IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 7, 2020, accepted October 11, 2020, date of publication October 21, 2020, date of current version November 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3032715

Artificial Bee Colony Optimization Algorithm Incorporated With Fuzzy Theory for Real-Time Machine Learning Control of Articulated Robotic Manipulators

HSU-CHIH HUANG, (Senior Member, IEEE), AND CHUN-CHIA CHUANG
Department of Electrical Engineering, National Ilan University, Yilan 260, Taiwan

Corresponding author: Hsu-Chih Huang (hchuang@niu.edu.tw)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 109-2221-E-197-032.

ABSTRACT This paper presents a real-time machine learning control (MLC) of articulated robotic manipulators using artificial bee colony optimization (ABC) algorithm incorporated with fuzzy theory. The modified ABC with dynamic weight is used to optimize the fuzzy structure and fractional order. The fractional parameters, fuzzy membership functions and rule base are determined by means of the ABC computation. This ABC-fuzzy hybrid learning algorithm is applied to real-time MLC of robotic manipulators by including fractional order proportional-integral-derivative (FOPID) control strategy. The MLC's control gain parameters are online tuned via the ABC-fuzzy optimization. With the kinematics analysis of a six-degree-of-freedom (DOF) articulated arm via reverse coordinates approach, an ABC-fuzzy MLC is developed to achieve motion control. A real-time operating system (RTOS) on a microprocessor collaborates with the ABC-fuzzy MLC to meet critical timing constraint by considering the dynamics of actuators. Finally, the mechatronic design and experimental setup of a six-DOF articulated robotic manipulator are constructed. Experimental results and comparative works are provided to demonstrate the merit of the proposed methods. Compared with the conventional control schemes, the proposed ABC-fuzzy MLC has theoretical and practice significance in term of real-time capability, online parameter tuning, convergent behavior and hybrid MLC. The proposed MLC methodologies are applicable to designing real-time modern controllers in both industry and academia.

INDEX TERMS Artificial bee colony optimization, fuzzy theory, machine learning control, robotic arm.

I. INTRODUCTION

Real-time systems have grown in demand in the market especially in industrial environments [1], [2]. They are time constrained and deterministic systems that respond within a specified time frame. To date, they are used to develop modern feedback control strategies that control actuators and sensors for industrial applications. Real-time control is one of the most common applications of real time systems. Although numerous real-time control methods have been presented and achieved exciting results [1]–[4], the machine learning technology is not included for high performance control. More importantly, they are still restricted with constant control gain, which may not be applicable for real-world engineering domains under time-varying uncertainties [5].

Machine learning is a subfield of artificial intelligence (AI) that provides systems the ability to automatically learn from data and improve through experience. This incredible form of artificial intelligence is already being used in various industries and professions where it is difficult or infeasible to develop conventional algorithms to perform the desired tasks [5]–[9]. By learning the previous patterns, it will learn the new process and execute the knowledge. This approach has proven to be one of the most challenging field of the past decade and is making a significant impact on the industry domains, including regression, image and speech recognition, medical diagnosis and etc.

MLC is a specific application of machine learning that employs data-driven methods for control design [10]–[14].

The associate editor coordinating the review of this manuscript and approving it for publication was Aysegul Ucar.

It is also a branch of intelligent control theory which solves optimal control problems using machine learning algorithms. This methodology has become particularly powerful for complex systems with strong nonlinearities where the conventional control schemes are not applicable. Taking the advantages of machine learning and intelligent control, MLC has tremendous potential and is a relatively new frontier in data-driven engineering [13], [14].

Evolutionary algorithms are powerful regression techniques because of their generality in optimizing both the structure and gain parameters associated with a controller [15]. Generally speaking, they provide an effective alternative search strategy to find optimal solutions in a high-dimensional search space [16], [17]. These computing paradigms form an important category of machine learning techniques that adapt and optimize through a nature-inspired process [14], [17]. They have been successful in many diverse control applications, including system identification, parameter tuning and optimal control. This methodology is a rapidly developing field at the intersection of control engineering and natural sciences [15]–[19].

Evolutionary MLCs are now pervading fields of academic and industrial research [20]. MLC feedback control laws with biological processes provide many advantages and is a challenging research in advanced intelligent control field. Some popular evolutionary algorithms, such as genetic algorithm (GA), ant colony optimization (ACO) and particle swarm optimization (PSO) are widely applied to design advanced MLCs. These algorithms are already central in control design and provide context for modern MLCs [20]–[24].

Among traditional metaheuristics, ABC is particularly promising for MLCs due to its strong global search capability [25]–[27]. ABC is one of the swarm intelligence algorithms that mimic honey bees' food search behavior to drive a search towards the optimization solution. This algorithm is proposed by Karaboga based on a particular foraging behavior of honey bee colonies on finding nectar and sharing the information of food sources to the bees in the hive [26], [27]. As the authors' best understanding, there has no attempt to design ABC-based real-time MLCs for robotic manipulators.

ABC algorithm has been shown to be competitive with other conventional bioinspired algorithms to solve complex engineering problems [28], [29]. However, this evolutionary computation has the difficulty of local optimum caused by the unbalance between global and local search. Some strategies have been employed to develop ABC variants to further improve performance [28]–[30]. Overall, these modified ABC algorithms are computationally extensive and not applicable to high-dimension optimization problems. This paper proposes an efficient self-adaptive ABC to not only increase the search diversity but also balance the exploitation and exploration. Moreover, this modified ABC is incorporated with fuzzy theory to present a ABC-fuzzy real-time MLC.

Compared with the conventional offline control schemes, real-time MLCs feature online tuning capability for engineering applications. In other words, the control gain parameters associated with the MLC are self-adjusted at every sampling, thus obtaining optimal performance. In this study, the metaheuristic ABC is utilized to optimize both fractional order parameters and fuzzy structure. This ABC-fuzzy approach is then employ to develop a real-time online FOPID MLC using a RTOS. The control parameters are online tuned by means of fuzzy logic. Taking the advantages of ABC-fuzzy MLC, online tuning and real-time control, the proposed ABC-fuzzy FOPID MLC is applied to industrial articulated robotic manipulators.

The articulated manipulator is an industrial robot which consist of mechanical links connected by rotating joints [31]. This kind of robot arm is ideally suited to industrial automation, such as material handling, assembly operations, welding, spray painting and medical surgery [31]–[33]. The main advantages of articulated robots are the flexibility and dexterity. They are extensively used to perform repetitive tasks with greater speed and consistency in the industrial manufacturing sector. Although numerous studies have been presented to cope with the design and control problems of articulated arms, the real-time optimal MLC deserves further investigations.

The reminder of this paper is organized as follows. Section II presents the hybrid ABC-fuzzy FOPID machine learning that includes ABC-fuzzy optimization and FOPID control theory. Section III describes the ABC-fuzzy real-time online MLC of robotic manipulators. Experimental results and comparative analysis are given in Section IV. Finally, Section V concludes this paper.

II. EVOLUTIONARY ABC-FUZZY FOPID MACHINE LEARNING CONTROL

A. SELF-ADAPTIVE ABC MACHINE LEARNING

1) ABC ALGORITHM

In nature, the exchange of information among honey bees is the most important occurrence in the formation of the collective knowledge [28]. Communication among bees related to the quality of food sources in the dancing area. ABC was inspired by this intelligent foraging behavior of honey bees. It is a swarm based metaheuristic algorithm for optimizing numerical problems. To date, this robust algorithm has been successfully utilized to solve combinatorial problems in science and engineering domains. It has been proven an efficient tool in solving complex and multimodal optimization problems.

In ABC algorithm, a colony of artificial forager bees (agents) search for rich artificial food sources (good solutions for a given problem). This computation consists of three essential agents: (1) employed bee, (2) onlooker bee (3) scout bee. Both onlookers and scouts are also called unemployed bees. The employed bees search for food and share the information to the onlooker bees. The onlooker bees select good food sources from those found by the employed bees and the scout bees are responsible for finding new food sources. In particular, the employed bee whose the food

source has been abandoned by the bees becomes a scout bee and continues the ABC search process.

To apply ABC, the considered optimization problem is first converted to the problem of finding the best parameter vector by optimizing a predefined fitness function. The position of food sources represents a possible solution and the nectar amount corresponds to the fitness of the associated solution. The artificial bees randomly discover a population of initial solution vectors and then iteratively improve them by employing the ABC strategies.

Taking the shared information of food sources from all employed bees through waggle dances, an onlooker bee chooses a food source with a probability related to its nectar amount, expressed by

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{1}$$

where fit_i is the fitness value of the ith solution in ABC swarm population and SN is the population size. In ABC algorithm, x_{ij} is the ith employed bee in the jth dimension, v_{ij} is a new solution for x_{ij} and x_{kj} is the neighbor of x_{ij}. The update rule is formulated as follows:

$$v_{ij} = x_{ij} + rand[-1, 1](x_{ij} - x_{kj}) \tag{2}$$

where rand[−1, 1] is a random number within [−1,1] to control the production of neighbor solutions around x_{ij}. $1 < k < SN$, $k \neq i$ and $1 < j < D$. If a position cannot be improved over a predefined number (called limit) of cycles, then the food source is abandoned and the scout bees will be sent to find a new food source. Then the food source found by the scout bee will take the place of the abandoned one. The movement of the scout bees is expressed by

$$x_{ij} = x_{\min}^{j} + rand[0, 1](x_{\max}^{j} - x_{\min}^{j}) \tag{3}$$

where rand[0, 1] is a random number within [0,1] based on a normal distribution and $x_{\min}^{j}$, $x_{\max}^{j}$are lower and upper boundaries of the jth dimension, respectively.

2) MODIFIED ABC WITH DYNAMIC INERTIA WEIGHT

To avoid the difficulty of local optimum caused by the unbalanced search and poor search diversity in classical ABC. This study presents a self-adaptive ABC with dynamic inertia weight and mutation operation to not only increases the search diversity but also balances the global and local search. In the proposed modified ABC, a dynamic inertia weight is added to balance exploitation and exploration in the ABC search space. The position update rule is expressed as follows.

$$v_{ij} = w(t)x_{ij} + rand[-1, 1](x_{ij} - x_{kj}) \tag{4}$$

where $w(t)$ is the weight at iteration t. This parameter is self-adaptive in the modified ABC computation, described by

$$w(t) = (w(0) - w(n_t))\frac{(n_t - t)}{n_t} + w(n_t) \tag{5}$$

Improved ABC algorithm

Initialization

(i) Set the ABC parameters: SN, $x_{\min}^{j}$, $x_{\max}^{j}$, n_t, $w(0)$ and $w(n_t)$.

(ii) Initialize the population of solutions x_{ij}.

REPEAT

(1) Evaluate the fitness value of each solution.
(2) Each ABC employed bee generates a new solution using (4) and (5).
(3) Calculate the fitness values for each solution using (1).
(4) Each ABC onlooker bee produces a new solution and evaluate it.
(5) If the solution has not been improved and exceeds a threshold *limit*, send the scouts into the search area for discovering new food sources.
(6) Memorize the best food source found so far.
(7) Check the termination condition, output the results or goto Step 2.

UNTIL(stopping criteria are met)

FIGURE 1. Scheme of the modified ABC algorithm.

where n_t is the maximum number of iteration, $w(0)$ is the initial weight and $w(n_t)$ is the final weight, $w(0) > w(n_t)$. To increase the search diversity of ABC, a single-point mutation strategy is employed in the scout bee phase. The general scheme of the improved ABC algorithm is presented in Fig. 1 that searches for the global optimum more efficiently. Notice that the detailed computational complexity analysis and statistical test of ABCs for optimization problems are presented in [25]–[30]. This study mainly contributes to the development of ABC-based real-time MLC of articulated robotic manipulators.

B. HYBRID ABC-FUZZY FOFPID MLC

1) TAKAGI-SUGENO (TS) FUZZY SYSTEM

Fuzzy systems are rule-based systems or knowledge-based systems by using fuzzy rule base and membership functions. The fuzzy IF-THEN rule bases are linguistic variables that describe the state of the system under given conditions and the membership functions define the degree of a particular measurement belongs to a particular state. This fuzzy-model-based system has been a promising research area and has attracted enormous efforts from fuzzy theory community.

Fig. 2 illustrates s the block diagram of the classical fuzzy system. It is composed of four main modules which interact with the crisp input to generate a desired output. There are four core modules in fuzzy systems, including fuzzifier, knowledge base, inference engine and defuzzifier. Fuzzification transforms the crisp input variables into fuzzy sets. Each fuzzy set is described by a varying membership function and quantified by a linguistic term. Knowledge base consists of a database and a rule base. They are linguistic variables that describe the state of the system under given conditions and

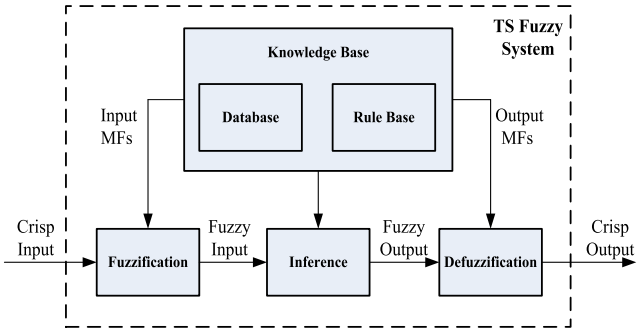

FIGURE 2. The structure of TS fuzzy systems.

the membership functions define the degree of a particular measurement belongs to a particular state.

The fuzzy rule base is composed of multiple "IF..., THEN..." inference sentences. The inference engine can apply reasoning to compute fuzzy outputs by considering the rule base. This paper considers the Takagi-Sugeno fuzzy system and the ith rule (R_i) is represented by

$$R_i : \text{IF } f(A_1 \text{ is } a_1, A_2 \text{ is } a_2, ..., A_n \text{ is } a_n) \text{ THEN C} = g(a_1, a_2, , \ldots, a_n,) \quad (6)$$

where *f()* and *g()* are mathematical functions, C is the consequent part, a_i is an antecedent or input variable. A_i is a fuzzy set represented by the membership function μ_{A_i}. Defuzzification is a fuzzy-to-crisp mathematical conversion procedure inferred by the inference engine into a real value. The firing strength α_k is calculated by

$$\alpha_k = \prod_{a_i \in A_k} \mu_{A_i}(a_i) \quad (7)$$

where A_k is the antecedents of rule k. The output of the fuzzy system is then obtained by

$$C = \frac{\sum_{k=1}^{L} \alpha_k g(a_1, \ldots, a_n)}{\sum_{k=1}^{L} \alpha_k} \quad (8)$$

2) INTELLIGENT ABC-FUZZY FOPID CONTROL

FOPID control introduced by Podlubny has received a considerable attention in the last years because it provides more flexibility and better response with respect to the standard PID controllers [34], [35]. In this study, the Riemann-Liouville method is used to define fractional calculus, formulated by

$$ {}_aD_t^q f(t) = \frac{d^q f(t)}{d(t-a)^q} = \frac{1}{\Gamma(n-q)} \frac{d^n}{dt^n} \int_0^t (t-\tau)^{n-q-1} f(\tau) d\tau \quad (9)$$

where ${}_aD_t^q$ is a differintegral operator that combines differentiation and integration, defined by

$$ {}_aD_t^q = \begin{cases} \frac{d^q}{dt^q}, & q > 0 \\ 1, & q = 0 \\ \int_0^t (d\tau)^{-q}, & q < 0 \end{cases} \quad (10)$$

n is an integer and $n-1 < q < n$. q is the fractional order. a and t are the limits of the operation. Γ is a Gamma function, expressed by

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt \quad (11)$$

A FOPID controller is an extension of the classical PID controller by involving an integrator of order λ and a differentiator of order μ, called $PI^\lambda D^\mu$ control. Using the fractional calculus, the generalized transfer function of the FOPID control law is described as follows.

$$G_c(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s^\lambda} + K_D s^\mu, \quad (\lambda, \mu \geq 0) \quad (12)$$

where $G_c(s)$ is the transfer function. $U(s)$ and $E(s)$ are the output and input signals in Laplace space, respectively. K_P, K_I and K_D are respectively the proportional gain, integration gain and derivative gain in FOPID control. To present a real-time FOPID controller with auto-tuning capability, the fuzzy theory is employed to adjust the control parameters at every sampling point and meet time constraint.

Fuzzy structure optimization is an essential issue in fuzzy applications. To address this concerning issue, this study presents a hybrid ABC-fuzzy FOPID control system, shown in Fig. 3. The triangular MFs with center c_i and width w_i are applied in the proposed ABC-fuzzy FOPID control system by taking the advantages of simplicity and easy implementation. The modified ABC algorithm is employed to optimize the fractional order and fuzzy structure. The parameters λ, μ, c_i, w_i and the number of rule L are considered in the proposed ABC-fuzzy FOPID MLC system.

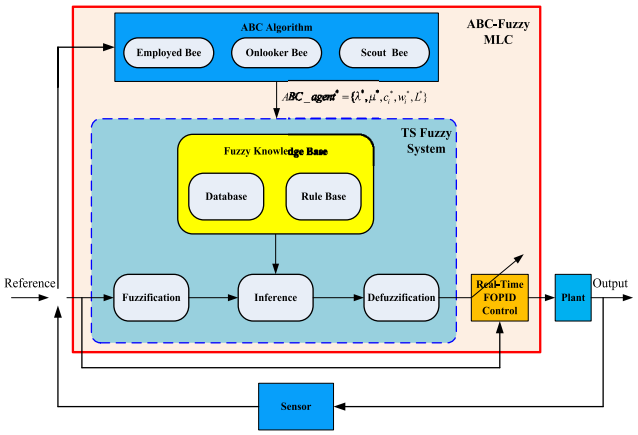

FIGURE 3. ABC-fuzzy MLC feedback control system.

In the proposed ABC-fuzzy FOPID control scheme, an ABC individual is defined by the parameters $ABC\_agent = \{\lambda, \mu, c_i, w_i, L\}$. The initial ABC bee population are randomly generated and the optimal FOPID model $ABC\_agent^* = \{\lambda^*, \mu^*, c_i^*, w_i^*, L^*\}$ is evolved through the ABC process. The fitness function (objective function) is defined by mean square error (MSE) with N_{ABC} sample to evaluate the ABC individuals, expressed by

$$MSE\_ABC = \frac{1}{N_{ABC}} \sum_{k=1}^{N_{ABC}} \left(y_{bee}{}^*(k) - y_{bee}(k)\right)^2 \quad (13)$$

where $y_{bee}(k)$ is the output in kth sampling data and $y_{bee}{}^*(k)$ is the predicted output. The seven fuzzy sets are defined by the linguistic values: Negative Large (NL), Negative Medium (NM), Negative Small (NS), Approximate Zero (AZ), Positive Small (PS), Positive Medium (PM), and Positive Large (PL). The MFs have sufficient overlap between adjacent MFs.

As shown in Fig. 3, the improved ABC algorithm is used to design optimal fuzzy structure and fractional order prior to motion control. Once the initial fuzzy structure and fractional order are optimized, the ABC-fuzzy computation is then utilized to online tune the FOPID parameters in the real-time MLC. Having the measured signals from plants, the error information is calculated and updated. Finally, the proposed ABC-fuzzy MLC system determines the real-time optimal control commands to the plants. This intelligent MLC approach is superior to the traditional fuzzy control systems because the fuzzy structure and fractional order are initially optimized through ABC process and the control gain parameters are online tuned.

III. APPLICATION TO REAL-TIME MLC OF ROBOTIC MANIPULATORS

This section aims to develop an ABC-fuzzy real-time MLC of six-DOF articulated robot manipulators. The forward kinematics is derived by means of Denavit-Hartenberg (D-H) convention. With the forward kinematic equations, the inverse kinematics is presented using reverse coordinates method. After kinematics analysis, the dynamic plant model and motion planning are employed to design a ABC-fuzzy MLC controller of robot arm with real-time FPGA implementation.

A. FORWARD KINEMATICS

Articulated robotic arm consists of a set of links connected by joints to form a kinematic chain. The D-H convention is employed to derive the forward kinematic equations of the six-DOF robot manipulator. According to classical D-H convention for robotic manipulators, the D-H matrix A_i from the coordinate *i-1* to the coordinate *i* is expressed by

$$^{i-1}A_i = Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,\alpha_i} \quad (14)$$

where *Rot* is a rotation matrix and *Trans* is a transformation matrix. The four quantities θ_i (joint angle), d_i (joint distance), a_i (link length) and α_i (link twist angle) are D-H parameters associated with link *i* and joint *i* of the six-DOF articulated manipulator. The homogeneous transformation matrix between adjacent coordinates is summarized by

$$\begin{aligned} {}^{i-1}A_i &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &\times \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (15)$$

Fig. 4 depicts the coordinate relationship of the articulated manipulator and Table 1 lists the D-H parameters of the proposed six-DOF articulated manipulator in this study. With

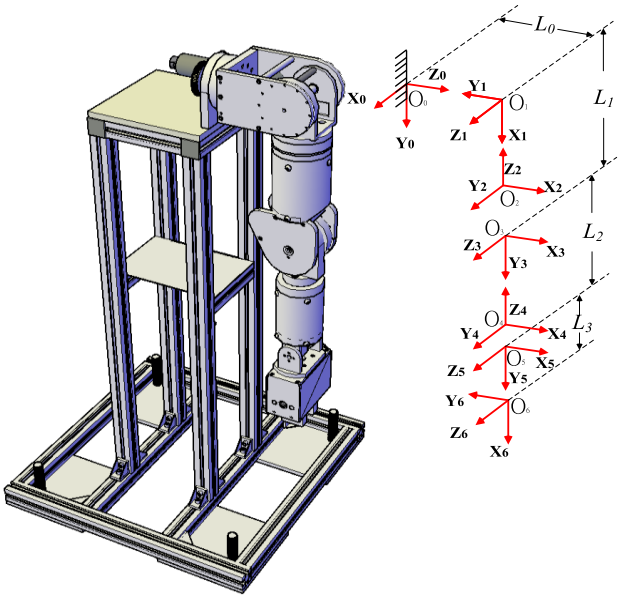

FIGURE 4. Coordinate of the six-DOF robotic manipulator.

TABLE 1. D-H parameters of the six-DOF manipulator.

Link *i*	θ_i (degree)	α_i (degree)	a_i (mm)	d_i (mm)
1	0	-90	0	110(L_0)
2	-90	90	0	0
3	0	-90	0	-265(-L_1)
4	0	90	0	0
5	0	-90	0	-235(-L_2)
6	90	0	70(L_3)	0

the geometric relationship and D-H parameters, the transformation matrices of adjacent joint are described as follows.

$$
{}^{0}A_1 = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_1 & 0 & \cos\theta_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{1}A_2 = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & 0 \\ \sin\theta_2 & 0 & -\cos\theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}
$$

$$
{}^{2}A_3 = \begin{bmatrix} \cos\theta_3 & 0 & -\sin\theta_3 & 0 \\ \sin\theta_3 & 0 & \cos\theta_3 & 0 \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{3}A_4 = \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 & 0 \\ \sin\theta_4 & 0 & -\cos\theta_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}
$$

$$
{}^{4}A_5 = \begin{bmatrix} \cos\theta_5 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_5 & 0 & \cos\theta_1 & 0 \\ 0 & -1 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{5}A_6 = \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & a_6\cos\theta_6 \\ \sin\theta_6 & \cos\theta_6 & 0 & a_6\sin\theta_6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}
$$

According to D-H convention, the transformation matrix H from link 0 to link 6 is described by

$$
H = {}^{0}A_1\,{}^{1}A_2\,{}^{2}A_3\,{}^{3}A_4\,{}^{4}A_5\,{}^{5}A_6 = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_0^6 & P_0^6 \\ 0 & 1 \end{bmatrix} \tag{19}
$$

where $P_0^6 = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T$ is the position vector and R_0^6 is the orientation matrix that includes yaw, pitch and roll angles. The D-H forward kinematics analysis for the articulated robot arm is completed in (19), namely that if the joint angles $\theta_1 \sim \theta_6$ are given, the pose of the end-effector is then determined.

B. ANALYTICAL INVERSE KINEMATICS ANALYSIS

This subsection is devoted to developing a closed-form inverse kinematics solution by means of the reverse coordinates methodology [36]. In reverse coordinates method, one should invert the base and the end of the articulated manipulator to obtain a new manipulator. The modified D-H (M-D-H) coordinates is established in Fig. 5 and the M-D-H parameters are listed in Table 2. The transformation matrix ${}^{(i-1)'}T_{i'}$ can be obtained from the inverse of ${}^{i-1}A_i$ as

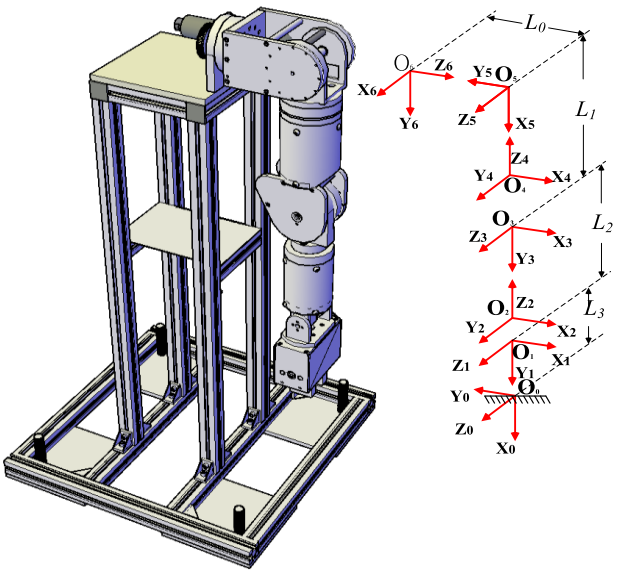

FIGURE 5. M-D-H coordinates system of the manipulator.

TABLE 2. M-D-H parameters of the manipulator.

link i'	θ_i' (degree)	α_i' (degree)	a_i' (mm)	d_i' (mm)
$1'$	-90	0	$-L_3$	0
$2'$	0	90	0	L_2
$3'$	0	-90	0	0
$4'$	0	90	0	L_1
$5'$	90	-90	0	0
$6'$	0	90	0	$-L_0$

follows:

$$
{}^{(i-1)'}T_{i'} = \begin{bmatrix} \cos\theta_i' & -\sin\theta_i' & 0 & a_i' \\ \sin\theta_i'\cos\alpha_i' & \cos\theta_i'\cos\alpha_i' & -\sin\alpha_i' & -d_i'\sin\alpha_i' \\ \sin\theta_i'\sin\alpha_i' & \cos\theta_i'\sin\alpha_i' & \cos\alpha_i' & d_i'\cos\alpha_i' \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}
$$

According to reverse coordinates approach, the position and orientation of end effector can be decoupled to perform inverse kinematics analysis. In doing so, Fig. 6 depicts the

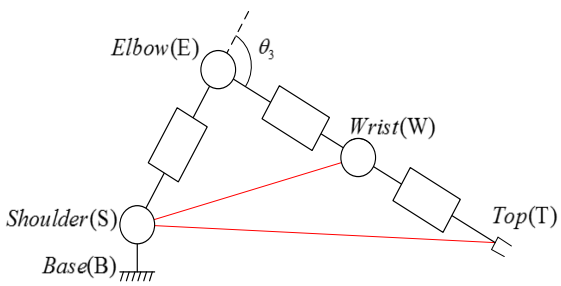

FIGURE 6. Simplified mode of the new manipulator.

simplified new manipulator that consists of base (B), shoulder (S), elbow (E), wrist (W) and top (T). The following relationships are obtained by means of reverse coordinates analysis.

$$\theta_3' = 180 - \cos^{-1}\left(\frac{\left\|{}^0X_{sw}\right\|^2 - L_{SE}^2 - L_{EW}^2}{2L_{SE}L_{EW}}\right) \tag{21}$$

where ${}^0X_{sw}$ is a vector from S to W and $\|\bullet\|$ is a norm operation, L_{SE} is the distance from S to E and L_{EW} is the distance from E to W. Using the geometric relationship in Fig. 6, the vector ${}^0X_{sw}$ is expressed by

$$\begin{aligned} {}^0X_{SW} &= \begin{bmatrix} L_1(\cos\theta_3'\sin\theta_1' + \cos\theta_1'\cos\theta_2'\sin\theta_3') + L_2\sin\theta_1' \\ -L_1(\cos\theta_1'\cos\theta_3' - \cos\theta_2'\sin\theta_1'\sin\theta_3') - L_2\cos\theta_1' \\ L_1\sin\theta_2'\sin\theta_3' \end{bmatrix} \\ &= \begin{bmatrix} {}^0X_{SW}(1) \\ {}^0X_{SW}(2) \\ {}^0X_{SW}(3) \end{bmatrix} \end{aligned} \tag{22}$$

Once θ_3' is determined, the next step is to compute θ_1' and θ_2' from (20). Based on (22), we have

$$\begin{bmatrix} \sin\theta_1' \\ \cos\theta_1' \end{bmatrix} = \begin{bmatrix} L_1\cos\theta_3' + L_2 & L_1\cos\theta_2'\sin\theta_3' \\ L_1\cos\theta_2'\sin\theta_3' & -L_1\cos\theta_3' - L_2 \end{bmatrix}^{-1} \times \begin{bmatrix} {}^0X_{SW}(1) \\ {}^0X_{SW}(2) \end{bmatrix} \tag{23}$$

and

$$\theta_1' = \tan^{-1}\left(\frac{\sin\theta_1'}{\cos\theta_1'}\right) \tag{24}$$

$$\theta_2' = \sin^{-1}\left(\frac{{}^0X_{SW}(3)}{L_1\sin\theta_3'}\right), \quad \text{if } \sin\theta_3' \neq 0 \tag{25}$$

The inverse kinematics solution θ_4', θ_5' and θ_6' can be derived by using the following rotation matrix ${}^0R_6$, formulated by

$${}^0R_6 = {}^0R_3^3R_6 \tag{26}$$

where

$$\begin{aligned} {}^3R_6 &= \left({}^0R_3^{-1}\right)\left({}^0R_6\right) \\ &= \begin{bmatrix} * & * & \cos\theta_4'\sin\theta_5' \\ \cos\theta_6'\sin\theta_5' & -\sin\theta_5'\sin\theta_6' & -\cos\theta_5' \\ * & * & \sin\theta_4'\sin\theta_5' \end{bmatrix} \end{aligned} \tag{27}$$

and $*$ are the ignored elements. From (27), the solution of θ_5' is described by

$$\theta_5' = \cos^{-1}\left(-{}^3R_{6\_23}\right) \tag{28}$$

and

$$\theta_4' = \tan^{-1}\left(\frac{{}^3R_{6\_33}}{{}^3R_{6\_13}}\right), \quad \text{if } \sin\theta_5' \neq 0 \tag{29}$$

$$\theta_6' = \cos^{-1}\left(\frac{{}^3R_{6\_21}}{\sin\theta_5}\right), \quad \text{if } \sin\theta_5' \neq 0 \tag{30}$$

where ${}^3R_{6\_ij}$ is the *(i, j)* element of matrix ${}^3R_6$. Notice that if $\sin\theta_5' = 0$, then θ_4' and θ_6' can be directly calculated. Comparing the parameters in Table 1 and Table 2, according to reverse coordinates method, one obtains the inverse closed-form kinematics solution of the six-DOF articulated manipulator, expressed by

$$\begin{aligned} &\theta_1 = -\theta_6', \quad \theta_2 = -\theta_5', \ \theta_3 = -\theta_4', \ \theta_4 = -\theta_3', \\ &\theta_5 = -\theta_2', \quad \theta_6 = -\theta_1' \end{aligned} \tag{31}$$

where $\theta_1 \sim \theta_6$ is the closed-form inverse kinematics solution of the original six-DOF robotic manipulator.

C. DYNAMIC MODELING OF ACTUATORS

In this study, the six-DOF articulated manipulator are actuated using electric motors. The generalized model with actuator dynamics and arm dynamics for n-link robot arms is derived as follows.

$$M(q)\ddot{q} + V_m(q,\dot{q})\,\dot{q} + F(\dot{q}) + G(q) = \tau \tag{32}$$

where $q \in R^n$ is the arm joint variable and τ is the generalized forces. $M(q)$ is the inertia matrix, $V_m(q,\dot{q})$ is the Coriolis /centripetal vector, $F(\dot{q})$ is the friction vector and $G(q)$ is gravity vector. The mathematical model of ith link is described by

$$J_{Mi}\ddot{q}_{Mi} + B_{Mi}\dot{q}_{Mi} = \tau_{Mi} - \tau_{Li} \tag{33}$$

where the ith electric motor has damping constant B_{Mi}, inertia J_{Mi}, rotor position angle q_{Mi}, motor's torque τ_{Mi} and load torque τ_{Li}. Considering the actuator friction vector F_{Mi}, torque constant K_{Mi}, rotor damping constant B_{Mi}, back emf constant K_{bi} and armature resistance R_{ai}.

The dynamics of the electric motors that drive the links are expressed by the n-decoupled equations:

$$J_M\ddot{q}_M + B\dot{q}_M + F_M + R\tau = K_M v \tag{34}$$

where $v = \{v_1, v_2, \ldots, v_n\} \in R^n$ is the motor voltage. $q_M = \{q_{M1}, q_{M2}, \ldots, q_{Mn}\} \in R^n$, $J_M = diag\{J_{Mi}\}$, $B = diag\{B_{Mi} + K_{Mi}K_{b_i}/R_{ai}\}$, $R = diag\{r_i\}$, $K_M = diag\{K_{Mi}/R_{ai}\}$, $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$. The gear ratio of ith link is described as follows.

$$q_i = r_i q_{M_i},\ q = Rq_M \tag{35}$$

Combing (33)-(35), one obtains the dynamics model of the articulated robotic manipulator, expressed by

$$\begin{aligned} \left(J_M + R^2M(q)\right)\ddot{q} + \left(B + R^2V_m(q,\dot{q})\right)\dot{q} + RF_M(q) \\ + R^2F(q) + R^2G(q) = RK_M v \end{aligned} \tag{36}$$

D. REAL-TIME ABC-FUZZY MLC OF ROBOTIC ARM

This subsection is devoted to developing an intelligent ABC-fuzzy MLC system for the six-DOF robotic manipulator with real-time capability. Having the ABC-fuzzy computation and robot arm model, the proposed MLC outputs the control command to drive the plant actuators in Fig. 7. This real-time FOPID MLC with online tuning has more practical

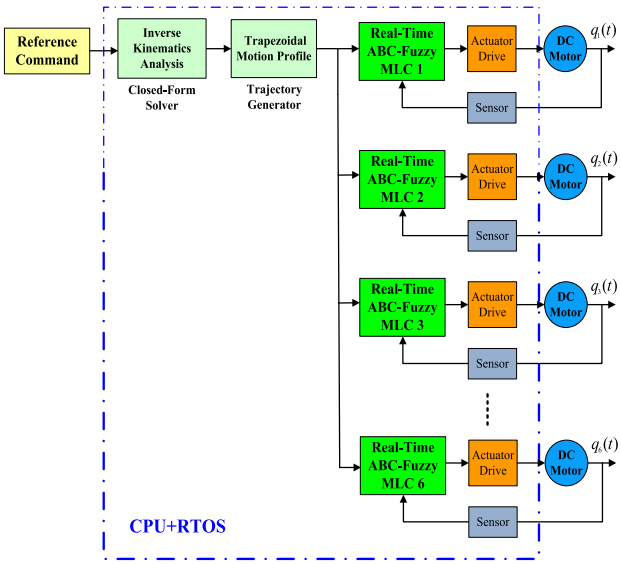

FIGURE 7. Real-time MLC system for robotic articulated arms.

significance because the fuzzy system is optimized via the modified ABC algorithm.

In Fig. 7, the controllers are implemented as several tasks on a microprocessor with a real-time operating system to meet critical timing constraint. The real-time kernel uses multiprogramming to multiplex the execution of the various tasks. In this study, the real-time ABC-fuzzy MLC of robotic manipulator is considered as a typical distributed feedback control system that consists of sensing (input), processing (computation), and actuation (output) components.

As shown in Fig. 7, once the reference command is received in Cartesian space, the proposed analytical inverse kinematics solver calculates the joint angles of each link. Then the trajectory profile is obtained by means of trapezoidal motion planning. In designing the feedback control of robot arm, the encoders are directly mounted on DC motors to provide sensed signal. An embedded processor with RTOS is employed to perform ABC-fuzzy MLC and output the actuation signal to DC motors.

There are six ABC-fuzzy MLCs in the proposed robotic system. Each controller unit is configured to support the RTOS and external communications with sensors/actuators. The configuration includes an interrupt controller, one timer clock, one processor, memory module and communication devices. The interrupt controller handles external interrupts according to priority to cope with external events or data arriving. All the inverse kinematics solver, motion profile and MLCs are implemented in one embedded processor to present a cost-effective robotic system.

IV. EXPERIMENTAL RESULTS, COMPARATIVE WORKS AND DISCUSSION

A. MECHATRONIC DESIGN AND EXPERIMENTAL SETUP

Fig. 8 presents the mechatronic design and experimental setup of the robotic manipulator used to validate the

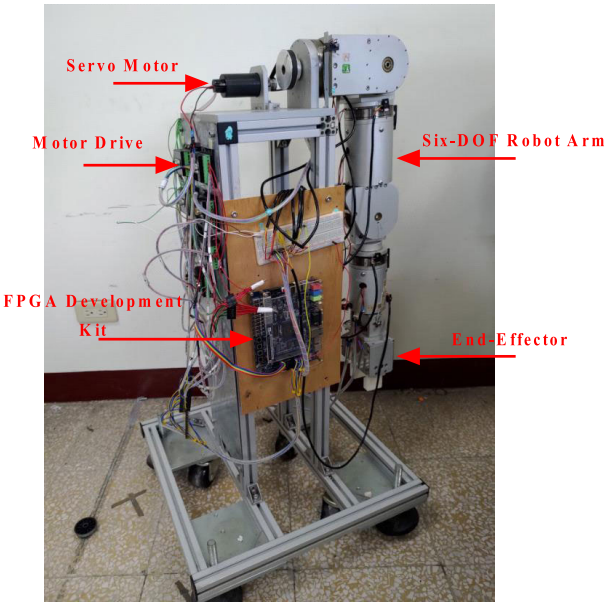

FIGURE 8. Picture of the six-DOF articulated robotic manipulator.

effectiveness and merit of the proposed real-time ABC-fuzzy MLC. As shown in Fig. 8, there are six servo motors with drives in the articulated robotic arm system. The two-phase encoders are directly mounted on the motors to provide sensor feedback signals. An Altera development board DE1-SoC with one embedded processor and RTOS is employed to perform the proposed real-time ABC-fuzzy MLC control law. The embedded processor collaborates with the RTOS to provide the FOPID control responses to the events within specific time.

In the proposed cost-effective MLC robotic system, the embedded dual-core ARM Cortex-A9 performs the inverse kinematics, motion profile and ABC-fuzzy FOPID control law in C/C++ language. The hardware components such as pulse width modulation (PWM) circuit and quadrature encoder pulse (QEP) module are implemented by Verilog hardware description language. Both the software modules and hardware components are integrated in one FPGA chip, thereby reducing the circuit size. Fig. 9 depicts the FPGA

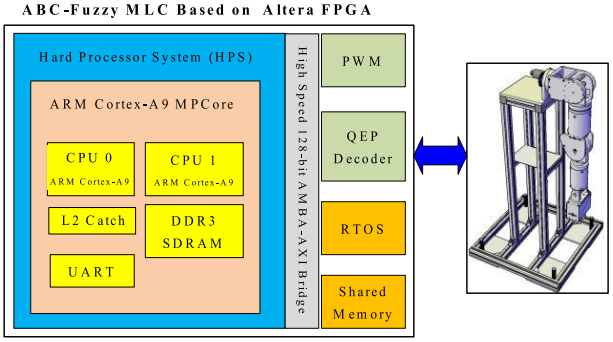

FIGURE 9. FPGA realization of the proposed real-time ABC fuzzy MLC for the six-DOF robot manipulator.

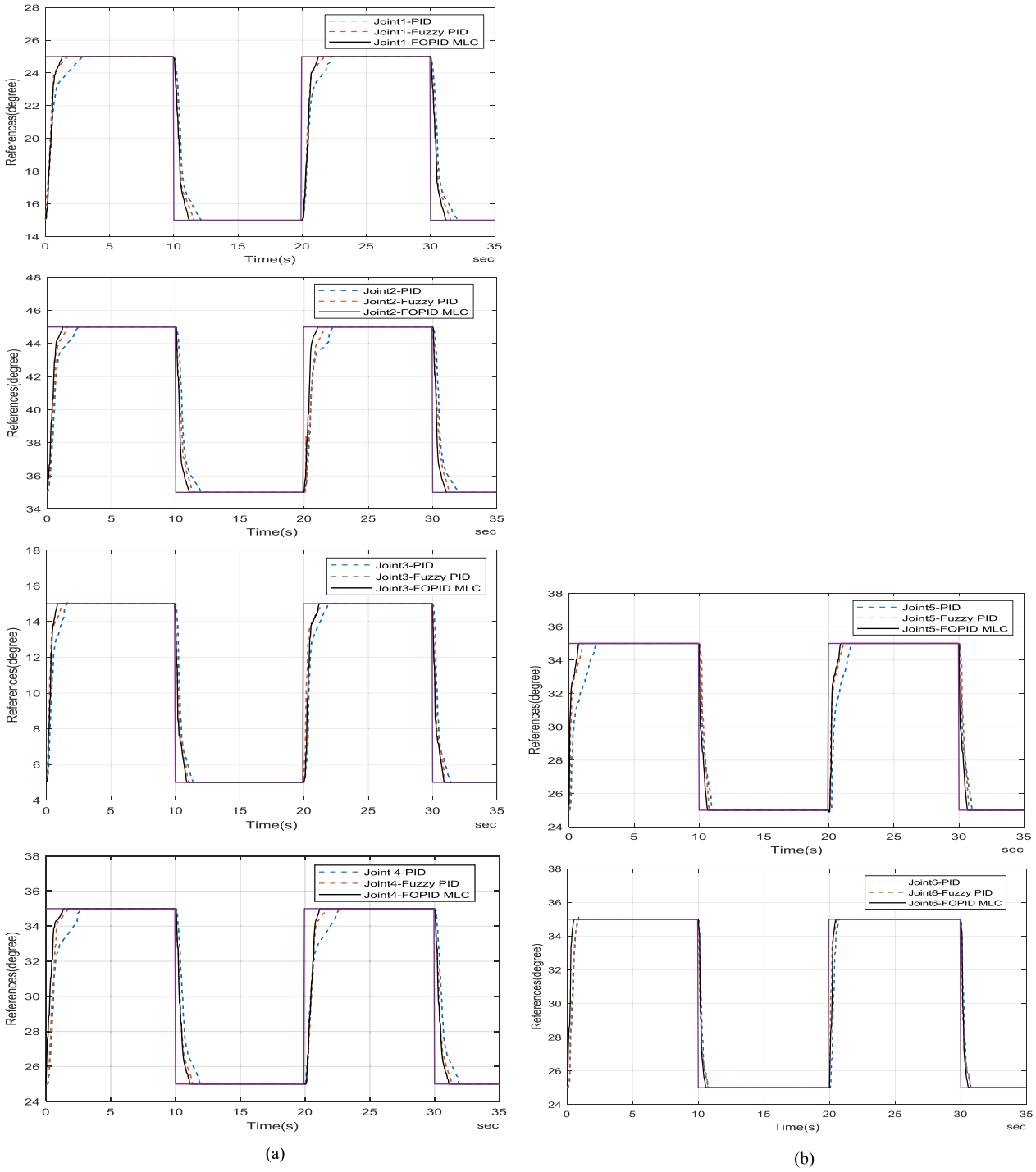

FIGURE 10. (a). Single-axis test for Joint1∼Joint4. (b). Single-axis test for Joint5∼Joint6.

realization of the proposed real-time ABC-fuzzy MLC for the six-DOF robot manipulator. The ARM hard processor system (HPS) consisting of processors, peripherals, and memory control. Although the HPS and the FPGA can operate independently, they are tightly coupled via a high-bandwidth system interconnect built from high-performance ARM AMBA AXI bus bridges. This FPGA realization outperforms the conventional controllers in term of reconfigurability, flexibility, time-to-product and reliability in designing modern MLCs for robotic systems.

B. ANALYTICAL INVERSE KINEMATICS SOLVER

This subsection aims at conducting several experiments to validate the effectiveness of the kinematics

TABLE 3. Test patterns for the six-DOF robot arm.

	θ_1 (degree)	θ_2 (degree)	θ_3 (degree)	θ_4 (degree)	θ_5 (degree)	θ_6 (degree)
1	-60.000	-60.000	-60.010	40.000	50.000	60.000
2	-70.000	-70.000	-70.000	50.000	-60.000	-70.000
3	-45.000	-25.00	-35.000	85.000	-65.000	-35.000
4	50.000	60.000	70.000	70.000	60.000	5.000
5	60.000	-50.000	80.000	60.000	50.000	20.000
6	-20.000	-75.000	40.000	30.000	10.000	-25.000
7	15.000	20.000	25.000	30.000	35.000	45.000
8	25.000	30.000	35.000	40.000	45.000	50.000
9	10.000	20.000	-30.000	0.000	5.000	-10.000
10	-30.000	-15.000	10.000	20.000	45.000	-20.000

TABLE 4. Inverse kinematics solutions.

	θ_1 (degree)	θ_2 (degree)	θ_3 (degree)	θ_4 (degree)	θ_5 (degree)	θ_6 (degree)
1	-60.006	-60.004	-60.014	40.004	49.994	60.003
2	-69.989	-69.993	-69.988	49.997	-60.005	-69.996
3	-45.001	-25.002	-35.002	85.001	-64.999	-35.001
4	49.999	60.002	69.998	69.999	60.001	5.001
5	59.999	-50.001	79.999	60.001	49.998	19.999
6	-19.994	-75.011	40.002	30.004	9.999	-25.002
7	15.001	20.001	25.004	29.999	34.999	45.001
8	25.000	30.001	35.004	40.000	44.997	49.999
9	10.099	20.193	25.124	-0.458	0.092	-9.757
10	-29.998	-14.996	9.991	19.995	45.010	-19.996

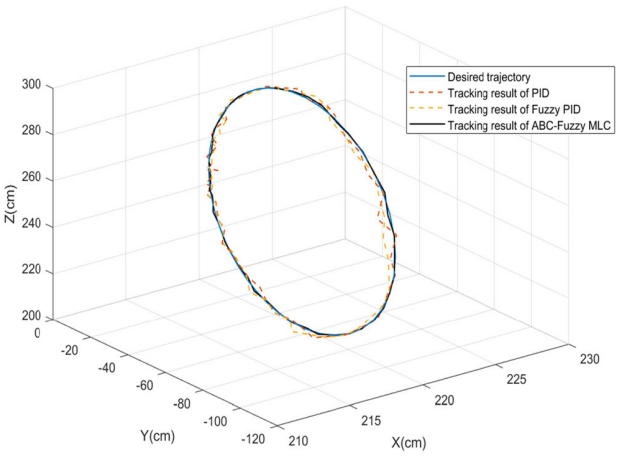

FIGURE 11. Experimental results of trajectory tracking using PID, fuzzy PID and ABC-fuzzy FOPID MLC.

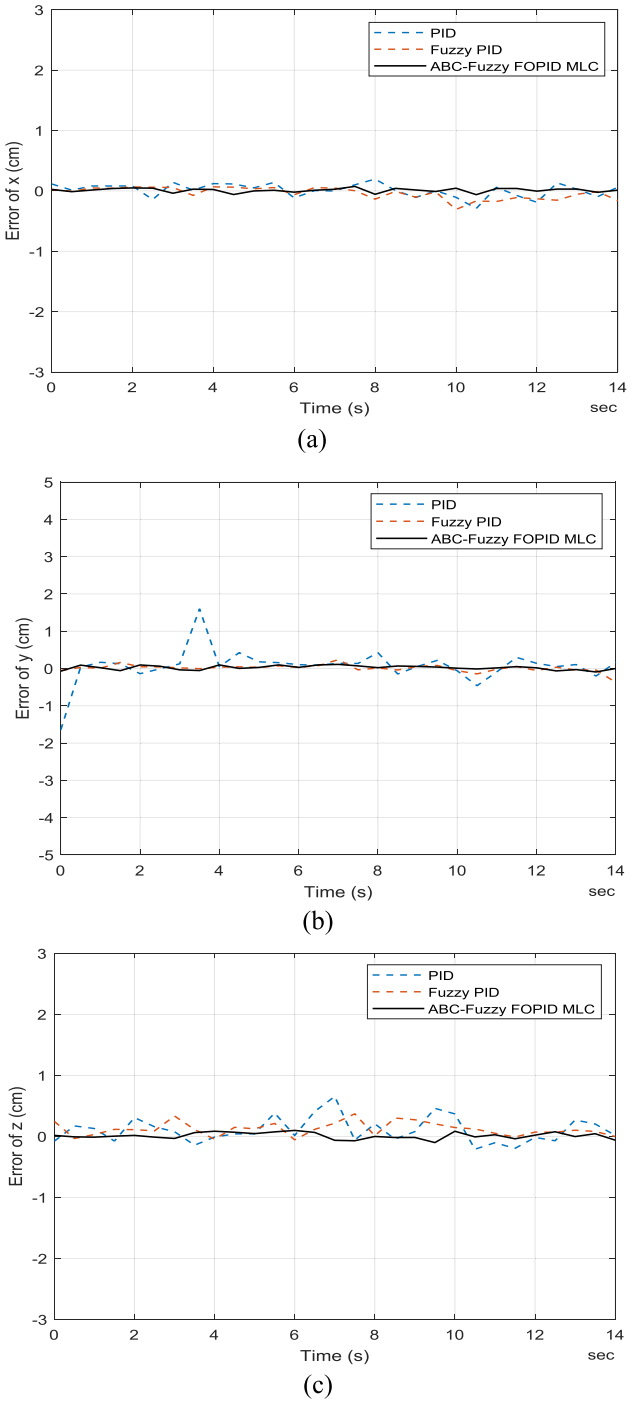

FIGURE 12. Experimental tracking errors of PID, fuzzy PID and ABC-fuzzy FOPID MLC (a)error of x (b)error of y (c) error of z.

analysis using D-H convention and reverse coordinates approach. Table 3 lists the ten patterns used to perform D-H forward kinematics analysis. Taking the calculated position and orientation matrices into the analytical inverse kinematics solver, Table 4 presents the results for all cases. Through this cross-checked experiments, the proposed closed-form solver provides inverse kinematics solutions of the six-DOF robotic manipulator successfully.

C. SINGLE-AXIS TEST AND COMPARATIVE WORKS

This subsection is devoted to conducting experiments to evaluate the control performance of the proposed ABC-fuzzy MLC for servo motors. Fig. 10 presents the experimental results of the six actuators using PWM H-bridge drives. The reference command for each axis is set as a square wave

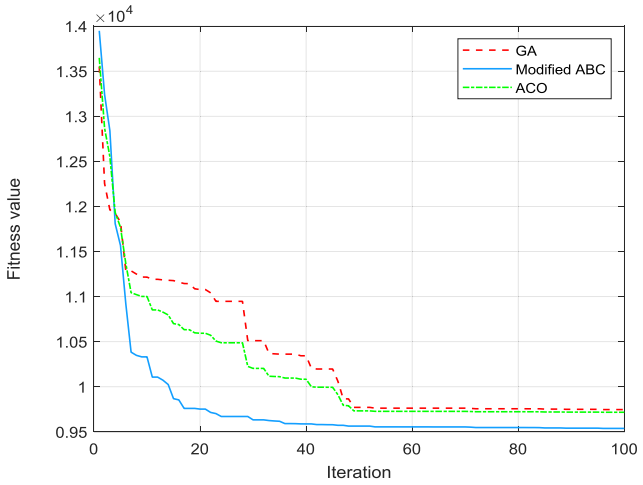

FIGURE 13. Convergent behavior of the fitness value using GA, ACO and modified ABC.

trajectory. As can be seen in Fig 10, the proposed ABC-fuzzy real-time MLC tracks the desired command successfully.

In order to illustrate the merit of the proposed real-time MLC over other existing control approaches, Fig. 10 also presents the comparison analysis. The traditional PID control, fuzzy-PID control schemes are also utilized to perform the same single-axis test of articulated robot arm. This comparison work clearly indicates that the proposed real-time ABC-fuzzy FOPID MLC scheme outperforms the conventional control methods.

D. EXPERIMENTAL RESULTS OF TRAJECTORY TRACKING

This subsection aims at presenting experimental results of trajectory tracking. After performing inverse kinematics analysis and trapezoidal motion profile, the robot manipulator is steered to achieve a circular trajectory tracking task. Fig. 11 depicts the tracking result using the proposed ABC-fuzzy FOPID control strategy. This experimental result clearly indicates that the proposed MLC achieves the tracking task successfully. To provide comparison analysis in Fig. 11, the traditional control laws are also employed to achieve the same tracking task. Fig. 12 depicts the experimental tracking errors of PID, fuzzy PID and ABC-fuzzy FOPID MLC. Through these experimental results and comparison works, the proposed ABC-fuzzy MLC is superior to the existing control schemes.

In this study, the modified ABC algorithm is utilized to determine the fractional order and fuzzy structure in the real-time MLC. The evolved agent $ABC\_agent^* = \{\lambda^*, \mu^*, c_i^*, w_i^*, L^*\}$ is employed to conduct the experiments. Fig. 13 presents the convergent behavior of the fitness value. To illustrate the merit of the proposed MLC, this result is compared with the ones using traditional GA, ACO based fuzzy systems. The GA and ACO parameters are determined using the robust design methods in [37], [38]. As shown in Fig. 13, the proposed MLC converges to optimum solution more quickly and the evolved solution is better than the ones obtained by GA and ACO evolutionary algorithms. Compared with the traditional MLCs using supervised, unsupervised and reinforcement learning [39], [40], the fuzzy structure is optimally determined via ABC computation to achieve optimal intelligent control.

V. CONCLUSION

This paper has presented a real-time MLC of articulated robotic manipulators using ABC algorithm incorporated with fuzzy theory. This ABC-fuzzy hybrid learning algorithm is applied to real-time MLC of robotic manipulators by including FOPID control strategy. The MLC's parameters are online tuned via the ABC-fuzzy optimization to meet critical timing constraint. Experimental results and comparative works clearly demonstrate the superiority of the proposed MLC methods against other existing approaches. The proposed ABC-fuzzy MLC has theoretical and practice significance in both industrial and academic applications.

REFERENCES

[1] H. Fu, J. Liu, Z. Han, and Z. Shao, "A heuristic task periods selection algorithm for real-time control systems on a multi-core processor," *IEEE Access*, vol. 5, pp. 24819–24829, 2017.
[2] G. Buttazzo, M. Velasco, and P. Marti, "Quality-of-control management in overloaded real-time systems," *IEEE Trans. Comput.*, vol. 56, no. 2, pp. 253–266, Feb. 2007.
[3] K.-D. Kim and P. R. Kumar, "Real-time middleware for networked control systems and application to an unstable system," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 1898–1906, Sep. 2013.
[4] D. G. Schmidt and F. Kuhns, "An overview of the real-time CORBA specification," *Computer*, vol. 33, no. 6, pp. 56–63, Jun. 2000.
[5] G. Krummenacher, C. S. Ong, S. Koller, S. Kobayashi, and J. M. Buhmann, "Wheel defect detection with machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1176–1187, Apr. 2018.
[6] D. Côté, "Using machine learning in communication networks," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 10, no. 10, pp. 1–10, 2018.
[7] M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang, "Disease prediction by machine learning over big data from healthcare communities," *IEEE Access*, vol. 5, pp. 8869–8879, 2017.
[8] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for Internet of Things data analysis: A survey," *Digit. Commun. Netw.*, vol. 4, no. 3, pp. 161–175, Aug. 2018.
[9] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, Jul. 2015.
[10] L. Nie, J. Guan, C. Lu, H. Zheng, and Z. Yin, "Longitudinal speed control of autonomous vehicle based on a self-adaptive PID of radial basis function neural network," *IET Intell. Transp. Syst.*, vol. 12, no. 6, pp. 485–494, Aug. 2018.
[11] J. H. Lee, J. Shin, and M. J. Realff, "Machine learning: Overview of the recent progresses and implications for the process systems engineering field," *Comput. Chem. Eng.*, vol. 114, pp. 111–121, Jun. 2018.
[12] S. Wang, W. Chaovalitwongse, and R. Babuska, "Machine learning algorithms in bipedal robot control," *IEEE Trans. Syst., Man, Cybern., C, Appl. Rev.*, vol. 42, no. 5, pp. 728–743, Sep. 2012.
[13] S. Jonic, T. Jankovic, V. Gajic, and D. Popvic, "Three machine learning techniques for automatic determination of rules to control locomotion," *IEEE Trans. Biomed. Eng.*, vol. 46, no. 3, pp. 300–310, Mar. 1999.
[14] T. Duriez, S. L. Brunton, and B. R. Noack, *Machine Learning Control: Taming Nonlinear Dynamics and Turbulence*. Cham, Switzerland: Springer, 2016.
[15] H. Chiroma, A. S. M. Noor, S. Abdulkareem, A. I. Abubakar, A. Hermawan, H. Qin, M. F. Hamza, and T. Herawan, "Neural networks optimization through genetic algorithm searches: A review," *Appl. Math. Inf. Sci.*, vol. 11, no. 6, pp. 1543–1564, Nov. 2017.
[16] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: A survey," *Control Eng. Pract.*, vol. 10, no. 11, pp. 1223–1241, Nov. 2002.

[17] K. Deep and M. Thakur, "A new mutation operator for real coded genetic algorithms," *Appl. Math. Comput.*, vol. 193, no. 1, pp. 211–230, Oct. 2007.
[18] H. Wei and X.-S. Tang, "A genetic-algorithm-based explicit description of object contour and its ability to facilitate recognition," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2558–2571, Nov. 2015.
[19] H. C. Lee, B. S. Roh, and B. H. Lee, "Multi-hypothesis map merging with sinogram-based PSO for multi-robot systems," *Electron. Lett.*, vol. 52, no. 14, pp. 1213–1214, Jul. 2016.
[20] L. A. Torres Salomao, M. Mahfouf, E. El-Samahy, and C.-H. Ting, "Psychophysiologically based real-time adaptive general type 2 fuzzy modeling and self-organizing control of operator's performance undertaking a cognitive task," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 1, pp. 43–57, Feb. 2017.
[21] T. Blackwell, J. Branke, and X. Li, "Particle swarms for dynamic optimization problems," *Swarm Intell.*, vol. 20, pp. 193–217, 2008.
[22] M. Dorigo and T. Stutzle, *Ant Colony Optimization*. Cambridge, MA, USA: MIT Press, 2004.
[23] M. Masood, M. M. Fouad, R. Kamal, I. Glesk, and I. U. Khan, "An improved particle swarm algorithm for multi-objectives based optimization in MPLS/GMPLS networks," *IEEE Access*, vol. 7, pp. 137147–137162, 2019.
[24] C. F. Wang and K. Liu, "A novel particle swarm optimization algorithm for global optimization," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–10, Jan. 2016.
[25] L. Cui, G. Li, Q. Lin, Z. Du, W. Gao, J. Chen, and N. Lu, "A novel artificial bee colony algorithm with depth-first search framework and elite-guided search equation," *Inf. Sci.*, vols. 367–368, pp. 1012–1044, Nov. 2016.
[26] B. Akay and D. Karaboga, "Artificial bee colony algorithm for large-scale problems and engineering design optimization," *J. Intell. Manuf.*, vol. 23, no. 4, pp. 1001–1014, Aug. 2012.
[27] L. Cui, G. Li, X. Wang, Q. Lin, J. Chen, N. Lu, and J. Lu, "A ranking-based adaptive artificial bee colony algorithm for global numerical optimization," *Inf. Sci.*, vol. 417, pp. 169–185, Nov. 2017.
[28] D. Karaboga and B. Gorkemli, "A quick artificial bee colony (qABC) algorithm and its performance on optimization problems," *Appl. Soft Comput.*, vol. 23, pp. 227–238, Oct. 2014.
[29] X. Zhang, X. Zhang, and L. Wang, "Antenna design by an adaptive variable differential artificial bee colony algorithm," *IEEE Trans. Magn.*, vol. 54, no. 3, pp. 1–6, Mar. 2018. Art. no. 7201704.
[30] L. Cui, G. Li, Z. Zhu, Q. Lin, Z. Wen, N. Lu, K.-C. Wong, and J. Chen, "A novel artificial bee colony algorithm with an adaptive population size for numerical function optimization," *Inf. Sci.*, vol. 414, pp. 53–67, Nov. 2017.
[31] K. Tahara, S. Arimoto, M. Sekimoto, M. Yoshida, and Z.-W. Luo, "On iterative learning control for simultaneous force/position trajectory tracking by using a 5 D.O.F. robotic thumb under non-holonomic rolling constraints," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2008, pp. 2611–2616.
[32] B.-P. Huynh, C.-W. Wu, and Y.-L. Kuo, "Force/position hybrid control for a hexa robot using gradient descent iterative learning control algorithm," *IEEE Access*, vol. 7, pp. 72329–72342, 2019.
[33] Z. Du, W. Wang, Z. Yan, W. Dong, and W. Wang, "Variable admittance control based on fuzzy reinforcement learning for minimally invasive surgery manipulator," *Sensors*, vol. 17, no. 4, pp. 1–15, 2017.
[34] P. Shah and S. Agashe, "Review of fractional PID controller," *Mechatronics*, vol. 38, pp. 29–41, Sep. 2016.
[35] M. P. Aghababa, "Optimal design of fractional-order PID controller for five bar linkage robot using a new particle swarm optimization algorithm," *Soft Comput.*, vol. 20, no. 10, pp. 4055–4067, Oct. 2016.
[36] L. Jiang, X. Huo, Y. Liu, and H. Liu, "An analytical inverse kinematic solution with the reverse coordinates for 6-DOF manipulators," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2013, pp. 1552–1558.
[37] G. Taguchi, *Introduction to Quality Engineering*. White Plains, NJ, USA: Quality Resources, 1986.
[38] J.-T. Tsai, J.-H. Chou, and T.-K. Liu, "Optimal design of digital IIR filters by using hybrid taguchi genetic algorithm," *IEEE Trans. Ind. Electron.*, vol. 53, no. 3, pp. 867–879, Jun. 2006.
[39] L. L. L. McLauchlan, R. Challoo, S. I. Omar, and R. A. McLauchlan, "Supervised and unsupervised learning applied to robotic manipulator control," in *Proc. Amer. Control Conf. (ACC)*, vol. 3, 1994, pp. 3357–3358.
[40] J.-B. Kim, H.-K. Lim, C.-M. Kim, M.-S. Kim, Y.-G. Hong, and Y.-H. Han, "Imitation reinforcement learning-based remote rotary inverted pendulum control in OpenFlow network," *IEEE Access*, vol. 7, pp. 36682–36690, 2019.

HSU-CHIH HUANG (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from National Chung Hsing University, Taichung, Taiwan, in 2009. He is currently a Professor with the Department of Electrical Engineering, National Ilan University, Yilan, Taiwan. His current research interests include intelligent control, mobile robots, embedded systems, the SoPC, and nonlinear control.

CHUN-CHIA CHUANG received the M.S. degree from the Department of Electrical Engineering, National Ilan University. His current research interests include intelligent control and SoPC.