/ijies2021.0630.05

Brunot, A. J. M. G. M. (2019). *Data Set and Reference Models of EMPS Data Set and Reference Models of EMPS*.

Chaudhary, N. I., & Raja, M. A. Z. (2015). Identification of Hammerstein nonlinear ARMAX systems using nonlinear adaptive algorithms. *Nonlinear Dynamics*, *79*(2), 1385–1397. https://doi.org/10.1007/s11071-014-1748-8

Chen, L., & Rezaei, T. (2021). A New Optimal Diagnosis System for Coronavirus (COVID-19) Diagnosis Based on Archimedes Optimization Algorithm on Chest X-Ray Images. *Computational Intelligence and Neuroscience*, *2021*. https://doi.org/10.1155/2021/7788491

Cuevas, E., Díaz, P., Avalos, O., Zaldívar, D., & Pérez-Cisneros, M. (2018). Nonlinear system identification based on ANFIS-Hammerstein model using Gravitational search algorithm. *Applied Intelligence*, *48*(1), 182–203. https://doi.org/10.1007/s10489-017-0969-1

Dehaene, S. (2003). The neural basis of the Weber-Fechner law: A logarithmic mental number line. *Trends in Cognitive Sciences*, *7*(4), 145–147. https://doi.org/10.1016/S1364-6613(03)00055-X

Desuky, A. S., Hussain, S., Kausar, S., Islam, M. A., & Bakrawy, L. M. E. (2021). EAOA: An Enhanced Archimedes Optimization Algorithm for Feature Selection in Classification.

IEEE Access, *9*(August), 120795–120814. https://doi.org/10.1109/ACCESS.2021.3108533

Ding, F., Liu, P., & Liu, G. (2011). Identification methods for Hammerstein nonlinear systems. *Digital Signal Processing*, *21*, 215–238. https://doi.org/10.1016/j.dsp.2010.06.006

Ding, J., Wang, Q., Zhang, Q., Ye, Q., & Ma, Y. (2019). A Hybrid Particle Swarm Optimization-Cuckoo Search Algorithm and Its Engineering Applications. *Mathematical Problems in Engineering*, *2019*, 1–12. https://doi.org/10.1155/2019/5213759

Eskinat, E., Johnson, S. H., & Luyben, W. L. (1991). Use of Hammerstein models in identification of nonlinear systems. *AIChE Journal*, *37*(2), 255–268. https://doi.org/10.1002/aic.690370211

Farahat, W., & Herr, H. (2005). A method for identification of electrically stimulated muscle. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, *7 VOLS*, 6225–6228. https://doi.org/10.1109/iembs.2005.1615918

Fathy, A., Alharbi, A. G., Alshammari, S., & Hasanien, H. M. (2022). Archimedes optimization algorithm based maximum power point tracker for wind energy generation system. *Ain Shams Engineering Journal*, *13*(2), 101548. https://doi.org/10.1016/j.asej.2021.06.032

Ganguli, S., Kaur, G., & Sarkar, P. (2019). A hybrid intelligent technique for model order reduction in the delta domain: a unified approach. *Soft Computing*, *23*(13), 4801–4814. https://doi.org/10.1007/s00500-018-3137-6

García, S., Molina, D., Lozano, M., & Herrera, F. (2009). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization. *Journal of Heuristics*, *15*(6), 617–644. https://doi.org/10.1007/s10732-008-9080-4

Ghazali, M. R., Ahmad, M. A., & Ismail, R. M. T. R. (2019). Data-Driven Neuroendocrine-PID Tuning Based on Safe Experimentation Dynamics for Control of TITO Coupled Tank System with Stochastic Input Delay. *Communications in Computer and Information Science*, *1015*, 1–12. https://doi.org/10.1007/978-981-13-7780-8_1

Hachino, T., Shimoda, K., & Takata, H. (2009). Hybrid Algorithm for Hammerstein System Identification Using Genetic Algorithm and Particle Swarm Optimization. *Engineering and Technology*, *3*(5), 499–504.

Hashim, F. A., Houssein, E. H., Hussain, K., Mabrouk, M. S., & Al-Atabany, W. (2022). Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. *Mathematics and Computers in Simulation*, *192*, 84–110. https://doi.org/10.1016/j.matcom.2021.08.013

Houssein, E. H., Helmy, B. E., Rezk, H., & Nassef, A. M. (2021). An enhanced Archimedes

optimization algorithm based on Local escaping operator and Orthogonal learning for PEM fuel cell parameter identification. *Engineering Applications of Artificial Intelligence*, *103*, 104309. https://doi.org/10.1016/j.engappai.2021.104309

Ismael, A. M., & Şengür, A. (2021). Deep learning approaches for COVID-19 detection based on chest X-ray images. *Expert Systems with Applications*, *164*(March 2020). https://doi.org/10.1016/j.eswa.2020.114054

Jui, J. J., & Ahmad, M. A. (2021). A hybrid metaheuristic algorithm for identification of continuous-time Hammerstein systems. *Applied Mathematical Modelling*, *95*, 339–360. https://doi.org/10.1016/j.apm.2021.01.023

Jui, J. J., Ahmad, M. A., & Rashid, M. I. M. (2022). Metaheuristics algorithms to identify nonlinear Hammerstein model: a decade survey. *Bulletin of Electrical Engineering and Informatics*, *11*(1), 454–465. https://doi.org/10.11591/eei.v11i1.3296

Jurado, F., Valverde, M., & Gómez, M. (2006). Identification of hammerstein model for solid oxide fuel cells. *Canadian Conference on Electrical and Computer Engineering*, 442–445. https://doi.org/10.1109/CCECE.2006.277371

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, *4*, 1942–1948. https://doi.org/10.1109/ICNN.1995.488968

Li, Y., Zhu, H., Wang, D., Wang, K., Kong, W., & Wu, X. (2021). Comprehensive optimization of distributed generation considering network reconstruction based on Archimedes optimization algorithm. *IOP Conference Series: Earth and Environmental Science*, *647*(1). https://doi.org/10.1088/1755-1315/647/1/012031

Madić, M., Marković, D., & Radovanović, M. (2013). Comparison of Meta-Heuristic Algorithms for Solving Machining Optimization Problems. *FACTA UNIVERSITATIS Series: Mechanical Engineering*, *11*(1), 29–44.

Manenti, F. (2011). Considerations on nonlinear model predictive control techniques. *Computers and Chemical Engineering*, *35*(11), 2491–2509. https://doi.org/10.1016/j.compchemeng.2011.04.009

Mangal, A., Kalia, S., Rajgopal, H., Rangarajan, K., Namboodiri, V., Banerjee, S., & Arora, C. (2020). *CovidAID: COVID-19 Detection Using Chest X-Ray*. 1–10. http://arxiv.org/abs/2004.09803

Mete, S., Ozer, S., & Zorlu, H. (2016). System identification using Hammerstein model optimized with differential evolution algorithm. *AEU - International Journal of Electronics and Communications*, *70*(12), 1667–1675. https://doi.org/10.1016/j.aeue.2016.10.005

Mirjalili, S., Mirjalili, S. M., & Hatamlou, A. (2016). Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, *27*(2), 495–513. https://doi.org/10.1007/s00521-015-1870-7

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*, *69*, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Mirjalili, S., Mohammad, S., & Lewis, A. (2014). Advances in Engineering Software Grey Wolf Optimizer. *ADVANCES IN ENGINEERING SOFTWARE*, *69*, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

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. https://doi.org/10.1016/j.eswa.2010.03.043

Panda, A., & Pani, S. (2014). A new model based on Colliding Bodies Optimization for identification of Hammerstein plant. *2014 Annual IEEE India Conference (INDICON)*, *4*, 1–5. https://doi.org/10.1109/INDICON.2014.7030381

Rauf, H. T., Shoaib, U., Lali, M. I., & Khan, M. A. (2020). *Particle Swarm Optimization With Probability Sequence for Global Optimization*. *8*, 110535–110549. https://doi.org/10.1109/ACCESS.2020.3002725

Schoukens, J., & Ljung, L. (2019). Nonlinear System Identification: A User-Oriented Road Map. *IEEE Control Systems*, *39*(6), 28–99. https://doi.org/10.1109/MCS.2019.2938121

Singh, R., & Kaur, R. (2022). A Novel Archimedes Optimization Algorithm with Levy Flight for Designing Microstrip Patch Antenna. *Arabian Journal for Science and Engineering*, *47*(3), 3683–3706. https://doi.org/10.1007/s13369-021-06307-x

Ting, T. O., Yang, X. S., Cheng, S., & Huang, K. (2015). Hybrid metaheuristic algorithms: Past, present, and future. *Studies in Computational Intelligence*, *585*, 71–83. https://doi.org/10.1007/978-3-319-13826-8_4

Toha, S. F., Julai, S., & Tokhi, M. O. (2012). Ant Colony Based Model Prediction of a Twin Rotor System. *Procedia Engineering*, *41*, 1135–1144. https://doi.org/10.1016/j.proeng.2012.07.293

Yan, Z., Chen, J., & Zhang, Z. (2017). Valve stiction detection using the bootstrap Hammerstein system identification. *2017 6th International Symposium on Advanced Control of Industrial Processes, AdCONIP 2017*, 84–89. https://doi.org/10.1109/ADCONIP.2017.7983760

Yang, X.-S. (2010). *Nature-Inspired Metaheuristic Algorithms: Second Edition*. Luniver Press.

Zou, Z., Yu, M., Wang, Z., Liu, X., Guo, Y., Zhang, F., & Guo, N. (2013). Nonlinear model algorithmic control of a pH neutralization process. *Chinese Journal of Chemical Engineering, 21*(4), 395–400. https://doi.org/10.1016/S1004-9541(13)60479-6