Original Paper

Social spider optimization algorithm for tuning parameters in PD-like Interval Type-2 Fuzzy Logic Controller applied to a parallel robot

Amjad J Humaidi[1] , **Huda T Najem[1], Ayad Q Al-Dujaili[2]** , **Daniel A Pereira[3], Ibraheem Kasim Ibraheem[4]** and **Ahmad Taher Azar[5]**

Abstract

This paper presents control design based on an Interval Type-2 Fuzzy Logic (IT2FL) for the trajectory tracking of 3-RRR (3-Revolute-Revolute-Revolute) planar parallel robot. The design of Type-1 Fuzzy Logic Controller (T1FLC) is also considered for the purpose of comparison with the IT2FLC in terms of robustness and trajectory tracking characteristics. The scaling factors in the output and input of T1FL and IT2FL controllers play a vital role in improving the performance of the closed-loop system. However, using trial-and-error procedure for tuning these design parameters is exhaustive and hence an optimization technique is applied to achieve their optimal values and to reach an improved performance. In this study, Social Spider Optimization (SSO) algorithm is proposed as a useful tool to tune the parameters of proportional-derivative (PD) versions of both IT2FLC and T1FLC. Two scenarios, based on two square desired trajectories (with and without disturbance), have been tested to evaluate the tracking performance and robustness characteristics of proposed controllers. The effectiveness of controllers have been verified via numerical simulations based on MATLAB/SIMULINK programming software, which showed the superior of IT2FLC in terms of robustness and tracking errors.

Introduction

The parallel robots are mechanisms with two or more kinematic chains linked to their end-effectors. They are better known because of the good performance when it takes care of acceleration, precision, stiffness, and capacity to carry heavy loads. The parallel robots are adequate instruments for applications in different fields such as in machining, welding, handling, flight simulators, and telescopes pointing. They have also advantages related to their sensitivity, which make them very suitable for applications in medicine, like endoscopy and other procedures that should be minimum invasive.[1–3]

The parallel manipulators have many positive features. Their actuators are fixed to the base, which decreases the mass and allows high acceleration; due to the parallel action, these actuators provide high stiffness and efficient load capacity. Theoretically, when compared to serial manipulators, their accuracy is better. The good precision is a result of the fact that parallel manipulators have errors in the position given by average errors of individual chains. Their repeatability is also high, which is due to kinematic chains that are closed. However, the parallel manipulators have some drawbacks: the workspace is limited and the behavior of singularities is more complicated than in serial manipulators.[1–6]

Parallel mechanisms can be used in humanoid robots in order to improve their structure and operation performance as inspired by human anatomy. In recent

[1]Control and Systems Engineering Department, University of Technology, Baghdad, Iraq
[2]Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq
[3]Federal University of Lavras, Lavras, Brazil
[4]Department of Electrical Engineering, College of Engineering, University of Baghdad, Iraq, Baghdad
[5]Faculty of Computers and Artificial Intelligence, Benha University, Benha 13518, Egypt.

Corresponding author:
Ayad Q. Al-Dujaili, Electrical Engineering Technical College, Middle Technical University, Al Doura, Baghdad 10022, Iraq.
Email: ayad.qasim@mtu.edu.iq

years significant advances have been made in Human-Robot Interaction (HRI).[7] Hand gestures have been investigated as the most natural tools of interaction for human beings, and particularly for disabled persons.[8] Some parallel mechanisms are utilized to mimicking human neck,[9] others are used to improve the imitation skill and children with autism.[10] Other parallel mechanisms are utilized for manufacturing anthropomorphic exoskeletons which matching and mimicking the real motions of some body parts, such as lower and upper limb exoskeletons, to be worn by elderly and disable persons.[11]

The redundant manipulators offer a safe physical collaborative flexible workspace for nurse or surgeon (assisting physicians, patient support) undergoing surgery. During the past decades, the development of surgical robot systems and teleoperated surgery, which mostly involve articulated (serial and parallel) robot configurations, has witnessed rapid growth due to the advancements in computing and sensor technologies.[12,13] In the tele-operated surgery, the surgical robots have to show both safe and flexible manipulation. On the contrary of serial robot manipulator, which are featured by flexible motion, the parallel robots suffers from limitation in workspace. However, the surgical parallel robots are characterized by higher accuracy and less vibration than serial counterparts. This makes them candidates for such medical applications that require high precision and safe operation.[14,15]

In this study, a planar parallel robot of the kind 3-Revolute–Revolute–Revolute (3-RRR) with 2-Degrees-Of-Freedom (DOF) will be considered. Many researchers have conducted their control design for tracking control of 2-DOF planar redundant parallel robot. Zhang et al. proposed an augmented PD controller with forward dynamic compensation for a redundant planar 2-DOF parallel manipulator. Better Accuracy has been achieved with the compensation of active joint friction and the proposed controller is easy to implement and it requires shorter sampling period and it shows better performance than simple PD controller.[16] Chen et al. proposed a simple scheme for computing the inverse dynamics of 2-DOF planar redundant parallel manipulator. The study developed four control algorithms, represented by two PD controls, an augmented PD control, and a computed-torque control scheme for control purpose. The experimental results showed that the model-based controllers have better than PD controllers.[17] Yu et al. have presented a novel Udwadia–Kalaba approach or 2-degrees of freedom redundant parallel manipulator. This approach can represent the direct and inverse dynamical models of the robot precisely and explicitly.[18] Sariyildiz et al.[19] applied a new Artificial Intelligent-based inverse kinematic solution method for the planar redundant robot manipulator using support vector machine method (SVM). Al-Mayyahi proposed fractional order proportional integral derivative (FOPID) controller for the path tracking control of the center of the 3-RRR planar parallel robot. The design parameters FOPID controller is optimized using the bat optimization algorithm for better improvement of controller's performance.[20] Liu focused on the solution of the singularity problem in Redundant Parallel Manipulator. Three kinds of redundant methods are developed and discussed. The kinematic and dynamic control methods are applied for trajectory tracking of redundantly actuated parallel mechanisms.[21] Shang and Cong applied a new robust nonlinear controller to a planar 2-DOF parallel manipulator. The controller combines nonlinear PD control with the robust dynamic compensation. The NPD control is used to compensate disturbances, unmodeled dynamics, and friction, while the robust control part is used to compensate the model uncertainties of the robot. The proposed controller showed better trajectory tracking accuracy as compared to augmented PD controller.[22] Sheng and Li have proposed optimized PID control design for trajectory tracking control of 3-RRR parallel robot. The Genetic Algorithm (GA) is applied for tuning the design parameters of PID controller in order to improve the robustness of controller and precision of trajectory tracking.[23] Moezi et al. presented robust optimal fuzzy logic controller for trajectory tracking of 3-RRR parallel robot based on PWM technique. The modified Cuckoo Optimization Technique has been applied for tuning the parameters of FL controller. The proposed control scheme showed better robustness characteristics and tracking performance when compared to optimal PID controller.[24] Noshadi et al. presented active force control (AFC) for 3-RRR planar parallel manipulator. The AFC has showed better robustness characteristics and better disturbance rejection capability as compared to conventional PID controller.[25]

Taking advantage of the fact that, compared with conventional controllers, the Fuzzy Logic Controllers (FLC) can be considered to be more robust and less sensitive to variations in parameters.[26] There are many researchers who have used and proved the efficacy of FLC in various applications like cruise control systems, cyber defense mechanism, and parallel robots.[27–29]

In the past few years, fuzzy control has been a topic of great interest in the area of robot control. In Moezi et al.,[24] the Cuckoo Optimization Technique has been used for tuning the design parameters of FLC to improve its tracking performance and robustness against variation of parameters. A fuzzy-based controller was proposed by Stanet et al. for an application in medicine with a 3-DOF parallel robot.[28] The fuzzy controller provided more accurate results than a classical proportional-integral-derivative controller. In Noshadi et al.,[30] Noshadi et al. designed two level fuzzy tuning resolved acceleration control (FLRAC) for trajectory tracking of 3-RRR planar parallel robot. The first level of FLC is used for acquiring the gains of PD controller. The second level is used for tuning the parameters of fuzzy controller to improve the performance. The FLRAC is combined with AFC to enhance the

robustness and accuracy. Lu et al. have proposed an optimization procedure to design and tune Interval Type-2 Fuzzy Logic Controller (IT2FLC) and PID IT2FLC for the tracking control of Delta parallel robot.[31,32] The aim was to improve the robot control accuracy; however, it was also considered that a good control program must-have characteristic such as simplicity, applicability, robustness, and stability.

The theory of fuzzy sets, proposed by Lotfi Zadeh in 1965, originated the Type-1 FLC (T1FLC), which was extended by Zadeh in 1975, originating the Type-2 FLC (T2FLC).[33,34] In T1FLC, the uncertainties related to the membership functions can be described in just two dimensions, whilst T2FLC can properly deal with them in three dimensions, which make it able to handle numerical uncertainties and also nonlinearities. So, T2FLC can replace T1FLC in problems with complex nonlinear systems.[35–37]

To improve the dynamic response of the closed-loop system, FLCs must have PI-type or PD-type control structures. Here in this work, a PD-like FLC structure is considered. Although, as in the classical PID controllers, the selection of the PD gain parameters using a trial and error procedure does not give an optimal solution. So, optimization techniques can be applied to tune these parameters.[38–41]

In the present work, the Social Spider Optimization (SSO) algorithm is applied to tune the scaling factors in the output and input of an Interval Type-2 PD-Like Fuzzy Logic Controller (IT2PDFLC) for the position control of parallel manipulator. So, the goal is to obtain optimal values toward the minimum value of the cost function and hence to improve the closed-loop system dynamics. It is also presented in this paper the design of a Type-1 PD-Like Fuzzy Logic Controller (T2PDFLC) for the position control of the 3-RRR parallel robot, was the SSO algorithm is also used to tune the design parameters. Therefore, the scope of this work can be summarized by:

☐ Design of IT2PDFLC for a 2-DOF planar 3-RRR parallel robot.
☐ Dynamic performance improvement of T1PDFLC and IT2PDFLC using SSO.
☐ Comparative analysis between T1PDFLC and IT2PDFLC controllers in terms of dynamic performance and robustness.

The model of planar 3-RRR parallel robot

In this section, the mathematical model of the planar 3-RRR parallel manipulator will be analyzed. For this kind of parallel robot, the solution is not unique for neither Direct Kinematics nor to Inverse Dynamics. So, there are discussed just Inverse Kinematic Model (IKM), Forward Kinematic Model (FKM), and Direct Dynamic Model (DDM).
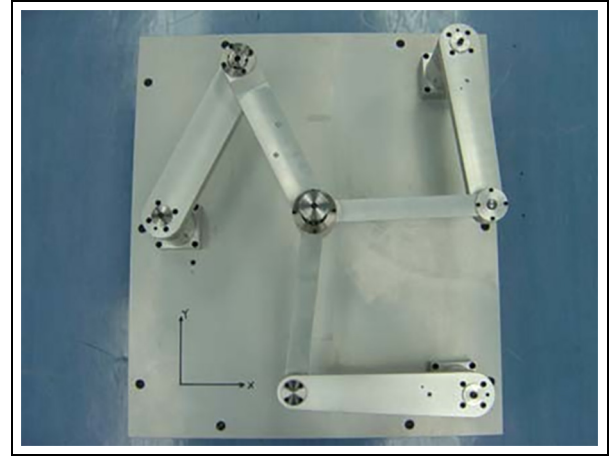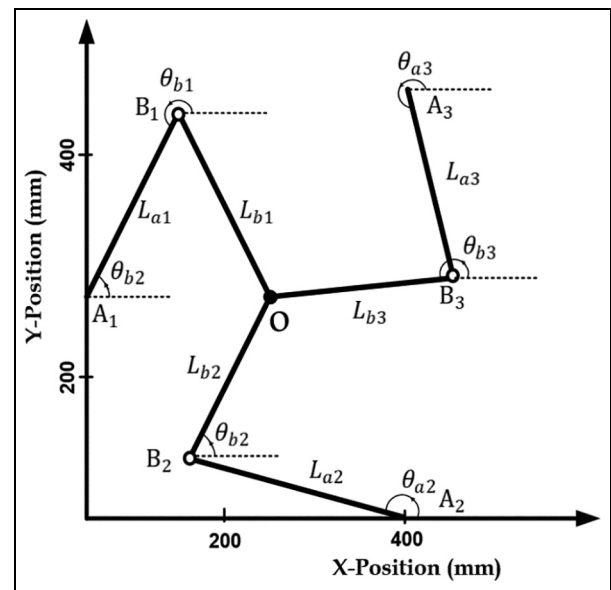

Figure 1. Mechanisms of the 3-RRR.

Figure 2. Scheme of a planar 3-RRR parallel robot.

Forward kinematic of (planar 3-RRR) parallel robot

The setup of 3-RRR parallel robot consists of three actuators (redundantly actuated) with tip point connected to three identical branches, being each branch planar with 2-DOF (translation along *x-y*) as shown in Figure 1. In such a system, the acceleration of the end-effector ranges between $(7.2–15)\,m/s^2$ and the velocity ranges between $(0.216–0.3)\,m/s$.[7,8] The general scheme of a redundant 2-DOF 3-RRR parallel robot is presented as shown in Figure 2. This parallel robot has three active links (L_{a1}, L_{a2}, L_{a3}), three passive links (L_{b1}, L_{b2}, L_{b3}), three active joints (A1, A2, A3), and three passive joints (B_1, B_2, B_3). The end-effector is positioned at *O*. As shown Figure 2, it is defined as the reference frame for the workspace of the manipulator. The desired trajectories for parallel robots should be described by the coordinates of the end-effector.

The problem of forward kinematic is related to the computation of the position in the Cartesian frame of

the end point C, regarding the joint angles θ_{a1}, θ_{a2}, and θ_{a3} as depicted in Figure 2. The index A_i is related to the ith actuated joint in the vector of coordinates

$$X_{Ai} = \begin{bmatrix} x_{Ai} \\ y_{Ai} \end{bmatrix}, \qquad (1)$$

where $i = 1, 2, 3$, The B_i represents the ith passive joint vector with coordinates

$$X_{Bi} = \begin{bmatrix} x_{Bi} \\ y_{Bi} \end{bmatrix} \qquad (2)$$

The point C describes the position of the end-effector, which is determined by:

$$X_C = \begin{bmatrix} x_C \\ y_C \end{bmatrix} \qquad (3)$$

From Figure 2, one can have

$$|X_C - X_{Bi}|^2 = L^2, i = 1, 2, 3 \qquad (4)$$

Equation (2) can be written as

$$X_{Bi} = \begin{bmatrix} x_{Bi} \\ y_{Bi} \end{bmatrix} = \begin{bmatrix} x_{Ai} + L\cos\theta_i \\ y_{Ai} + L\sin\theta_i \end{bmatrix} i = 1, 2, 3 \qquad (5)$$

Taking equations (3) and (5) and finding a solution for x_C and y_C leads to equations of the forward kinematics

$$x_C{}^2 - 2x_C x_{Bi} + x_{Bi}{}^2 + y_C{}^2 - 2y_C y_{Bi} + y_{Bi}{}^2 = L^2 \qquad (6)$$

Using equation (5), the norm of X_{Bi} is given by

$$x_{Bi}{}^2 + y_{Bi}{}^2 = \|X_{Bi}\|^2 \qquad (7)$$

The substitution of equation (7) into equation (6) provides the equations of links (1), (2), and (3);

$$x_C{}^2 + y_C{}^2 - 2x_C x_{B1} - 2y_C y_{B1} + \|X_{B1}\|^2 = L^2 \quad (8)$$
$$x_C{}^2 + y_C{}^2 - 2x_C x_{B2} - 2y_C y_{B2} + \|X_{B2}\|^2 = L^2 \quad (9)$$
$$x_C{}^2 + y_C{}^2 - 2x_C x_{B3} - 2y_C y_{B3} + \|X_{B3}\|^2 = L^2 \quad (10)$$

Taking equation (8) and subtracting equations (9) and (10), respectively, gives:

$$2x_C(x_{B2} - x_{B1}) + 2y_C(y_{B2} - y_{B1}) = \|X_{B2}\|^2 - \|X_{B1}\|^2 \qquad (11)$$

$$2x_C(x_{B3} - x_{B1}) + 2y_C(y_{B3} - y_{B1}) = \|X_{B3}\|^2 - \|X_{B1}\|^2 \qquad (12)$$

Using a matrix notation, equations (11) and (12) can be written together as:

$$\begin{bmatrix} 2(x_{B2} - x_{B1}) & 2(y_{B2} - y_{B1}) \\ 2(x_{B3} - x_{B1}) & 2(y_{B3} - y_{B1}) \end{bmatrix} \begin{bmatrix} x_C \\ y_C \end{bmatrix} = \begin{bmatrix} \|X_{B2}\|^2 - \|X_{B1}\|^2 \\ \|X_{B3}\|^2 - \|X_{B1}\|^2 \end{bmatrix} \qquad (13)$$

It can be easy shown that x_C can be calculated by

$$x_C = \frac{\|X_{B1}\|^2(y_{B2} - y_{B3}) + \|X_{B2}\|^2(y_{B3} - y_{B1})\|X_{B3}\|^2(y_{B1} - y_{B2})}{2|x_{B1}(y_{B2} - y_{B3}) + x_{B2}(y_{B3} - y_{B1}) + x_{B3}(y_{B1} - y_{B2})|} \qquad (14)$$

Using equation (13) and follow the same analysis as above, one can find the y_C

$$y_C = \frac{\|X_{B1}\|^2(x_{B3} - x_{B2}) + \|X_{B2}\|^2(x_{B1} - x_{B3})\|X_{B3}\|^2(x_{B2} - x_{B1})}{2|x_{B1}(y_{B2} - y_{B3}) + x_{B2}(y_{B3} - y_{B1}) + x_{B3}(y_{B1} - y_{B2})|} \qquad (15)$$

It is worthy to note that without the third leg, it is possible to find solutions to any two-leg planar manipulator considering the forward kinematic. These constraints lead to a unique solution of the forward kinematic problem.

Inverse kinematic

The problem of the inverse kinematic can be characterized by the computation of the actuator joint angles $\theta_{a1}, \theta_{a2}, \theta_{a3}$ as a function of a given position in Cartesian space of the end-effector C. Taking any triangle in Figure 2 gives:

$$L_{bi}^2 = L_{ai}^2 + L_i^2 - 2L_{ai}L_{bi}\cos(\theta_{ai} - \theta_{bi}) \qquad (16)$$

where,

$$\theta_{bi} = \arctan\frac{y_c - y_{Ai}}{x_c - x_{Ai}} \qquad (17)$$

Hence θ_{ai} can be solved as follows:

$$\theta_{ai} = \arccos\frac{L_{ai}^2 + L_i^2 - L_{bi}^2}{2L_{ai}L_i} + \theta_{bi} \qquad (18)$$

There are two solutions related to equation (18), which result from the "arc cos" function. As there are two solutions for each leg, it means that the manipulator has eight solutions. It is also important to notice that the actuated joints are placed on the vertices of a triangle.

Direct dynamic model

In this work, the Euler-Lagrange equation is considered for the development of the dynamic model of the 3-RRR parallel manipulator. The general motion of the 3-RRR manipulator may be represented by the serial kinematics chains subjected to constraints in the closed loop. Also, the parallel manipulator dynamics can be given by the combination of the dynamics of the serial kinematic chains with their related force constraints.

Dynamic model of serial kinematic chain. The 3-RRR parallel robot can be divided in three serial kinematic chains based on its cutting at the end-effector O, being each chain a planar 2-DOF serial robot with two links and
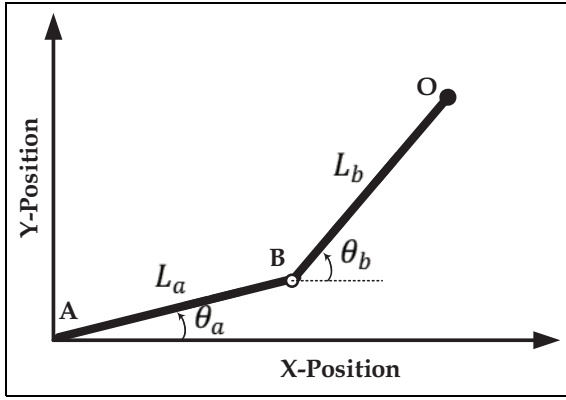
Figure 3. Scheme for the serial kinematic chain.

two joints. A scheme for the serial kinematic chain is presented as shown in Figure 3.

L_a, L_b denote the links and θ_a, θ_b denote the joint angles. A is the base of the kinematic chain and O is the position of the end-effector. The two links have the same length L and are considered to be ideal rigid bodies.

As the motion of the kinematic chain is performed in the horizontal plane, it is possible to ignore the gravity effect. Thus, the links have kinetic energy KE_a and KE_b given by:

$$KE_a = \frac{1}{2}J_a\dot{\theta}_a^2 + \frac{1}{2}m_a(\dot{x}_{ca}^2 + \dot{y}_{ca}^2) \tag{19}$$

$$KE_b = \frac{1}{2}J_b\dot{\theta}_b^2 + \frac{1}{2}m_b(\dot{x}_{cb}^2 + \dot{y}_{cb}^2) \tag{20}$$

where m_a and m_b represent masses of the links, J_a and J_b denote the inertia moments of the links with respect to the centers of masses, and (x_{ca}, y_{ca}) and (x_{cb}, x_{cb}) represent the coordinates of the centers of mass. The symbols r_a and r_b represent the distances from the joints to the centers of mass. Moreover, the coordinates of the centers of mass can be expressed by:

$$x_{ca} = r_a cos\theta_a,\ y_{ca} = r_a \sin\theta_a,\ x_{cb} = lcos\theta_a + r_b cos\theta_b,$$
$$x_{cb} = lsin\theta_a + r_b sin\theta_b \tag{21}$$

Taking the derivative of equation (21) and substituting into equations (19) and (20) give

$$KE_a = \frac{1}{2}J_a\dot{\theta}_a^2 + \frac{1}{2}m_a r_a^2 \dot{\theta}_a^2 \tag{22}$$

$$KE_b = \frac{1}{2}J_b\dot{\theta}_b^2 + \frac{1}{2}m_b(l^2\dot{\theta}_a^2 + r_b^2\dot{\theta}_b^2) + m_b.l.r_b cos(\theta_a - \theta_b)\dot{\theta}_a\dot{\theta}_b \tag{23}$$

The kinematic chain kinetic energy is derived from equations (22) and (23):

$$KE = \frac{1}{2}\alpha\dot{\theta}_a^2 + \frac{1}{2}\beta\dot{\theta}_b^2 + \gamma cos(\theta_a - \theta_b)\dot{\theta}_a\dot{\theta}_b \tag{24}$$

where the symbols α, β, γ are defined as follows:

$$\alpha = J_a + m_a r_a^2 + m_b l^2,\ \beta = J_b + m_b r_b^2,\ \gamma = m_b l r_b$$

Let $\theta = [\theta_a \theta_b]^T$ be a vector containing the joint angles, $\tau = [\tau_a \tau_b]^T$ the vector with the torques of the joints. So, the Euler-Lagrange formulation for the equation of the kinematic chain is given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\theta}\right) - \frac{\partial L}{\partial\theta} = \tau \tag{25}$$

Expanding equation (25), the kinematic chain has a dynamic that can be expressed by:

$$M\ddot{\theta} + C\dot{\theta} = \tau \tag{26}$$

where the matrices M and C are given by;

$$\mathbf{M} = \begin{bmatrix} \alpha & \gamma\cos(\theta_a - \theta_b) \\ \gamma\cos(\theta_a - \theta_b) & \beta \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} 0 & \gamma\sin(\theta_a - \theta_b)\dot{\theta}_b \\ -\gamma\sin(\theta_a - \theta_b)\dot{\theta}_a & 0 \end{bmatrix}$$

Dynamic model of parallel manipulator. The 3-RRR parallel manipulator has a dynamic model derived from the serial manipulator dynamic model. As such, the parameters of the serial kinematic chains are

$$\alpha_i = J_{ai} + m_{ai}r_{ai}^2 + m_{bi}l^2,$$
$$\beta_i = J_{bi} + m_{bi}r_{bi}^2,\ \gamma_i = m_{bi}lr_{bi}$$

where $i = 1, 2, 3$, m_{ai} and m_{bi} refer to the masses of links, J_{ai} and J_{bi} represent the links' inertia moments with respect to the center of mass, r_{ai} and r_{bi} denote the distances from the joints to the centers of mass, and L is the links' length. The combination of the three serial chains dynamics, taking into consideration their force constraints, leads to the formulation of the parallel manipulator dynamic model represented by:

$$M\ddot{\theta} + C\dot{\theta} = \tau \tag{27}$$

where $\theta = [\theta_{a1}, \theta_{a2}, \theta_{a3}, \theta_{b1}, \theta_{b2}, \theta_{b3}]^T$, $\tau = [\tau_{a1}, \tau_{a2}, \tau_{a3}, \tau_{b1}, \tau_{b2}, \tau_{b3}]^T$ denote the vector of joint positions and input torques, respectively. M is the matrix of inertia and C is the matrix of Coriolis, which are given by

$$\mathbf{M} = \begin{bmatrix} \alpha_1 & 0 & 0 & \gamma_1 C_{ab1} & 0 & 0 \\ 0 & \alpha_2 & 0 & 0 & \gamma_2 C_{ab2} & 0 \\ 0 & 0 & \alpha_3 & 0 & 0 & \gamma_3 C_{ab3} \\ \gamma_1 C_{ab1} & 0 & 0 & \beta_1 & 0 & 0 \\ 0 & \gamma_2 C_{ab2} & 0 & 0 & \beta_2 & 0 \\ 0 & 0 & \gamma_3 C_{ab3} & 0 & 0 & \beta_3 \end{bmatrix} \tag{28}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & \gamma_1 S_{ab1}\dot{\theta}_{b1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \gamma_2 S_{ab2}\dot{\theta}_{b2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \gamma_3 S_{ab3}\dot{\theta}_{b3} \\ -\gamma_1 S_{ab1}\dot{\theta}_{a1} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma_2 S_{ab2}\dot{\theta}_{a2} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\gamma_3 S_{ab3}\dot{\theta}_{a3} & 0 & 0 & 0 \end{bmatrix} \quad (29)$$

In equations (28) and (29), the symbol C_{abi} denotes $\cos(\theta_a - \theta_b)$, $i = 1, 2, 3$, while S_{abi} defines $\cos(\theta_a - \theta_b)$, $i = 1, 2, 3$. Therefore, equations (27)–(29) describe the dynamic model of 3-RRR planar parallel robot represented by Figure 2 and they will be used as the robotic system dynamic to be controlled by both IT2FLC and T1FL controllers.

Type-2 Fuzzy Logic Control

Type-2 fuzzy logic theory

The membership function in the fuzzy sets represents the degree of membership of an element. The Type-1 Fuzzy Set (T1 FS) named A comprises the domain of real numbers U (which is the universe of discourse) and also the membership function (MF) (μ_A): $U \rightarrow [0; 1]$. For each value of x, $\mu_A(x)$ is the membership grade or degree of membership, of x in A. When U is continuous, A is written as[37,42]

$$A = \int_U \mu_A(x)/x \quad (30)$$

A type-2 fuzzy set (T2 FS) $\tilde{A}$ is the extension of the concept of T1 FS. It introduces the concept of uncertainty in membership grades of T1 FSs. Mathematically, a type-2 fuzzy set denoted as $\tilde{A}$, is characterized by a type-2 membership function $\mu_{\tilde{A}}(x, u)$, where $x \in X$ and $u \in J_x \subseteq [0.1]$, that is,

$$\tilde{A} = \left\{ ((x, u).\mu_{\tilde{A}}) | \forall x \in X, \forall u \in J_x \subseteq [0.1] \right\}$$

where X is the universe of discourse (UOD) and $0 \leqslant u_{\tilde{A}} \leqslant 1$. $\tilde{A}$ can also be expressed as:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \frac{\mu_{\tilde{A}}(x, u)}{(x, u)} J_x \subseteq [0, 1] \quad (31)$$

where $\int\int$ represents the union overall admissible u and x. The J_x is referred to as primary membership of x.

Figure 4 shows the membership function (MF) of Type-2 Fuzzy Logic set. A vertical slice of the T2 MF at $x = x'$ defines the secondary membership function at that point given by

$$\mu_{\tilde{A}} = \int_{u \in J_{x'}} \frac{\mu_{\tilde{A}}(x', u)}{u} J_x \subseteq [0, 1] \quad (32)$$

The secondary membership function associated with $x = x'$, for a given $x \in X$, is the type-1 membership function defined by $\mu_{\tilde{A}}(x = x', u)$, $u \in J_x$. The Type-2 Fuzzy Set has an uncertainty area $\tilde{A}$ that is bounded that is known as footprint of uncertainty (FOU). This area characterizes the union of all the primary membership functions, which are given by[42]:

$$FOU(\tilde{A}) = \bigcup_{x \in X} J_x \quad (33)$$

The lower and upper MF of $\tilde{A}$ are two T1 FS, while the $FOU(\tilde{A})$ has boundaries for T2 FS $\tilde{A}$ that are the lower and upper bounds of T1 FS. The lower MF is represented by $\tilde{\mu}_{\tilde{A}}(x)x \in X$, and the upper MF is defined by $\overline{\mu}_{\tilde{A}}(x)x \in \overline{X}$, which means

$$\overline{\mu}_{\tilde{A}}(x) = \overline{FOU(\tilde{A})}, \; \tilde{\mu}_{\tilde{A}}(x) = \underline{FOU(\tilde{A})} \quad (34)$$

Structure of IT2FL control

The IT2FLC has a structure that is almost the same of the T1FLC, but the membership functions are different. This structure can be seen in Figure 5. The fundamental form of the IT2FLC are Fuzzifier, Rule base, Inference engine, Type-reduction, and Defuzzification.[26,37]
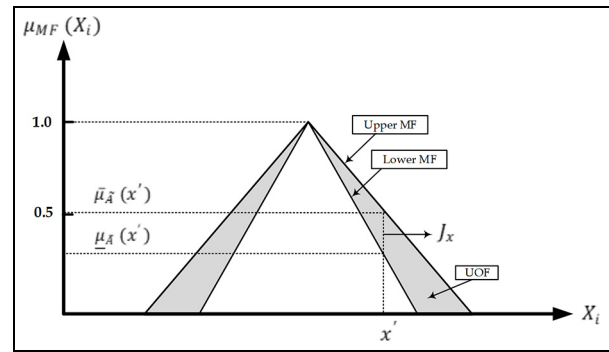

Figure 4. The structure of membership function for Type-2 FLC.
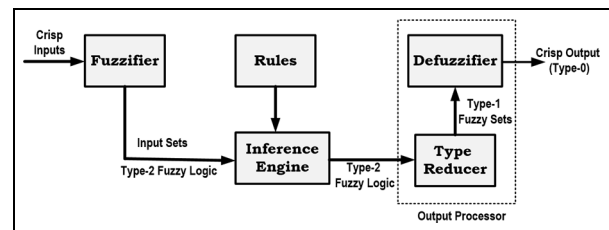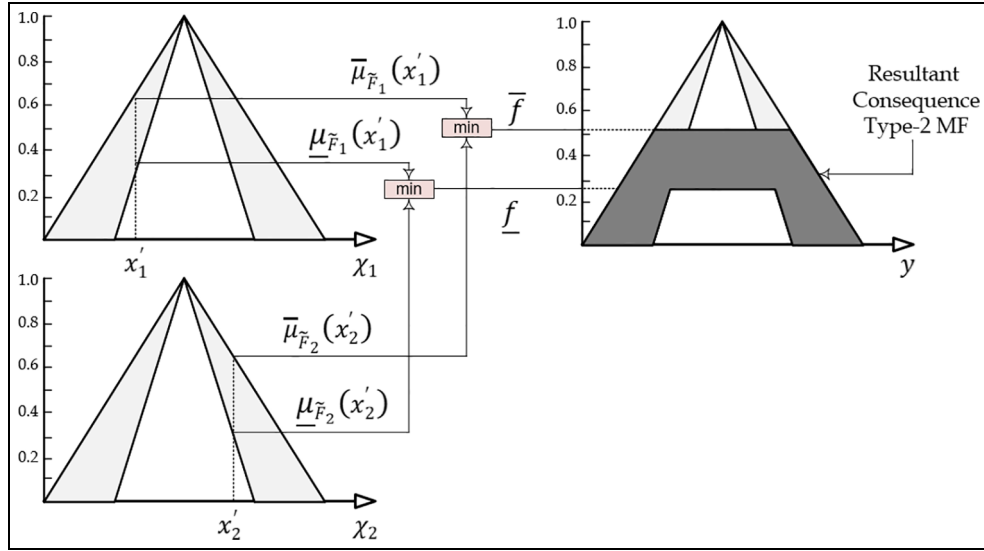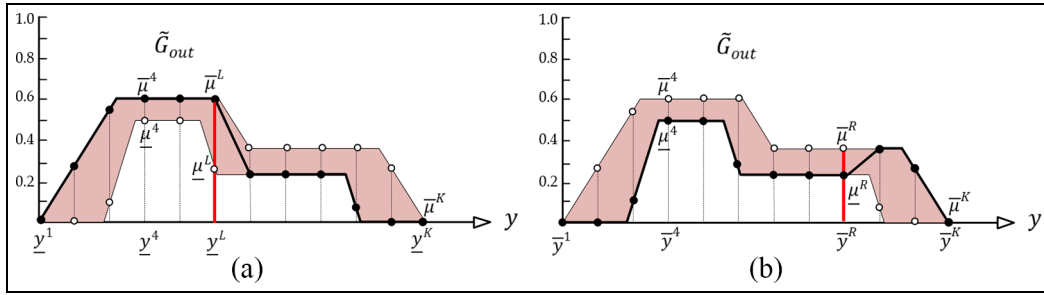

Figure 5. Structure of T2FLC.

Figure 6. Mamdani inference minimum operation used by Type-2 FSs.

Figure 7. Switching points in computing y_l and y_r.

Fuzzifier. Inputs are mapped, by the fuzzifier, into Type 2 Fuzzy Sets (T2 FS), starting the inference engine. The implementation of the fuzzifier for the T2 FS can be performed in a simple manner by the mapping of a crisp input in a Singleton FS, as represented by[37,42]

$$\mu_{\tilde{A}_x} = \begin{cases} 1, & x = x' \\ 0, & otherwise \end{cases} \qquad (35)$$

Rule base. In Type-2 FLSs the Rule-Base is also synthesized in a set of If-Then rules, which establish the relationship between the input and output of the system. The Type-2 FLS rule can be represented as follows[37]:

$$R^i : \text{ if } x_1 \text{ is } \tilde{F}_1^i \text{ and } \cdots \text{ and } x_j \text{ is } \tilde{F}_j^i, \text{ Then } y^i \text{ is } \tilde{G}^i$$

where R^i denotes the ith fuzzy rule, $\tilde{F}_j^i$ and $\tilde{G}^i$ are linguistical terms described by Interval T2FLS, $i = \{1, \cdots, M\}$, being M the amount of rules, $J = \{1, \cdots, N\}$, being N the amount of antecedents, x_j representing the inputs of the FLS and y^i the output of the rule.

Inference engine. Fuzzy inputs are assigned to fuzzy outputs by the Inference Engine (IE), which is performed using the operators and base rules, including union and intersection operators. The IE is the essential difference between T2 and T1 FLS. In IT2FS, the resultant operation is described as an interval depicted by $\tilde{\mu}_j^i$

$$\tilde{\mu}_{F_j^i}(x_j) = \left[\underline{\mu}_{\tilde{F}_j^i}(x_j), \bar{\mu}_{\tilde{F}_j^i}(x_j) \right] \qquad (36)$$

where x_j is the jth FLS system input. Taking the lower and upper bounds of the IT2FS, the intersection is obtained adopting the t-norm and considering the antecedent FS, begin represented by:

$$\underline{f}^i = T_{j=1}^N \left(\underline{\mu}_{\tilde{F}_j^i}(x_j) \right), \bar{f}^i = T_{j=1}^N \left(\bar{\mu}_{\tilde{F}_j^i}(x_j) \right) \qquad (37)$$

being T a t-norm (product or minimum). Taking the minimum t-norm, the result for input and antecedent operations can be seen in Figure 6.

In this work, IT2FLS applies the Mamdani minimum implication method, as the *t-norm* operator, to the rule's firing level $\tilde{f}^i$ and the consequent $\tilde{G}^i$ taking into account the upper and lower bounds of $\tilde{f}^i$ and $\tilde{G}^i$ separately. This minimum t-norm can be described in Figure 7.

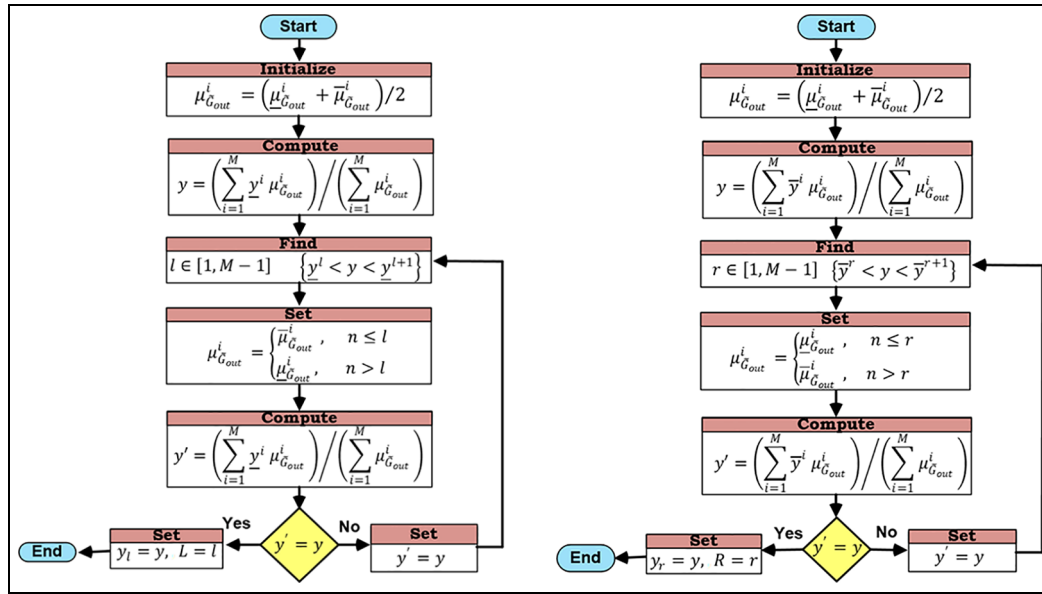*Type reduction.* In this operation, all the fired FS must be combined in order to find the crisp value. With this

Figure 8. Iterative Karnik-Mendel algorithm.

aim, the centroid of a T2 FS must be described as an interval that is a type-reduced set. This work adopts an extension of Type-1 defuzzification method called Karnik-Mendel (KM) algorithm.[37] It has been shown that this iterative algorithm is a TR method with a great accuracy. The TR Centroid begins by the definition of K samples from a T2 FS. The TR method has firstly to obtain two Type-1 FS whose centroid best approximates the upper and lower bounds of the Type-2 FS centroid. Considering the switching points (L, R), Figure 7 illustrates the procedure to find the optimal values.

The candidate points are obtained as follows:

$$y_l(k) = \left(\sum_{i=1}^{k} \underline{y}^i \bar{\mu}_{\tilde{G}_{out}}^i + \sum_{i=k+1}^{K} \underline{y}^i \underline{\mu}_{\tilde{G}_{out}}^i\right)/ \left(\sum_{i=1}^{k} \bar{\mu}_{\tilde{G}_{out}}^i + \sum_{i=k+1}^{K} \underline{\mu}_{\tilde{G}_{out}}^i\right) \quad (38)$$

$$y_r(k) = \left(\sum_{i=1}^{k} \bar{y}^i \underline{\mu}_{\tilde{G}_{out}}^i + \sum_{i=k+1}^{K} \bar{y}^i \bar{\mu}_{\tilde{G}_{out}}^i\right)/ \left(\sum_{i=1}^{k} \underline{\mu}_{\tilde{G}_{out}}^i + \sum_{i=k+1}^{K} \bar{\mu}_{\tilde{G}_{out}}^i\right) \quad (39)$$

where k is an integer in $[1, K-1]$ interval, and K represents the number of discretization points. Then, the optimal bounds can be getting by y_l and y_r,

$$y_l = \min_{k \in [1, M-1]} y_l(k) \equiv y(L) \equiv$$
$$\left(\sum_{i=1}^{L} \underline{y}^i \bar{\mu}_{\tilde{G}_{out}}^i + \sum_{i=L+1}^{K} \underline{y}^i \underline{\mu}_{\tilde{G}_{out}}^i\right)/ \left(\sum_{i=1}^{L} \bar{\mu}_{\tilde{G}_{out}}^i + \sum_{i=L+1}^{K} \underline{\mu}_{\tilde{G}_{out}}^i\right) \quad (40)$$

$$y_r = \max_{k \in [1, M-1]} y_r(k) \equiv y(R) \equiv$$
$$\left(\sum_{i=1}^{R} \bar{y}^{i-} \mu_{\tilde{G}_{out}}^i + \sum_{i=R+1}^{K} \bar{y}^i \bar{\mu}_{\tilde{G}_{out}}^i\right)/ \left(\sum_{i=1}^{R} {}^{-}\mu_{\tilde{G}_{out}}^i + \sum_{i=R+1}^{K} \bar{\mu}_{\tilde{G}_{out}}^i\right) \quad (41)$$

where R and L are switch points satisfying

$$y^L \leqslant y_l \leqslant y^{L+1}, \ y^R \leqslant y_r \leqslant y^{R+1}$$

Since the approach of the search for centroid candidates (y_l, y_r) is computationally ineffective, an iterative method is to detect the optimal switching points as illustrated in Figure 8.

Defuzzification. After applying the TR method, the obtained IFS needs to be converted into a crisp number. However, this step is fairly straightforward, in which the defuzzified value can simply be obtained by calculating the average of the intervals left and right endpoints as follows:

$$y_{out} = \frac{y_r + y_l}{2} \quad (42)$$

SSO-based IT2FL control of 3-RRR planar Robot

The SSO algorithm takes the assumption that the search space is considered a communal web, the place for the communication of the social spiders. Each solution inside the search space express the position of the spider inside the communal web. A weight, compatible with the fitness value of the solution, is attributed to each spider. There are two variance search spiders in

the algorithm: males and females. It starts defining the number of female and male spiders inside the search space. The number N_f of females is selected randomly inside the domain of 65%–90% of the whole society N_s in chosen ancient times. Therefore, N_f can measure using the following[43–48]:

$$N_f = floor[(0.9 - 0.25 \times rand) \cdot N_s] \quad (43)$$

where $floor(.)$ maps a real number to an integer number, whereas $rand$ is a random number lies between (0, 1). The number of male spiders N_m is calculated based on N_f and N_s as follows[48]:

$$N_m = N_s - N_f \quad (44)$$

Thus, the N_s elements of the entire population S is splitted into two groups F and M, where F collects the female individuals ($f = \{f_1, f_2, \ldots, f_{Nf}\}$) whereas M groups the male members ($M = \{m_1, m_2, \ldots, m_{Nf}\}$), where $S = f \cup M$ ($S = \{s_1, s_2, \ldots, s_{Nf}\}$), such that $S = \{s_1 = f_1, s_2 = f_2, s_{Nf} = f_{Nf}, s_{Nf+1} = m_1, s_{Nf+2} = m_2, \ldots, s_N = m_{Nm}\}$.

Assignation of fitness

Each spider in the population S is quantified based on its fitness value and weight. The weight w_i assigned to each spider qualifies the spider i of the population. The weight of every spider is assessed based on the following equation[48]:

$$w_i = \frac{J(s_i) - (worst)_s}{(best)_s - (worst)_s} \quad (45)$$

where $J(s_i)$ is the fitness value evaluated at the spider position s_i concerning the objective function $J(\_)$. The values $(worst)_s$ and $(best)_s$ are defined by;

$$best_s = \max_{k \in (1, 2, \ldots N)} J_{(sk)} \quad (46)$$

$$worst_s = \min_{k \in (1, 2, \ldots N)} J_{(sk)} \quad (47)$$

Modeling of the vibrations

In the colony, the information among members is encoded based on small vibrations. Such vibrations are employed to initiate the spider and they are related to spider weight and the distance between spiders. The model of the vibration process can be described by;

$$Vb_{i,j} = w_j e^{-d_{i,j}^2}$$

where i and j denote the indices of spider individuals, $d_{i,j}$ is the Euclidian distance between the spiders j and i defined by[48]:

$$d_{i,j} = \|s_i - s_j\|$$

Initialization of population

The SSO algorithm is iterative and the complete population (females and males) should be initialized randomly. The procedure is started with the initialization of the position of the N spiders and the set S. The position f_i of each female spider (or male m_i) is represented by an n-dimensional vector, with parameters values that must be optimized. This value is randomly and uniformly distributed by initial parameters between predefined upper (PjHIGH) and lower (PJLOW) bounds, as represented by[43,48]:

$$f_{i,j}^0 = p_j^{low} + rand\left(p_j^{high} - p_j^{low}\right) \quad (48)$$

$$m_{i,j}^0 = p_j^{low} + rand\left(p_j^{high} - p_j^{low}\right) \quad (49)$$

where $i = 1, 2, \ldots, N_f, j = 1, 2, \ldots, n, k = 1, 2, \ldots, N_m$, and $f_{i,j}$ ($m_{i,j}$) represents the jth parameter of the ith female (male) spider position.

Cooperative operator

Female cooperative operator. Female spiders are attracted or repulsed by other spiders, it doesn't matter their gender, which is generally defined by their vibrations transmitted on the communal net and elements like cycle of reproduction, curiosity, and random phenomena.

The attraction movement is activated if a uniform random number r_m, generated within (0, 1), is smaller than a threshold PF. If this is satisfied, the attraction movement is produced based on the following formula[48]:

$$f_i^{k+1} = f_i^k + \alpha.V_i bc_i.\left(s_c - f_i^k\right) + \beta.V_i bb_i.\left(s_b - f_i^k\right) + \delta.(rand - 0.5) \quad (50)$$

Otherwise, a repulsion movement is generated as

$$f_i^{k+1} = f_i^k + \alpha.V_i bc_i.\left(s_c - f_i^k\right) - \beta.V_i bb_i.\left(s_b - f_i^k\right) + \delta.(rand - 0.5) \quad (51)$$

where β, α, δ, and $rand(.)$ are randomly generated within the interval $[0, 1]$, being t the number of the iteration. s_b represent the individual that is the nearest to the member i that has the uppermost weight and s_c is the best one in the population S.

Male cooperative operator. The social cooperation has biological features that allows the classification of the spider's male population into Dominant (*D*) or non-dominant (*ND*), which is performed in terms of the position of the median member. When it comes to weight and size, the *D* male spiders have fitness characteristics that are better than *ND* ones. Moreover, within the communal web, the closest female spider attracts the *D* males. A dominant (*D*) individual male is defined as a member with weight value that is higher than the male population median weight. On the other side, a non-dominant (*ND*) individual male is that one with

weight above the median value. The position change of male spider can be modeled using the following formula[43–48]:

$$m_i^{k+1} =$$
$$\begin{cases} m_i^k + \alpha.Vbf_i.\left(s_f - m_i^k\right) + \delta.(rand - 0.5) \ if \ w_{Nf+i} > w_{Nf+m} \\ m_i^k + \alpha.\left(\frac{\sum_{h=1}^{Nm} m_h^k.w_{Nf+h}}{\sum_{h=1}^{Nm} w_{Nf+h}} - m_i^k\right) \ if \ w_{Nf+i} \leqslant w_{Nf+m} \end{cases}$$
$$(52)$$

where w_{Nf+m} denotes the median male member, s_f is the nearest individual of female to the male member i, and $\frac{\sum_{h=1}^{Nm} m_h^k.w_{Nf+h}}{\sum_{h=1}^{Nm} w_{Nf+h}} - m_i^k$ represents the weighted mean of the male society M.

Mating operator

Mating in a social spider colony is done through dominant female male members. If the dominant male m_g spider ($g \in D$) locates a set E^g of female members within a specific range re (range of mating), it mates, forming a new brood s_{new} which is generated taking into account all the elements of the set T^g; that is, a new brood is generated by the union $E^g \cup^{m_g}$. The search area is limited by a radius re, which can be calculated according to[48]:

$$re = \frac{\sum_{j=1}^n (P_j^{high} - P_j^{low})}{2n} \qquad (53)$$

One aspect in the mating operation is that each spider involved in the process (elements inside T^g) has a weight defining the probability of impact of every individual on the created brood. Spiders with smaller weights have lower probability to affect the new product. The probability of influence P_{si} of each individual can be assigned by the roulette method[45]:

$$P_{si} = \frac{w_i}{\sum_{j \in T^k} w_j} \qquad (54)$$

where i $\in T^g$. After the shaping of the new spider, it can be performed a comparison between the worst spider s_{wo} and the novel candidate s_{new} in the community, considering their weights $w_{wo} = \underset{l \in (1,2,...N)}{min}(w_l)$. When the new spider is considered to be better than the worst one, it replaces the worst spider. If not, the new spider is discarded the society remains without changes. When there is a replacement, the new spider assumes the index and the gender of the replaced one. This ensure that the population S continues with the same original rate between female and male members. The SSO algorithm is described in the flowchart presented in Figure 9.

Design of optimal IT2FLC of planar 3-RRR parallel robot

When it comes to the Interval Type-2 PD-Like Fuzzy Logic Controller, it is considered the Mamdani system. So on, it will require seven input-output membership
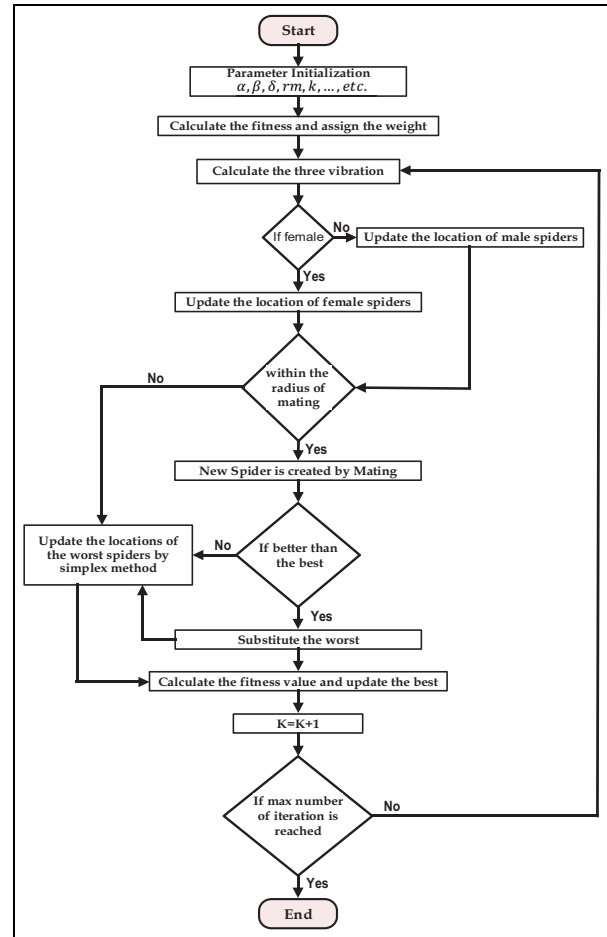

Figure 9. SSO algorithm in a flowchart.

functions of triangular type, as presented in Figure 10. The universe of discourse should be inside the range $(-1, 1)$ for both input and output. The chosen procedure for defuzzification is the Centroid. A rule-base for the proposed IT2FLC is presented in Table 1. A reduction method for the IT2FLC is implemented by the KM algorithm shown in Figure 8.

The SSO algorithm for the optimal tuning of the IT2FLC applied to the 3-RRR parallel manipulator was shown in Figure 11. This algorithm is responsible to tune and optimize nine gains (six inputs and three outputs) of the IT2FLC and also for the T1FLC. The desired positions (x_d, y_d) in the workspace are converted to the joint space $(\theta_{d1}, \theta_{d2}, \theta_{d3})$, that can be compared to the current angular positions $(\theta_1, \theta_2, \theta_3)$ in order to find the joint errors.

Computer simulation

Each controller structure has design parameters related to the SSO algorithm that should be defined by a trial-and-error procedure. However, the tuning criteria is based on the minimization of the tracking error. The MATLAB/Simulink (R2015b) environment was used for the simulations that allowed the examination of the

Table 1. Rule base of IT2FLC.

		e						
		NB	NM	NS	Z	PS	PM	PB
Δe	NB	PB	PB	PB	PB	PM	PS	Z
	NM	PB	PB	PB	PM	PS	Z	NS
	NS	PB	PM	PM	PS	Z	NS	NS
	Z	PM	PM	PS	Z	NS	NM	NM
	PS	PS	PS	Z	NS	NM	NM	NB
	PM	PS	Z	NS	NM	NB	NB	NB
	PB	Z	NS	NM	NB	NB	NB	NB
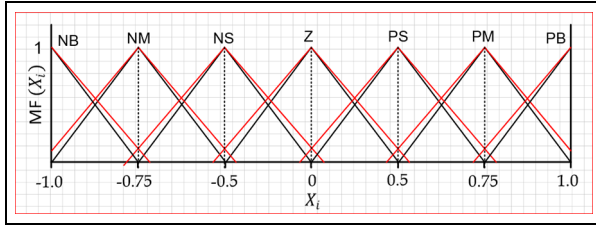
Figure 10. Input/output MFs.

Table 2. Parameters setting for the SSO algorithm.

Parameter	Value
Number of spiders	30
Dimension	9
Upper bound	50
Lower bound	0.01
Number of Iteration	30

controllers' effectiveness. These simulations were performed considering a sampling period of 0.5 ms.

The SSO algorithm is dedicated to optimization of the output and input gains of FLC. As it was defined

one single fuzzy logic structure for each active joint, the 3-RRR parallel manipulator has three fuzzy logic structures. Each FLC has three gains, which are the Proportional and Derivative input gains and the output gain. So, there are nine gains that the SSO algorithm is responsible to tune, which are the proportional gains (K_{Pi1}, K_{Pi2}, and K_{Pi3}), the derivative gains (K_{di1}, K_{di2}, and K_{di3}), and the output gains (K_{o1}, K_{o2}, and K_{o3}). The parameters settled for the SSO algorithm can be found on Table 2.

Tables 3 and 4 list the optimal values of total gains for T1FL and IT2FL controllers, which are tuned based on SSO algorithm.

Two scenarios are presented; one for the case of square trajectory without load application and the other with the applied load.

Case I: A square trajectory without disturbance

The steps of robot movement is performed in the following sequence:

The position of the robot gripper has to move from (0.1, 0.27) to (0.17, 0.27) m, and the end-effector goes from (0.17, 0.27) to (0.17, 0.33) m. So, the center of the gripper goes from (0.17, 0.33) m to (0.1, 0.33) m. Then, the end-effector goes from (0.1, 0.33) m to the initial
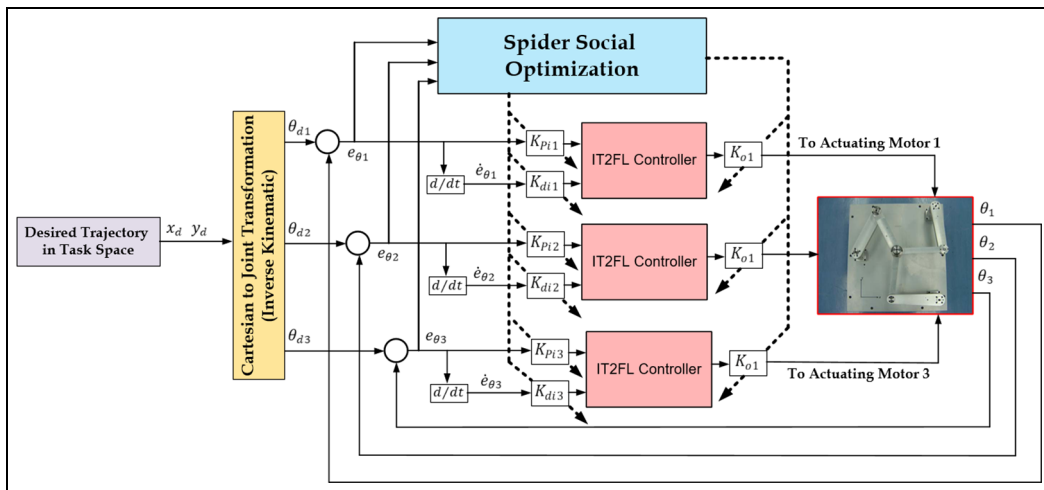
Figure 11. SSO based IT2FLC of planar 3-RRR parallel robot.

Table 3. Design parameters of the T1FL.

| Gain description	Symbol	Value
P input gain of FLC for active Joint one	K_{Pi1}	5.423
P input gain of FLC for active Joint two	K_{Pi2}	20.876
P input gain of FLC for active Joint three	K_{Pi3}	11.213
D input gain of FLC for active Joint one	K_{di1}	0.203
D input gain of FLC for active Joint two	K_{di2}	0.543
D input gain of FLC for active Joint three	K_{di3}	1.445
Output gain of FLC for active Joint one	K_{o1}	10.021
Output gain of FLC for active Joint two	K_{o2}	10.200
Output gain of FLC for active Joint three	K_{o3}	5.432

Table 4. Design parameters of the IT2FL.

Gain description	Symbol	Value
P input gain of FLC for active Joint one	K_{Pi1}	33.5731
P input gain of FLC for active Joint two	K_{Pi2}	35.2345
P input gain of FLC for active Joint three	K_{Pi3}	37.4066
D input gain of FLC for active Joint one	K_{di1}	1.5655
D input gain of FLC for active Joint two	K_{di2}	2.4370
D input gain of FLC for active Joint three	K_{di3}	0.2795
Output gain of FLC for active Joint one	K_{o1}	41.1167
Output gain of FLC for active Joint two	K_{o2}	0.279565
Output gain of FLC for active Joint three	K_{o3}	49.9957

position $(0.1, 0.27)$ m. However, the initial conditions begins at $x(0) = y(0) = 0.125m$.

Such a trajectory is suitable for high-velocity and high-acceleration robots. It is commonly adopted in applications in industry, like the assembly of electronic components, cutting by laser, and food packing. With this algorithm, continuity in acceleration is assured when dealing with robots of high-speed. Figure 12 illustrates the desired and actual trajectories along x, y axes, with both IT2FL and T1FL controllers. Figure 13 shows the tracking response of the end-effector in Cartesian coordinates, represented by $x(t)$ and $y(t)$.

The Cartesian errors e_x, and e_y based on T1FLC and Type-2 FLC are illustrated in Figure 14.

The performance in Cartesian space of T1FL and IT2FL controllers can be evaluated by the Root Mean Square Error (RMSE). The controller with the smaller RMSE value is considered the best one. The calculation of the RMSE in Cartesian space is given by:
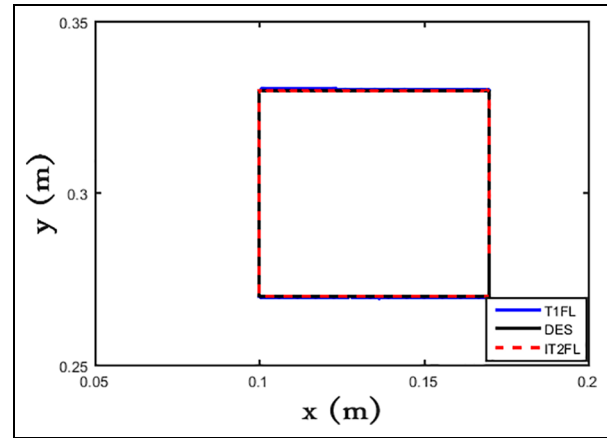
Figure 12. View of the desired square trajectory in the $x - y$ plane without disturbance.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(e_x^2(j) + e_y^2(j) \right)}$$

This RMSE value can be rewritten as:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(x^d(k) - x(k) \right)^2 + \left(y^d(k) - y(k) \right)^2}$$

being k the discrete-time sample related to the actual and desired trajectories, N the total number of samples, e_x and e_y the errors of tracking in both x and y axes. The RMSE values in Cartesian space computed considering the controllers T1FL and IT2FL are presented in Table 5.

The resulting RMSE values along the x and y axes are much better for the IT2FL controller when it is compared with the T1FL controller, as it is presented in Table 5. On the x axis, there is an improvement of 97.571%, while on the y axis the improvement is of 94.3227%.

The errors e_{θ_1}, e_{θ_2}, and e_{θ_3} of the active joints, considering both T1FL and IT2FL controllers, are presented in Figure 15. The RMSE value can be used again for the performance evaluation of these errors, being the best controller the one with the smaller RMSE value. The formula for the RMSE computation in joint space is:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(e_{\theta_1}^2(j) + e_{\theta_2}^2(j) + e_{\theta_3}^2(j) \right)}$$

This RMSE value in joint space can be rewritten as:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(\theta_1^d(j) - \theta_1(j) \right)^2 + \left(\theta_2^d(j) - \theta_2(j) \right)^2 + \left(\theta_3^d(j) - \theta_3(j) \right)^2}$$

being k the discrete-time sample related to the actual and desired trajectories, N the total number of samples, e_{θ_1}, e_{θ_2}, and e_{θ_3} the tracking errors along each joint. The
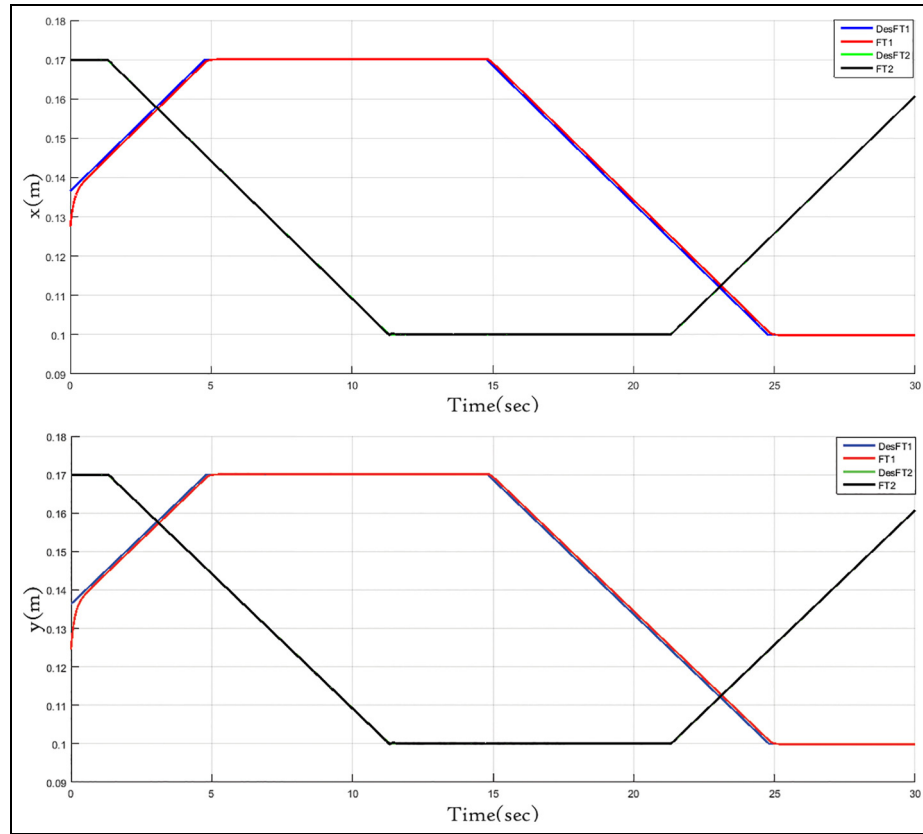
Figure 13. Trajectory tracking of end effector Cartesian positions $y(t)$ and $x(t)$ based on T1FL controller (red-colored line) and IT2FL controller (black-colored line).
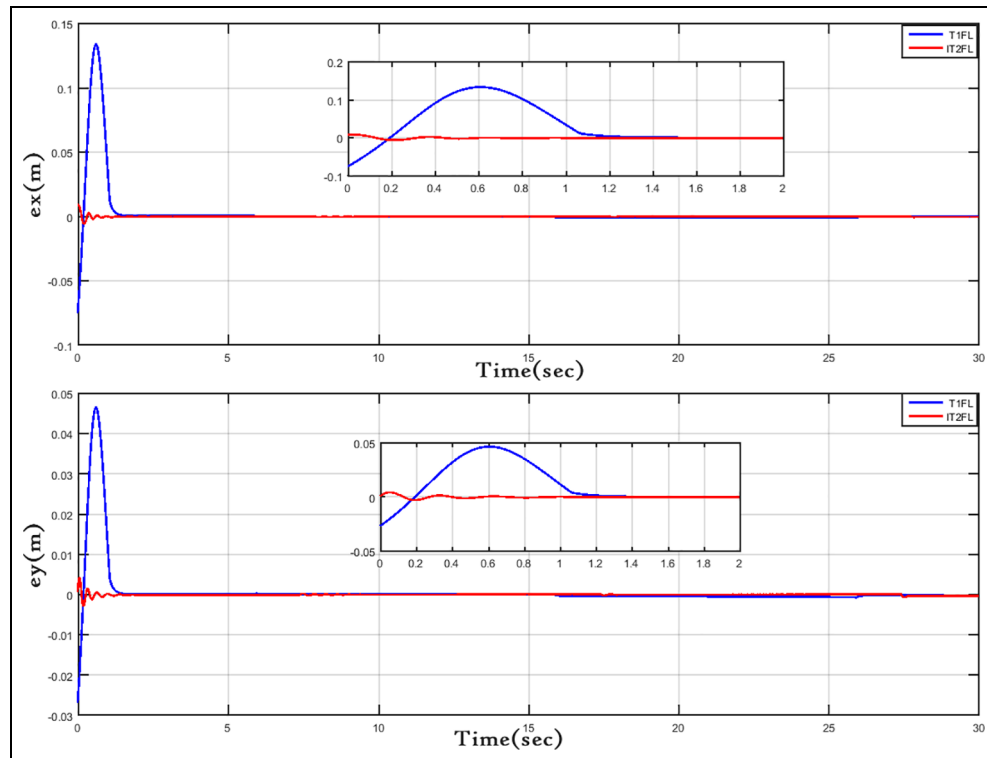
Figure 14. Cartesian position errors e_x and e_y based on the joint T1FL controller (blue-colored line) and the IT2Fl controller (red-colored line).
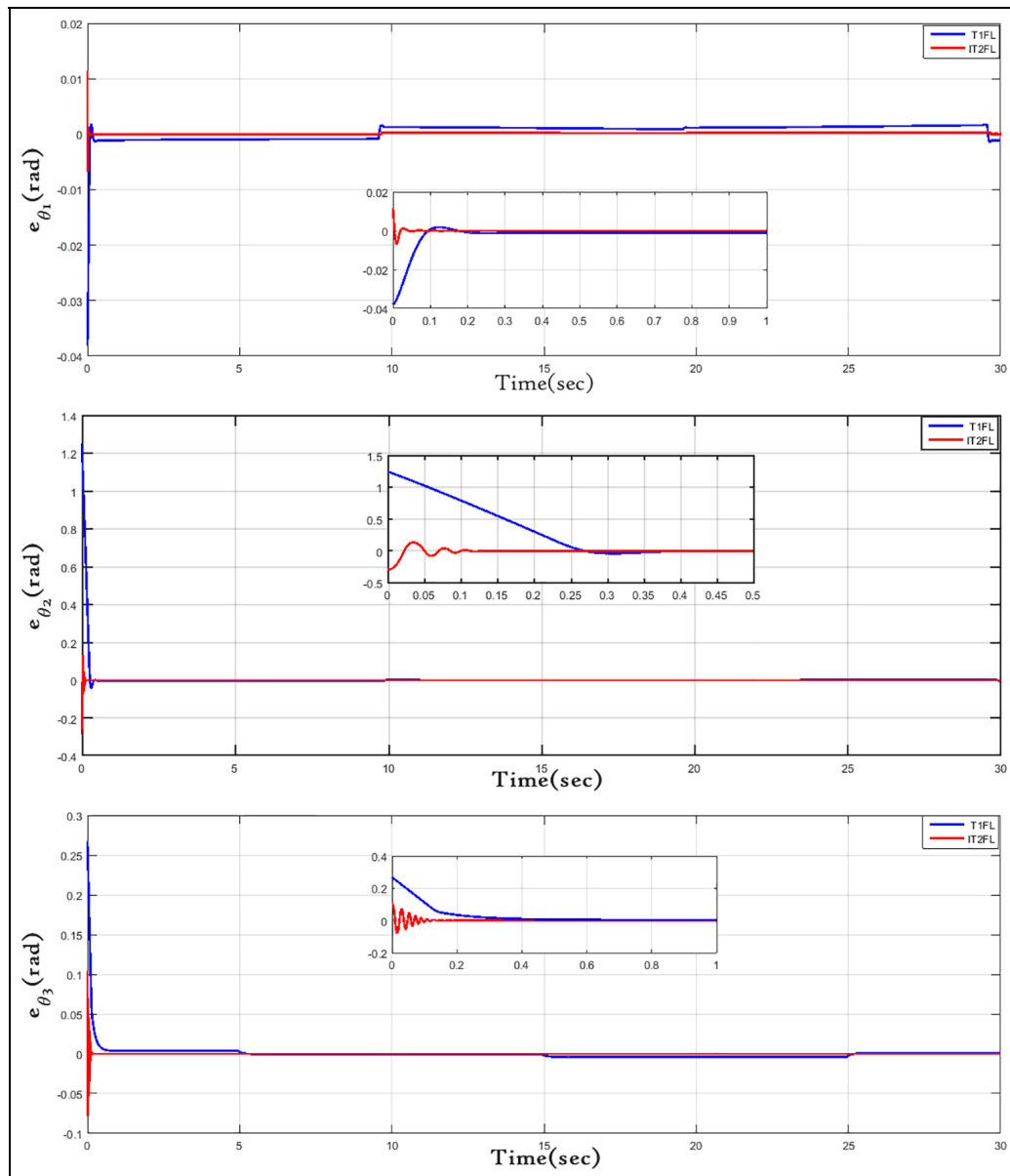
Figure 15. Joints errors of active joints based on T1FL controller (blue-colored line) and IT2FL controller (red-colored line).

Table 5. RMSE values in Cartesian space for IT2FL and T1FL controllers.

Cartesian position error	Type-1 FL controller	Interval Type-2 FL controller	Improvement %
X	0.1466	0.0036	97.571%
Y	0.5926	0.0336	94.3227%

Table 6. The RMS values of errors in joint space resulting from T1FL and IT2FL controllers.

Joint position errors	Type-1 FL controller	Type-2 FL controller	Improvement %
Along Joint 1	0.2952	0.1157	88.53%
Along with Joint 2	0.2688	6.543×10^{-2}	86.527%
Along Joint 3	0.0777	0.0083	89.3218%

RMSE values in joint space computed considering the controllers T1FL and IT2FL are presented in Table 6.

In joint space, the IT2FL controller provided better performance in terms of RMSE values, as it can be clearly seen in Table 6. The RMSE values for the T1FL controller are high and there is a significant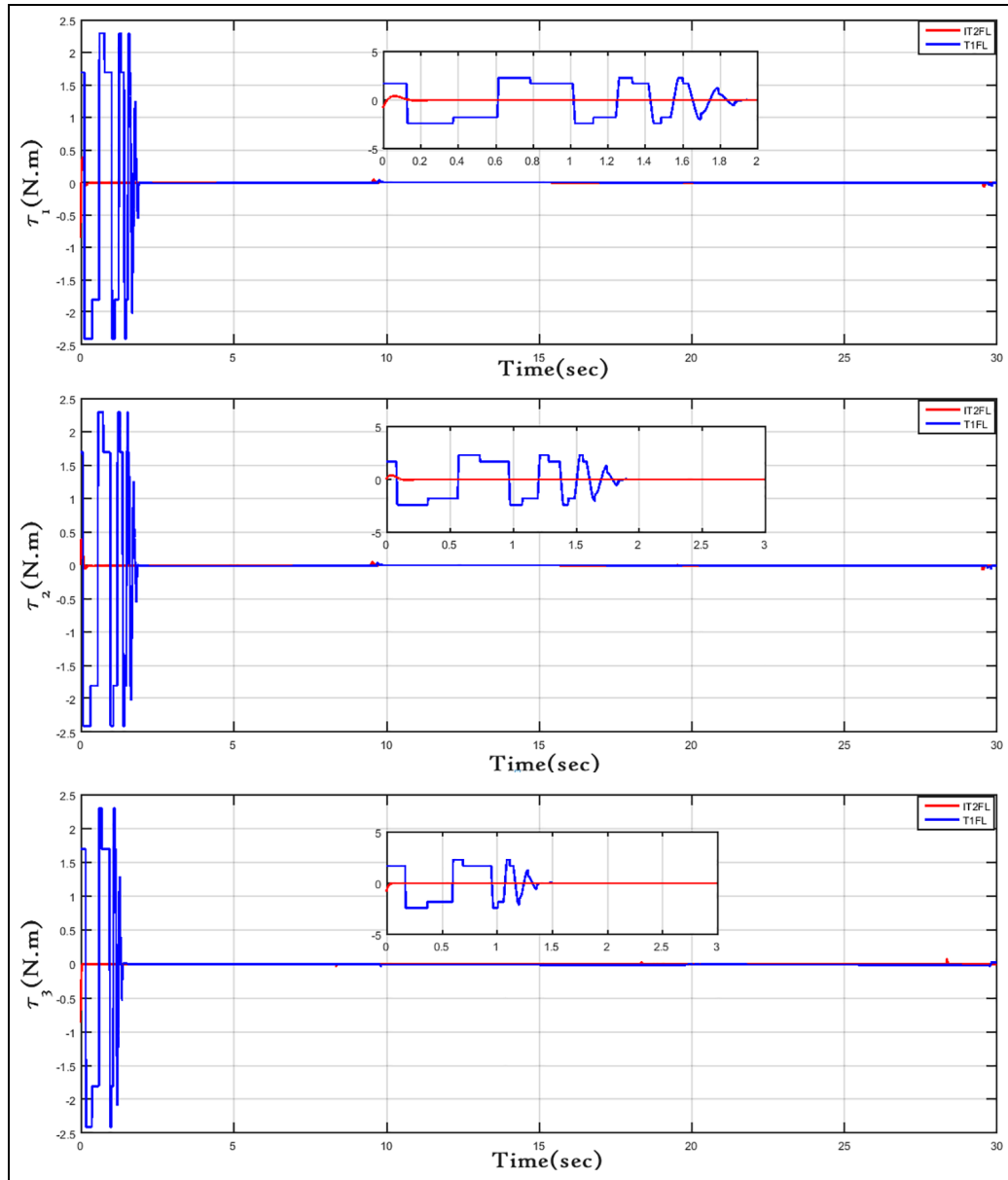 improvement when the IT2FL controller is considered. So, in both joint and Cartesian spaces, it can be concluded that the tracking performance is better for the IT2FL controller than for the T1FL controller.

The control signals, represented by the torques of the three actuator motors, are presented in Figure 16.

Figure 16. Torque responses actuated by parallel robot motors due to T1FL and IT2FL controllers in joint space.

Considering the same task for the system, first with the T1FL controller and then with the IT2FL controller, it can be clearly seen in Figure 16 that the IT2FL controller required a smaller torque than the T1FL controller. This means that the control effort provided by the IT2FL controller was smaller for a same task.

Case II: A square trajectory with disturbance

The steps of robot movement is performed in the following sequence:

The position of the robot gripper has to move from (0.1, 0.27) to (0.17, 0.27) m, and the end-effector goes from (0.17, 0.27) to (0.17, 0.33) m. So, the center of the gripper goes from (0.17, 0.33) m to (0.1, 0.33) m. Then, the end-effector goes from (0.1, 0.33) m to the initial
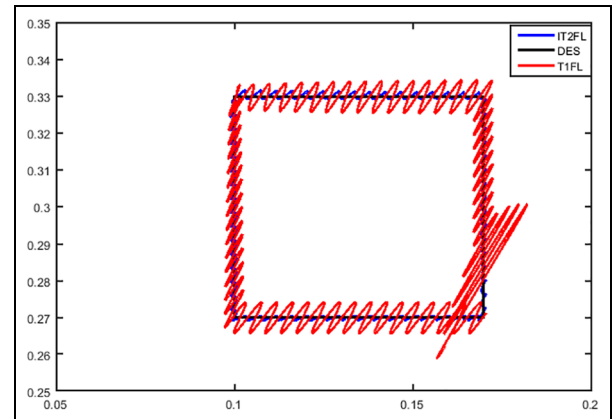

Figure 17. Top view of the desired square trajectory in the *x-y* plane with disturbance.
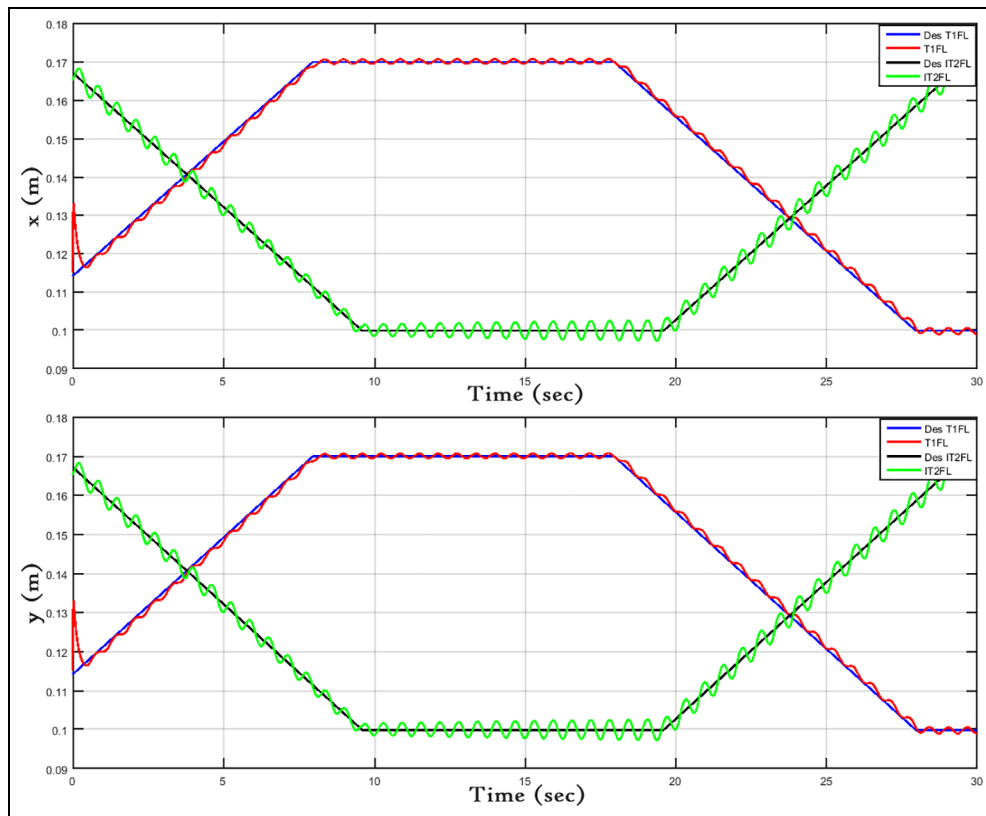
Figure 18. Trajectory tracking of end effector Cartesian positions $y(t)$ and $x(t)$ based on T1FL controller (red-colored line) and IT2FL controller (black-colored line).
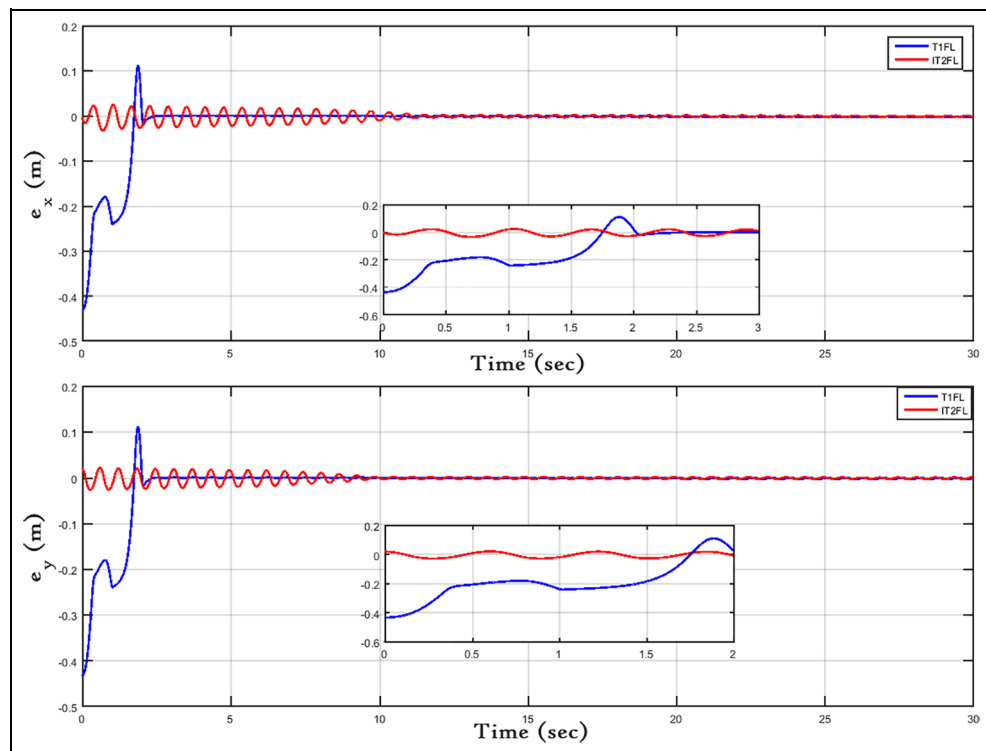
Figure 19. Cartesian position errors e_x and e_y based on the joint T1FL controller (red-colored line) and the IT2FL controller (blue-colored line).
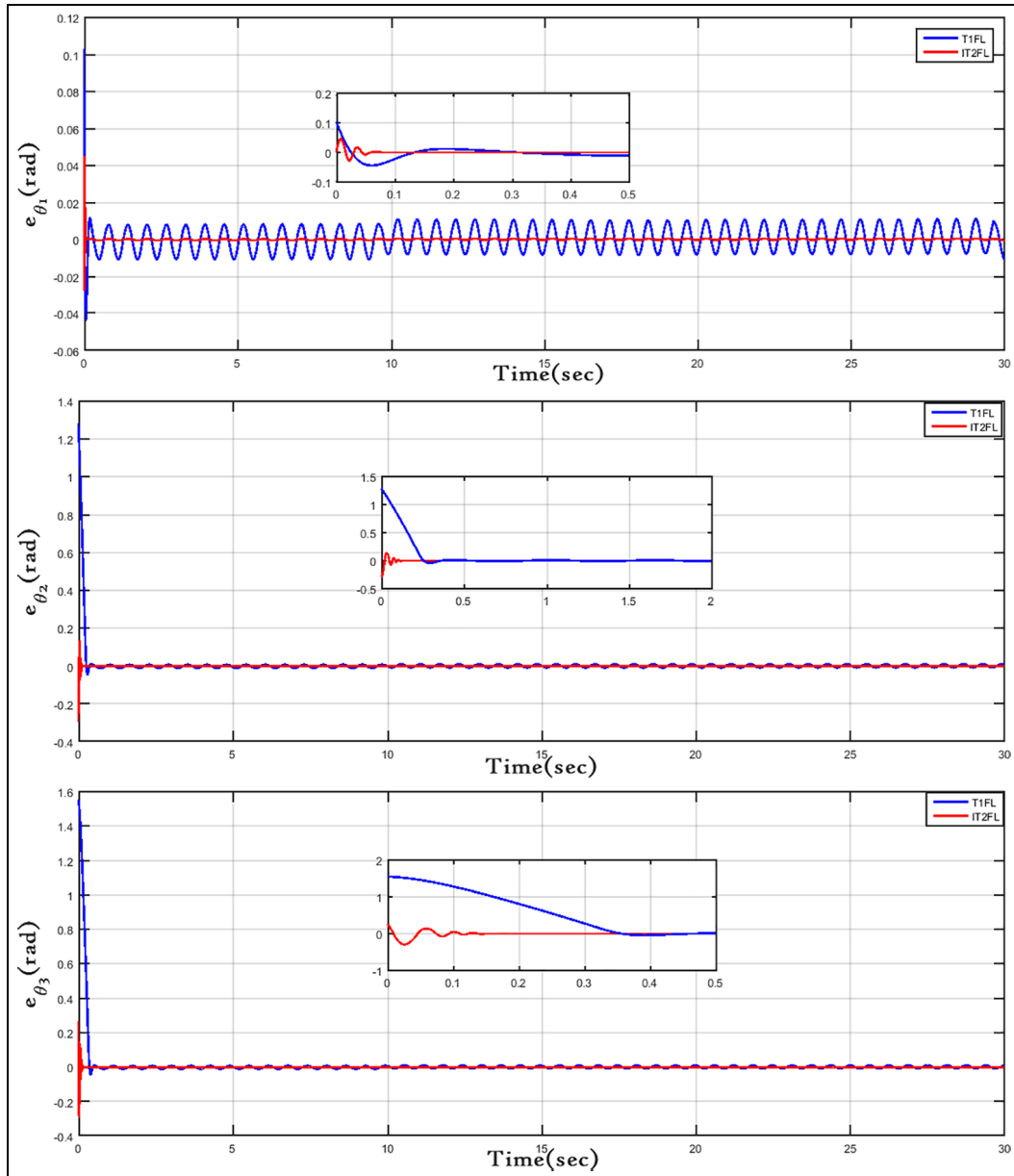
Figure 20. Position and joints errors based on T1FL controller in joint space (blue line) and IFT2 controller in joints (red line).

Table 7. The values of RMS errors in Cartesian space resulting in fromT1FL and IT2FL controllers.

Cartesian position error	Type-1 FL controller	Interval Type-2 FL controller	Improvement %
X	0.2564	0.0264	89.7035%
Y	0.5756	0.0954	83.4320%

Table 8. The RMS values of errors in joint space resulting from T1FL and Interval Type-2 FL controllers with disturbance.

| Joint position errors	Type-1 FL controller	Type-2 FL controller	Improvement %
Along with Joint 1	0.3524	0.1225	78.478%
Along with Joint 2	0.5692	0.2109	81.955%
Along with Joint 3	0.3652	6.4217×10^{-2}	81.777%

position $(0.1, 0.27)$ m. However, the initial conditions begins at $x(0) = y(0) = 0.125m$.

For this case, the robustness to external disturbances will be considered in the performance evaluation. So, a harmonic torque disturbance given by $0.5sin(10t)$ $N.m$ will be applied in each actuated joint. The robustness is quantitatively evaluated by the calculated variance between the nominal response and the deviated one that arises from the application of the external disturbance. For the purpose of comparison, a controller providing a response with smaller variance can be
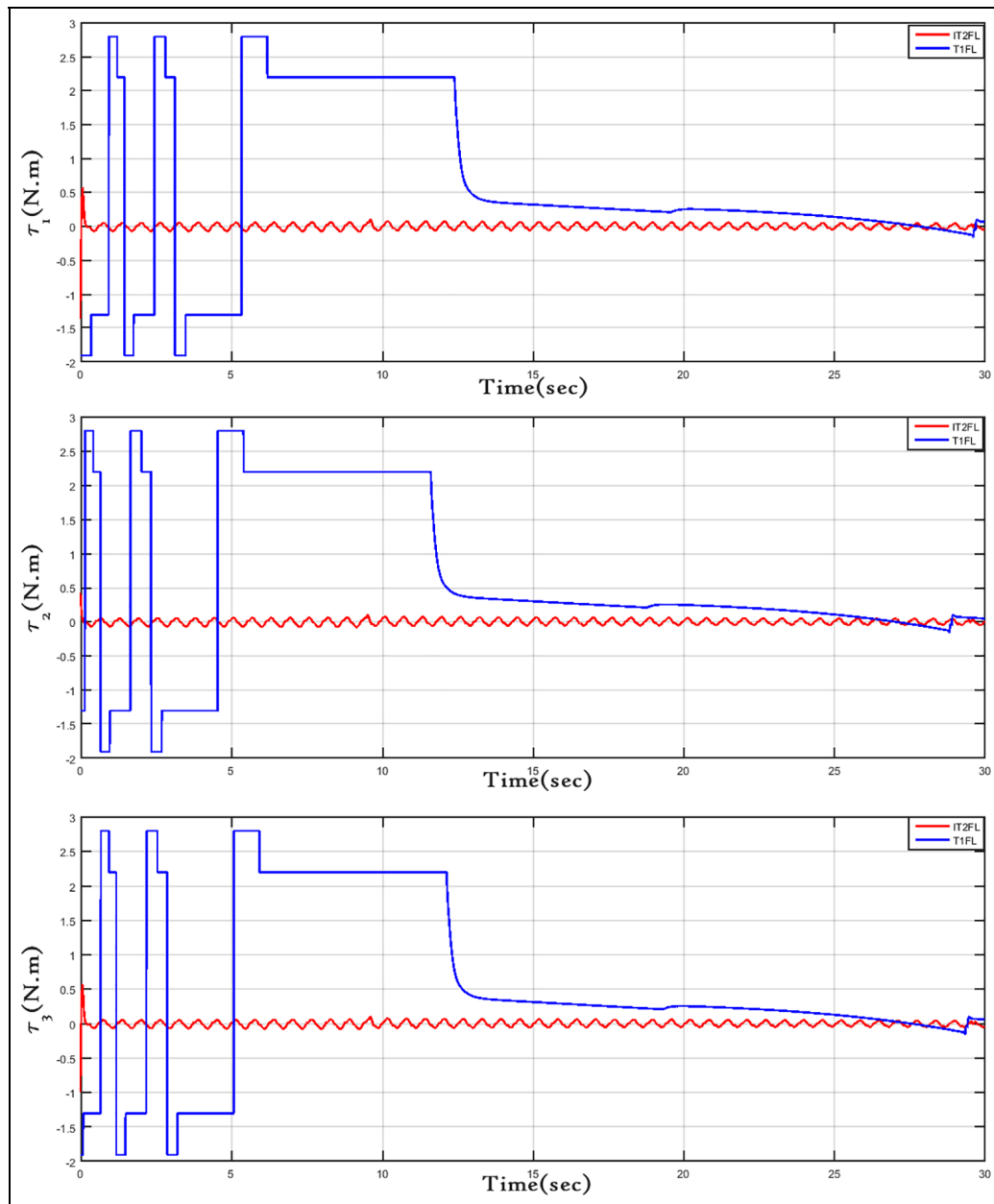
Figure 21. Torque responses actuated by robot motor due to T1FL and IT2FL controllers in joint space.

considered more robust than the others. The tracking of a square trajectory, considering the harmonic disturbance, is shown in Figure 17.

The actual and desired trajectories in a square are presented in Figure 18, for the system with both T1FL and IT2FL joint controllers under effect of external harmonic disturbances. It can be seen that, under disturbances, the IT2FL controller provided a transient response that remains relatively smooth and unchanged in terms of tracking, whereas the transient response with the T1FL controller is highly rippled and distorted. This primary analysis indicates a more robust IT2FL controller than the T1FL controller.

The position errors in Cartesian space, e_x and e_y, are shown in Figure 19 for the T1FL and IT2FL controllers. It can be seen that the tracking performance with

the IT2FL controller is better, in terms of robustness, when the external harmonic disturbance is applied. The RMSE value can be used to evaluate the tracking performance of the system with this trajectory, that is different from the previous one, and under disturbances. The calculated RMSE values are presented in Table 7, showing that there is an improvement in the tracking performance with the IT2FL controller over the T1FL controller.

The errors e_{θ_1}, e_{θ_2}, and e_{θ_3} relative to the active joints, considering external harmonic disturbances and both IT2FL and T1FL controllers, are presented in Figure 20. It can be visually seen that the IT2FL controller provided smaller errors, in transient and also steady-state response. In order to properly quantify the performance in joint space, the RMSE values are

presented in Table 8. The IT2FL controller provided an improved performance in terms of RMSE, when compared with the T1FL controller. So, it can be said that the IT2FL controller outperformed the T1FL controller in both Cartesian and joint spaces.

The control signals, represented by the torques of the three actuator motors, are presented in Figure 21. Considering the same task for the system, first with the T1FL controller and then with the IT2FL controller, it can be clearly seen in Figure 21 that the IT2FL controller required a smaller torque than the T1FL controller. This means that the control effort provided by the IT2FL controller was smaller for a same task.

Conclusions

It has been shown that the SSO algorithm is a viable alternative for the optimal tuning of PD design parameters in both T1FL and IT2FL controllers, avoiding traditional time consuming and tedious trial-and-error procedures. Moreover, it was verified that this optimization strategy confirms that, in general, the IT2FL controller provides better results than the T1FL controller.

Considering no exogenous disturbance, it was achieved a better performance, in both joint and Cartesian spaces, with the IT2FL controller when compared with the T1FL controller. Moreover, the control efforts, in terms of torque, are smaller for the IT2FL controller in comparison with the T1FL controller.

When exogenous disturbances are considered, the simulations presented that a better tracking behavior was achieved, in terms of RMSE, with the IT2FL controller when compared with the T1FL controller. Also, the IT2FL controller provided lower control effort and presented more robust characteristics than the T1FL controller.

This study can be implemented in real-time to verify experimentally the effectiveness of proposed controller. In addition, other extension of the work can be made by including other optimization techniques for the sake of comparison.[49–55] Cooperative control can be developed for this type of 3-RRR planar parallel robot for the surgical application.[12,14,56]

Authors' Note

Ahmad Taher Azar is also affiliated with College of computer and Information Sciences, Prince Sultan University, Riyadh.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

ORCID iDs

Amjad J Humaidi https://orcid.org/0000-0002-9071-1329
Ayad Q Al-Dujaili https://orcid.org/0000-0002-1126-3290
Ibraheem Kasim Ibraheem https://orcid.org/0000-0001-7009-3634

References

1. Amjad JH and Hanan AH. Adaptive control of parallel manipulator in Cartesian space. In: *IEEE international conference on electrical, computer and communication technologies (ICECCT)*, Coimbatore, India, 20–22 February 2019.
2. Merlet JP. *Parallel robots*. 2nd ed. Sophia-Antipolis. Springer, 2006.
3. Humaidi AJ and Abdulkareem AI Design of augmented nonlinear PD controller of delta/Par4-like robot. *J Control Sci Eng* 2019; 2019: DOI:10.1155/2019/7689673.
4. Kuen YY. *Geometry, dynamic and control of parallel manipulator*. PhD Thesis, Hong Kong University, Hong Kong, 2002.
5. Liu GF, Wu YL and Li ZX. Analysis and control of redundant parallel manipulators. In: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*, Seoul, South Korea, 21–26 May 2001, vol. 4, pp.3748–3754. New York: IEEE.
6. Humaidi AJ, Oglah A.A, Saad JA, et al. Optimal augmented linear and nonlinear PD control design for parallel robot based on PSO tuner. *Int Rev Model Simul* 2019; 12(5): 281. DOI: 10.15866/iremos.v12i5.16298
7. Su H, Qi W, Yang C, et al. Deep neural network approach in human-like redundancy optimization for anthropomorphic manipulators. *IEEE Access* 2019; 7: 124207–124216.
8. Bulgarelli A, Toscana G, Russo LO, et al. A low-cost open source 3D-printable dexterous anthropomorphic robotic hand with a parallel spherical joint wrist for sign languages reproduction. *Int J Adv Robot Syst* 2016; 13: 126.
9. Fujimoto I, Matsumoto T, Ravindra P, et al. Mimicking and evaluating human motion to improve the imitation skill of children with autism through a robot. *Int J Soc Robot* 2011; 3: 349–357.
10. Jiang L, Gao B and Zhao J. Kinematic and static analysis of a cable-driven parallel robot with a flexible link spine. In: *Proceedings of the 2015 IEEE conference on robotics and biomimetics*, Zhuhai, China, 6–9 December, 2015.
11. Ren B, Liu J, Luo X, et al. On the kinematic design of anthropomorphic lower limb exoskeletons and their matching movement. *Int J Adv Robot Syst* 2019; 16(5): 1–9.
12. Su H, Sandoval J, Vieyres P, et al. Safety-enhanced collaborative framework for tele-operated minimally invasive surgery using a 7-DoF torque-controlled robot. *Int J Control Autom Syst* 2018; 16: 2915–2923.
13. Avgousti S, Christoforou EG, Panayides AS, et al. Medical telerobotic systems: current status and future trends. *Biomed Eng Online* 2016; 15: 96.
14. Su H, Danioni A, Mira RM, et al. Experimental validation of manipulability optimization control of a 7-DoF serial manipulator for robot-assisted surgery. *Int J Med Robot* 2021; 17: e2193.

15. Ceccarelli M, Russo M and Morales-Cruz C. Parallel architectures for humanoid robots. *Robotics* 2020; 9: 75.

16. Zhang YX, Cong S, Shang WW, et al. Modeling, identification and control of a redundant planar 2-DOF parallel manipulator. *Int J Control Automat Syst* 2007; 5: 559–569.

17. Cheng H, Yiu YK and Li Z. Dynamics and control of redundantly actuated parallel manipulators. *IEEE/ASME Trans Mechatron* 2003; 8(4): 483–491.

18. Yu R, Zhao H, Zhen S, et al. A novel approach for 2-degrees of freedom redundant parallel manipulator dynamics. *Adv Mech Eng* 2017; 9(6): 1–12.

19. Sariyildiz E, Ucak K, Oke G, et al. A trajectory tracking application of redundant planar robot arm via support vector machines. In: Bouchachia A (ed.) *Adaptive and intelligent systems. ICAIS 2011. Lecture notes in computer science*, vol. 6943. pp 192–202 Berlin, Heidelberg: Springer, 2011.

20. Al-Mayyahi A, Aldair AA and Chatwin C. Control of a 3-RRR planar parallel robot using fractional order PID controller. *Int J Autom Comput* 2020; 17: 822–836.

21. Liu GF, Wu YL, Wu XZ, et al. Analysis and control of redundant parallel manipulators. In: *Proceedings of the 2001 IEEE international conference on robotics & automation*, Seoul, Korea, 21–26 May 2001.

22. Shang W and Cong S. Robust nonlinear control of a planar 2-DOF parallel manipulator with redundant actuation. *Robot Comput Integr Manuf* 2014; 30: 597–604.

23. Sheng L and Li W. Optimization design by genetic algorithm controller for trajectory control of a 3-RRR parallel robot. *Algorithms* 2018; 11(1): 7.

24. Moezi SA, Rafeeyan M, Zakeri E, et al. Simulation and the experimental control of a 3-RPR parallel robot using optimal fuzzy controller and fast on/off solenoid valves based on the PWM wave. *ISA Trans* 2016; 61: 265–286.

25. Noshadi A, Mailah M and Zolfagharian A. Active force control of 3-RRR planar parallel manipulator. In: *Proceedings of the IEEE international conference on mechanical and electrical technology*, Singapore, 10–12 September 2010, pp.77–81. New York: IEEE.

26. Abed HY, Humod AT and Humaidi AJ. Type 1 versus Type 2 fuzzy logic speed controllers for brushless DC motors. *Int J Electr Comput Eng* 2020; 10(1): 265–274.

27. Prabhakar G, Selvaperumal S and Nedumal Pugazhenthi P. Fault data injection attack on car-following model and mitigation based on interval type-2 fuzzy logic controller. *IET Cyber Phys Syst Theory Appl* 2019; 4(2): 128–138.

28. Stan SD., Gogu G., Manic M., Balan R., Rad C. (2010) Fuzzy Control of a 3 Degree of Freedom Parallel Robot. In: Iskander M., Kapila V., Karim M. (eds) *Technological Developments in Education and Automation*. Springer, Dordrecht. https://doi.org/10.1007/978-90-481-3656-8_79.

29. Prabhkar G, Selvaperumal S and Nedumal Pugazhenthi P. Fuzzy PD plus I control-based adaptive cruise control system in simulation and real-time environment. *IETE J Res* 2019; 65(1):69–79.

30. Noshadi A, Mailah M and Zolfagharian A. Intelligent active force control of a 3-RRR parallel manipulator incorporating fuzzy resolved acceleration control. *Appl Math Model* 2012; 36(6): 2370–2383.

31. Lu XG and Liu M. Optimal design and tuning of PID-type interval type-2 fuzzy logic controllers for delta parallel robots. *Int J Adv Robot Syst* 2016; 13(3): 96.

32. Lu XG, Liu M and Liu JX. Design and optimization of interval type-2 fuzzy logic controller for delta parallel robot trajectory control. *Int J Fuzzy Systems* 2017; 19(1): 190–206.

33. Passino KM and Yurkovich S. *Fuzzy control*. Addison-Weslwey Longman Inc, 1998. Menlo Park, California.

34. Wu D and Mendel JM. Approaches for reducing the computational cost of interval Type-2 fuzzy logic systems: overview and comparisons. *Inf Sci* 2013; 21: 80–90.

35. Ozek MB and Akpolat ZH. A software tool: Type-2 fuzzy logic toolbox. *Comput Appl Eng Educ* 2008; 16(2): 137–146.

36. Mendel JM, John RI and Liu F. Interval Type-2 fuzzy logic systems made simple. *IEEE Trans Fuzzy Syst* 2006; 14: 808–821.

37. Jerry MM, Hagras H, Tan WW, et al. *Introduction to Type-2 fuzzy logic control: theory and applications*. John Wiley & Sons, 2014. United States.

38. Bingül Z and Karahan O. A fuzzy logic controller tuned with PSO for 2-DOF robot trajectory control. *Expert Syst Appl* 2011; 38(1): 1017–1031.

39. Humaidi AJ and Hasan AF. Particle swarm optimization–based adaptive super-twisting sliding mode control design for 2-degree-of-freedom helicopter. *Meas Control* 2019; 52: 1403–1419.

40. Premkumar K and Manikandan BV. Bat algorithm optimized fuzzy PD based speed controller for brushless direct current motor. *Eng Sci Tech* 2011; 19(2): 8018–8040.

41. Humaidi AJ and Hameed M. Development of a new adaptive backstepping control design for a non-strict and under-actuated system based on a PSO tuner. *Information* 2019; 10(2): 38 https://doi.org/10.3390/info10020038

42. Wu D and Mendel JM. Approaches for reducing the computational cost of interval Type-2 fuzzy logic systems: overview and comparisons. *Inf Sci* 2013; 21: 80–90.

43. Al-Azza AA, Al-Jodah AA and Harackiewicz FJ. Spider monkey optimization: a novel technique for antenna optimization. *IEEE Antennas Wirel Propag Lett* 2016; 15: 1016–1019.

44. Zhao Z, Ruxin A, Luo Q, et al. Elite opposition-based social Spider optimization algorithm for global function optimization. *Algorithms* 2007; 10(1): 9.

45. Zhou Z, Yongquan M, Zhou Y, et al. A simplex method-based social spider optimization algorithm for clustering analysis. *Eng Appl Artif Intell* 2017; 64: 67–82.

46. Gupta K., Deep K. (2016) Tournament Selection Based Probability Scheme in Spider Monkey Optimization Algorithm. In: Kim J., Geem Z. (eds) *Harmony Search Algorithm*. Advances in Intelligent Systems and Computing, vol 382. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-47926-1_23

47. Kien LC, Nguyen TT, Hien CT, et al. A novel social spider optimization algorithm for large-scale economic load dispatch problem. *Energies* 2019; 12: 1075.

48. Al-Dujaili AQ, Falah A, Humaidi AJ, et al. Optimal super-twisting sliding mode control design of robot manipulator: design and comparison study. *Int J Adv Robot Syst*. Epub ahead of print November 2020. DOI: 10.1177/1729881420981524.

49. Gao ZM and Zhao J. An improved grey wolf optimization algorithm with variable weights. *Comput Intell Neurosci* 2019; 2019: 2981282.

50. Humaidi AJ, Kadhim SK and Gataa AS. Development of a novel optimal backstepping control algorithm of magnetic impeller-bearing system for artificial heart ventricle pump. *Cybern Syst* 2020; 51(4): 521–541. https://doi.org/10.1080/01969722.2020.1758467

51. Prabhakar G, Selvaperumal S, Nedumal Pugazhenthi P, et al. Online optimization based model predictive control on two wheel Segway system. *Mat Today* 2020; 33(Part 7): 3846–3853.

52. Nasiri J and Khiyabani FM. A whale optimization algorithm (WOA) approach for clustering. *Cogent Math Stat* 2018; 5: 1.

53. Mirjalili S. A sine cosine algorithm for solving optimization problems. *Knowl Based Syst* 2016; 96: 120–133.

54. Su H, Yang C, Ferrigno G, et al. Improved human-robot collaborative control of redundant robot for teleoperated minimally invasive surgery. *IEEE Robot Autom Lett* 2019; 4(2): 1447–1453.

55. Ibraheem K. I., Ajeil F. H, Path Planning of an autonomous Mobile Robot using Swarm based Optimization Techniques, *Al- Kawarizmi Engineering Journal*, Vol. 12, No. 4, pp. 12–25, 2016.

56. Humaidi AJ; Badr HM; Hameed A H, PSO-Based Active Disturbance Rejection Control for Position Control of Magnetic Levitation System, *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 922–928, doi:10.1109/CoDIT.2018.8394955.