Arab J Sci Eng (2017) 42:739–750
DOI 10.1007/s13369-016-2306-0

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

Optimum Design of Fractional-Order Hybrid Fuzzy Logic Controller for a Robotic Manipulator

Richa Sharma[1,2] · **Prerna Gaur**[2] · **A. P. Mittal**[2]

Received: 25 January 2016 / Accepted: 23 August 2016 / Published online: 15 October 2016

Abstract The robotic manipulators are complex and coupled nonlinear systems. Therefore, the designing of an effective controller for these systems is quite complicated. The main hurdle in operating these systems is the interlinkage between the links, and this can be removed by using any decoupling method. The decoupling between the links is not a good idea from the viewpoint of practical applications. In this paper, a fractional-order hybrid fuzzy logic controller (FOHFLC) scheme is developed for a two-degree-of-freedom rigid planar robotic manipulator with payload (2-DOF RPRMWP) plant for the trajectory tracking. The cuckoo search algorithm (CSA) is utilized for finding the optimal parameters of the proposed approach. For witnessing the effectiveness, the performance of proposed FOHFLC scheme is compared with integer-order hybrid FLC (IOHFLC) approach and conventional PID controller. The robustness testing is investigated for parameter variations and disturbance rejection for the proposed controller schemes.

Keywords Robotic manipulator · Fuzzy logic controller · Fractional-order operators · Cuckoo search algorithm · Trajectory tracking

✉ Richa Sharma
richasharma_7@yahoo.co.in

Prerna Gaur
prernagaur@yahoo.com

A. P. Mittal
mittalap@gmail.com

[1] Electrical and Instrumentation Engineering Department, Thapar University, Patiala 147004, India

[2] Instrumentation and Control Engineering Division, Netaji Subhas Institute of Technology, Dwarka, New Delhi 110078, India

1 Introduction

In the recent times, various industries such as process industries, nuclear plants, and medical fields have significantly been captured by the robotic manipulators. These systems have features such as fast and accurate positioning which are possible only when the end-effectors are effectively controlled. The manipulators are multi-input–multi-output (MIMO), nonlinear, inter-acting, and highly uncertain systems. Therefore, designing an effective control approach for the manipulators is not a cakewalk for the control experts. The major problem with these systems is the coupling between the links as the position of one link cannot remain undisturbed during the movement of other link. It is an essential requirement to use effective decoupling techniques to get rid of these undesirable coupling effects but these techniques are unacceptable for the real-time control applications. The essential demand for designing such decoupling approaches is to find an accurate dynamic model of the manipulator systems which seems impossible for such complex systems. Hence a model-free decoupling approach is suitable for obtaining the exact model of the system to be controlled [1].

Over the time, the FLC has been dominated on the conventional methods due to its unparallel properties such as association of human expertise and being a model-free and flexible approach. It can provide effective control to the systems having uncertainties and nonlinearities [2–4]. Various authors have been working on FLC-based control schemes for different MIMO, nonlinear, and uncertain plants. Xu and Shin [5] presented an adaptive multi-level FLC for multi-input–single-output systems wherein the system model is modeled with fuzzy domain. Huo et al. [6] investigated an adaptive FLC scheme for the MIMO nonlinear systems with actuator faults and unmeasured states. Tong et al. [7] presented a partial tracking error-based FLC output feedback

dynamic surface control scheme for the uncertain MIMO plant. Yadav and Gaur [8] proposed an improved self-tuning FLC-based control scheme for the speed control of nonlinear hybrid electric vehicle. Su et al. [9] investigated an indirect adaptive FLC approach for the MIMO nonlinear nonaffine plant having actuator saturator and unknown dynamics for the green house climate control task.

For the past few decades, the FLC schemes have been utilized in different structures and combinations for designing efficient controllers for the robotic plants by several researchers. Song et al. [10] presented a hybrid controller scheme for the robotic manipulator having the classical computed torque control approach in combination with model-free FLC approach. Meza et al. [11] investigated a fuzzy self-tuning PID control approach for robotic manipulator using experimental results. Chu et al. [12] proposed an adaptive disturbance observer-based output feedback controller in which the parameters are tuned online using FLC for the electrically driven free-space manipulator. Chiu [13] presented a mixed feedforward- and feedback-based adaptive FLC approach for the two nonlinear MIMO plants namely inverted pendulum and two-link robotic system. Lian and Lin [1] investigated a mixed FLC approach having two FLCs for each link. The first FLC is the classical one and other acts as the coupling FLC. The control approach was designed to remove the coupling effects between the links. Baghli et al. [14] presented a MIMO FLC approach for the two-degree-of-freedom robotic arm.

Recently, various researchers have been working toward the use of fractional-order mathematics in the controller design. The fractional-order mathematics have successfully been integrated with FLC. Das et al. [15,16] proposed a fractional-order FLC (FOFLC) scheme for the delayed open-loop unstable and nonlinear systems, and claimed it to be better than the other controllers such as fuzzy PID, fractional-order PID (FOPID), and classical PID controllers. They also presented a comparison among different structures of FOFLC scheme for oscillatory fractional-order plant with dead time. Sharma et al. [17] presented FOFLC approach for the robotic manipulator and found it effective and robust than other potential controllers such as fuzzy PID, FOPID, and classical PID controllers. Hajiloo [18] presented the design of a FOPID controller whose parameters are tuned using FLC approach. The multi-objective optimization technique is used to design the required FLC for the presented scheme. Therefore, it is concluded that the integration of fractional-order mathematics with FLC may be worth to design an effective controller.

In this paper, the FOHFLC approach is developed for a 2-DOF RPRMWP for trajectory tracking problem. The rationale behind this work is to provide an effective control approach for eliminating the coupling effects between the links and also, to provide flexibility to the user in selection of controller parameters. The coupling FOFLC (C-FOFLC) is used along with the FOFLC to overcome the coupling between the links. The fractional-order operators are used for providing the extra tunable parameters in the controller design. The parameters of the proposed control approach is obtained with an effective optimization tool named as CSA. A comparative study of the proposed FOHFLC approach is carried out with integer-order design and conventional PID controller for trajectory tracking problem. The robustness of the proposed FOHFLC scheme is also investigated for the parameter variations and external disturbance rejection.

2 Mathematical Model of Robot Manipulator

The mathematical model of SCARA-type 2-DOF RPRMWP has been described by Lin [19] and is expressed in (1). The robotic manipulator with two rigid links having a payload at the end of the second link is shown in Fig. 1. Also, Table 1 lists the parameters of 2-DOF RPRMWP plant used for simulation.

$$\begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} + \begin{bmatrix} V_{11} \\ V_{22} \end{bmatrix} + \begin{bmatrix} g_{11} \\ g_{22} \end{bmatrix} = \begin{bmatrix} \tau_{FOHFLC_1} \\ \tau_{FOHFLC_2} \end{bmatrix} \tag{1}$$

where

$$\begin{aligned}
z_{11} &= I_{11} + I_{22} + m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 \\
&\quad + 2m_2 l_1 l_{c2} \cos\theta_2 + m_o l_1^2 + m_o l_2^2 + 2m_o l_1 l_2 \cos\theta_2 \\
z_{12} &= I_{22} + m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(\theta_2) + m_o l_2^2 \\
&\quad + m_o l_1 l_2 \cos(\theta_2) \\
z_{21} &= z_{12} \\
z_{22} &= I_{22} + m_2 l_{c2}^2 + m_o l_2^2 \\
B_{11} &= -m_2 l_1 l_{c2} \left(2\dot{\theta}_1 + \dot{\theta}_2\right) \dot{\theta}_2 \sin\theta_2 \\
&\quad - m_o l_1 l_2 \left(2\dot{\theta}_1 + \dot{\theta}_2\right) \dot{\theta}_2 \sin\theta_2 \\
B_{21} &= m_2 l_1 \dot{\theta}_1^2 l_{c2} \sin\theta_2 + m_o l_1 \dot{\theta}_1^2 l_2 \sin\theta_2 \\
V_{11} &= b_{11} \dot{\theta}_1 \\
V_{22} &= b_{21} \dot{\theta}_2 \\
g_{11} &= m_1 l_{c1} g \cos(\theta_1) + m_2 g \left(l_{c2} \cos(\theta_1 + \theta_2) + l_1 \cos(\theta_1)\right) \\
&\quad + m_o g \left(l_2 \cos(\theta_1 + \theta_2) + l_1 \cos(\theta_1)\right) \\
g_{22} &= m_2 l_{c2} g \cos(\theta_1 + \theta_2) + m_o l_2 g \cos(\theta_1 + \theta_2)
\end{aligned}$$

where θ_1 and θ_2 represent the positions of links; τ_{FOHFLC_1} and τ_{FOHFLC_2} are the torques; m_1 and m_2 are masses; l_1 and l_2 represent the lengths; I_{11} and I_{22} express lengthwise centroid inertia; l_{c1} and l_{c2} denotes the distances from the joint to links

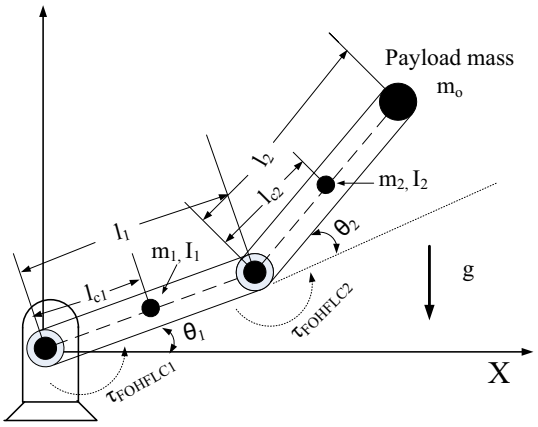

Fig. 1 A 2-DOF RPRMWP with payload at tip

Table 1 Parameters for a 2-DOF RPRMWP plant

Parameters	Link1	Link2
Mass	0.392924 kg	0.094403 kg
Acceleration due to gravity (g)	9.81 m/s^2	9.81 m/s^2
Length	0.2032 m	0.1524 m
Distance from the joint of link to its center of gravity	0.104648 m	0.081788 m
Lengthwise centroid inertia of link	0.0011411 kg m^2	0.0020247 kg m^2
Friction at joints	0.141231 N-m/radian/s	0.3530776 N-m/radian/s

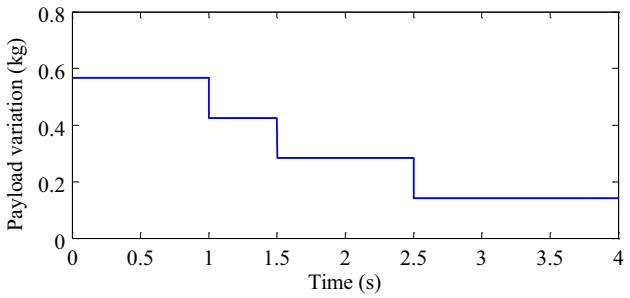

Fig. 2 Payload variations

to their center of gravity; b_{11} and b_{21} are the coefficients of friction at joints. Also, m_o represents the mass of a payload and its value is varied from 0.56699 to 0.14174 kg in the entire 4s as shown in Fig. 2.

3 Controller Design

In this section, the design and implementation of the proposed FOHFLC approach is presented. After this, the design method of fractional-order operators is explained, and finally, the design approach of FLC employed in the proposed controller scheme is given.

3.1 Basic Design and Implementation of FOHFLC Approach

The basic scheme of the proposed FOHFLC approach has two FLCs as shown in Fig. 3. The FLCs employed are fractional order in nature which means the fractional-order differentiators and integrators are used instead of the traditional ones. There are two FOFLCs used in the controller design: One is the FOFLC which is a fractional-order version of fuzzy PID-type controller and another is the coupling FOFLC (C-FOFLC) which is a fractional-order version of fuzzy PD-type controller design.

In the proposed controller scheme, a coupling FOFLC (C-FOFLC) is used in the design to reduce the effect of one link over the other [1]. The basic scheme of the proposed FOHFLC controller scheme for a 2-DOF RPRMWP is given in Fig. 3. The details of implementation of proposed scheme are shown in Fig. 4. There are overall four FOFLCs employed in the controller design in Fig. 4 in which there is one FOFLC and one C-FOFLC employed for each link of the robotic manipulator. The terms K_{p_1}, K_{r_1}, K_{d_1}, and K_{u_1} are the scaling gains for FOFLC1 and the terms K_{p_2}, K_{r_2}, and K_{u_2} are the scaling gains for C-FOFLC1. Similarly, the terms K_{p_3}, K_{r_3}, and K_{u_3} are the scaling gains for C-FOFLC2, and the terms K_{p_4}, K_{r_4}, K_{d_2}, and K_{u_4} are the scaling gains for FOFLC2. Also, γ_i and δ_i are the values of fractional-order differentiator and integrators, respectively.

The C-FOFLCs employed in the design are fractional version of conventional fuzzy PD-type controllers and the outputs from C-FOFLC1 and C-FOFLC2 are $\tau_{C-\mathrm{FOFLC}_1}$ and $\tau_{C-\mathrm{FOFLC}_2}$ respectively. The relation between input and output for a conventional fuzzy PD design is expressed as follows:

$$\tau_{\mathrm{FLC}_i}(k) = K_{p_i} e_i(k) + K_{r_i} \frac{\left(1 - z^{-1}\right)}{T} e_i(k) \tag{2}$$

Therefore, the control law for C-FOFLC design can be written as follows:

$$\tau_{C-\mathrm{FOFLC}_i}(k) = K_{p_i} e_i(k) + K_{r_i} [\text{Differentiator}]^{\gamma_i} e_i(k) \tag{3}$$

The FOFLCs employed in the design of proposed scheme are of fuzzy PID type, for which a single rule is able to provide aggregate control action [9]. The following equation represents the control law or input–output relation for a con-

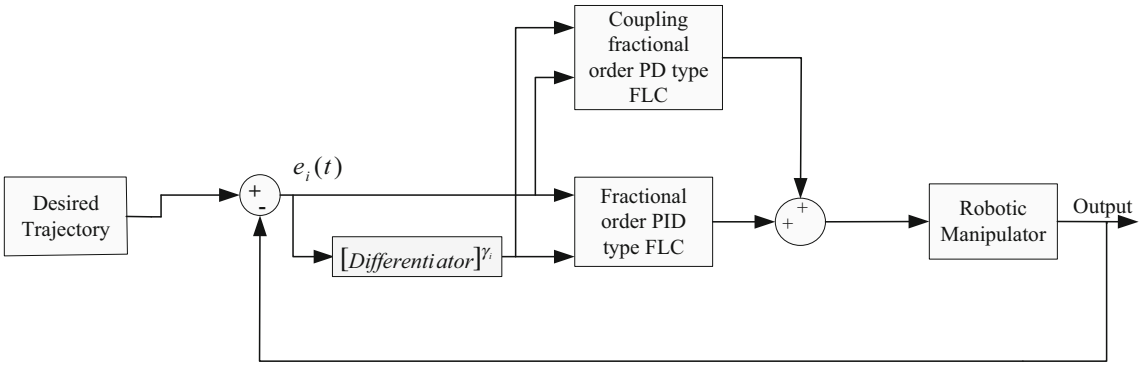

Fig. 3 Basic design scheme of FOHFLC approach

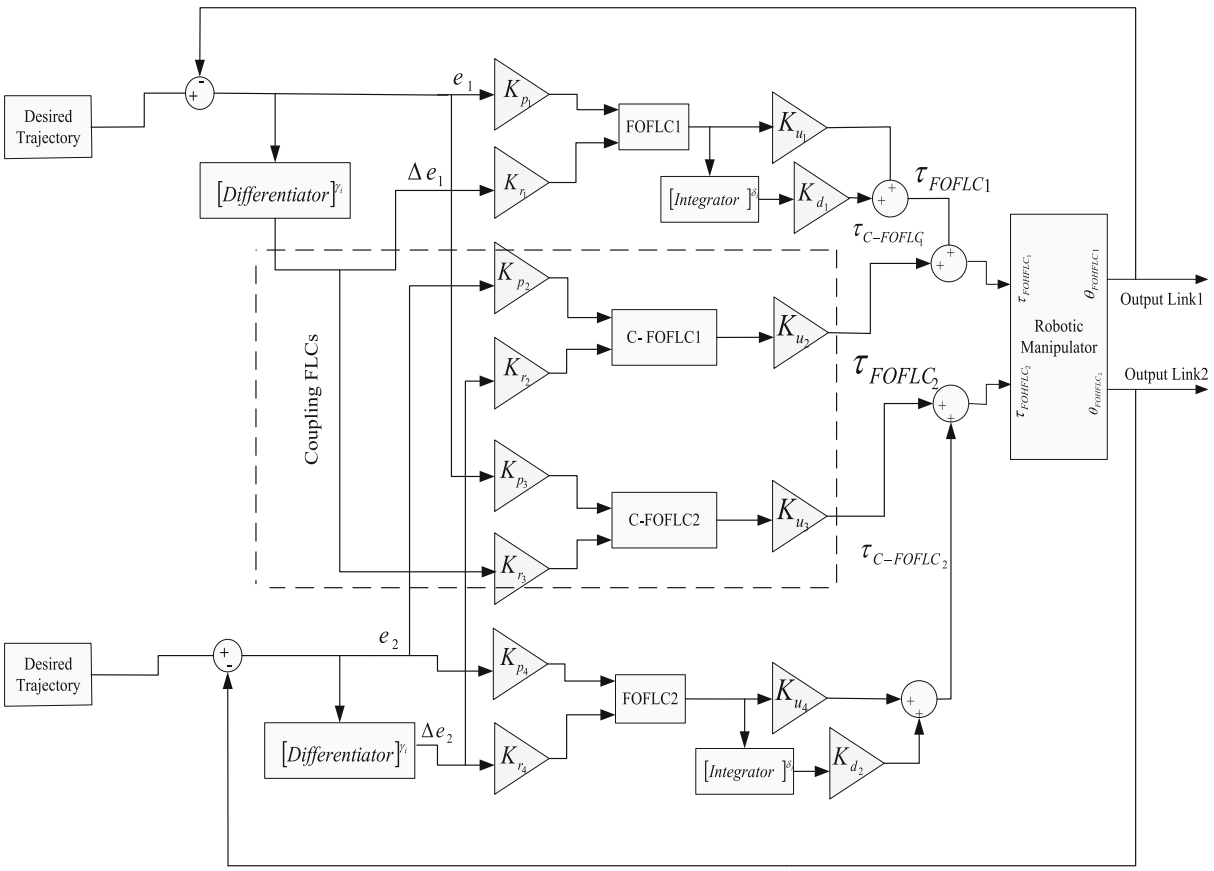

Fig. 4 Block diagram of implementation of FOHFLC approach for a 2-DOF RPRMWP plant

ventional fuzzy PID controller approach:

$$\tau_{\mathrm{FOFLC}_i}(k) = \frac{T}{\left(1-z^{-1}\right)} K_{d_1} \Delta\tau_{\mathrm{CFLC}_1}(k) + K_{u_1}\tau_{\mathrm{CFLC}_1}(k) = K_{p_1} e_1(k) + K_{r_1}\frac{\left(1-z^{-1}\right)}{T} e_1(k) \quad (4)$$

where $\frac{(1-z^{-1})}{T} e_1(k)$ is rate of change of error and $\Delta\tau_{\mathrm{CFLC}_1}(k) = \frac{(1-z^{-1})}{T}\tau_{\mathrm{CFLC}_1}(k)$ is the incremental controller output.

Furthermore, for the design of fractional-order controller, i.e., FOFLC, the following input output relations are used:

$$\tau_{\mathrm{FOFLC}_1}(k) = \left[\mathrm{Integrator}\right]^{\delta_i} K_{d_1}\Delta\tau_{\mathrm{FOFLC}_1}(k) + K_{u_1}\tau_{\mathrm{FOFLC}_1}(k) = K_{p_1} e_1(k) + K_{r_1}\left[\mathrm{differentiator}\right]^{\gamma_i} e_1(k) \quad (5)$$

where the order of integrator δ_i and differentiator γ_i represent the additional parameters in the design and are tuned properly to obtain the desired response.

Therefore, the overall controller output of the proposed scheme is given as follows:

$$\tau_{\mathrm{FOHFLC}_i}(k) = \tau_{C-\mathrm{FOFLC}_i}(k) + \tau_{\mathrm{FOFLC}_i}(k) \tag{6}$$

where $i = 1, 2$ represents Link1 and Link2, respectively.

3.2 Fractional-Order Implementation

To get an idea about the implementation of proposed control scheme, the understanding of fractional-order integrators and differentiators are necessary, and the method used for implementation of these operators is named as Oustaloup's approximation. It provides the recursive distribution of zeros and poles in the form of an approximating transfer function which is equivalent to the fractional operator s^c where c is a fractional-order value [20].

$$s^c = k_{fu} \prod_{k_u=-N_{fu}}^{N_{fu}} \frac{s + w_{ff_{zu}}}{s + w_{ff_{pu}}} \tag{7}$$

where k_{fu} is gain of the filter, $w_{ff_{zu}}$ presents zeros, and $w_{ff_{pu}}$ presents poles of the filter, and these parameters can obtained as given below [21,22]:

$$w_{ff_{pu}} = w_{b_{ff}} \left(\frac{w_{h_{ff}}}{w_{b_{ff}}} \right)^{\frac{k_{fu}+N_{fu}+\frac{1}{2}+\frac{c}{2}}{2N_{fu}+1}} \tag{8}$$

$$w_{ff_{zu}} = w_{b_{ff}} \left(\frac{w_{h_{ff}}}{w_{b_{ff}}} \right)^{\frac{k_{fu}+N_{fu}+\frac{1}{2}-\frac{c}{2}}{2N_{fu}+1}} \tag{9}$$

$$k_{fu} = w_{hff}^{c} \tag{10}$$

Thus, c represents the order of fractional-type differentiator; $2N_{fu} + 1$ is the order of approximation; $\{w_{h_{ff}}, w_{b_{ff}}\}$ provides the frequency range [21]. This technique has a conflict issue between the value of N_{fu} and its performance [21,22]. However, this method is chosen over other competitive methods as it has the ease and possibilities of implementing in real hardware design in terms of higher-order infinite impulse response digital or analog filters [21].

3.3 Fuzzy Logic Controller

The FLCs used in the proposed controller schemes are four in numbers: two are FOFLC and other two are C-FOFLC. A general FLC has four core components namely fuzzification, inference engine, rule base, and defuzzification.

1. Fuzzification: It produces the linguistic equivalents of the crisp values. The formulation of membership functions (MFs) is the prime step and is generally based on human expertise as well as experience. In the presented work, two FOFLCs are used for each link. For all FOFLCs, the error and fractional rate of change of error are the input variables as shown in the controller scheme in Fig. 4. The input and output variables for the FOFLCs are represented by seven triangular MFs such as positive small (PS), positive medium (PM), positive large (PL), zero (ZO), negative large (NL), negative medium (NM), and negative small (NS). For the C-FOFLCs, only three triangular MFs such as negative large (NL), zero (ZE), and positive large (PL) are used for both input and output operators [1]. The 50 % overlapping is adopted for all the MFs of all FOFLCs. The operating range of all MFs, both for inputs and for outputs, are chosen as $[-1, 1]$ as shown in Fig. 5. The surface graphs of the input and output variables for both FLCs are shown in Fig. 6.
2. Rule base: The rule base is the key part in the design of a FLC and is developed with the human expertise and trends in the system output characteristics. For the FOFLCs, the rule base is in 7×7 matrix format which means overall 49 rules as given in Table 2 are formulated between two input variables namely error and fractional rate of change in the error with seven MFs each.

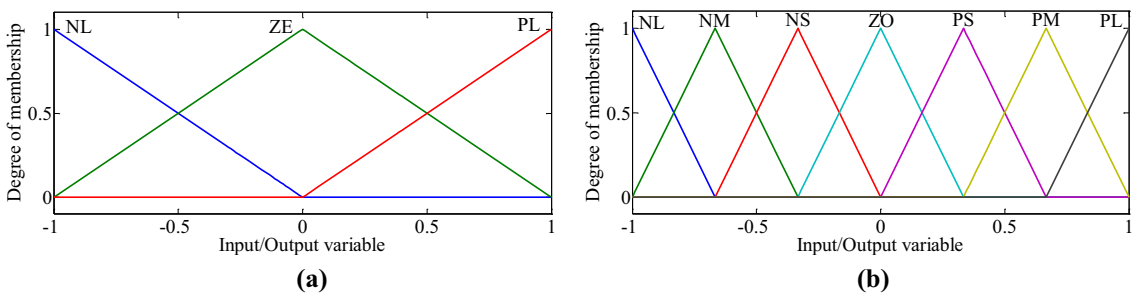

Fig. 5 MFs for input/output variables for (**a**) C-FOFLC (**b**) FOFLC

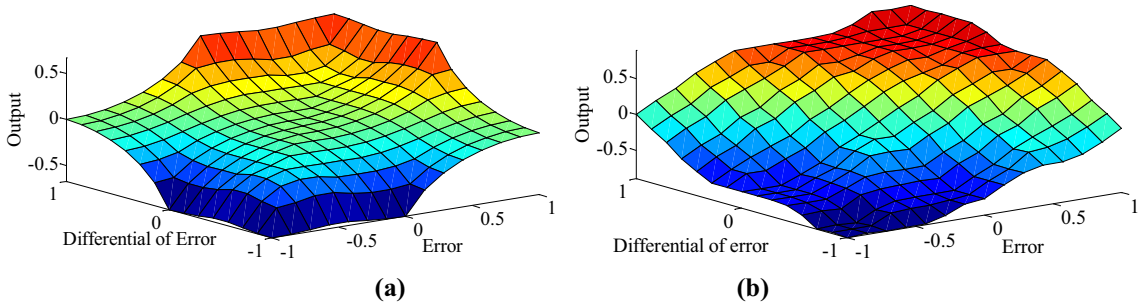

Fig. 6 Surface diagram for input and output variables for (**a**) C-FFOLC (**b**) FOFLC

Table 2 Rule base for FOFLC

Error							
	NL	NM	NS	ZO	PS	PM	PL
Fractional rate of change of error							
NL	NL	NL	NL	NL	NM	NS	ZO
NM	NL	NL	NL	NM	NS	ZO	PS
NS	NL	NL	NM	NS	ZO	PS	PM
ZO	NL	NM	NS	ZO	PS	PM	PL
PS	NM	NS	ZO	PS	PM	PL	PL
PM	NS	ZO	PS	PM	PL	PL	PL
PL	ZO	PS	PM	PL	PL	PL	PL

Table 3 Rule base for C-FOFLC

Error			
	NL	ZE	PL
Fractional rate of change of error			
NL	NL	NL	ZE
ZE	NL	ZE	PL
PL	ZE	PL	PL

The rule base for the C-FOFLCs is given in Table 3, which is a 3 × 3 matrix and overall nine rules are generated between two input variables and single output variable with 3 MFs [1]. The rules are expressed with IF–THEN statements.

3. Fuzzy inference and defuzzification: The inference engine is employed to get an appropriate control action on the basis of apportion of each rule. The Mamdani min–max fuzzy inference approach is used here. The output from the FLCs need to be changed into crisp values so that it can actually given to the plant and this conversion action is named as defuzzification. For the present control approach, the center of gravity method is employed to defuzzify the fuzzy output.

4 Control Objectives and Tuning with Cuckoo Search Algorithm

4.1 Control Objective Function

The control objective functions COF chosen for the proposed work is the weighted sum of integral of absolute error (IAE) of Link1 and Link2 for the purpose of minimization and are expressed by (11) and (12), respectively. The aggregate objective function COF is designed as the weighted sum of IAE of both the links. The reason behind the selection of these objective functions is to reduce the error between actual and desired trajectories.

$$\mathrm{Cof}_1 = \int |e_1(t)|\mathrm{d}t \tag{11}$$

$$\mathrm{Cof}_2 = \int |e_2(t)|\mathrm{d}t \tag{12}$$

$$\mathrm{COF} = w_1\mathrm{cof}_1 + w_2\mathrm{cof}_2 \tag{13}$$

where w_1 and w_2 are the weights assigned to cof_1 and cof_2, respectively.

4.2 Cuckoo Search Algorithm

CSA is a new meta-heuristic optimization technique based on the parasitic breeding characteristics of the cuckoo birds. The implementation of CSA was invented by Yang and Deb [23]. The basic plot of this algorithm is based on the finding of the other species nest by the cuckoo's search, and the other thing is that the cuckoo birds are in search of a nest wherein other host bird has just laid its eggs [24]. In order to understand the implementation of CSA, it is necessary to get an idea about cuckoo birds characteristics and the reason behind the formulation of this technique.

4.2.1 Characteristics of Cuckoo Birds

The cuckoo birds have abilities which make them different and cunning from other birds. These birds can mimic the call

of host bird's chick. These birds have the ability to copy the color and pattern of eggs of other birds so that they may easily survive in other birds' nests. The cuckoo's chicks show the wittiness and throw the eggs of the host birds out from occupied nest, and then they can easily survive there by having all the benefits and care from the host parent bird. The cuckoo chicks have the sense to frequently call the parent host birds for grabbing their attention and to take all the benefits of the food. However, some parent host birds may discover the foreign eggs and they throw them out from their own habitat or they may themselves vacate the nest [23,25].

4.2.2 Literature Outcomes for CSA

Despite its infant stage, CSA has been emerged as a popular optimization technique due to its incomparable features. The initialization parameters for this algorithm are lesser than the other heuristic and nature-inspired technique such as GA and PSO. The convergence rate is insensitive toward its parameters which makes it available to many optimization problems [24,26]. This algorithm has large steps for searching the space therefore make it more useful than other nature-inspired techniques. It also has the significant elitism property and allow the best solution in the next generation [26].

The CSA performs better than both GA and PSO for searching the best solutions or values for the location and size of the distributed generation system [27]. The CSA may be used for finding the optimal cutting parameters of any milling process and it performs superior to other popular optimization methods such as hybrid PSO, feasible direction technique, hybrid immune algorithm, standard GA and ant colony optimization etc. [28]. This algorithm has also showed its potential toward different mechanical problems such as spring design, speed reducer, gear train and welded beam system etc. [24,26] and may be applied to other similar kind of mechanical structures. Bulatovic et al. [29] also used this optimization method for the six-bar double dwell linkage.

4.2.3 Lévy Flight

The birds or insects move in the search space in order to search food using Lévy flight. Lévy flight are the random steps based on the current location and the transition probability for the next location and are presented as follows [23]:

$$x_{f_{\text{csa}}}(t_f + 1) = x_{f_{\text{csa}}}(t_f) + \beta_{f_{\text{csa}}} \oplus L\acute{e}vy(\lambda_{f_{\text{csa}}}) \tag{14}$$

where $\beta_{f_{\text{csa}}}$ provides the step size of Lévy [17]. The general flowchart for CSA is shown in Fig. 4 [23,26].

4.2.4 Significant Rules

This algorithm is based on hunting for the best nest with optimal solutions. The following three rules are the basic building blocks of CSA algorithm [26]:

1. Every cuckoo bird lays one egg and choose a nest randomly to place it.
2. The best nests with potential solutions are carried over to the next generation.
3. The host nests are restricted and the probability of discovering the foreigner eggs '$P_{f_{\text{csa}}}$' is in [0, 1] range.

4.3 Implementation Procedure for Finding the Controller Parameters

From Fig. 4, it is clearly visible that there are overall eighteen scaling gains of the four FLCs employed in the proposed controller scheme that need to be tuned for obtaining an effective response. The step-by-step implementation of CSA for finding the optimal solutions are presented as follows and are also given in Fig. 7:

Step 1: Formulate the objective function as given in (13). Randomly initialize a population of f host nests z_{f_i} $(i = 1, 2, \ldots\ldots, 25)$. Set the termination criteria as maximum number of iterations = 100 and also set $P_{f_{\text{csa}}} = 0.25$.
Step 2: Obtain a cuckoo cs randomly with the Lévy flight as given in (14) and calculate its fitness f_{cs} in reference to the objective function chosen for the presented problem.
Step 3: Randomly choose a nest cus from the generated population f and compute its fitness as f_{cus}.
Step 4: Compare these two fitness values f_{cs} and f_{cus}, and if $f_{\text{cs}} > f_{\text{cus}}$, then the host nest cus is replaced by the new nest cs that is obtained using Lévy flight.
Step 5: A fraction '$P_{f_{\text{csa}}}$' of the align nests are abandoned and new nests z_n are generated randomly at new locations using the Lévy flight.
Step 6: Compute the fitness of all the new generated nests.
Step 7: Update the best nest for the present iteration.
Step 8: The best nest z_{f_b} obtained in the present iteration is replaced by the best nest of the iteration z_{f_q}, if the fitness value $f_{z_{f_b}}$ of the best nest z_{f_b} is greater than the fitness $f_{z_{f_q}}$ of the best nest z_q of the iteration.
Step 9: Repeat steps 2–8 till the termination criteria is reached. The best nest obtained at the final iteration provides the optimal solution for the problem taken.

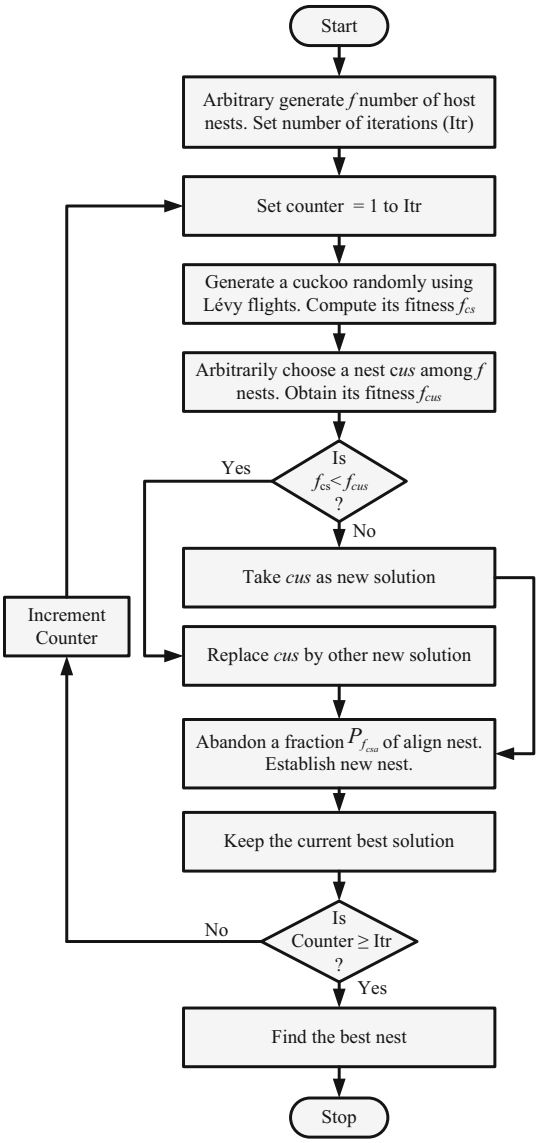

Fig. 7 Flowchart for the CSA implementation

5 Simulation Results

In this section, the results obtained for trajectory tracking, parameter variations, and disturbance rejection for the proposed FOHFLC, IOHFLC, and conventional PID controller schemes are presented. The numerical simulations are performed with the MATLAB version R2009b using algebraic solver Runge–Kutta 4 method. The proposed controller scheme is implemented as shown in Fig. 4 and its integer-order design IOHFLC is implemented by keeping the orders of integrators and differentiators to unity. For the implementation of IOHFLC scheme, the orders δ_i and γ_i of the integrator and differentiators in the control law in (5) are kept as unity, respectively. It means that the orders of integrator and differentiator in IOHFLC approach are of fixed ones, whereas the orders in FOHFLC scheme are tunable parameters and provide flexibility to user in selection of controller parameters. The scaling gains of all FLCs for the both controllers schemes are presented in Table 4. For the presented work, the trajectory used is of cubic polynomial nature and is given as follows [30]:

$$\theta_{\mathrm{Ref}_{\mathrm{flc}_i}}(t_z) = z_{\mathrm{flc}_0} + z_{\mathrm{flc}_1}(t_z) + z_{\mathrm{flc}_2}(t_z)^2 + z_{\mathrm{flc}_3}(t_z)^3 \quad (15)$$

the constraints are

$$\dot{\theta}_{\mathrm{Ref}_{\mathrm{flc}_i}}(t_z) = z_{\mathrm{flc}_1} + 2z_{\mathrm{flc}_2}(t_z) + 3z_{\mathrm{flc}_3}(t_z)^2 \quad (16)$$

$$\ddot{\theta}_{\mathrm{Ref}_{\mathrm{flc}_i}}(t_z) = 2z_{\mathrm{flc}_2} + 6z_{\mathrm{flc}_3}(t_z) \quad (17)$$

where $\theta_{\mathrm{Ref}_{\mathrm{flc}_i}}$ represent the reference positions; $i = 1, 2$ for Link1 and Link2, respectively; $\theta_{\mathrm{Ref}_{\mathrm{flc}_i}} = 1$ radian and $\theta_{\mathrm{Ref}_{\mathrm{flc}_2}} = 2$ radian for $t_z = 2s$; $\theta_{\mathrm{Ref}_{\mathrm{flc}_1}} = 0.5$ radian and $\theta_{\mathrm{Ref}_{\mathrm{flc}_2}} = 4$ radian for $t_z = 4s$; $\theta_{R_{\mathrm{flc}_i}} = 0$ radian$/s$ for both $t_z = 2s$ and $t_z = 4s$.

The graphs for the trajectory tracking, control output, positional error, and path tracked by the end-effector are shown

Table 4 FLCs scaling gains and IAE for FOHFLC and IOHFLC approaches

Parameters	Link1		Parameters	Link2	
	FOHFLC	IOHFLC		FOHFLC	IOHFLC
K_{p_1}	116.5261	143.9640	K_{p_3}	165.0072	272.7322
K_{r_1}	0.7034	0.000100	K_{r_3}	0.1145	0.0007660
K_{u_1}	300.0000	160.02250	K_{u_3}	0.3300	0.0044820
K_{p_2}	51.8432	102.303900	K_{p_4}	72.1687	274.8904
K_{r_2}	0.1275	0.001919	K_{r_4}	0.0768	0.0001
K_{u_2}	0.3815	0.004482	K_{u_4}	0.4875	0.005235
K_{d_1}	0.9685	0.006343	K_{d_2}	6.1595	0.8856
δ_1	0.4240	1.000000	δ_2	0.8600	1.0000
γ_1	0.0001	1.000000	γ_1	0.9999	1.00000
IAE	7.022×10^{-5}	1.08×10^{-4}	IAE	2.137×10^{-3}	4.467×10^{-3}

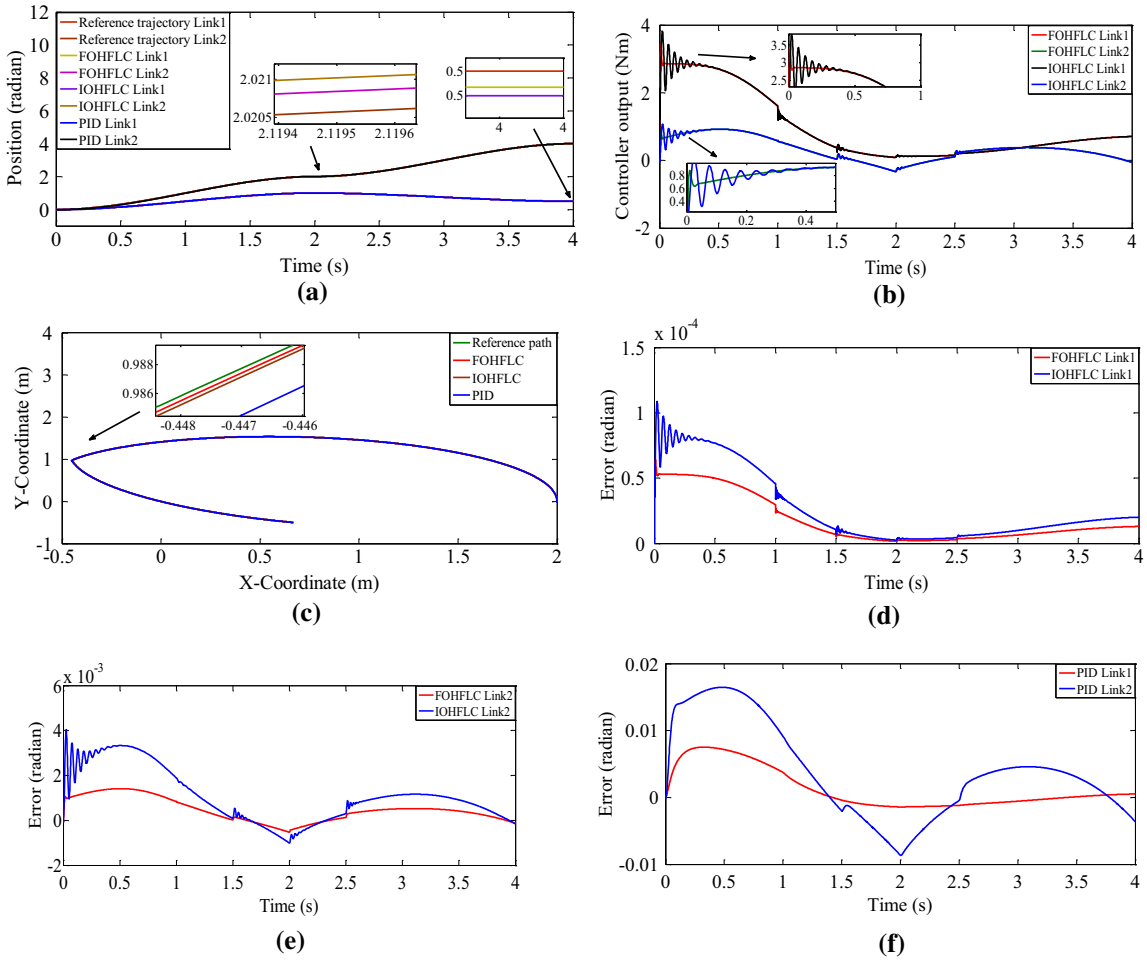

Fig. 8 **a** Trajectory tracking performances. **b** Controller output signal. **c** Path tracked by end-effector for FOHFLC, IOHFLC and PID controllers. **d** Position errors for Link1. **e** Position errors for Link2 for FOHFLC and IOHFLC controllers. **f** Position errors for Link1 and Link2 for PID controller

in Fig. 8. The IAE values for PID controllers for Link1 and Link2 are 8.599×10^{-3} and 2.472×10^{-2} respectively. The IAE values for the Link1 and Link2 of the FOHFLC approach are 7.022×10^{-5} and 2.137×10^{-3} respectively whereas the IAE values for IOHFLC approach for Link1 and Link2 are 1.08×10^{-4} and 4.467×10^{-3} respectively as in Table 4. Therefore, it can be clearly visible that the IAE values for the FOHFLC approach is smaller as compared to those of IOHFLC and conventional PID controller approaches. From Fig. 8d, e, it is clearly visible that positional errors for Link1 and Link2 of the FOHFLC scheme are lesser as compared to those of IOHFLC and conventional PID controller schemes. Therefore, it can be concluded that the performance of FOHFLC scheme is more effective than its integer-order design as well as conventional PID controller for the trajectory tracking task.

5.1 Parameter Variations

In this section, the study of parameter variations on the performance of proposed controller approaches is presented. The parameter variations include change in two significant parameters namely mass and coefficient of friction and are listed in Table 5. The IAE values for FOHFLC, IOHFLC, and conventional PID controller schemes for both cases of parameter variations, i.e., change in link mass and coefficient of friction are listed in Table 5.

From Table 5, it can be clearly indicated that the IAE values for proposed FOHFLC scheme are smaller than those of its integer-order design, IOHFLC, and conventional PID controllers. Therefore, it is inferred that the proposed FOHFLC scheme is superior to its integer-order design for the parameter variations.

Table 5 IAE values for Link1 and Link2 for ±5 % change in parameter values

Parameter variation (5 %)	FOHFLC		IOHFLC		PID	
	Link1 $\times 10^{-5}$	Link2 $\times 10^{-3}$	Link1 $\times 10^{-4}$	Link2 $\times 10^{-3}$	Link1 $\times 10^{-3}$	Link2 $\times 10^{-2}$
Decrease						
Parameter 1: m_1	6.907	2.137	1.062	4.467	8.556	3.214
Parameter 2: m_2	6.978	2.136	1.073	4.466	8.556	2.465
Parameter 3: m_1, m_2	6.863	2.136	1.056	4.466	8.513	2.465
Parameter 4: b_{11}	7.016	2.137	1.079	4.467	8.580	2.472
Parameter 5: b_{21}	7.022	2.055	1.080	4.281	8.598	2.403
Parameter 6: b_{11}, b_{21}	7.016	2.055	1.079	4.281	8.579	2.403
Increase						
Parameter 1: m_1	7.137	2.137	1.098	4.467	8.643	2.472
Parameter 2: m_2	7.066	2.138	1.087	4.469	8.643	2.479
Parameter 3: m_1, m_2	7.182	2.138	1.105	4.469	8.686	2.479
Parameter 4: b_{11}	7.029	2.137	1.081	4.467	8.618	2.472
Parameter 5: b_{21}	7.022	2.221	1.080	4.657	8.600	2.542
Parameter 6: b_{11}, b_{21}	7.029	2.221	1.081	4.657	8.619	2.542

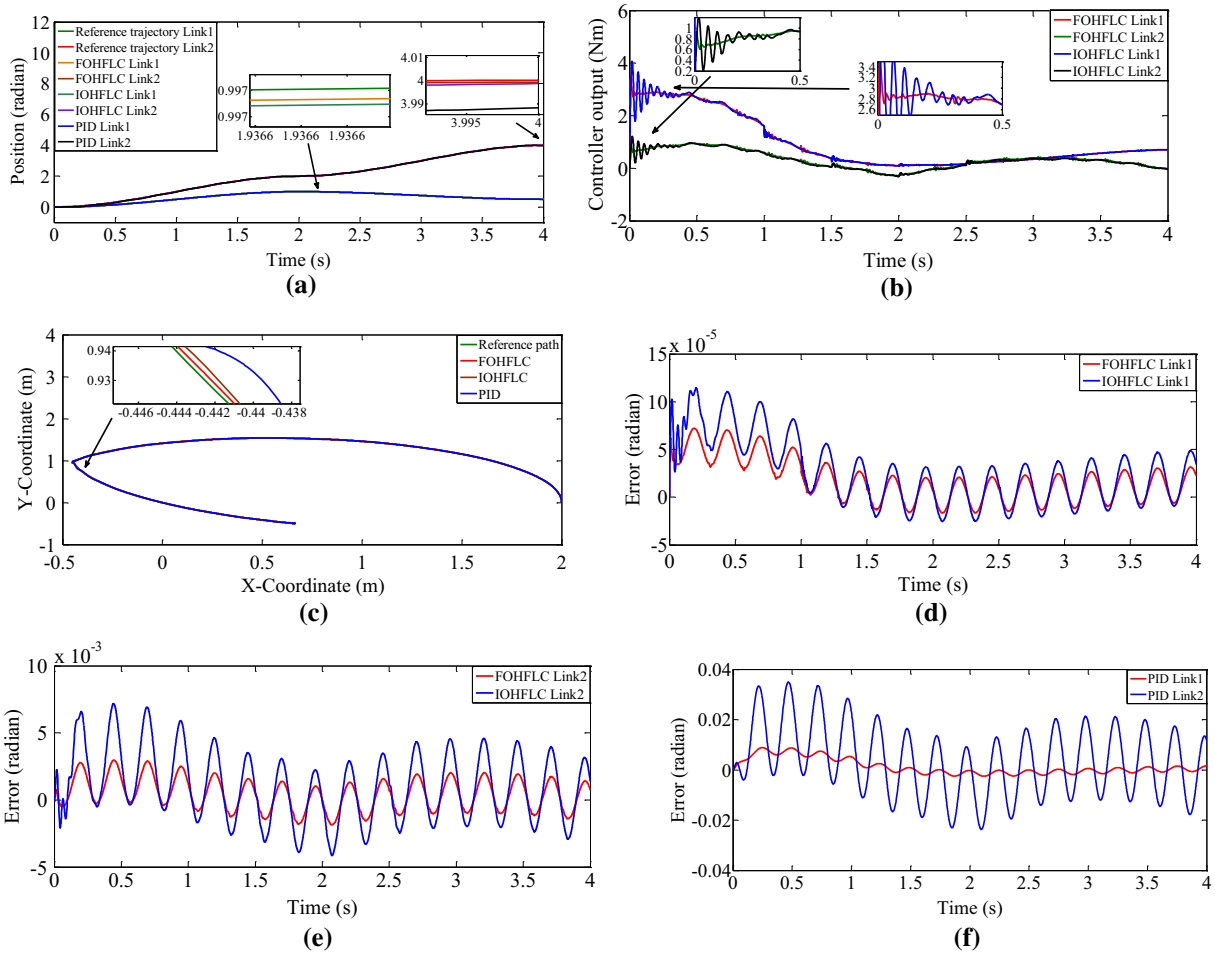

Fig. 9 **a** Trajectory tracking performances. **b** Controller output signal. **c** Path tracked by end-effector for FOHFLC, IOHFLC and PID controllers. **d** Position errors for Link1. **e** Position errors for Link2 for FOHFLC and IOHFLC controllers. **f** Position errors for Link1 and Link2 for PID controllers for adding disturbances

Table 6 IAE values for FOHFLC, IOHFLC, and PID controller schemes for disturbance 1.0 sin25t N-m

Disturbances (N-m)	FOHFLC		IOHFLC		PID	
	Link1 $\times 10^{-5}$	Link2 $\times 10^{-3}$	Link1 $\times 10^{-4}$	Link2 $\times 10^{-3}$	Link1 $\times 10^{-3}$	Link2 $\times 10^{-2}$
Link1	8.714	2.137	1.337	4.467	9.122	2.472
Link2	7.023	4.182	1.080	9.924	8.599	4.788
Both links	8.725	4.182	1.340	9.126	9.165	4.796

5.2 Disturbance Rejection

The effects of external disturbances on the performance of the proposed controller schemes are presented in this section. The sinusoidal disturbances of 1.0 sin25t N-m is added to the controller outputs for the entire 4 s. The graphs for trajectory tracking, controller outputs, path tracked by the end-effector and the positional error for three controller schemes for addition of 1.0sin25t N-m disturbance to both the links are shown in Fig. 9.

The IAE values for adding disturbances to the controller outputs are listed in Table 6. From Table 6, it is clearly indicated that the IAE values for the proposed FOHFLC scheme are lesser as compared to those of its integer-order design and conventional PID controllers. Therefore, the proposed controller approach FOHFLC performs better than IOHFLC scheme as well as conventional PID controllers in the presence of external disturbances.

From the overall results, it can be inferred that the IAE values for FOHFLC approach are lesser than those of its integer-order design and conventional PID controller for trajectory tracking, parameter variations and external disturbance. Therefore, the FOHFLC approach is more robust and effective than its integer-order design, i.e., IOHFLC and conventional PID controller for a complex 2-DOF RPRMWP plant.

6 Conclusion

In this work, the FOHFLC approach is implemented for a 2-DOF RPRMWP plant for trajectory tracking problem. The proposed controller scheme incorporates the use of two FOFLCs wherein one is a conventional FOFLC and the other is a compensating FOFLC. The compensating FOFLC has been incorporated in the design to counterbalance the coupling effects between the two links of the plant used. The fractional-order operators have enhanced the flexibility in the controller parameter choice. For the effective control action, the tuning of the parameters of any control scheme is essential therefore CSA has been used to find the optimal parameters of the proposed FOHFLC controllers. A comparative study of the FOHFLC scheme has been carried out with its integer-order design and conventional PID controller for trajectory tracking task. The robustness analysis of the proposed FOHFLC approach for parameter variations and external disturbances has also been investigated. From the results obtained, it is clearly visible that FOHFLC is more effective and robust controller approach as compared to its integer-order design and conventional PID controller.

The proposed FOHFLC scheme may be explored for its real-time implementation in the near future. The proposed approach may also be investigated for manipulator systems with flexible link which may find applications in areas such as space exploration and nuclear plants.

References

1. Lian, R.-J.; Lin, B.-F.: Design of a mixed fuzzy controller for multiple-input multiple-output systems. Mechatronics **15**, 1225–1252 (2005)
2. Lee, C.C.: Fuzzy logic in control systems: fuzzy logic controller-part 1. IEEE Trans. Syst. Man Cybern. **20**(2), 404–418 (1990)
3. Ohtani, Y.; Yoshimura, T.: Fuzzy control of a manipulator using the concept of sliding mode. Int. J. Syst. Sci. **27**(2), 179–186 (1996)
4. Hazzab, A.; Bousserhane, I.K.; Zerbo, M.; Sicard, P.: Real-time implementation of fuzzy gain scheduling of PI controller for induction motor machine control. Neural Process. Lett. **24**, 203–215 (2005)
5. Xu, C.; Shin, Y.C.: A multilevel fuzzy control design for a class of multi input single-output systems. IEEE Trans. Ind. Electron. **59**(8), 3113–3123 (2012)
6. Huo, B.; Li, Y.; Tong, S.: Fuzzy adaptive fault-tolerant output feedback control of multi-input and multi-output non-linear systems in strict-feedback form. IET Control Theory Appl. **6**(17), 2704–2715 (2012)
7. Tong, S.; Sui, S.; Li, Y.: Fuzzy adaptive output feedback control of MIMO nonlinear systems with partial tracking errors constrained. IEEE Trans. Fuzzy Syst. **23**(4), 729–742 (2015)
8. Yadav, A.K.; Gaur, P.: An optimized and improved STF-PID speed control of throttle controlled HEV. Arab. J. Sci. Eng. **41**(9), 1–12 (2016)
9. Su, Y.; Xu, L.; Li, D.: Adaptive fuzzy control of a class of MIMO nonlinear system with actuator saturation for greenhouse climate control problem. IEEE Trans. Autom. Sci. Eng. **13**(2), 772–788 (2016)
10. Song, Z.; Yi, J.; Zhao, D.; Li, X.: A computed torque controller for uncertain robotic manipulator systems: fuzzy approach. Fuzzy Sets Syst. **154**, 208–226 (2005)
11. Meza, J.L.; Santianez, V.; Soto, R.; Llama, M.A.: Fuzzy self-tuning PID semiglobal regulator for robotic manipulators. IEEE Trans. Ind. Electron. **59**(6), 2709–2717 (2012)

12. Chu, Z.Y.; Cui, J.; Sun, F.: Fuzzy adaptive disturbance-observer based robust tracking control of electrically driven free-floating space manipulator. IEEE Syst. J. **8**(2), 343–351 (2014)
13. Chiu, C.-S.: Mixed feedforward/feedback based adaptive fuzzy control for a class of MIMO nonlniear systems. IEEE Trans. Fuzzy Syst. **14**(6), 716–727 (2006)
14. Baghli, F.Z.; Bakkali, L.E.; Lakhal, Y.: Multi-input multi-output fuzzy logic controller for complex system: application on two-links manipulator. Proc. Technol. **19**, 607–614 (2015)
15. Das, S.; Pan, I.; Das, S.; Gupta, A.: A novel fractional order fuzzy PID controller and its optimal time domain tuning based on integral performance indices. Eng. Appl. Artif. Intell. **25**, 430–442 (2012)
16. Das, S.; Pan, I.; Das, S.: Performance comparison of optimal fractional order hybrid fuzzy PID controllers for handling oscillatory fractional order processes with dead time. ISA Trans. **52**, 550–566 (2013)
17. Sharma, R.; Rana, K.P.S.: Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator. Expert Syst. Appl. **41**, 11335–11346 (2014)
18. Hajiloo, A.; Xie, W.-F.: Fuzzy fractional-order PID controller design using multi-objective optimization. In: Proceedings of IFSA World Congress and NAFIPS Annual meeting 2013 Joint, Edmonton, 24–28 June 2013
19. Lin, F.: Robust Control Design: An Optimal Control Approach. Wiley, Chichester (2007)
20. Oustaloup, A.; Levron, F.; Mathieu, B.; Nanot, F.M.: Frequency-band complex noninteger differentiator: characterization and synthesis. IEEE Trans. Circuits Syst. I: Fundam. Theory Appl. **47**(1), 25–39 (2000)
21. Tang, Y.; Cui, M.; Hua, C.; Li, L.; Yang, Y.: Optimum design of fractional order $PI^{\lambda}D^{\mu}$ controller for AVR system using chaotic ant swarm. Expert Syst. Appl. **39**, 6887–6896 (2012)
22. Pan, I.; Das, S.: Chaotic multi-objective optimization based design of fractional order $PI^{\lambda}D^{\mu}$ controller in AVR system. Electr. Power Energy Syst. **43**, 393–407 (2012)
23. Yang, X.S.; Deb, S.: Cuckoo Search via Lévy flights. In: Proceedings World Congress on Nature and Biologically Inspired Computing India, pp. 210–214 (2009)
24. Gandomi, A.H.; Yang, X.-S.; Alavi, A.H.: Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng. Comput. **29**, 17–35 (2013)
25. Rajabioun, R.: Cuckoo optimization algorithm. Appl. Soft Comput. **11**, 5508–5518 (2011)
26. Yang, X.-S.; Deb, S.: Engineering optimization by cuckoo search. Int. J. Math. Modell. Numer. Optim. **1**(4), 330–343 (2010)
27. Tan, W.S.; Hassan, M.Y.; Majid, M.S.; Rahman, H.A.: Allocation and Sizing of DG using cuckoo search algorithm. In: 2012 IEEE International Conference on Power and Energy (PEcon) Kota Kinabalu Sabah, Malaysia, pp. 133–8 (2012)
28. Yildiz, A.R.: Cuckoo search algorithm for the selection of optimal machining parameters in milling operations. Int. J. Adv. Manuf. Technol. **64**(1–4), 55–61 (2013)
29. Bulatovic, R.R.; Dordevic, S.R.; Dordevic, V.S.: Cuckoo search algorithm: a metaheuristic approach to solving the problem of optimum synthesis of a six-bar double dwell linkage. Mech. Mach. Theory **61**, 1–13 (2013)
30. Ayala, H.V.H.; Coelho, L.D.S.: Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator. Expert Syst. Appl. **39**, 8968–8974 (2012)