ISA Transactions 129 (2022) 572–593

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Practice article

Grey-box modelling and fuzzy logic control of a Leader–Follower robot manipulator system: A hybrid Grey Wolf–Whale Optimisation approach

Ololade O. Obadina [a], Mohamed A. Thaha [b,c], Zaharuddin Mohamed [d], M. Hasan Shaheed [a,*]

[a] School of Engineering and Materials Science, Queen Mary University of London, UK
[b] Blizard Institute, Barts and The London School of Medicine & Dentistry, Queen Mary University of London, UK
[c] Department of Colorectal Surgery, Royal London Hospital, Barts Health NHS Trust, Whitechapel, London, UK
[d] School of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia

ARTICLE INFO

Article history:
Received 6 July 2021
Received in revised form 4 January 2022
Accepted 15 February 2022
Available online 21 February 2022

Keywords:
Grey-box modelling
White-box modelling
Leader–follower robot manipulator
Fuzzy logic control
Optimisation
Membership functions

ABSTRACT

This study presents the development of a grey-box modelling approach and fuzzy logic control for real time trajectory control of an experimental four degree-of-freedom Leader–Follower Robot (LFR) manipulator system using a hybrid optimisation algorithm, known as Grey Wolf Optimiser (GWO) - Whale Optimisation Algorithm (WOA). The approach has advantages in achieving an accurate model of the LFR manipulator system, and together with a better trajectory tracking performance. In the first instance, the white box model is formed by modelling the dynamics of the follower manipulator using the Euler–Lagrange formulation. This white-box model is then improved upon by re-tuning the model's parameters using GWO-WOA and experimental data from the real LFR manipulator system, thus forming the grey-box model. A minimum improvement of 73.9% is achieved by the grey-box model in comparison to the white-box model. In the latter part of this investigation, the developed grey-box model is used for the design, tuning and real-time implementation of a fuzzy PD+I controller on the experimental LFR manipulator system. A 78% improvement in the total mean squared error is realised after tuning the membership functions of the fuzzy logic controller using GWO-WOA. Experimental results show that the approach significantly improves the trajectory tracking performance of the LFR manipulator system in terms of mean squared error, steady state error and time delay.

1. Introduction

Tasks such as grasping and manipulation using laparoscopic instruments have been made feasible through a technique known as minimally invasive surgery (MIS), which goes as far back as the 1960s. Reported benefits such as minimal scarring, shorter recovery time and less post-surgery pain are a few rewards that MIS presents for the patients [1]. However, some issues related to limited dexterity, poor hand-eye coordination and difficulty in sensing forces exerted by these laparoscopic instruments limits the efficiency of the traditional MIS technique. The drawbacks of using the traditional MIS technique can be resolved by incorporating robotic systems [2], of which a common type is widely known as the Master-Slave (MS) robotic system [3]. In this paper, the MS robotic system will be referred to as the leader–follower robot (LFR) manipulator system.

Leader–follower robotic systems typically comprise of two sub-systems — the leader and follower robot manipulators [4]. In operation, the leader manipulator is controlled by a human user, where position instructions are sent to the follower manipulator which then transforms the hand movements of the user into precise, real time and scalable motions [3]. In the medical field for instance, existing leader–follower robot technology is used particularly for minimally invasive surgeries, where the most common type of robot system used is the da Vinci Surgical System that is manufactured by Intuitive Surgical [5]. The widely reported advantages of the da Vinci system include a reduced risk of complications and blood loss. While these benefits are attractive, a handful of drawbacks exist. For example, the da Vinci surgical system is large and cost prohibitive. In addition, there is a steep learning curve for the surgeon and the speed of operation of these systems is limited due to the size of the actuator thus increasing the amount of energy consumption.

The overall objective of the LFR manipulator system is for the follower robot manipulator to accurately track the trajectory of the leader robot manipulator in real time, with minimal delay. It is required for the time delay to be a small as practically possible to prevent dire consequences in the applications of LFR system, especially in a surgery [6]. To achieve the control

* Corresponding author.
E-mail address: m.h.shaheed@qmul.ac.uk (M.H. Shaheed).

https://doi.org/10.1016/j.isatra.2022.02.023

objectives, an accurate dynamic model of the leader–follower robot system is vital, together with an efficient control algorithm. The fundamental idea behind modelling is to obtain an accurate representation of a real system's dynamics that will be suitable for controller design. Substantial research that addresses the modelling of robot manipulators using analytical and data-based modelling approaches has been reported. Three common techniques used in dynamic modelling exist and they include white-box, black-box, and grey-box modelling approaches.

White box models are derived analytically and are based on a priori knowledge of system parameters, which are often challenging to quantify due to the physical constraints of the system [7]. Black box models on the other hand are developed using experimental input/output data of a system. This approach, however, lacks in accuracy to a degree as important structural information will be lost during the mapping of experimental data to nonlinear differential equations of a system [8]. The grey box modelling technique combines the properties of both white-box and black-box modelling by utilising a priori knowledge of the system and identifying immeasurable parameters using experimental data.

The successful application of the grey-box modelling approach on dynamic systems has been widely addressed in literature. For instance, a grey-box approach using genetic algorithm (GA) is proposed for the application on an aerodynamic system in [9]. This method estimates unknown parameters of the model using experimental data and a priori system information. Premkumar et al. [10] have proposed an enhanced chaotic JAYA algorithm for the parameter identification of photovoltaic models. Al-Messabi et al. [11] have proposed the grey-box identification of photovoltaic power systems using particle swarm optimisation. Jizhen et al. [12] have presented the grey-box dynamic modelling of a wind turbine generation system using genetic algorithms. For robot manipulator systems, Gao et al. [13] have proposed a grey-box parametric identification approach for a 6 degree-of-freedom (DOF) industrial robot. The investigation considers the development of a kinematic model of the robot system using the Denavit–Hartenberg technique and then uses an alternative identification algorithm to estimate the structural parameters of the robot. Similarly, another investigation which uses a particle swarm search approach to identify the kinematic model parameters of a two-link robot manipulator is presented in [14]. Both methods employ kinematic models which do not take into account the joint actuator torques that result in the motion of the robots. Other grey-box modelling approaches found in literature use analytical techniques such as the H_∞-norm local approach [15], linear parameter varying method [16] and instrumental variable techniques [17].

Research on the application of fuzzy logic control on robotic systems is mature. In the work carried out by Li et al. [18], a hybrid controller comprising of an incremental fuzzy logic proportional controller and an integral and derivative (Fuzzy P + ID) controller is proposed for application on a robot manipulator. The fuzzy controller is used to replace the proportional component in the traditional PID but benefits from the properties of the individual components such as improving rise time, reducing steady state error and ensuring overall stability of the system. Simulation studies on the application of fuzzy logic control on a 6-DOF robot manipulator has also been presented by Alavandar and Nigam [19]. The authors have designed a fuzzy PD + I controller which consist of two inputs for the proportional and derivative (PD) gains with a linear integral control added at the output. The feasibility of the fuzzy PD + I controller is tested in terms of trajectory tracking and results show that the performance of the proposed controller improves noticeably in comparison to its traditional PID counterpart.

The application of evolutionary algorithm techniques on fuzzy logic control has received an increased interest by researchers particularly in the robotics field. This is because the tuning process of fuzzy logic controllers (FLCs) is time consuming and often requires an expert when the number of inputs and outputs of a system are increased [20]. Studies involving evolutionary algorithms such as particle swarm optimisation (PSO), genetic algorithm (GA) and bee's algorithm (BA) have been proven to be effective in tuning the parameters of FLCs. More recently, the trend of hybridising two or more algorithms resulting in better-quality solutions for practical applications has risen. A broad range of scientific contributions that have applied optimisation algorithms for various applications including fuzzy logic control tuning are presented in Table 1.

It is found that grey-box model identification of a LFR system that takes into consideration the joint actuator torques and optimisation algorithms is very limited in literature. In addition, designing an effective FLC by optimising the membership functions (MFs) can be explored especially for control of a LFR manipulator system. This paper therefore seeks to fill this research gap by presenting the grey-box model identification and fuzzy logic control of a 4-DOF LFR manipulator system based on a hybrid Grey Wolf Optimiser (GWO)–Whale Optimisation Algorithm (WOA) approach. The dynamic model is derived using the Euler–Lagrange approach, and the optimisation approach is employed to optimally identify the parameters of the derived model. The accuracy of the proposed model is verified by comparing with the actual response of the system. Using the developed grey-box model, a FLC design is proposed, and the MFs are tuned optimally. The FLC is implemented on an experimental LFR manipulator system to verify the performance of the controller in terms of input tracking, steady-state error and time delay.

The main contributions of this work are given as follows:

a. Development of a unique hybrid GWO-WOA approach for a grey-box model identification of a LFR manipulator system. Parameters of the system's mathematical model are optimally identified, resulting in a more accurate model. The approach can be utilised for accurate modelling of other nonlinear dynamic systems.
b. Design of a FLC method for control of a LFR system with optimised MF parameters using the unique GWO-WOA approach. The proposed control technique is beneficial in finding effective FLC controllers.

The breakdown of this paper is given as follows: Section 2 introduces the experimental LFR manipulator system and gives the mathematical equation of motion of the follower robot manipulator derived using the Euler–Lagrange method. Section 3 discusses the proposed method of grey-box modelling using the hybrid GWO-WOA algorithm and comparative results. Section 4 present the design, development and tuning of the proposed FLC. Section 5 presents the implementation and experimental results obtained in this investigation. Finally, Section 6 concludes the paper and suggests future work.

2. The leader–follower robot manipulator system

A prototype of the experimental LFR manipulator system under investigation in this study is shown in Fig. 1. This system consists of a leader robot manipulator and a follower robot manipulator with 4-DOF respectively, a PC interface, a DC power supply and an Arduino-based control system. The principle of operation of the LFR manipulator system is similar to unilateral teleoperation where no haptic feedback is present between the leader and follower manipulators, as illustrated in Fig. 2. The joints of the leader manipulator are fitted with sensors (i.e., potentiometers) and the follower manipulator's joints are fitted with modified servo-motor actuators.

Table 1
Scientific contributions that apply optimisation algorithms to various applications.

Ref.	Technique	Application
[21]	Hybridisation of harmony search algorithm and cuckoo search (HS/CS)	Benchmark evaluation of fourteen objective functions including the Ackley, Rastrigin and Rosenbrock functions.
[22]	Hybridisation of particle swarm optimisation and gravitational search algorithm (PSO-GSA)	Benchmark evaluation of twenty-three objective functions.
[23]	Hybridisation of whale optimisation algorithm and simulated annealing (WOA-SA)	Feature selection evaluation which uses a custom objective function that depends on a KNN classifier.
[24]	Hybridisation of grey wolf optimiser and flower pollination algorithm (GWO-FPA)	Benchmark evaluation of six objective functions.
[25]	Hybridisation of grey-wolf optimiser and bat algorithm (GWOBA)	Benchmark evaluation of thirty objective functions consisting of unimodal, multi-modal, hybrid and composition functions to simulate real life applications problems.
[26]	Hybridisation of particle swarm optimisation and grey wolf optimiser (HPSOGWO)	Benchmark evaluation of twenty-three objective functions comprising of unimodal, multimodal, and fixed dimension multimodal functions.
[27]	Hybridisation of whale optimiser algorithm with mean strategy of grey wolf optimiser (HAGWO)	– Benchmark evaluation of twenty-three objective functions comprising of unimodal, multimodal, and fixed dimension multimodal functions. – Bio-medical science real life problems using Iris, XOR, baloon and breast cancer datasets. – Welded beam design problem using a multi-objective cost function and seven constraints to minimise the cost of production. – Design of a cylindrical pressure vessel using a customised objective function to minimise the total cost of design.
[28]	Hybridisation of grey wolf optimiser and artificial bee colony algorithm (GWO-ABC)	– Benchmark evaluation of twenty-seven functions comprising of unimodal, multimodal, and composite objective functions. – Trajectory tracking control of a 2-DOF robot manipulator with the integral time absolute error (ITAE) objective function for optimising the FOPID controller.
[29]	Hybridisation of grey-wolf optimiser and whale optimisation algorithm (GWO-WOA)	– Benchmark evaluation of ten functions comprising of unimodal and multimodal objective functions. – Model identification of a 4-DOF master–slave robotic manipulator with the mean squared error (MSE) objective function.
[30]	Hybridisation of cuckoo optimisation algorithm and harmony search (COA-HS)	Gene selection for cancer classification which uses a custom objective function that depends on the accuracy of a classifier and the number of selected genes.
[20]	Fuzzy logic control tuning using particle swarm optimisation	Trajectory control of a 2-DOF robot using three objective functions independently namely mean of root of squared error (MRSE), mean of absolute value of the error (MAE) and reference-based error with control effort (RBECE).
[31]	Scaling factors tuning of a fuzzy logic controller using genetic algorithm	Set-point tracking control of a single flexible link manipulator using an objective function that is dependent on the tracking error and overshoot of the system.
[32]	Fractional PD+I fuzzy logic control tuning using genetic algorithm	Numerical simulations of a high order plant system using the integral time absolute error (ITAE) criterion as the objective function.
[33]	Scaling factors and membership functions tuning of a fuzzy logic controller using bee's algorithm	Position and vibration control of a single link flexible robot using a custom weighted objective function that is dependent on the time delay, rise time, peak time, maximum overshoot, steady state error and the robot's deflection amplitude.
[34]	Membership functions tuning of a fuzzy logic controller using bee colony optimisation	Estimation of the average energy consumption of freight trains using the energy to be minimised as the objective function.
[35]	Membership functions tuning of a hybrid fuzzy-computed torque controller using genetic algorithm	Position tracking control of a 2-DOF robot manipulator using a custom objective function that is dependent on the applied torque on each of the robot's joints.
[36]	Scaling factors tuning of a fuzzy fractional order PID controller using a hybrid artificial bee-colony-genetic algorithm (ABC-GA)	Trajectory tracking control of a 2-DOF robot manipulator using the ITAE criterion as the objective function.
[37]	Type-1 and type-2 fuzzy logic controller tuning using a hybrid particle swarm optimisation-genetic algorithm (PSO-GA)	Trajectory tracking control of an autonomous mobile robot using the mean of the absolute error criterion as the objective function.
[38]	Scaling factors tuning of a neuro-fuzzy subsets controller using genetic algorithm	Position tracking control of a humanoid robot manipulator using the absolute sum of errors as the objective function.
[39]	Sliding surface gains tuning of a robust adaptive type-2 fuzzy neural network controller using cuckoo algorithm	Trajectory tracking control of a quadrotor using the mean squared tracking error as the objective function.

In this study, the dynamic model of the leader robot manipulator is not required for the synchronisation of the LFR manipulator system as the measured joint positions of the leader robot are used. Therefore, only the mathematical model of the follower manipulator will be presented. The 4-DOF correspond to the base, shoulder, wrist pitch and wrist roll joints and their angles are denoted by $\theta_0, \theta_1, \theta_2$ and θ_3 respectively. A schematic diagram of the follower manipulator is shown in Fig. 3.

For a satisfactory control performance, the accurate modelling of the LFR manipulator system's dynamics is vital. In this study, the Euler–Lagrange approach is used to formulate the dynamics of the LFR manipulator system. Considering the follower robot

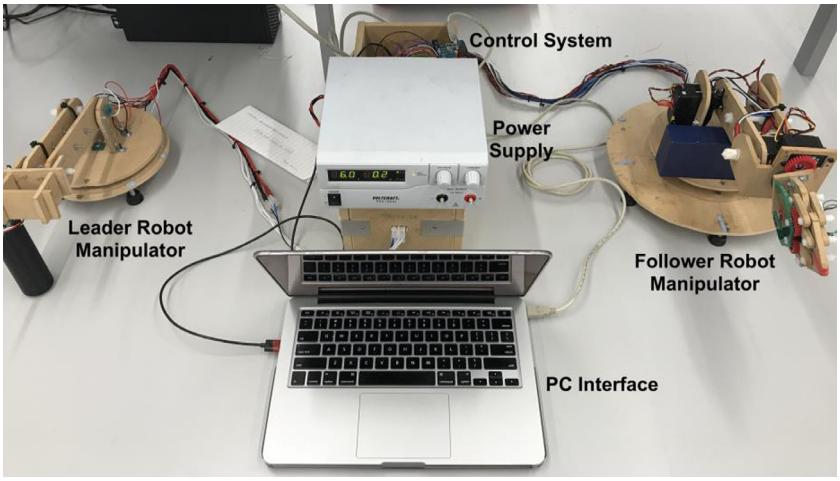


Fig. 1. Prototype of the experimental 4-DOF leader–follower robot manipulator system.

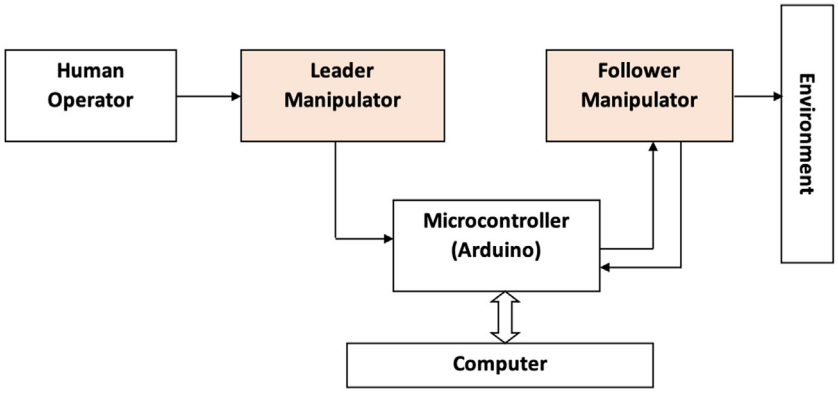


Fig. 2. Schematic diagram of the leader–follower robot manipulator system.

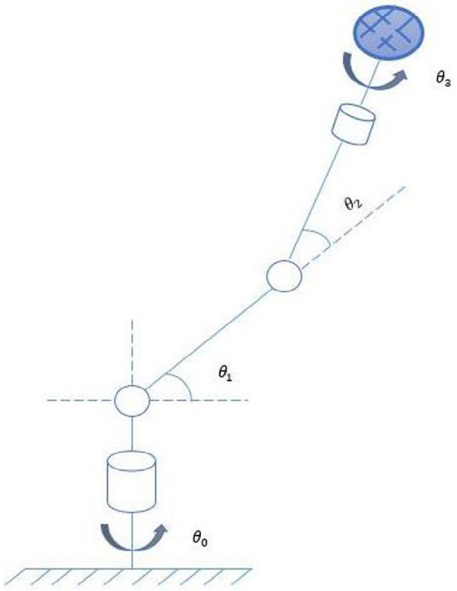


Fig. 3. Schematic of the follower manipulator in 2-dimension (2D).

manipulator is driven by direct current (DC) servomotors, the dynamics of these actuators are also accounted for in the follower robot's overall equation of motion. The overall equation of motion of the 4-DOF follower robot manipulator (including the actuators' dynamics) is derived as:

$$\underline{U} = \boldsymbol{K_T}^{-1} \cdot \left(\boldsymbol{R} \cdot \left[\left[\boldsymbol{A}\left(\boldsymbol{\theta}\right) + \boldsymbol{J_m} r\right]\underline{\ddot{\theta}} + \underline{C}\left(\theta, \dot{\theta}\right) + \underline{G}\left(\theta\right)\right] + \boldsymbol{K_T K_E} r\underline{\dot{\theta}} + \boldsymbol{B_m R} r\underline{\dot{\theta}}\right) \quad (1)$$

where:

$\boldsymbol{A}(\boldsymbol{\theta})$	4×4 matrix of the follower robot's inertia,
$\underline{C}(\theta, \dot{\theta})$	4×1 vector of the Coriolis and centrifugal forces,
$\underline{G}(\theta, \dot{\theta})$	4×1 vector of the gravitational forces,
$\underline{U}$	4×1 vector of the actuators input voltages,
r	Servomotors' gear ratios,
$\boldsymbol{R}$	4×4 diagonal matrix of the actuators' armature resistances,
$\boldsymbol{K_T}$	4×4 diagonal matrix of the actuators' torque constants,
$\boldsymbol{K_E}$	4×4 diagonal matrix of the actuators' electromotive force constants,
$\boldsymbol{J_m}$	4×4 diagonal matrix of the actuators' motor moment of inertia,
$\boldsymbol{B_m}$	4×4 diagonal matrix of the actuators' motor viscous friction constants, and
$\theta, \dot{\theta}$ and $\ddot{\theta}$	Joint angle positions, velocities, and accelerations of the follower robot respectively.

It is important to note that the bold parameters in Eq. (1) signify matrices while the other parameters are vectors. After expansion, the parameters $\boldsymbol{A}(\boldsymbol{\theta})$, $\underline{C}(\theta, \dot{\theta})$ and $\underline{G}(\theta, \dot{\theta})$ are simplified as shown:

$$\boldsymbol{A}(\boldsymbol{\theta}) = \begin{bmatrix} A_{11} & 0 & 0 & A_{14} \\ 0 & A_{22} & A_{23} & 0 \\ 0 & A_{32} & A_{33} & 0 \\ A_{41} & 0 & 0 & A_{44} \end{bmatrix} \tag{2}$$

$$\underline{C}(\theta, \dot{\theta}) = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} \tag{3}$$

$$\underline{G}(\theta, \dot{\theta}) = \begin{bmatrix} 0 \\ G_2 \\ G_3 \\ 0 \end{bmatrix} \tag{4}$$

where:

$$A_{14} = A_{41} = I_{xx3}\ \sin(\theta_1 + \theta_2)$$

$$A_{44} = I_{xx3}$$

$$A_{23} = A_{32} = I_{zz2} + I_{zz3} + M\left(L_2^2 + L_3^2\right) + m_3L_2^2 + m_2L_{2cg}^2 + m_3L_{3cg}^2 + 2ML_2L_3 + 2m_3L_2L_{3cg} + \cos\theta_2\left(ML_1L_2 + ML_1L_3 + m_3L_1L_2 + m_2L_1L_{2cg} + m_3L_1L_{3cg}\right)$$

$$A_{33} = I_{zz2} + I_{zz3} + M\left(L_2^2 + L_3^2\right) + m_3L_2^2 + m_2L_{2cg}^2 + m_3L_{3cg}^2 + 2ML_2L_3 + 2m_3L_2L_{3cg}$$

$$A_{11} = 2I_{xx1} + 2I_{xx2} + 2I_{xx3} + 2I_{yy1} + 2I_{yy2} + 2I_{yy3} + 2 + 2M(L_1^2 + L_2^2 + L_3^2) + 2m_2L_1^2 + 2m_3L_1^2 + 2m_3L_2^2 + 2m_1L_{1cg}^2 + 2m_2L_{2cg}^2 + 2m_3L_{3cg}^2 + 4ML_2L_3 + 4m_3L_2L_{3cg} + \cos 2\theta_1\left(2I_{yy1} - 2I_{xx1} + 2m_2L_1^2 + 2m_3L_1^2 + 2\,m_1L_{1cg}^2 + 2ML_1^2\right) + \cos\theta_2\left(4ML_1L_2 + 4ML_1L_3 + 4m_3L_1L_2 + 4m_2L_1L_{2cg} + 4m_3L_1L_{3cg}\right) + \cos(2\theta_1 + 2\theta_2)\left[2I_{yy2} - 2I_{xx2} - 2I_{xx3} + 2I_{yy3} + 2M\left(L_2^2 + L_3^2\right) + 2m_3L_2^2 + 2m_2L_{2cg}^2 + 2m_3L_{3cg}^2 + 4ML_2L_3 + 4m_3L_2L_{3cg}\right] + \cos(2\theta_1 + \theta_2)\left[4ML_1L_2 + 4ML_1L_3 + 4m_3L_1L_2 + 4m_2L_1L_{2cg} + 4m_3L_1L_{3cg}\right]$$

$$A_{22} = I_{zz1} + I_{zz2} + I_{zz3} + M(L_1^2 + L_2^2 + L_3^2) + m_2L_1^2 + m_3L_1^2 + m_3L_2^2 + m_1L_{1cg}^2 + m_2L_{2cg}^2 + m_3L_{3cg}^2 + 2ML_2L_3 + 2m_3L_2L_{3cg} + \cos\theta_2\left(2ML_1L_2 + 2ML_1L_3 + 2m_3L_1L_2 + 2m_2L_1L_{2cg} + 2m_3L_1L_{3cg}\right)$$

For the Coriolis and centrifugal vector, C_1, C_2, C_3 and C_4 are defined as:

$$C_1 = \dot{\theta}_0\dot{\theta}_1\left[\sin 2\theta_1\left(I_{xx1} - I_{yy1} - ML_1^2 - m_2L_1^2 - m_3L_1^2 - m_1L_{1cg}^2\right) + \sin(2\theta_1 + 2\theta_2)\left(I_{xx2} + I_{xx3} - I_{yy2} - I_{yy3} - ML_2^2 - ML_3^2 - m_3L_2^2 - m_2L_{2cg}^2 - m_3L_{3cg}^2 - 2ML_2L_3 - 2m_3L_2L_{3cg}\right) + \sin(2\theta_1 + \theta_2)\left(-2ML_1L_2 - 2ML_1L_3 - 2m_3L_1L_2 - 2m_2L_1L_{2cg} - 2m_3L_1L_{3cg}\right)\right] + \dot{\theta}_0\dot{\theta}_2\left[\sin(2\theta_1 + 2\theta_2)\left(I_{xx2} + I_{xx3} - I_{yy2} - I_{yy3} - ML_2^2 - ML_3^2 - m_3L_2^2 - m_3L_{3cg}^2 - 2ML_2L_3 - 2m_3L_2L_{3cg}\right) + \sin(2\theta_1 + \theta_2)\left(-m_2L_{2cg} - ML_1L_2 - ML_1L_3 - m_3L_1L_2 - m_2L_1L_{2cg} - m_3L_1L_{3cg}\right) + \sin\theta_2\left(-m_3L_1L_2 - m_2L_1L_{2cg} - m_3L_1L_{3cg} - ML_1L_2 - ML_1L_3\right)\right] + \dot{\theta}_1\dot{\theta}_3\left[I_{xx3}\cos(\theta_1 + \theta_2)\right] + \dot{\theta}_2\dot{\theta}_3\left[I_{xx3}\cos(\theta_1 + \theta_2)\right]$$

$$C_2 = \dot{\theta}_0^2\left[\sin(2\theta_1 + 2\theta_2)\left(2I_{yy2} + 2I_{yy3} + 2ML_2^2 + 2ML_3^2 + 2m_3L_2^2 + 2m_2L_{2cg}^2 + 2m_3L_{3cg}^2 + 4ML_2L_3 + 4m_3L_2L_{3cg} - 2I_{xx2} - 2I_{xx3}\right) + \sin(2\theta_1)\left(2I_{yy1} + 2ML_1^2 + 2m_2L_1^2 + 2m_3L_1^2 + 2m_1L_{1cg}^2\right) + \sin(2\theta_1 + \theta_2)\left(4ML_1L_2 + 4ML_1L_3 + 4m_3L_1L_2 + 4m_2L_1L_{2cg} + 4m_3L_1L_{3cg}\right)\right] + \dot{\theta}_1^2\left[-0.5I_{xx1}\sin(2\theta_1)\right] + \dot{\theta}_2^2\left[\sin(\theta_2)\left(-m_3L_1L_2 - m_2L_1L_{2cg} - m_3L_1L_{3cg} - ML_1L_2 - ML_1L_3\right)\right] + \dot{\theta}_1\dot{\theta}_2\left[\sin(\theta_2) \times \left(-2m_3L_1L_2 - 2m_2L_1L_{2cg} - 2m_3L_1L_{3cg} - 2ML_1L_2 - 2ML_1L_3\right)\right] + \dot{\theta}_0\dot{\theta}_3\left[-I_{xx3}\cos(\theta_1 + \theta_2)\right]$$

$$C_3 = \dot{\theta}_0^2\left[\sin(2\theta_1 + 2\theta_2)\left(2I_{yy2} + 2I_{yy3} + 2ML_2^2 + 2ML_3^2 + 2m_3L_2^2 + 2m_2L_{2cg}^2 + 2m_3L_{3cg}^2 + 4ML_2L_3 + 4m_3L_2L_{3cg} - 2I_{xx2} - 2I_{xx3}\right) + \sin(2\theta_1 + \theta_2)\left(4ML_1L_2 + 4ML_1L_3 + 4m_3L_1L_2 + 4m_2L_1L_{2cg} + 4m_3L_1L_{3cg}\right) + \sin(\theta_2)\left(2m_3L_1L_2 + 2m_2L_1L_{2cg} + 2m_3L_1L_{3cg} + 2ML_1L_2 + 2ML_1L_3\right)\right] + \dot{\theta}_1^2\left[\sin(\theta_2)\left(2m_3L_1L_2 + 2m_2L_1L_{2cg} + 2m_3L_1L_{3cg} + 2ML_1L_2 + 2ML_1L_3\right)\right] + \dot{\theta}_0\dot{\theta}_3\left[-I_{xx3}\cos(\theta_1 + \theta_2)\right]$$

$$C_4 = \dot{\theta}_0\dot{\theta}_1\left[I_{xx3}\cos(\theta_1 + \theta_2)\right] + \dot{\theta}_0\dot{\theta}_2\left[I_{xx3}\cos(\theta_1 + \theta_2)\right]$$

and, the gravity vector parameters G_2 and G_3 are expressed as:

$$G_2 = g\left[\cos(\theta_1 + \theta_2)\left[ML_2 + ML_3 + m_3L_2 + m_2L_{2cg} + m_3L_{3cg}\right] + \cos\theta_1\left[ML_1 + m_2L_1 + m_3L_1 + m_1L_{1cg}\right]\right]$$

$$G_3 = g\left[\cos(\theta_1 + \theta_2)\left[ML_2 + ML_3 + m_3L_2 + m_2L_{2cg} + m_3L_{3cg}\right]\right]$$

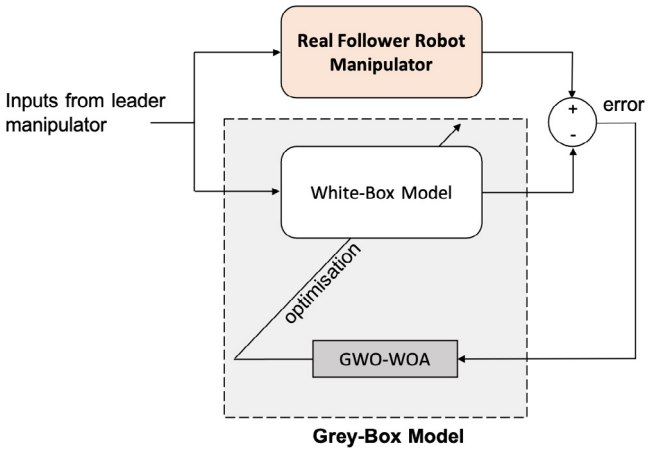


Fig. 4. Grey box model identification procedure.

where:

m_i	Link mass for the ith link,
M	Mass of the end effector,
L_i	Link length of the ith link,
$L_{i,cg}$	Centre of gravity (i.e., midpoint) of link length L_i
g	Gravity constant
$I_{xx,i}$, $I_{yy,i}$ and $I_{zz,i}$	Mass moment of inertia in the x, y and z-axis respectively for the ith link

Some assumptions were made in deriving the equation of motion for the follower robot manipulator. These assumptions include the following:

a. Point masses are located at the centre of each link of the robot and the mass of each link is considered as a point mass.
b. The effects of friction and transmission losses, and armature inductance of the actuators are negligible.
c. The gearbox coupling ratio is 1.

While some parameters such as the link lengths and armature resistances, can be quantified using specialist devices, it is assumed that these devices are inaccessible. As such, all physical parameters of the follower robot manipulator are given an estimated value, which are then included in the optimisation process to ultimately improve the model's accuracy.

3. Grey-box model identification

Grey box modelling is an ingenious method that merges the benefits of white and black box modelling techniques by utilising a priori knowledge of the system and identifying immeasurable parameters using experimental data. Broadly speaking, this approach is seen to be superior in performance compared to each of the white and black box techniques when used separately [8].

From an algorithm's perspective, a cost function can either be minimised or maximised. Also, defining limits of unknown parameters to be estimated such as lower bound (lb) and upper bound (ub), is essential as this provides a defined search space for an algorithm to optimally find parameters within given boundaries. In this study, the objective is to minimise the mean squared error (MSE) function chosen as the cost function for this model identification task. The MSE function is described as:

$$MSE = \frac{1}{N}\left(\sum_{i=1}^{N}\left(\theta_0 - \hat{\theta}_0\right)^2 + \sum_{i=1}^{N}\left(\theta_1 - \hat{\theta}_1\right)^2 + \sum_{i=1}^{N}\left(\theta_2 - \hat{\theta}_2\right)^2 + \sum_{i=1}^{N}\left(\theta_3 - \hat{\theta}_3\right)^2\right) \tag{5}$$

where $\theta_0, \theta_1, \theta_2$ and θ_3 are the joint angle positions of the real LFR manipulator system and $\hat{\theta}_0, \hat{\theta}_1, \hat{\theta}_2$ and $\hat{\theta}_3$ are the joint angle positions of the predicted grey-box model for joints 0, 1, 2 and 3 respectively. N is the number of collected output data samples.

Let us define a vector containing the unknown parameters to be identified as $\boldsymbol{v}$, where:

$$v = \{v_1, v_2, v_3, \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.., v_{19}, v_{20}\}$$

and vector v represents $\{J_0, Ra_0, Kt_0, Bm_0, r_0, J_1, Ra_1, Kt_1, Bm_1, r_1, J_2, Ra_2, Kt_2, Bm_2, r_2, J_3, Ra_3, Kt_3, Bm_3$ and $r_3\}$ respectively. These are the parameters related to motors of Joint 0, 1, 2 and 3, in which, five parameters are identified for each joint. They are moment of inertia, armature resistance, torque constant, viscous friction constant and reduction gear ratio.

Fig. 4 illustrates the grey box model identification procedure where the same inputs from the leader manipulator are fed to both the real follower robot manipulator and the previously developed Euler–Lagrange white box. The calculated position errors from these outputs are then fed into an optimising algorithm, which then minimises the cost function and finds the unknown parameters optimally. In this work, a hybrid GWO-WOA is designed and utilised to obtain optimal parameter values of the follower robot manipulator.

3.1. The hybrid GWO-WOA algorithm

GWO and WOA are simple to implement and have only a few modifiable parameters [40] amongst other attributes. However, several papers in literature have reported the main drawback of GWO is the algorithm's ease in falling into a local optimum particularly in complex situations that involve optimising high dimensional data [27–29]. WOA on the other hand has good exploitation abilities [41] but has a relatively poor convergence rate.

Obadina et al. [29] have proposed a hybrid GWO-WOA algorithm which inherits the benefits of original algorithms GWO [42] and WOA [43]. The hybrid GWO-WOA is based on merging the

Initialize random grey wolves population X_i where $i = (1,2,3,\ldots,N)$

Initialize grey wolf parameters $\vec{a}, \vec{A}$ *and* $\vec{Q}$ using equations (6), (7) and (8)

Calculate the fitness of each grey wolf
where:

$\vec{X}_\alpha$ = position of alpha wolf (i.e. best solution)
$\vec{X}_\beta$ = position of beta wolf (i.e. second best solution)
$\vec{X}_\delta$ = position of delta wolf (i.e. third best solution)

while $(t < T_{max})$
  for every search agent:
    Update the position of the alpha wolf using equations (9) and (12)
    Update the positions of other grey wolves using equations (10), (11), (13) − (15)
  end for
    Update grey wolf parameters $\vec{a}, \vec{A}$ *and* $\vec{Q}$
    Calculate the fitness of all grey wolves
    $t = t + 1$
end while
  return position of alpha wolf (i.e. best solution), $\vec{X}_\alpha$

Fig. 5. Hybrid GWO-WOA pseudo-code.

functionalities of the Grey Wolf Optimiser and the logarithmic spiral equation of the WOA. By embedding the logarithmic spiral behaviour of WOA into the GWO algorithm, the hybrid algorithm benefits from an increased efficiency, a faster convergence rate and accuracy in finding global optima solutions. The performance of GWO-WOA has been studied in [29] and compared with state-of-art algorithms such as particle swarm optimisation (PSO), GWO and WOA respectively, of which the superiority of GWO-WOA is demonstrated.

The hybrid GWO-WOA considers all possible solutions as the positions of hunting grey-wolves in a pack, which are guided by the alpha wolf (i.e., the best solution). As the position of the target prey is unknown, the location of this target prey is regarded as the optimal solution. The second and third best solutions are known as the beta and delta wolves respectively. The grey wolves deviate from each other to search for the prey and converge to attack the prey [42]. The encircling behaviour of the alpha grey wolf has been enhanced by integrating the spiral logarithmic equation found in WOA [43].

In the first iteration of the hybrid algorithm, an initial population of grey wolves (i.e., solutions) is randomly generated and the vector parameters $\vec{a}, \vec{A}$ *and* $\vec{Q}$ are initialised as [42]:

$$\vec{a} = 2 - \frac{t}{T_{max}} \tag{6}$$

$$\vec{A} = 2\vec{a} * \overrightarrow{r_1} - \vec{a} \tag{7}$$

$$\vec{Q} = 2\overrightarrow{r_2} \tag{8}$$

where $\vec{a}$ is a vector that linearly decreases from 2 to 0 over the course of iterations; $\vec{A}$ and $\vec{Q}$ are coefficient vectors; t is the current iteration; T_{max} is the maximum iteration; $\overrightarrow{r_1}$ and $\overrightarrow{r_2}$ are random vectors between [0,1] and $*$ is an element-by-element multiplication.

The objective function for each grey wolf is computed where the best, second best and third best fitness scores represent the alpha, beta and delta grey wolves respectively. The other wolves update their positions based on the locations of these top three search agents. The alpha wolf's location is very important to note because the hunt for prey (i.e., the best solution) is commonly guided by the alpha wolf.

In the hybrid GWO-WOA algorithm, the alpha wolf can attack its prey in either an encircling manner or using a spiral logarithmic method. Thus, the position update vector $\vec{d}_\alpha$ of the alpha wolf has been modified as follows:

$$\vec{d}_\alpha = \begin{cases} \Upsilon * \left|\vec{Q}_1 * \vec{X}_\alpha(t) - \vec{X}(t)\right| & \text{if } p_1 < 0.5 \\ \rho * e^{bl} * \cos(2\pi l) * \left|\vec{Q}_1 * \vec{X}_\alpha(t) - \vec{X}(t)\right| & \text{if } p_1 \geq 0.5 \end{cases} \tag{9}$$

where $\vec{X}_\alpha$ is the position vector of the alpha wolf; $\vec{X}$ is the position vector of a grey wolf; Υ and ρ are random numbers between [0, 1]; b is a constant for defining the logarithmic spiral and l is a random number between $[-1, 1]$. p_1 is a random number between [0, 1] and represents the probability of the alpha wolf choosing the encircling method or the bubble-net mechanism to update its position.

The position update vectors of the beta and delta wolves are given as:

$$\vec{d}_\beta = \left|\vec{Q}_2 * \vec{X}_\beta(t) - \vec{X}(t)\right| \tag{10}$$

$$\vec{d}_\delta = \left|\vec{Q}_3 * \vec{X}_\delta(t) - \vec{X}(t)\right| \tag{11}$$

where $\overrightarrow{X_\beta}$ and $\overrightarrow{X_\delta}$ represent the position vectors of the beta and delta wolves respectively.

The positions of the best three search agents (i.e., alpha, beta and delta wolves) are therefore calculated as:

$$\vec{X}_1(t) = \vec{X}_\alpha(t) - \vec{A}_1 * \vec{d}_\alpha \tag{12}$$

$$\vec{X}_2(t) = \vec{X}_\beta(t) - \vec{A}_2 * \vec{d}_\beta \tag{13}$$

$$\vec{X}_3(t) = \vec{X}_\delta(t) - \vec{A}_3 * \vec{d}_\delta \tag{14}$$

The locations of the other grey wolves in the pack are then updated based on the positions of the best three search agents using the update equation:

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1}(t) + \overrightarrow{X_2}(t) + \overrightarrow{X_3}(t)}{3} \tag{15}$$

The pseudo-code for the hybrid GWO-WOA algorithm is shown Fig. 5.

3.2. Model identification results

In this section, the accuracy of the LFR manipulator's grey-box model with GWO-WOA is analysed by comparing the joints' output responses with the white-box model, and responses of the real LFR manipulator system. The white-box model in this investigation is obtained by using the equation of motion derived

Table 2
Algorithm parameters for grey box modelling.

Algorithms' parameters	Value
Number of search agents	10
Maximum iteration	50
Number of variables	20
Objective function	Mean Squared Error (MSE)
Lower bounds	{0, 0.01, 0, 0, 0, 0, 0.01, 0, 0, 0, 0, 0.01, 0, 0, 0, 0, 0.01, 0, 0, 0}
Upper bounds	{100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100}

Table 3
Physical characteristics of the follower robot manipulator.

Symbol	Parameter	Value	Unit
m_0	Mass of link 0	0.21	kg
m_1	Mass of link 1	0.13	kg
m_2	Mass of link 2	0.11	kg
m_3	Mass of link 3	0.2148	kg
M	Mass at tip	0	kg
L_0	Length of link 0	0.086	m
L_1	Length of link 1	0.14	m
L_2	Length of link 2	0.049	m
L_3	Length of link 3	0.172	m
$L_{0,cg}$	Position of point mass of link 0	0.043	m
$L_{1,cg}$	Position of point mass of link 1	0.07	m
$L_{2,cg}$	Position of point mass of link 2	0.0245	m
$L_{3,cg}$	Position of point mass of link 3	0.086	m
g	Gravity constant	9.81	m/s^2
I_{xx0}	Mass moment of inertia in the x-axis (Joint 0)	1.34×10^{-4}	kg/m^2
I_{xx1}	Mass moment of inertia in the x-axis (Joint 1)	3.17×10^{-5}	kg/m^2
I_{xx2}	Mass moment of inertia in the x-axis (Joint 2)	8.96×10^{-5}	kg/m^2
I_{xx3}	Mass moment of inertia in the x-axis (Joint 3)	1.37×10^{-4}	kg/m^2
I_{yy0}	Mass moment of inertia in the y-axis (Joint 0)	8.26×10^{-3}	kg/m^2
I_{yy1}	Mass moment of inertia in the y-axis (Joint 1)	3.98×10^{-4}	kg/m^2
I_{yy2}	Mass moment of inertia in the y-axis (Joint 2)	2.37×10^{-5}	kg/m^2
I_{yy3}	Mass moment of inertia in the y-axis (Joint 3)	2.78×10^{-4}	kg/m^2
I_{zz0}	Mass moment of inertia in the z-axis (Joint 0)	1.24×10^{-4}	kg/m^2
I_{zz1}	Mass moment of inertia in the z-axis (Joint 1)	3.98×10^{-4}	kg/m^2
I_{zz2}	Mass moment of inertia in the z-axis (Joint 2)	1.13×10^{-4}	kg/m^2
I_{zz3}	Mass moment of inertia in the z-axis (Joint 3)	2.45×10^{-4}	kg/m^2

in the previous section and applying estimated motor and other physical system parameters. The parameters to be identified were originally estimated in a previous work carried out by Bernth [44], where the author conducted a friction and inertia experiment to determine the motor moment of inertia and viscous friction constant values for each joint of the LFR manipulator system. The armature resistance values were derived by using a multi-meter directly. It was also assumed that the motor torque constant is equal to the electromotive force constant. The parameters that have been used to run the optimisation algorithm are given in Table 2. The white box model system parameters and motor data are given in Tables 3 and 4 respectively. Two separate datasets named *Dataset 1* and *Dataset 2* are used in this work to validate the dynamic models. These datasets are real-time input and output joint position data of all joints of the LFR manipulator system, which were recorded via a serial Arduino measurement set-up during excitation of the leader manipulator.

An improvement in the accuracy of the developed white-box model is then considered by identifying the unknown motor parameters using the hybrid GWO-WOA, thus forming the grey-box model. The 20 unknown parameters of the follower robot manipulator as described in Section 3 are to be identified optimally using the hybrid GWO-WOA. The grey-box model parameters that were identified using the hybrid GWO-WOA are shown in Table 4, and it is noted that they were different from the white-box model parameters. The convergence curve of GWO-WOA obtained while identifying the unknown parameters of the LFR manipulator system is shown in Fig. 6. It is observed that the technique converges to a fixed value at the 47th iteration. A detailed comparative analysis of the hybrid GWO-WOA for model identification is given in [29] where the hybrid algorithm has proven to be advantageous compared to other state-of-the-art techniques.

Figs. 7 and 8 present the time domain predicted output responses of the developed grey-box and white box models in terms of 4-DOF system dynamics, together with the response of the LFR manipulator system using *Dataset 1* and *Dataset 2* respectively. The input and output of the LFR manipulator system are from manually exciting each joint of the leader robot manipulator in real-time. For consistency, the same input is introduced to the white-box and grey-box models to obtain both models' predicted output responses. Noticeable improvements of the grey-box model can be seen in majority of the plots as the grey-box model's responses closely match that of the experimental LFR manipulator system.

From analysing the MSE values in Table 5, the grey-box model is seen to show an 82.3% improvement with a total MSE value of 0.0715 compared to the white-box model's MSE value of 0.4042, using *Dataset 1*. Similarly, the total MSE of the grey-box model is 0.1121, which is a 73.9% improvement compared to the white-box model at 0.4293 using *Dataset 2*. These results therefore strengthen the premise that grey box models are better than white-box models. Consequently, the developed GWO-WOA based grey-box model will now be used for the development and synthesis of sophisticated control algorithms such as FLCs.

4. Fuzzy logic control design and tuning using GWO-WOA

Fuzzy control systems imitate the intuitive decision-making process of human beings. These decisions are based on an expert knowledge base which consist of fuzzy IF-THEN rules [45].

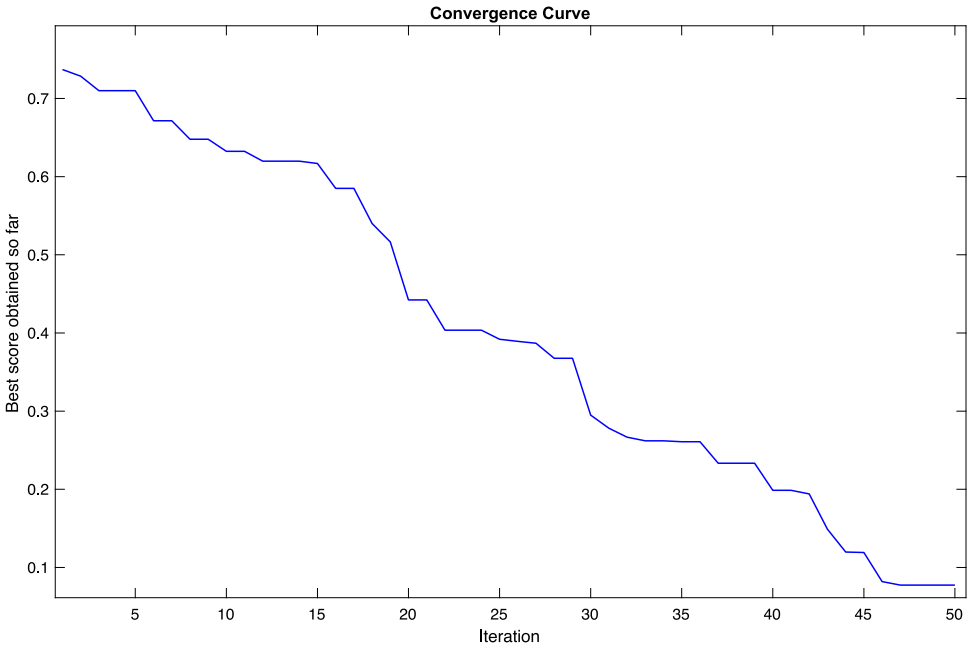


Fig. 6. Convergence curve of GWO-WOA for model identification.

Table 4
White-box and grey-box model parameters of the leader–follower robot manipulator.

Symbol	Parameter	Value	
		White-box model	Grey-box model
Joint 0 (Base Joint)			
J_0	Motor moment of inertia	0.034 kg m^2	18.302 kg m^2
Ra_0	Motor armature resistance	3.6 Ω	0.029 Ω
Kt_0	Motor torque constant	0.9 N m/A	6.661 N m/A
Bm_0	Motor viscous friction constant	0.046 N m s	0.495 N m s
r_0	Motor reduction gear ratio	0.8	1.429
Joint 1 (Shoulder Joint)			
J_1	Motor moment of inertia	0.055 kg m^2	4.477 kg m^2
Ra_1	Motor armature resistance	0.8 Ω	0.092 Ω
Kt_1	Motor torque constant	1.2 N m/A	1.124 N m/A
Bm_1	Motor viscous friction constant	0.063 N m s	5.731 N m s
r_1	Motor reduction gear ratio	0.8	2.921
Joint 2 (Wrist Joint)			
J_2	Motor moment of inertia	0.025 kg m^2	0.918 kg m^2
Ra_2	Motor armature resistance	2.4 Ω	0.275 Ω
Kt_2	Motor torque constant	1.0 N m/A	8.922 N m/A
Bm_2	Motor viscous friction constant	0.037 N m s	20.405 N m s
r_2	Motor reduction gear ratio	0.8	0.897
Joint 3 (Wrist – Twist Joint)			
J_3	Motor moment of inertia	0.009 kg m^2	0.176 kg m^2
Ra_3	Motor armature resistance	6.3 Ω	0.486 Ω
Kt_3	Motor torque constant	0.8 N m/A	1.189 N m/A
Bm_3	Motor viscous friction constant	0.025 N m s	0.457 N m s
r_3	Motor reduction gear ratio	0.8	6.854

The overall fuzzy logic control process can be summed up into three (3) stages namely fuzzification, fuzzy inference and defuzzification processes. The fuzzification step converts crisp inputs into fuzzy sets through input membership functions. The fuzzy inference system (FIS) is the knowledge base which consists of linguistic control rules. Finally, the defuzzification step generates crisp control outputs from output fuzzy sets using either the Mamdani or Sugeno fuzzy inference system (FIS).

The proposed Fuzzy PD + I controller in this work is a Mamdani-type that consist of two inputs and an intermediate output. The overall controller output comprises the sum of this intermediate output and an integral term. This phenomenon is illustrated in Fig. 9. This Fuzzy PD + I controller is simple to implement as its structure is similar to the traditional PID controller. Considering the leader and follower robot manipulators have four joints separately, each joint will be controlled in joint space. This implies that four FLCs are required for the position control of the whole experimental leader–follower manipulator system. A block diagram of the overall LFR manipulator control scheme is demonstrated in Fig. 10.

In terms of 1-DOF, the first input of the FLC is the tracking error, which is defined as the difference between the desired joint angle position from the leader manipulator, $\theta_m(k)$ and the actual joint angle position of the follower manipulator, $\theta_s(k)$. The second

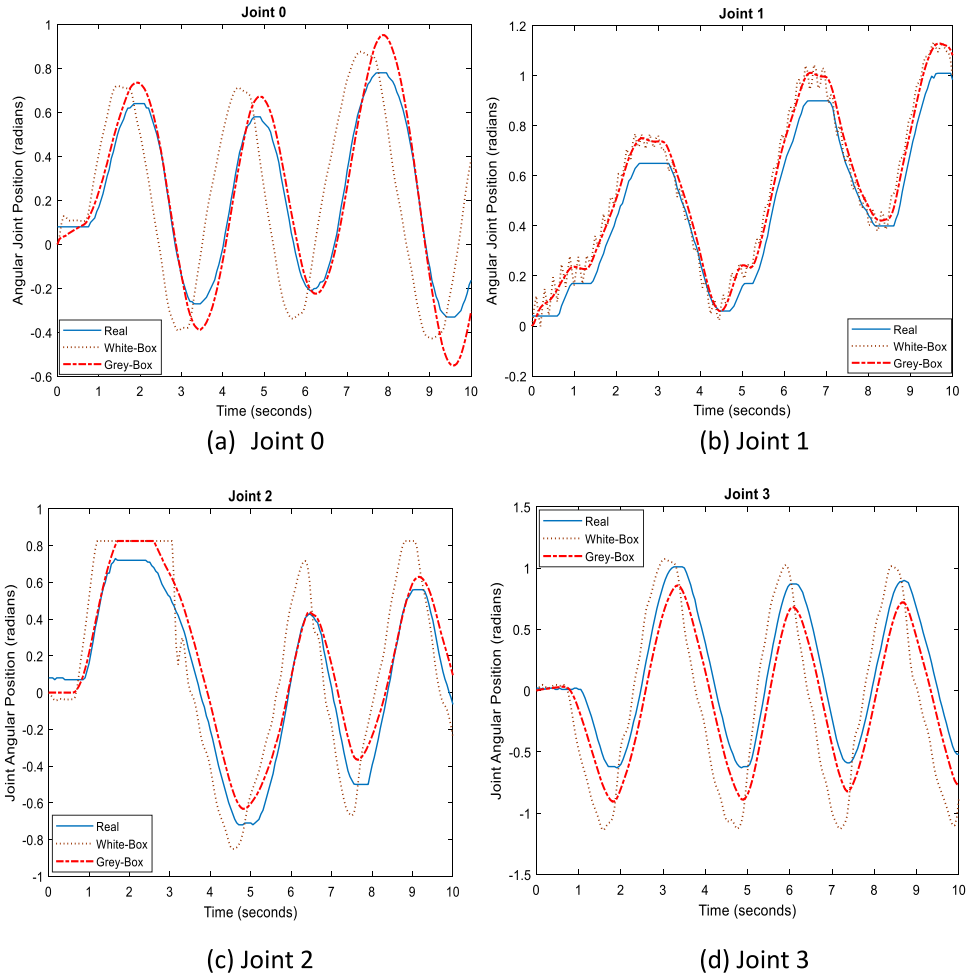


Fig. 7. Time domain response plots using dataset 1.

Table 5
Mean squared error comparison of white-box and grey-box models.

Dataset 1		
	White-box model	Grey-box model
Joint 0	0.1080	**0.0077**
Joint 1	0.0090	**0.0062**
Joint 2	0.0554	**0.0120**
Joint 3	0.2318	**0.0456**
All joints	**0.4042**	**0.0715**
Dataset 2		
	White-box model	Grey-box model
Joint 0	0.0949	**0.0307**
Joint 1	0.0081	**0.0050**
Joint 2	0.0527	**0.0122**
Joint 3	0.2736	**0.0643**
All joints	**0.4293**	**0.1122**

input of the FLC is the change in tracking error, while the overall output of the FLC is the control voltage from the servo motor that drives the robot manipulator.

The tracking error, $e\,(k)$ and change in tracking error, $ce\,(k)$ are defined as

$$e\,(k) = \theta_m\,(k) - \theta_s(k) \tag{16}$$

$$ce\,(k) = e\,(k) - e(k-1) \tag{17}$$

The overall output of the Fuzzy PD + I controller is a function of position tracking error and angular velocity. This function is, thus, expressed mathematically as:

$$u\,(k) = K_u * \left[f\left(K_p e\,(k)\,, K_d ce\,(k)\right)\right] + K_i \int e\,(k)\,dk \tag{18}$$

where

$e\,(k)$	Tracking error
$ce\,(k)$	Change in tracking error
$u(k)$	Control voltage
K_p	Proportional term scaling factor
K_i	Integral term scaling factor
K_d	Derivative term scaling factor
K_u	Control output scaling factor

The fuzzification of inputs and defuzzification of the output are characterised by a combination of trapezoidal and triangular MFs, as shown in Fig. 11. The newness of this proposed FLC is in the design of the output MFs where triangular MFs of the control output are used to mimic singleton MFs. The advantage of this method is that the computational time of this fuzzy logic controller is improved while also providing the full benefits of Mamdani-type systems. In addition, the rising concern that increasing the number of MFs of a FLC increases the total computation time and by extension, the overall performance of the robot, is addressed by using only nine MFs per FLC.

The input and output variables each have three MFs, which are defined in the universe of discourse [−1,1]. The input scaling factors are used to regulate the tracking error and angular velocity

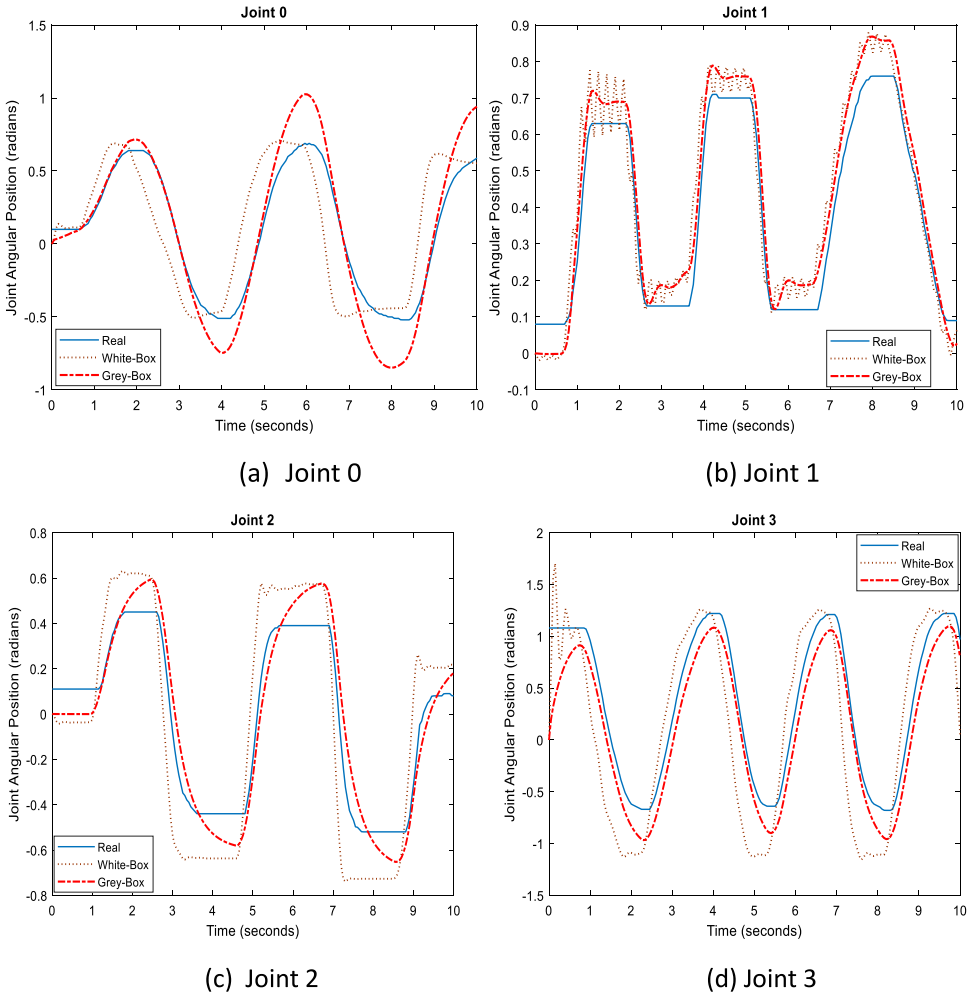


Fig. 8. Time domain response plots using dataset 2.

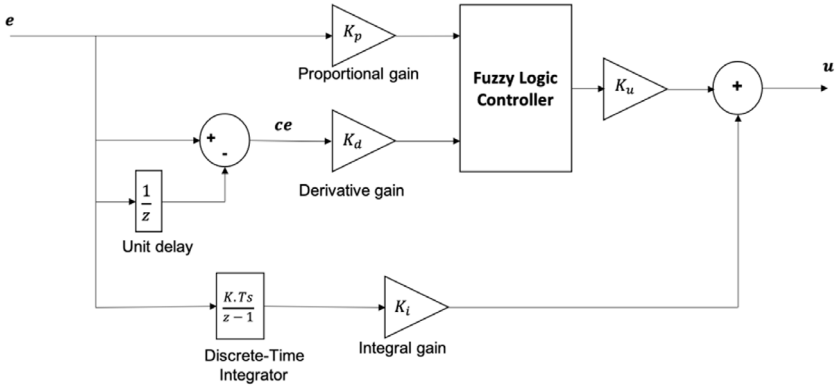


Fig. 9. Fuzzy PD + I control structure.

within the range [−1, 1] before going into the FLC. Similarly, the output scaling factor is used to convert the normalised output value to a real control voltage that will be fed into the robot manipulator. In Fig. 11, the linguistic set {NE, ZE, PE} is assigned as negative error, zero error and positive error. The sets {NCE, ZCE and PCE} and {NU, ZU, PU} are assigned as negative change in error, zero change in error and positive change in error, and negative output voltage, zero output voltage and positive output voltage respectively. The designed Mamdani-type FLC uses a min–max method of inferencing comprising of nine fuzzy control rules. These rules are presented in Table 6 in the form: if (e is XY) and (ce is XZ) then (u is YZ).

While FLCs have several advantages, the challenge of deciding the membership functions' parameters still exists. The trial-and-error method is often used which can become burdensome. This task can be solved by tuning the FLC parameters using optimisation algorithms to obtain optimum solutions based on a given cost objective function [46]. This section, therefore, explores the optimisation of the designed FLCs parameters using the hybrid GWO-WOA algorithm, for application on the experimental 4-DOF LFR manipulator system. With the optimal parameters of the

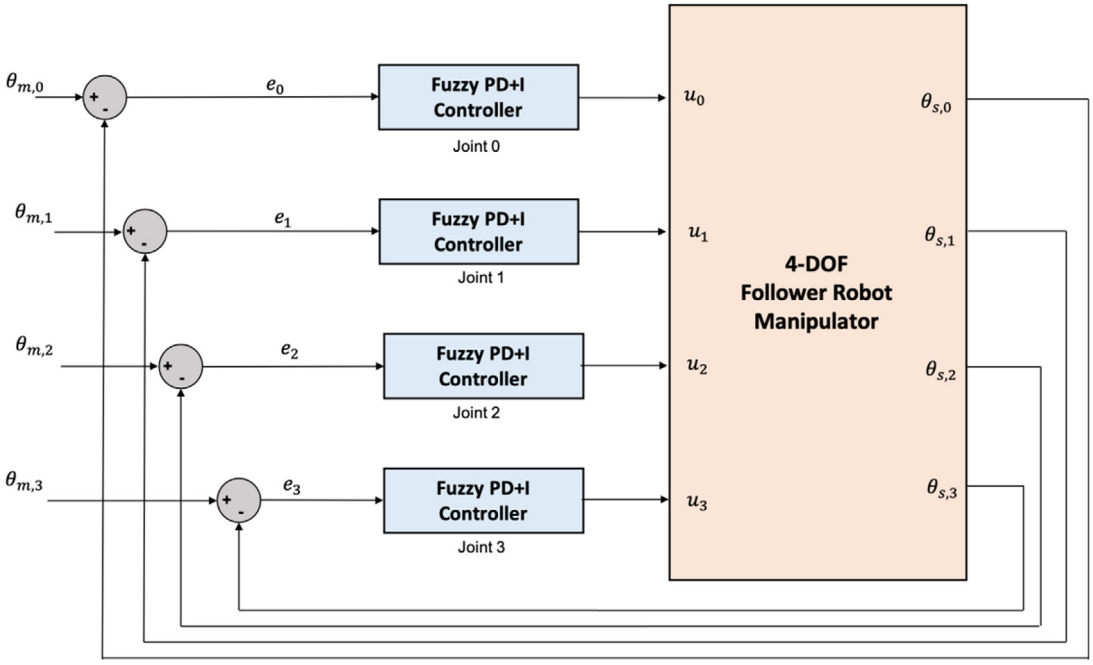
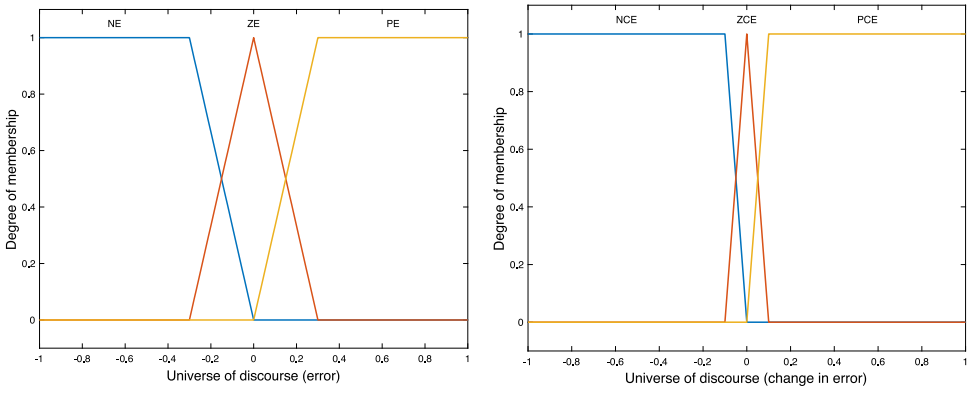
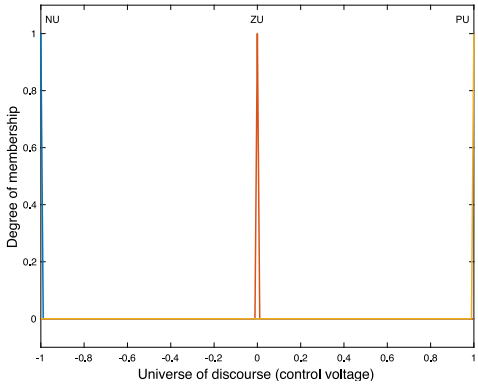


Fig. 10. Overall leader–follower robot manipulator control scheme.



(a) Position tracking error (rad) (b) Angular velocity (rad/s)



(c) Control voltage (V)

Fig. 11. Membership functions before tuning.

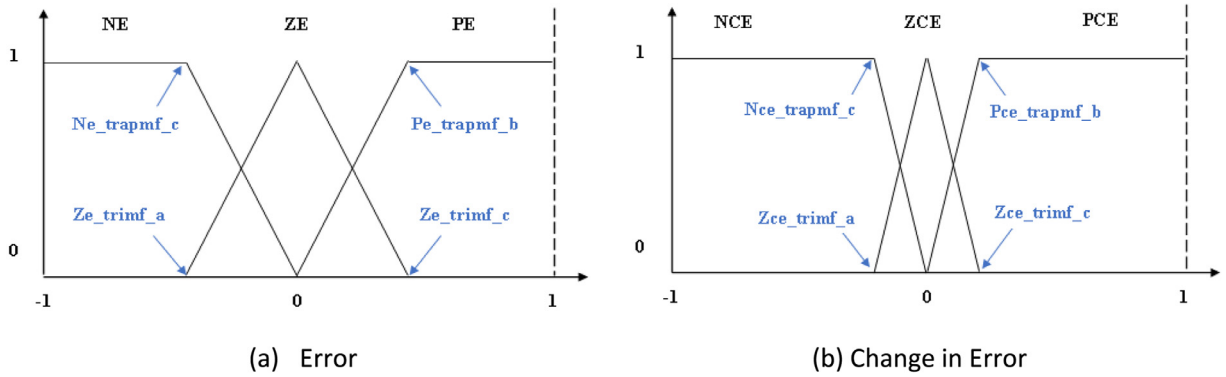


Fig. 12. Tuneable parameters of input membership functions.

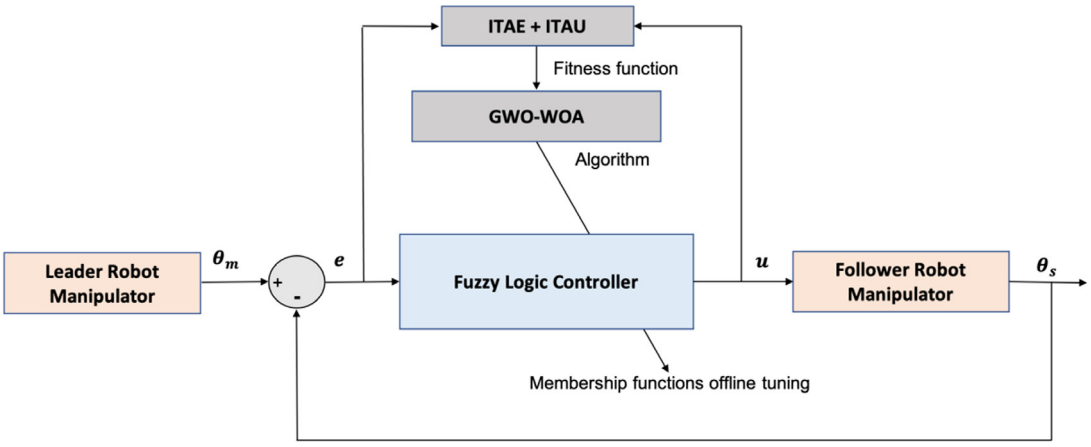


Fig. 13. Leader–follower closed loop control system.

Table 6
Fuzzy logic controller rule base.

If (*e* is *NE*) and (*ce* is *NCE*) then (*u* is *NU*)
If (*e* is *NE*) and (*ce* is *ZCE*) then (*u* is *NU*)
If (*e* is *NE*) and (*ce* is *PCE*) then (*u* is *ZU*)
If (*e* is *ZE*) and (*ce* is *NCE*) then (*u* is *NU*)
If (*e* is *ZE*) and (*ce* is *ZCE*) then (*u* is *ZU*)
If (*e* is *ZE*) and (*ce* is *PCE*) then (*u* is *PU*)
If (*e* is PE) and (*ce* is *NCE*) then (*u* is *ZU*)
If (*e* is *PE*) and (*ce* is *ZCE*) then (*u* is *PU*)
If (*e* is *PE*) and (*ce* is *PCE*) then (*u* is *PU*)

MFs, the proposed FLC can be enhanced to minimise the position tracking errors, steady state errors and the time delay between the leader and follower's joints of the LFR manipulator system.

For accurate control of the LFR manipulator system, optimal parameters of the FLCs, i.e., input MF variables are determined through optimisation offline, using the hybrid GWO-WOA algorithm in MATLAB/Simulink environment. The chosen objective function for this tuning process is based on minimising the sum of integral time weighted absolute error (ITAE) and integral time weighted absolute control signal (ITAU). They are defined as:

$$ITAE = \int_0^\infty k\,|e(k)|\,dk \tag{19}$$

$$ITAU = \int_0^\infty k\,|u(k)|\,dk \tag{20}$$

Therefore, the overall objective function, J is expressed as:

$$J = \int_0^\infty k\,|e(k)|\,dk + \int_0^\infty k\,|u(k)|\,dk \tag{21}$$

A closed loop system is typically driven by error i.e., the difference between the setpoint and process output signals. As the objective functions ITAE and ITAU depend on the absolute error and absolute control signal respectively, which is weighted with the time of occurrence of the error and control, it is believed that using the sum of ITAE and ITAU could provide an optimal performance during the optimisation process. In Eq. (21), the weightings used for the cost functions ITAE and ITAU are set to 1 respectively.

Different techniques of tuning the fuzzy logic controllers were investigated in this study to determine the best way of achieving a great control performance on the LFR manipulator system. These methods include:

a. Tuning of the FLC's scaling factors while keeping all MFs constant.
b. Tuning of the FLC's input MFs parameters while keeping the output MFs and scaling factors constant
c. Tuning the FLC's scaling factors and input MFs parameters simultaneously.

Simulation results of the follower robot manipulator responses to a multi-step reference input from the leader robot manipulator using all the tuning techniques are shown and discussed in the next section. It was discovered that tuning the FLCs' input MFs while keeping the output membership ('singleton') functions and scaling factors constant produced the best results. Fig. 12 highlights the tuneable parameters of the input membership functions for each FLC. The fuzzy control performance is improved by optimising the input membership function parameters Ne_trapmf_c, Ze_trimf_a, Pe_trapmf_b, Ze_trimf_c, Nce_trapmf_c, Zce_trimf_a, Pce_trapmf_b and Zce_trimf_c using GWO-WOA. The boundary

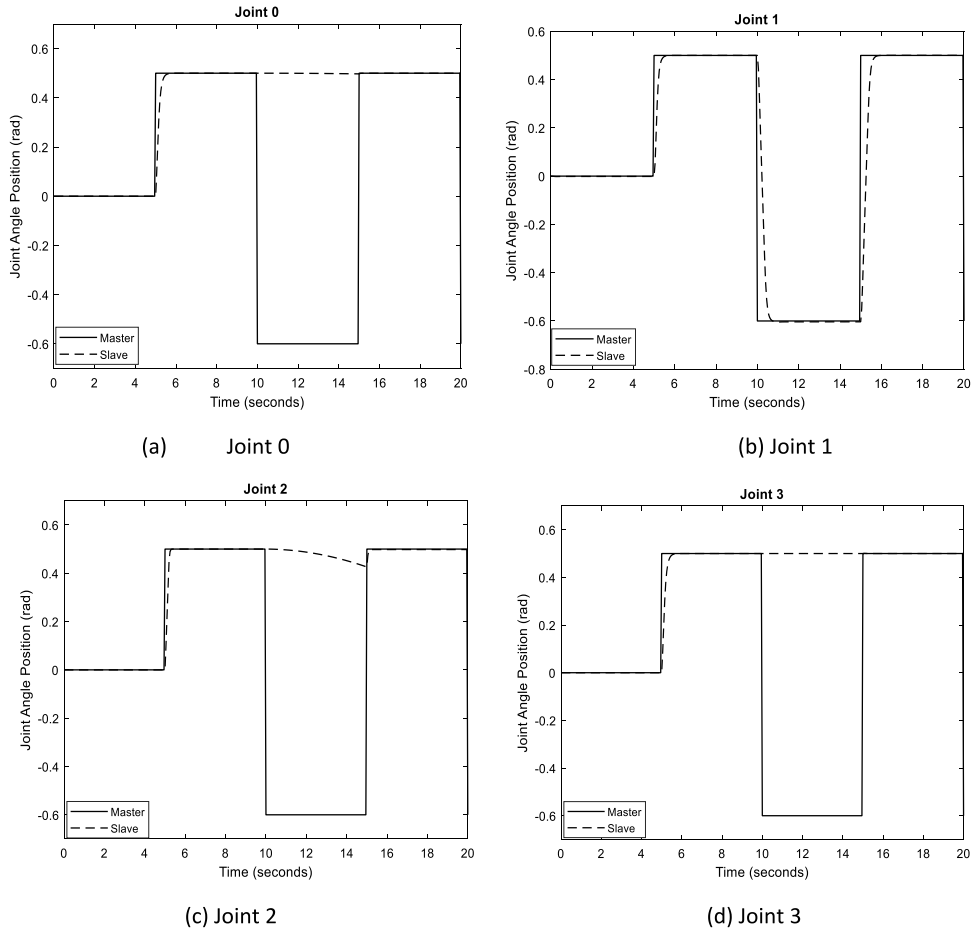


Fig. 14. Simulated multi-step output response using tuned scaling factors.

of each tuneable parameter is based on the symmetry and structure of the membership functions. For instance, the parameters {Ne_trapmf_c, Ze_trimf_a, Nce_trapmf_c and Zce_trimf_a} are within the range [−1,0] while the parameters {Pe_trapmf_b, Ze_trimf_c, Pce_trapmf_b and Zce_trimf_c} are between [0,1].

These bounds will be used as part of the GWO-WOA algorithms' parameters to find the optimal parameters of the MFs. It is important to note that the four different FLCs for the individual robot joints will be optimised independently. Using optimisation lingo, the aim of this 'problem' is to find optimal MFs parameters by minimising the objective function using an algorithm and within given constraints. The algorithm parameters that have been used for tuning the MFs of the FLCs are given in Table 7. Let us assume the vector for optimisation is w, and this vector is defined as:

$$w = \{w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8\}_i \qquad \text{for}(i = 0, 1, 2, 3)$$

where $\{w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8\}_i$ represents {Ne_ trapmf _c, Ze_trimf_a, Ze_trimf_c, Pe_trapmf_b, Nce_trapmf_c, Zce_trimf _a, Zce_trimf_c and Pce_trapmf_b} respectively for each joint, *i* of the LFR manipulator system.

Table 7
GWO-WOA parameters for tuning MFs.

Algorithms' parameters	Value
Number of search agents	10
Maximum iteration	30
Number of variables	8
Objective function	ITAE + ITAU
Lower bounds	{-1, −1, 0, 0, −1, −1, 0, 0}
Upper bounds	{0, 0, 1, 1, 0, 0, 1, 1}

5. Implementation and results

5.1. Implementation

The fuzzy PD + I controller designed based on the GWO-WOA is implemented on the experimental 4-DOF LFR manipulator system for verification of their real-time performance. Fig. 13 shows the block diagram of the closed-loop control during the optimisation process of the FLCs. The developed GWO-WOA based grey-box model discussed earlier is used to determine the optimal MFs of the input variables. This optimisation process is carried out offline using the hybrid GWO-WOA algorithm in MATLAB/Simulink environment, to ensure an effective control of the LFR manipulator system.

In this investigation, three measures have been used to analyse the effectiveness of the fuzzy PD + I controller on all joints of the experimental LFR manipulator system. Moreover, a comparative analysis on the performance of the LFR manipulator system before and after tuning the fuzzy PD + I controllers is also carried out. The measures are:

- Input tracking: This is measured by obtaining the mean squared error (MSE) between the actual response and the input command. A low MSE is desirable as this indicates closeness to the desired response.
- Steady-state error: The steady state error is calculated by finding the difference between the desired final position

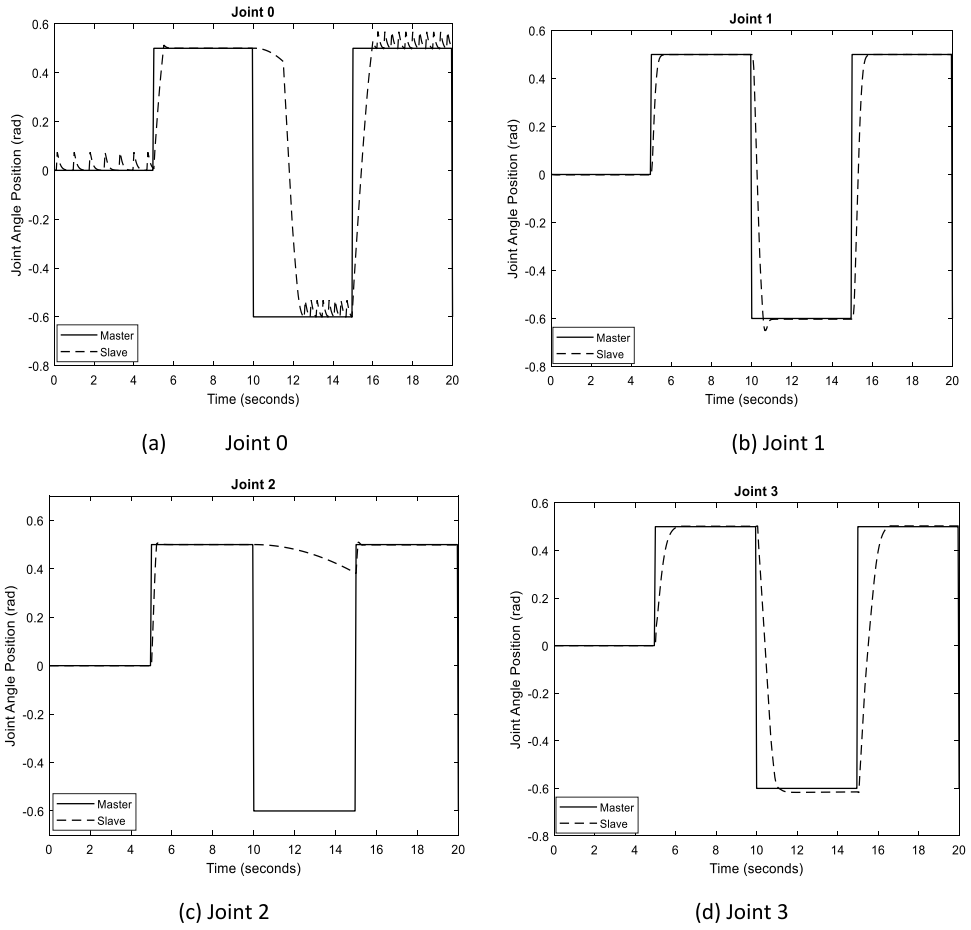


Fig. 15. Simulated multi-step output response using tuned scaling factors and membership functions.

from the leader robot manipulator and the follower manipulator's actual position in steady state.

- Time delay: This is calculated by working out the difference between the times that the leader robot manipulator gives a position command and the time the follower robot manipulator begins to respond.

For comparison of the control performance before and after optimisation, it is intended to use the same input commands for each of the joints from the leader robot. However, in the actual experiments, it is not possible to get the exact same trajectory, as the leader robot is manually excited. To mitigate this, similar command trajectories from the leader manipulator have been used. For example, the step and square responses have similar time periods. The feasibility of the designed FLC is tested initially by using step signals that are given by manually exciting the leader robot manipulator. To further investigate the robustness of the fuzzy PD + I controller, another dataset which includes multi-step trajectories is tested. The purpose of using different trajectories and signals is to ensure that the designed and optimised controllers can cope with uncertainties, which mimic real-life scenarios.

5.2. Comparative simulation study on different FLC tuning methods

In this section, the best tuning method for optimising the FLC is investigated through simulations. Validation tests using a multi-step trajectory are carried out on the three FLC tuning approaches described in Section 4. Subsequent to observing the behaviour of the LFR manipulator via simulation tests, the best approach for tuning the FLCs will be implemented on the experimental LFR manipulator system.

Table 8
Tuned scaling factors of the FLCs.

	Joint 0	Joint 1	Joint 2	Joint 3
K_p	1.0427	0.754	1.8448	0.9957
K_d	0.0018	0.3097	0.0089	0.0008
K_i	0.002	0.5871	0.0449	0.0004
K_u	20	20	20	20

5.2.1. FLC tuning of scaling factors

The scaling factors of the FLC were tuned using the GWO-WOA algorithm offline while keeping the MFs constant. The tuned K_p, K_d, K_i and K_u scaling factors obtained after optimisation are given in Table 8. The tuned scaling factors are then applied to the LFR manipulator's simulation model to give the multi-step output responses for all joints, as shown in Fig. 14. From the output responses, a change in direction of the leader robot's multi-step trajectory shows how the follower's joints 0, 2 and 3 struggle to track the leader's movements with large tracking errors. It is clear that tuning the scaling factors of the FLCs alone does not provide a robust control performance for the LFR manipulator system.

5.2.2. FLC tuning of scaling factors and membership functions

Another simulation test was carried out to check the efficacy of tuning the FLC's MFs and scaling factors simultaneously using multi-step trajectories, and the obtained parameters are given in Table 9. Similar to the results in Section 5.2.1, the follower

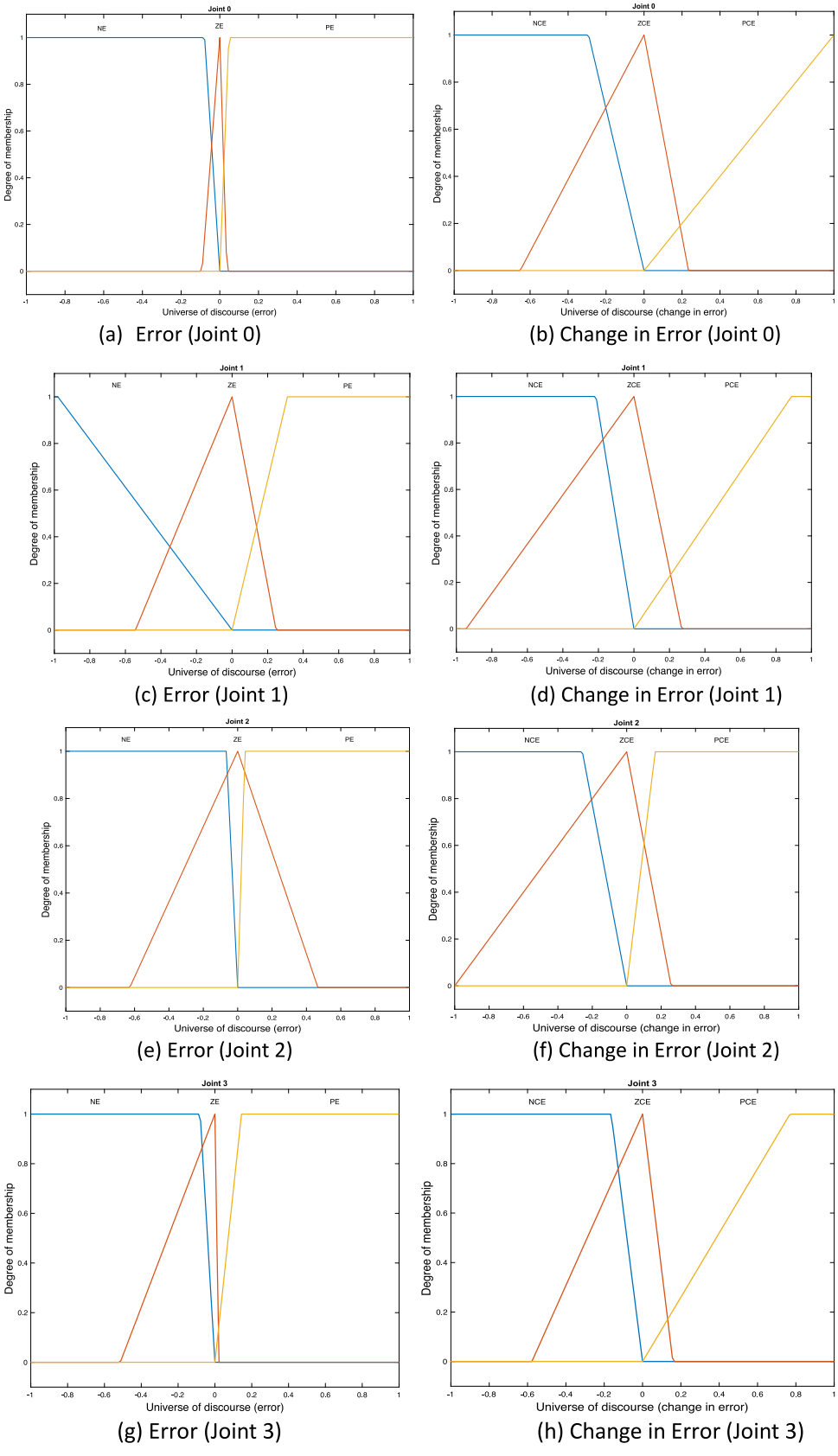


Fig. 16. Fuzzy logic input membership functions after tuning.

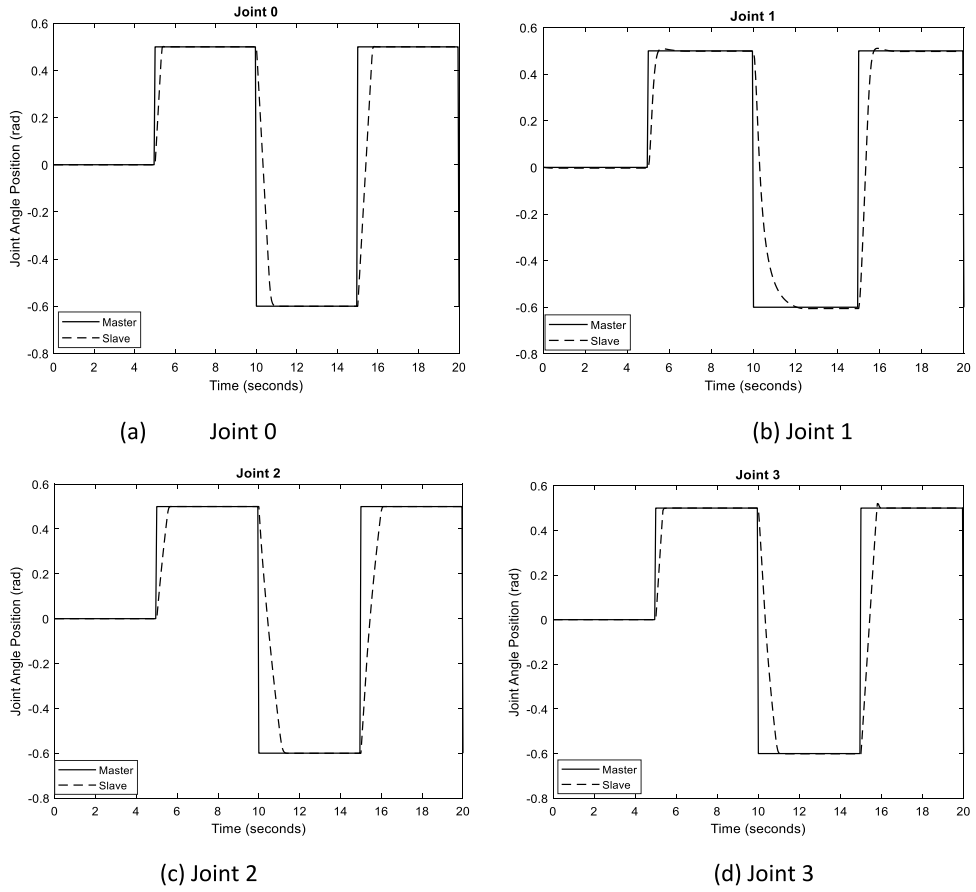


Fig. 17. Simulated multi-step output response using tuned membership functions.

Table 9
Tuned scaling factors and input membership functions of the FLCs.

MFs parameters	Joint 0	Joint 1	Joint 2	Joint 3
K_p	0.956	0.783	1.853	0.824
K_d	0.243	1.699	0.017	1.223
K_i	0.461	0.343	0.075	2.072
K_u	15.3961	19.8269	20.0	18.802
Ne_trapmf_c	−0.1675	−0.1142	−0.7425	−0.0852
Ze_trimf_a	−0.4407	−0.3600	−0.1322	−0.2222
Ze_trimf_c	0.0	0.4083	0.3893	0.6173
Pe_trapmf_b	1.0	0.3134	0.1116	0.3293
Nce_trapmf_c	−0.0943	−0.5980	−0.1630	−0.5694
Zce_trimf_a	−0.6021	−0.7104	−0.2780	−0.1694
Zce_trimf_c	0.3860	0.4352	0.7679	0.3421
Pce_trapmf_b	0.1119	0.7732	0.6847	0.2051

manipulator struggles to track the multi-step trajectory given by the leader manipulator thus resulting in large tracking errors and steady state errors between joints 0 and 2. The output responses are illustrated in Fig. 15.

5.2.3. FLC tuning of membership functions

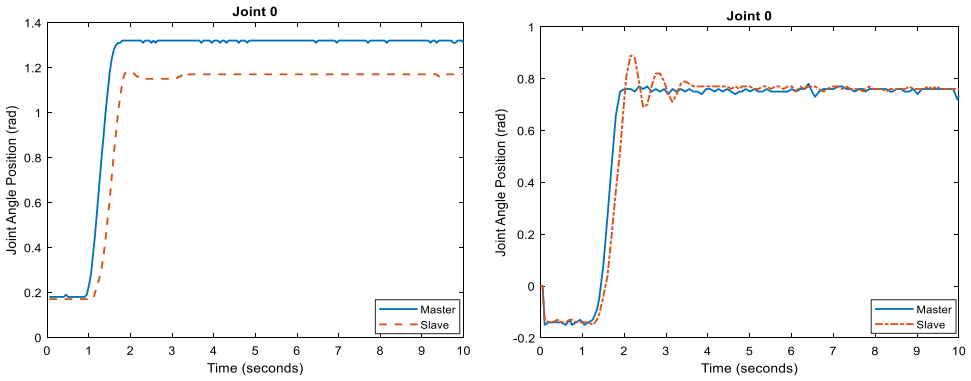
The input MFs of the FLC are tuned using the GWO-WOA algorithm offline while keeping the output MFs and scaling factors constant. The scaling factors K_p, K_d, K_i and K_u used for joints 0, 1, 2 and 3 are {0.513, 0.160, 0.313, 15.0}, {0.590, 0.147, 0.201, 11.55}, {0.630, 0.244, 0.251, 13.15} and {0.782, 0.169, 0.233, 14.5} respectively, and the optimised input MFs for all joints are illustrated in Fig. 16. Table 10 presents the optimised input MFs parameters after tuning with GWO-WOA offline, while keeping the output MFs and scaling factors constant. Simulation results in Fig. 17 reveal a favourable performance of the tuned FLCs across all joints as the follower manipulator is able to efficiently track the multi-step trajectories given by the leader with no tracking errors or steady state errors.

5.3. Experimental results

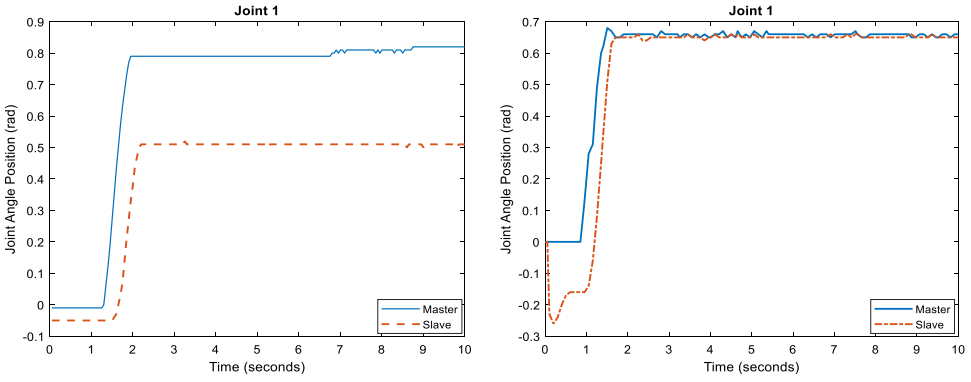
The main idea behind using the hybrid GWO-WOA for tuning the MFs of the FLC is to search a global optimisation space to find a set of parameters that will produce the best control performance of the fuzzy logic controller. The tuned MFs parameters given in Table 10 have been used for the real-time fuzzy logic control of the experimental LFR manipulator system. A comparison of the experimental step output responses for all joints of the LFR manipulator system before and after tuning the MFs using the GWO-WOA algorithm is depicted in Fig. 18. The performances of the FLCs are quantified and analysed in terms of MSE, steady-state error and time delay, as shown in Table 11.

The results depicted in Fig. 18 and Table 11 demonstrate a significant improvement in the trajectory tracking performance of the FLC using the GWO-WOA approach, particularly in terms of MSE, steady state error and time delays between the leader and follower manipulator joints. For example, the total MSE of the entire LFR manipulator system after tuning is 0.0405 in comparison to the MSE value of 0.1858 before tuning. This shows a 78% improvement which is due to the improved exploration capability of the hybrid GWO-WOA. The hybrid algorithm can find optimum parameters of the FLC's MFs that minimises its objective function and updates its best solution using the top three fitness scores.

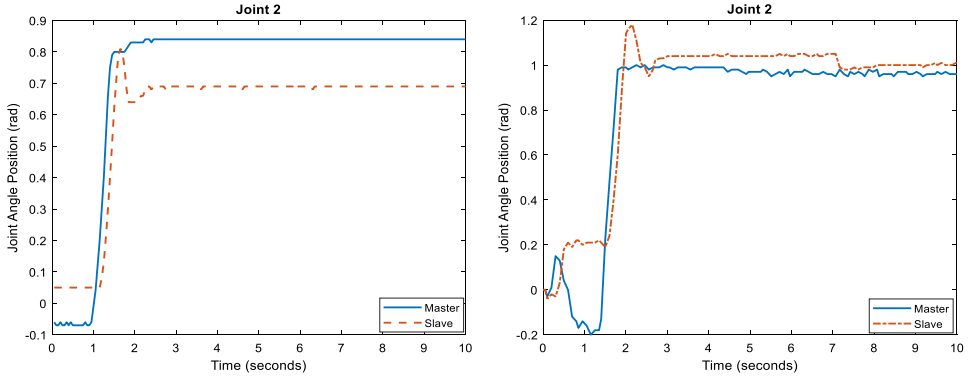
Similarly, a significant reduction in the steady state errors and time delay across all joints of the LFR manipulator system was achieved with the GWO-WOA approach. Joint 1's steady state



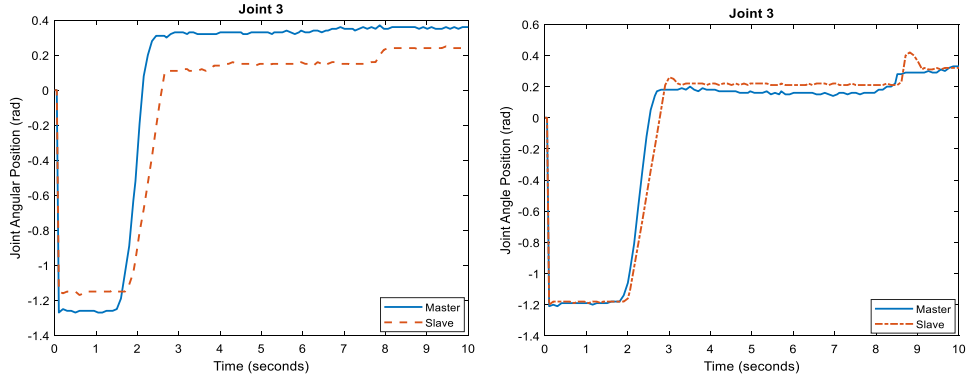
(a) Before and After Optimisation of Joint 0



(b) Before and After Optimisation of Joint 1



(c) Before and After Optimisation of Joint 2



(d) Before and After Optimisation of Joint 3

Fig. 18. Step response performance of the LFR manipulator system before and after GWO-WOA optimisation.

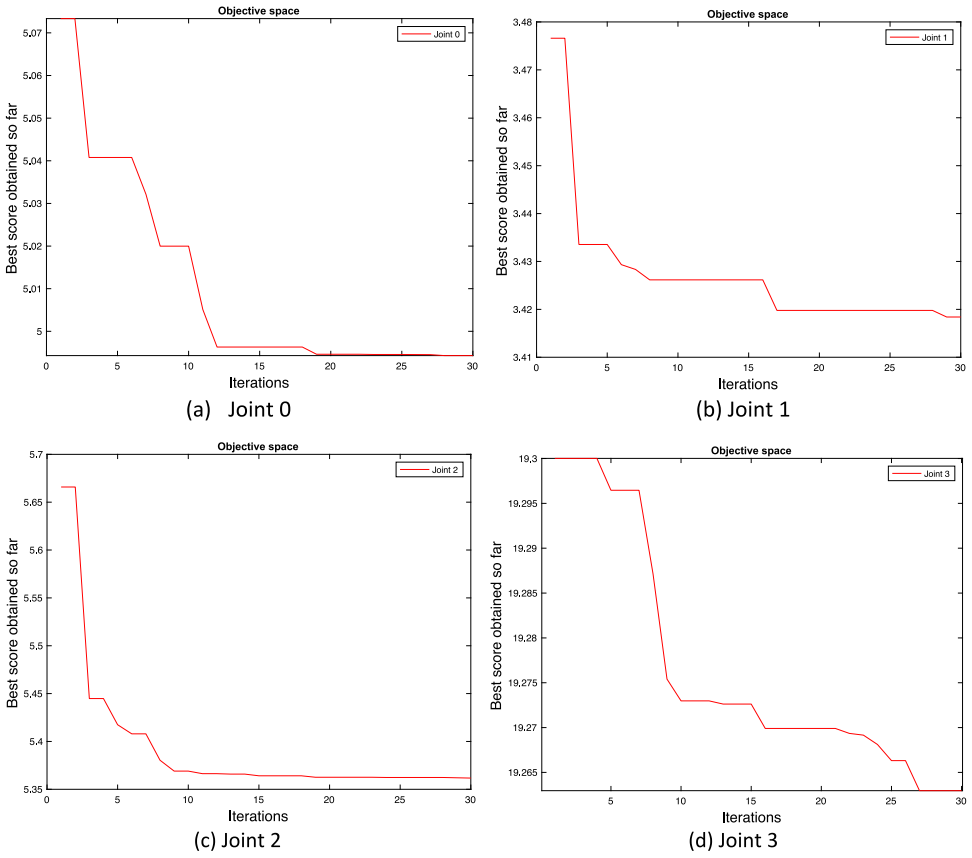


Fig. 19. Convergence curves of FLCs after GWO-WOA optimisation.

Table 10
Tuned inputs' membership functions parameters of the FLCs.

MFs parameters	Before tuning	Joint 0	After tuning (J1)	After tuning (J2)	After tuning (J3)
Ne_trapmf_c	−0.3	−0.0785	−0.9801	−0.0659	−0.0800
Ze_trimf_a	−0.3	−0.0920	−0.5437	−0.6279	−0.5139
Ze_trimf_c	0.3	0.0363	0.2473	0.4669	0.0212
Pe_trapmf_b	0.3	0.0466	0.3113	0.0423	0.1444
Nce_trapmf_c	−0.1	−0.2918	−0.2138	−0.2578	−0.1637
Zce_trimf_a	−0.1	−0.6524	−0.9454	−1.0	−0.5776
Zce_trimf_c	0.1	0.2340	0.2689	0.2580	0.1579
Pce_trapmf_b	0.1	1.0	0.8877	0.1657	0.7692

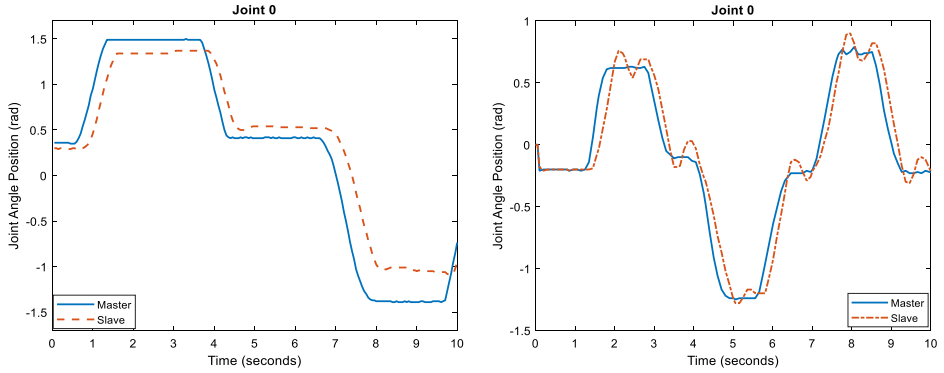
error was reduced by 96.7% which is substantial as large steady state errors could become catastrophic in surgical applications, for instance. Also, a reduction in time delay between the leader and follower's responses ensures that the LFR manipulator system remains stable always, as perturbation and disturbances introduced to the system can be acted upon quickly. The convergence curves of all joints of the LFR manipulator system while tuning the FLC parameters using GWO-WOA are given in Fig. 19. Joints 0 and 2 converge to the final solution after 20 iterations while Joints 1 and 3 converge to the final solution after iterations 29 and 27 respectively.

5.4. Robustness test

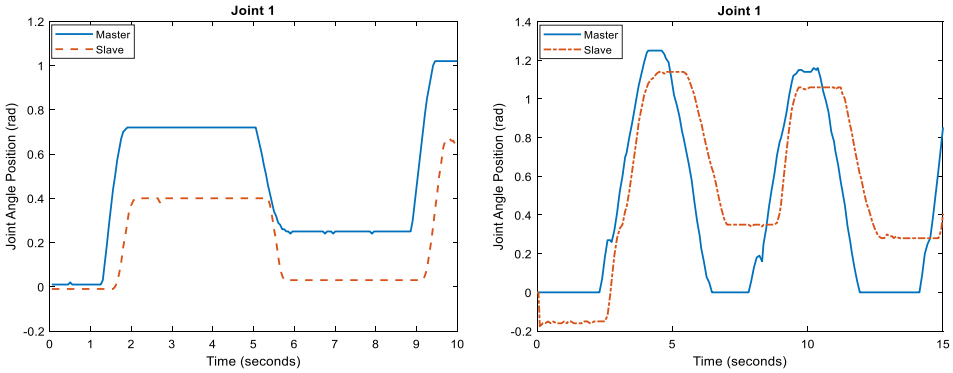
The multi-step (or square) signals have also been used to test the robustness of the tuned FLC for application on the experimental LFR manipulator system in real time. These multi-square trajectory signals are multi-step in nature and are introduced by exciting the leader manipulator while the follower manipulator is expected to follow the given trajectory. Fig. 20 shows the multi-step responses of the LFR manipulator system for all joints before and after optimisation.

Table 11
Performance comparison of the fuzzy logic controllers before and after tuning.

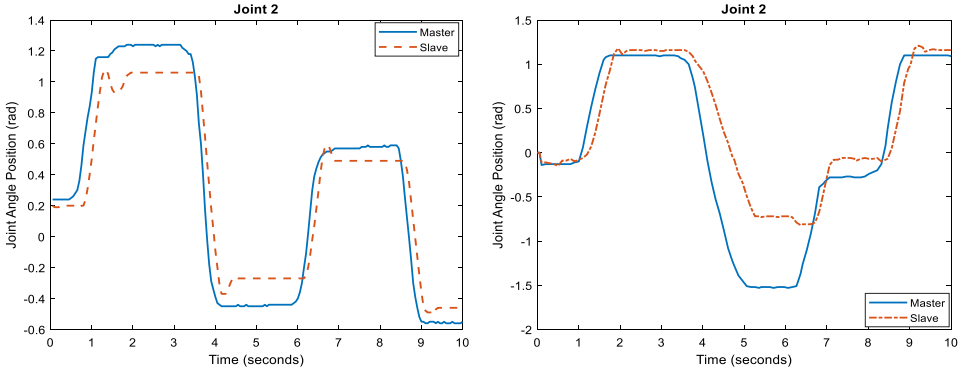
	Before tuning	After tuning
Mean squared error		
Joint 0	0.0324	**0.0034**
Joint 1	0.0808	**0.0106**
Joint 2	0.0235	**0.0180**
Joint 3	0.0492	**0.0086**
All joints	**0.1858**	**0.0405**
Steady-state error (radians)		
Joint 0	0.14	**0.02**
Joint 1	0.31	**0.01**
Joint 2	0.15	**0.03**
Joint 3	0.12	**0.01**
Time delay (seconds)		
Joint 0	0.20	**0.10**
Joint 1	0.25	**0.10**
Joint 2	0.25	**0.15**
Joint 3	0.30	**0.10**



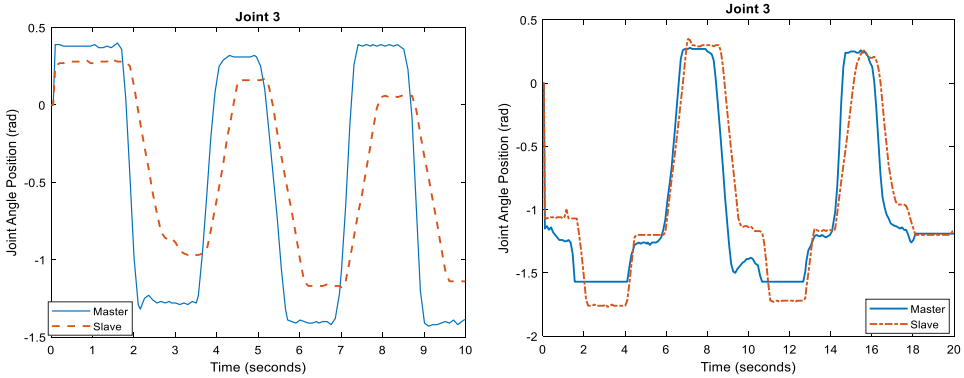
(a) Before and After Optimisation of Joint 0



(b) Before and After Optimisation of Joint 1



(c) Before and After Optimisation of Joint 2



(d) Before and After Optimisation of Joint 3

Fig. 20. Multi-step response performance of all joints of LFR manipulator system before and after GWO-WOA optimisation.

Table 12
MSE comparison of experimental multi-step response before and after GWO-WOA tuning the FLCs.

MSE	Before tuning	After tuning
Joint 0	0.1021	**0.0250**
Joint 1	0.1040	**0.0974**
Joint 2	**0.0384**	0.2364
Joint 3	0.2974	**0.0772**
All joints	**0.5420**	**0.4360**

Although some tracking errors exist across the joints in Fig. 20, the tuned FLCs are observed to still perform satisfactorily as the follower manipulator joints are seen to follow the trajectories of the leader robot's joints closely in real time. The tracking behaviour of the LFR manipulator system after implementing the tuned FLCs is evaluated in Table 12 using the MSE measure. It is observed that the fuzzy PD + I controller performance of the LFR manipulator system after tuning is better and more robust in Joints 0, 1 and 3. The total MSE for all joints after tuning is 0.4360 while the MSE before tuning is 0.5420. This 19.6% improvement implies that the tuned FLC is robust enough to handle system uncertainties such as unfamiliar trajectories, when tuned appropriately.

6. Conclusions

This proposed work considers the grey-box modelling and optimal control design of a FLC for application on an experimental 4-DOF LFR manipulator system, using a hybrid grey wolf–whale optimisation algorithm (GWO-WOA). Analysis on the performance of the GWO-based grey-box model in comparison to the Euler–Lagrange based white box model has clearly shown the advantage of utilising the grey-box modelling technique. The fuzzy PD + I controller presented in this paper provides a practical approach for the real time trajectory tracking and position control of a LFR system. Parameters of the input MFs for the fuzzy logic controller have been obtained by minimising the sum of ITAE and ITAU objective functions using the hybrid GWO-WOA. Experimental results have shown a satisfactory tracking ability, and a reduction in the steady state errors and time delay across all joints between the leader and follower robot manipulators. These results guarantee the performance of GWO-WOA in determining optimal system and control parameters for use in position control and trajectory tracking applications. In future research, GWO-WOA can be employed for the online tuning of the fuzzy controller's scaling factors and membership functions.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Vitiello V. Emerging robotic platforms for minimally invasive surgery. IEEE Rev Biomed Eng 2013;6:111–26.

[2] Sang H, Wang S, Zhang L, He C, Zhang L, Wang X. Control design and implementation of a novel master – slave surgery robot system, MicroHand A. Int J Med Robot Comput Assist Surg 2011;(7):334–47. http://dx.doi.org/10.1002/rcs.

[3] Baek YM, et al. Highly precise master–slave robot system for super micro surgery. In: Proceedings - IEEE, RAS & EMBS international conference on biomedical robotics and biomechatronics; 2010, p. 740–5.

[4] Wu W, Liu DJ, Liu JS, Wu J. Master–slave intelligent robot telepresence system. Comput Ind Eng 1996;31(3–4):703–6.

[5] Guthart GS, Salisbury JK. The intuitive telesurgery system: overview and application. In: Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings, Vol. 1, No. April; 2000, p. 618–21.

[6] Al Mashagbeh M, Khamesee MB. Unilateral teleoperated master–slave system for medical applications. IFAC-PapersOnLine 2015;28(3):784–7. http://dx.doi.org/10.1016/j.ifacol.2015.06.178.

[7] Spännar J, Wide P, Sohlberg B. Estimation of a non-measurable variable by using grey box modelling. In: Conference record - IEEE instrumentation and measurement technology conference, Vol. 2. 2001, p. 778–83. http://dx.doi.org/10.1109/IMTC.2001.928184.

[8] Rahideh A, Shaheed MH. Dynamic modelling of a twin rotor MIMO system using grey box approach. In: 5th international symposium on mechatronics and its applications. 2008, p. 1–6. http://dx.doi.org/10.1109/ISMA.2008.4648835.

[9] Rahideh A, Shaheed MH. Grey-box modelling of a non-linear aerodynamic system using genetic algorithms. Proc Inst Mech Eng G 2011;225(8):863–73. http://dx.doi.org/10.1177/0954410011403817.

[10] Premkumar M, Jangir P, Sowmya R, Madurai R, Kumar BS. Enhanced chaotic JAYA algorithm for parameter estimation of photovoltaic cell/ modules. ISA Trans 2021;116:139–66. http://dx.doi.org/10.1016/j.isatra.2021.01.045.

[11] Al-Messabi N, Goh C, Li Y. Grey-box identification for photovoltaic power systems via particle-swarm algorithm. In: 21st international conference on automation & computing (ICAC 2015). 2015, p. 1–7. http://dx.doi.org/10.1109/IConAC.2015.7313980.

[12] Jizhen L, Junlin G, Yang H, Juan W, Hong L. Dynamic modeling of wind turbine generation system based on grey-box identification with genetic algorithm. In: Proceedings of the 36th Chinese control conference; 2017, p. 2038–42.

[13] Gao G, Sun G, Na J, Guo Y, Wu X. Structural parameter identification for 6 DOF industrial robots. Mech Syst Signal Process 2017;113:145–55. http://dx.doi.org/10.1016/J.YMSSP.2017.08.011.

[14] Yan D, Lu Y, Levy D. Parameter identification of robot manipulators: A heuristic particle swarm search approach. PLoS One 2015;10(6). http://dx.doi.org/10.1371/journal.pone.0129157.

[15] Vizer D, Mercère G, Laroche E. Gray-box LPV model identification of a 2-DoF surgical robotic manipulator by using an h∞-norm-based local approach. IFAC-PapersOnLine 2015;48(26):79–84. http://dx.doi.org/10.1016/j.ifacol.2015.11.117.

[16] Mercere G, Laroche E, Prot O. Analytical modelling and grey-box identification of a flexible arm using a linear parameter-varying model. IFAC Proc Vol 2012;45(16):1251–6. http://dx.doi.org/10.3182/20120711-3-BE-2027.00070.

[17] Brunot M, Janot A, Young PC, Carrillo F. An improved instrumental variable method for industrial robot model identification. Control Eng Pract 2018;(74):107–17. http://dx.doi.org/10.1016/j.conengprac.2018.02.010.

[18] Li W, Chang XG, Wahl FM, Farrell J. Tracking control of a manipulator under uncertainty by FUZZY P+ID controller. Fuzzy Sets and Systems 2001;122(1):125–37.

[19] Alavandar S, Nigam MJ. Fuzzy PD+I control of a six DOF robot manipulator. Ind Robot Int J 2008;35(2):125–32. http://dx.doi.org/10.1108/01439910810854610.

[20] Bingül Zafer, Karahan O. A fuzzy logic controller tuned with PSO for 2-DOF robot trajectory control. Expert Syst Appl 2010;38(1):1017–31.

[21] Wang G-G, Gandomi AH, Zhao X, Chu HCE. Hybridizing harmony search algorithm with cuckoo search for global numerical optimization. Soft Comput 2016;20(1):273–85. http://dx.doi.org/10.1007/s00500-014-1502-7.

[22] Mirjalili S, Hashim SZM. A new hybrid PSOGSA algorithm for function optimization. In: Proceedings of ICCIA 2010-2010 international conference on computer and information application. 2010, p. 374–7. http://dx.doi.org/10.1109/ICCIA.2010.6141614.

[23] Mafarja MM, Mirjalili S. Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing 2017;260:302–12. http://dx.doi.org/10.1016/j.neucom.2017.04.053.

[24] Pan J-S, Dao T-K, Chu S-C, Nguyen T-T. A novel hybrid GWO-FPA algorithm for optimization applications. In: Smart innovation, systems and technologies, Vol. 2018. p. 274–81. http://dx.doi.org/10.1007/978-3-319-70730-3_33.

[25] ElGayyar M, Emary E, Sweilam NH, Abdelazeem M. A hybrid grey wolf-bat algorithm for global optimization. In: Advances in intelligent systems and computing. 2018, p. 3–12. http://dx.doi.org/10.1007/978-3-319-74690-6_1.

[26] Singh N, Singh SB. Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. J Appl Math 2017;2017:1–15. http://dx.doi.org/10.1155/2017/2030489.
[27] Singh N, Hachimi H. A new hybrid whale optimizer algorithm with mean strategy of grey wolf optimizer for global optimization. Math Comput Appl 2018;23(1):14. http://dx.doi.org/10.3390/mca23010014.
[28] Gaidhane PJ, Nigam MJ. A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems. J Comput Sci 2018;27:284–302. http://dx.doi.org/10.1016/j.jocs.2018.06.008.
[29] Obadina OO, Thaha MA, Althoefer K, Shaheed MH. Dynamic characterization of a master–slave robotic manipulator using a hybrid grey wolf-whale optimization algorithm. J Vib Control 2021. http://dx.doi.org/10.1177/10775463211003402.
[30] Elyasigomari V, Lee DA, Screen HRC, Shaheed MH. Development of a two-stage gene selection method that incorporates a novel hybrid approach using the cuckoo optimization algorithm and harmony search for cancer classification. J Biomed Inform 2017;67:11–20. http://dx.doi.org/10.1016/j.jbi.2017.01.016.
[31] Alam MS, Tokhi MO. Hybrid fuzzy logic control with genetic optimisation for a single-link flexible manipulator. Eng Appl Artif Intell 2008;21(6):858–73. http://dx.doi.org/10.1016/j.engappai.2007.08.002.
[32] Jesus IS, Barbosa RS. Genetic optimization of fuzzy fractional PD+I controllers. ISA Trans 2015;57:220–30. http://dx.doi.org/10.1016/j.isatra.2015.01.006.
[33] Pham DT, Kalyoncu M. Optimisation of a fuzzy logic controller for a flexible single-link robot arm using the bees algorithm. In: 2009 7th IEEE international conference on industrial informatics. 2009, p. 475–80. http://dx.doi.org/10.1109/INDIN.2009.5195850.
[34] Nikolić M, Šelmić M, Macura D, Ćalić J. Bee colony optimization metaheuristic for fuzzy membership functions tuning. Expert Syst Appl 2020;158:113601. http://dx.doi.org/10.1016/j.eswa.2020.113601.
[35] Pizarro-Lerma AO, García-Hernández R, Santibáñez VA. Fine-tuning of a fuzzy computed-torque control for a 2-DOF robot via genetic algorithms. IFAC-PapersOnLine 2018;51(13). http://dx.doi.org/10.1016/j.ifacol.2018.07.299.
[36] Kumar A, Kumar V. Hybridized ABC-GA optimized fractional order fuzzy pre-compensated FOPID control design for 2-DOF robot manipulator. AEU Int J Electron Commun 2017;79:219–33. http://dx.doi.org/10.1016/j.aeue.2017.06.008.
[37] Martínez-Soto R, Castillo O, Aguilar LT. Type-1 and type-2 fuzzy logic controller design using a hybrid PSO-GA optimization method. Inform Sci 2014;285(1):35–49. http://dx.doi.org/10.1016/j.ins.2014.07.012.
[38] Refoufi S, Benmahammed K. Control of a manipulator robot by neuro-fuzzy subsets form approach control optimized by the genetic algorithms. ISA Trans 2018;77:133–45. http://dx.doi.org/10.1016/j.isatra.2018.03.023.
[39] Shirzadeh M, Amirkhani A, Tork N, Taghavifar H. Trajectory tracking of a quadrotor using a robust adaptive type-2 fuzzy neural controller optimized by cuckoo algorithm. ISA Trans 2021;114:171–90. http://dx.doi.org/10.1016/j.isatra.2020.12.047.
[40] Teng Z, Lv J, Guo L. An improved hybrid grey wolf optimization algorithm. Soft Comput 2019;23:6617–31. http://dx.doi.org/10.1007/s00500-018-3310-y.
[41] el Aziz MA, Ewees AA, Hassanien AE. Whale optimization algorithm and moth–flame optimization for multilevel thresholding image segmentation. Expert Syst Appl 2017;83:242–56. http://dx.doi.org/10.1016/j.eswa.2017.04.023.
[42] Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Softw 2014;69:46–61. http://dx.doi.org/10.1016/j.advengsoft.2013.12.007.
[43] Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw 2016;95:51–67. http://dx.doi.org/10.1016/j.advengsoft.2016.01.008.
[44] Bernth J. Design, construction and control of a master-slave robotic manipulator (BEng Thesis), Queen Mary University of London; 2014.
[45] Dao T, Chen C. Tuning fuzzy-logic controllers. In: Dadios E, editor. Fuzzy logic - controls, concepts, theories and applications. InTech; 2012, p. 428. http://dx.doi.org/10.5772/36946.
[46] Pham DT, Darwish AH, Eldukhri EE, Otri S. Using the bees algorithm to tune a fuzzy logic controller for a robot gymnast. In: 3rd virtual int. conf. on innovative production machines and systems (IPROMS 2007). 2007, p. 546–51.