IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 September 2022, accepted 10 October 2022, date of publication 1 November 2022,
date of current version 30 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3215980

APPLIED RESEARCH

Simple Neural Network Compact Form Model-Free Adaptive Controller for Thin McKibben Muscle System

MUHAMAD HAZWAN ABDUL HAFIDZ[1], AHMAD ATHIF MOHD FAUDZI[1,2], (Senior Member, IEEE), NOR MOHD HAZIQ NORSAHPERI[3], (Member, IEEE), MOHD NAJEB JAMALUDIN[4], (Member, IEEE), DAYANG TIAWA AWANG HAMID[5], AND SHAHROL MOHAMADDAN[6]

[1]Faculty of Engineering, School of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia
[2]Centre for Artificial Intelligence and Robotics, Universiti Teknologi Malaysia, Kuala Lumpur 54100, Malaysia
[3]Department of Electrical and Electronics Engineering, Faculty of Engineering, Universiti Putra Malaysia, Seri Kembangan, Selangor 42400, Malaysia
[4]Sports Innovation and Technology Centre, Institute Human Centered Engineering, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia
[5]Faculty of Islamic Technology, Sultan Sharif Ali Islamic University, Bandar Seri Begawan BE1310, Brunei
[6]Department of Bioscience and Engineering, College of Systems Engineering and Science, Shibaura Institute of Technology, Saitama 337-8570, Japan

Corresponding author: Ahmad Athif Mohd Faudzi (athif@utm.my)

This work was supported by the Ministry of Higher Education Malaysia (MOHE) through Fundamental Research Grant Scheme under Grant FRGS/1/2019/TK04/UTM/02/41.

ABSTRACT This paper proposes a simple neural network compact form model-free adaptive controller (NNCFMFAC) for a single thin McKibben muscle (TMM) system. The main contribution of this work is the simplification of the current neural network (NN) based compact form model-free adaptive controller (CFMFAC), which requires only two adaptive weights. This is achieved by designing a NN topology to specifically enhance the CFMFAC response. The prominent control parameters of the CFMFAC are combined and an adaptive weight is used for self-tuning, while the second adaptive weight is used to minimize the offset at each operating point. Hence the issues of redundant adaptive weights in complex neuro-based CFMFACs and slow response of the CFMFAC are significantly addressed. The idea is proven in three ways: analytically, simulation on a nonlinear system and experiments on a TMM platform. Experimental results demonstrating the superiority of the proposed method over the conventional CFMFAC is confirmed by a 76% improvement in convergence speed and a 60% reduction in root mean square error (RMSE). It is envisaged that the proposed controller can be very useful for TMM driven applications as it is model-independent, has fast response, high tracking accuracy, and minimal complexity.

INDEX TERMS Artificial neural networks, hydraulic/pneumatic actuators, model-free adaptive controller, modeling, control and learning for soft robots.

I. INTRODUCTION

Pneumatic artificial muscle (PAM) is a type of actuation in soft robots [1]. Due to the similarities of PAM with biological muscles, PAM is increasingly popular in research and industrial application such as rehabilitation, robotic manipulators, and exploration [2]. The PAM structure usually contains an inflatable inner elastic tube enclosed in braided mesh sleeves and fittings are used to secure the ends. When pressurized air is pumped into the PAM body, the inner tube expands in the radial direction and contracts axially, thus producing an axial contractile force used for actuation. PAM actuators are used for their high power to weight ratio, highly compliant, and their soft body allows safer interaction with the environment [1], [3].

Thin McKibben muscles (TMMs) are a type of PAM developed by [4] to further enhance the capabilities of conventional PAM actuators in terms of flexibility, compactness, and are

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

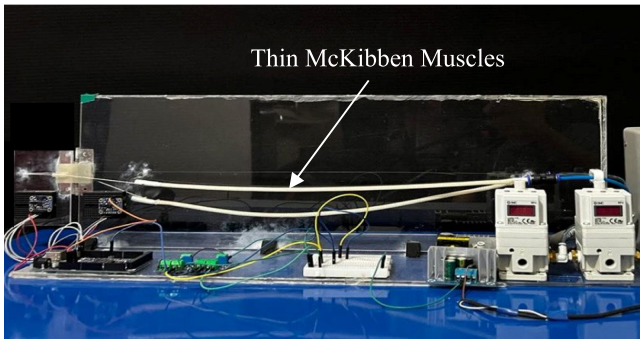

FIGURE 1. TMM platform.

significantly lighter in weight [4], [5]. Applications of TMM include continuum manipulators [6], wearable devices [5], and bio-inspired robots [7].

Despite many advantages in using PAMs as actuators, characteristics of PAM driven systems such as large dead-zones due to slack, elasticity of inner rubber tube, a complex structure of braided sleeve, and aging of rubber material causes the actuator to suffer from highly nonlinear characteristics and time-varying parameters [8]. Furthermore, in contrast to rigid actuators, the soft body of PAMs continuously deforms with infinite degrees of freedom [2]. Thus, obtaining a perfect model for PAM-driven systems is extremely complex.

Recently, various model-based control strategies have achieved regulation and tracking objectives in PAM-related research such as sliding mode control [9] and other Lyapunov-based controller designs [10]. The study in [11] modeled the motion mechanism of PAM as a dynamic nonlinear system and proposed an adaptive control method based on a nonlinear extended state observer to estimate total disturbances for a PAM system. Phenomenological modeling of PAM was combined with a neuroadaptive controller in [8]. The findings showed that a neural network (NN) can be used to approximate the unknown nonlinear characteristics in a PAM system and show that tracking error converges through PAM experiments. In [12], a dynamic model of the PAM mechanism was introduced, and an adaptive extended state observer is used to estimate the disturbances and states of the mechanism to deal with the uncertainties in PAMs. Diverse models were mathematically formulated, and control strategies were designed based on the models. However, due to the complexities in obtaining an accurate model of PAM systems, the use of model-based controllers may cause unexpected system behaviors and could form an unstable closed-loop system [13], [14].

To deal with modeling-related issues such as accurate model unavailability or systems that involve large uncertainties, data-driven strategies have been implemented where only the input and output data of the system is used to form the controller [13]. Among various data-driven control strategies, the model-free adaptive control (MFAC) is a popular method used in industrial process control [15]. Recently, improved MFAC schemes to control PAM systems are introduced. A high-order pseudo-partial derivative-based MFAC was implemented on a PAM platform and showed impressive tracking abilities [14]. However, iterative pre-training is needed which is more suitable for repetitive applications such as in rehabilitation exercises. In [16], an improved MFAC for PAM actuated systems for rehabilitation robots which do not require any pre-training is constructed. However, the controller could only manage reasonable tracking in low-speed systems. Other findings have also proven that although MFAC does have advantages of being model-independent, it has slow tracking speed [17].

The study in [15] has shown that a neural network combined with a partial form MFAC can be used to increase the response speed. Even so, the use of the partial form dynamic linearization method is more complex compared to the compact form model-free adaptive controller (CFMFAC) which is relatively simpler and preferable [13]. In [18], CFMFAC which has characteristics of straightforward operation is combined with a parameter self-tuning mechanism using backpropagation neural networks (BPNN). However, by combining a CFMFAC with a neural network with long-term short-term memory architecture, authors in [19] has shown better tracking performances compared to [18]. Unfortunately, many adaptive weights were used which in turn, increases the computational load. With the objective of decreasing the computational load, authors in [20] used a general regression neural network (GRNN) combined with an improved CFMFAC to estimate the pseudo-partial derivative (PPD). Despite the promising work, the computational load of the controller still depends on the number of neurons in the pattern layer which in turn affects the controller accuracy when low numbers of neurons are used [20]. Furthermore, the GRNN requires an additional layer for the architecture compared to the BPNN used in [18]. Together, these characteristics results in a larger NN architecture. The major advantages of implementing neural networks with smaller architecture include overfitting avoidance, augmentation of the generalized ability of the network and fewer calculations, thus accelerating the process of weight adaption [21]. Additionally, due to the black box-like nature of neural networks, troubleshooting a neural network is a complex process due to the numerous learnable parameters [22]. Previous studies have also shown that there is no linear relationship between the complexity of the NN architecture and accuracy [23]. Therefore, it can be concluded that a neural network with a simple architecture is preferable as it is faster, easier to maintain, avoids overfit and can achieve high approximation accuracy.

Based on the merits and drawbacks of the CFMFAC and neural network combination, the main motivation of this study is to assess the performance of this combination with a neural network with a specific network topology with reduced weights, without compromising the speed and accuracy of the response. Furthermore, there exists trivial experimental evidence that can confirm the simulation findings from [15], [17], [19], and [20] can perform in PAM driven systems.

To our best knowledge, no previous study has assessed the CFMFAC method on a TMM system. Hence, this study intends to explore the potential of this model-free controller on a TMM driven system due to the advantages of the actuator over conventional PAMs.

Taking everything into consideration, this study contributes to the expansion of knowledge in this field by addressing three important issues. First, the performance of a neural network algorithm with reduced number of total adaptive weights in enhancing the performance of the CFMFAC. Second, the potential of the proposed model-free NNCFMFAC control strategy for nonlinear systems, and finally the comprehensive tracking experimental validation of the designed controller on a single-input single-output (SISO) TMM system.

The following section will be discussed as follows: Section II, the designed controller is introduced. Section III, proof of convergence and boundedness are shown. Section IV, the proposed controller is evaluated in simulation and TMM experiments, and the key findings are then discussed. Finally, Section IV concludes the study and recommend future work.

II. CONTROL STRATEGY

A. COMPACT FORM MODEL-FREE ADAPTIVE CONTROLLER (CFMFAC)

The general discrete time SISO nonlinear system can be described as in (1).

$$y(k+1) = f\left(y(k), .., y\left(k-n_y\right), v(k), .., v(k-n_v)\right) \tag{1}$$

where $y(k) \in \mathbb{R}$ and $v(k) \in \mathbb{R}$ are the output of the control system and output of the CFMFAC at instant k. Meanwhile, n_y and n_v represent the unknown system order of system output and controller output respectively, and $f(\cdots)$ is an unknown nonlinear function. The following assumptions are made on system (1) to allow transformation into a compact form data linearization model.

Assumption 1: The partial derivative of $f(\cdots)$ with respect to $v(k)$ is continuous.

Assumption 2: Equation (1) satisfies the general *Lipschitz* condition for all k with finite exception according to the following condition in (2):

$$|y(k_1+1) - y(k_2+1)| \leq L\,|v(k_1) - v(k_2)| \tag{2}$$

for any $k_1 \neq k_2$, $k_1, k_2 \geq 0$ and $v(k_2) \neq v(k_1)$, where the Lipschitz constant, L is a positive constant.

For a general control system design, *Assumption 1* is a typical constraint and is a reasonable assumption. *Assumption 2* imposes an upper bound on the change rate of the output driven by the change of the input. From an energy point of view, the energy of the system is bounded when produced by bounded input energy changes.

Satisfying the assumptions mentioned, Equation (1) can be transformed into the following compact form data linearization model (CFDL) in (3):

$$y(k+1) = y(k) + \phi(k)\,\Delta v(k) \tag{3}$$

where $|\phi(k)| \leq L$ is the pseudo partial derivative (PPD) at time instant k and $\Delta v(k) = v(k) - v(k-1)$. Further proof can be found in [13]. Next, the one-step-ahead prediction error cost function is used to design the controller algorithm as in (4). This cost function is used to minimize the square of error and maintain the smoothness in the change of the control input.

$$J(v(k)) = \left|y^*(k+1) - y(k+1)\right|^2 + \lambda\,|\Delta v(k)|^2 \tag{4}$$

where $\lambda > 0$ is a tunable control parameter added to enhance or reduce the changing rate of the control input and y^* is the desired output. By substituting (3) into (4), (5) is obtained.

$$J(v(k)) = \left|y^*(k+1) - y(k) - \phi(k)\,\Delta v(k)\right|^2 + \lambda\,|\Delta v(k)|^2 \tag{5}$$

The derivation of (5) is as (6), and an optimized solution is obtained when $\partial J(v(k))/\partial v(k) = 0$, resulting in the CFMAC output in (7).

$$\frac{\partial J(v(k))}{\partial v(k)} = -2\phi(k)\left(y^*(k+1) - y(k) - \phi(k)\,v(k) + \phi(k)\,v(k-1)\right) + 2\lambda v(k) - 2\lambda v(k-1) \tag{6}$$

$$v(k) = v(k-1) + \frac{\rho\phi(k)}{\lambda + |\phi(k)|^2}\left(y^*(k+1) - y(k)\right) \tag{7}$$

where $\rho \in (0, 1]$ is a tunable control parameter introduced to increase flexibility of the controller. Subsequently, the cost function in (8) can be used to minimize the square of the error between the actual output of the system and the output of the CFDL model.

$$J(\phi(k)) = |y(k) - y(k-1) - \phi(k)\,\Delta v(k-1)|^2 + \mu\left|\phi(k) - \hat{\phi}(k-1)\right|^2 \tag{8}$$

where $\mu > 0$ is a tunable control gain and $\hat{\phi}$ is an estimate of the PPD. From the cost function used, $\hat{\phi}(k)$ can be obtained according to the cost function in (8) by minimizing $\partial J(\phi(k))/\partial\phi(k) = 0$. While the tracking error is minimized, $\hat{\phi}(k)$ converges to $\phi(k)$. The estimated PPD is obtained as in (9) and (10).

$$\frac{\partial J(\phi(k))}{\phi(k)} = -2\Delta v(k-1)(\Delta y(k) - \phi(k)\Delta v(k-1) + 2\mu\left(\phi(k) - \hat{\phi}(k-1)\right) \tag{9}$$

$$\hat{\phi}(k) = \hat{\phi}(k-1) + \frac{\eta\Delta v(k-1)}{\mu + \Delta v(k-1)^2} \times \left(\Delta y(k) - \hat{\phi}(k-1)\,\Delta v(k-1)\right) \tag{10}$$

where $\eta \in (0, 2]$ can be adjusted to make the controller more general. To ensure the controller follows the set conditions, the reset algorithm is formulated as (11).

$$\hat{\phi}(k) = \hat{\phi}_0, \quad if\ \hat{\phi}(k) \leq \varepsilon or\ |\Delta v(k-1)| \leq \varepsilon \tag{11}$$

where ε is a small positive constant and $\hat{\phi}_0$ is the initial value of the PPD estimate. The CFMFAC consist of the CFMFAC output algorithm, PPD estimation algorithm and reset algorithm in (7), (10), and (11) respectively. From the derivations shown, it can be observed that the control method only utilizes the input and output of the closed-loop control system, hence it is a model-free adaptive control method.

B. NEURAL NETWORK COMPACT FORM MODEL FREE ADAPTIVE CONTROLLER (NNCFMFAC)

The choice of the NN architecture is mainly problem specific [24]. The aim of this section is to design a NN architecture with reduced complexity yet able to improve the response of CFMFAC. In this study, a feedforward NN (FFNN) is used. A feedback NN utilizes additional adaptive weights, therefore the FFNN is preferred to reduce the complexity of the network. The FFNN model complexity is analyzed according to the well-established feedforward architecture criterion (FFAC) [21]. According to the FFAC, the complexity of a feed forward neural network depends on the total number of weights and biases in the network [25] and is defined as (12).

$$FFAC = \gamma e^{f(x)} \tag{12}$$

where γ is a constant and $f(x)$ is a function of the total number of weights and biases. From (12), the FFAC suggests that the penalty value increases exponentially with the number of total weights and biases. Thus, a neural network architecture with fewer weights and bias is preferred in reducing the complexity of the network.

The neural network architecture is defined by three critical components: network topology, node character, and learning rules [26]. The topology is designed as a "2-1" structure which means a two-layer NN consisting of the input layer with two nodes and an output layer with one node. The first input node is designed to approximate the relationship of CFMFAC output to the system error and the second is introduced with a tunable bias to shift the NN output for a faster response. The single output node is used to convey the control signal to the system. The layers are fully interconnected and assigned with adaptive weights which adjust according to the stochastic gradient descent learning rule [27]. The mean squared error is used as the cost function as in (13).

$$J = \frac{1}{2}e^2(k) \tag{13}$$

where $e(k)$ is the difference between the desired system output and the actual output at instant k. Linear activation function is chosen as to avoid gradient vanishing problems associated with other activation functions such as the Sigmoid and Tanh function [28]. The designed NN architecture is shown in Fig. 2.

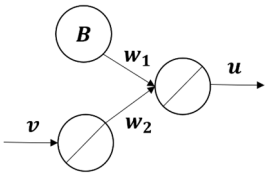

FIGURE 2. Neural network architecture for proposed controller.

Since a linear activation function is used, the control law of the proposed NNCFMFAC can be calculated directly from the summation of weighted input and weighted bias and is defined as (14).

$$u(k) = g(Bw_1(k) + v(k)w_2(k)) \tag{14}$$

where $u(k)$ is the control input, $g(\cdots)$ is a linear activation function, $B \in \mathbb{R}$ is a tunable bias introduced to the network and $w_1 \in [-1, 1]$ and $w_2 \in [0, 1]$ are the adaptive weights associated with the bias and the CFMFAC output respectively. These weights can be computed as (15).

$$w_n(k) = w_n(k-1) + \Delta w_n(k) \tag{15}$$

where $n = 1, 2$ is the weight number and $\Delta w_n(k)$ is the change of weight number n, which is calculated as (16) and (17) with regards to the following assumption:

Assumption 3: The partial derivatives associated with the NN is continuous.

This is a reasonable assumption as discrete-time models are numerical discretization of continuous-time models, and the approximation accuracy increases when using small time steps [29].

$$\Delta w_n(k) = -\eta_n \frac{\partial J}{\partial w_n} \tag{16}$$

$$\frac{\partial J}{\partial w_n} = \frac{\partial J}{\partial e} \cdot \frac{\partial e}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial w_n} \tag{17}$$

where $\eta_n \in \mathbb{R}$ is the learning rate at weight number $n = 1, 2$. Equation (17) is further simplified as (18) and (19).

$$\frac{\partial J}{\partial w_1} = -e \cdot sgn\left(\frac{\partial y}{\partial u}\right) \cdot B \tag{18}$$

$$\frac{\partial J}{\partial w_2} = -e \cdot sgn\left(\frac{\partial y}{\partial u}\right) \cdot v \tag{19}$$

The designed NN is combined with the CFMFAC to form the NNCFMFAC. Although CFMFAC has several tunable control parameters that can be adjusted for better performance, control parameters ρ and λ have greater influences on the output response of CFMFAC operation where both parameters affect the response speed [30]. Experiments conducted in [15] and [19] also show that both control parameters produce a similar trend during parameter self-tuning operation. To reduce complexity of the neural network, instead of separating the two control gains for parameter self-tuning, the output of the CFMFAC can be directly connected to the neural

network input node to self-tune the speed of the controller. The tunable bias connected with an adaptive weight helps offset the control input which also suppresses effects of dead-zones in TMM systems. The proposed NNCFMFAC control strategy is shown in Fig. 3.

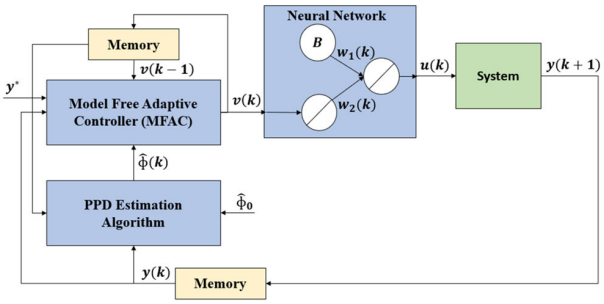

FIGURE 3. Block diagram of proposed NNCFMFAC.

III. CONVERGENCE ANALYSIS

A nonlinear system satisfying *Assumptions 1*, *2*, and *3* is controlled by the proposed NNCFMFAC for a regulation problem where the desired signal is a constant y^*, then the following theorem can be obtained for any $\lambda > \lambda_{min}$ and $\eta_n < \eta_{max}$.

Theorem 1: For a SISO discrete nonlinear system, the NNCFMFAC has the following properties.

1) The tracking error for the system output converges, and $\lim_{k\to\infty} |e(k)| = 0$.

2) For all k, the output, $y(k)$, and input, $u(k)$ are bounded.

The tracking error is defined as (20).

$$e(k+1) = y^* - y(k+1) \tag{20}$$

Substituting the CFDL as in (3) into (20) and taking the absolute value leads to (21).

$$|e(k+1)| = \left|y^* - y(k) - \phi(k)\Delta u(k)\right| \tag{21}$$

where $\Delta u(k)$ can be derived from (14) and (15) as in (22).

$$\begin{aligned}\Delta u(k) &= B\Delta w_1(k) + v(k)\, w_2(k) - v(k-1)\, w_2(k-1)\\ &= B\Delta w_1(k) + v(k)\,\Delta w_2(k) + \Delta v(k)\, w_2(k-1)\end{aligned} \tag{22}$$

Substituting (22) into (21), the following is obtained.

$$\begin{aligned}|e(k+1)| = |e(k) - \phi(k)\,(&B\Delta w_1(k) + v(k)\,\Delta w_2(k)\\ &+ \Delta v(k)\, w_2(k-1))|\end{aligned} \tag{23}$$

From (7), Δv can be obtained and substituted in (23) to produce (24).

$$|e(k+1) = \left|e(k) - \phi(k)\left(B\Delta w_1(k) + v(k)\,\Delta w_2(k) + \left(\frac{\rho\hat{\phi}(k)}{\lambda + \left|\hat{\phi}(k)\right|^2} e(k)\right) w_2(k-1)\right)\right| \tag{24}$$

and from (16), $\Delta w_1(k)$ and $\Delta w_2(k)$ is substituted in (24), hence (25) can be derived.

$$\begin{aligned}|e(k+1)| = \Bigg|1 - \frac{w_2(k-1)\,\phi(k)\,\rho\hat{\phi}(k)}{\lambda + \left|\hat{\phi}(k)\right|^2}\\ - \phi(k)\left(B^2\eta_1 + v(k)^2\eta_2\right)\Bigg| |e(k)|\\ |e(k+1)| \le |1 - w_2(k-1)\, d_1 - d_2|\, |e(k)|\end{aligned} \tag{25}$$

where

$$\begin{aligned}d_1 &= \frac{\phi(k)\,\rho\hat{\phi}(k)}{\lambda + \left|\hat{\phi}(k)\right|^2}\\ d_2 &= \phi(k)\, B^2\eta_1 + \phi(k)\, v(k)^2\eta_2\end{aligned} \tag{26}$$

From assumptions that the reset algorithm in (11), $\hat{\phi}(k) > \varepsilon$ is guaranteed. Let $\lambda_{min} = L^2/4$ and $\phi(k) \le L$ from *Assumption 2*, $0 \le \rho < 1$ and $\left(\sqrt{\lambda}\right)^2 + \left|\hat{\phi}(k)\right|^2 \ge 2\sqrt{\lambda}\hat{\phi}(k)$, the following inequality is formed.

$$0 < d_1 \le \frac{\phi(k)\,\rho\hat{\phi}(k)}{\lambda + \left|\hat{\phi}(k)\right|^2} \le \frac{L\rho\hat{\phi}(k)}{2\sqrt{\lambda}\hat{\phi}(k)} \le \frac{L\rho}{2\sqrt{\lambda}} < 1 \tag{27}$$

Since $\eta_n < \eta_{nmax}$, let $\eta_{1max} = a_1/LB^2$, $\eta_{2max} = a_2/Lv(k)^2$, and $0 \le a_1 + a_2 < 1$, therefore $0 \le d_2 < 1$ and the following inequality can be formed from (25).

$$\begin{aligned}|e(k+1)| &\le |d_3|\, |e(k)|\\ &\le |d_3^2|\, |e(k-1)| \le \cdots \le |d_3^k|\, |e(1)|\end{aligned} \tag{28}$$

where $d_3 = |1 - w_2(k-1)\, d_1 - d_2|$. Since $0 \le w_2(k-1) \le 1$, $0 \le d_1 \le 1$, and $0 \le d_2 \le 1$, therefore $0 < d_3 < 1$. This guarantees that $\lim_{k\to\infty} |e(k)| = 0$. As $y^*(k)$ is a constant value, convergence of $e(k)$ indicates that $y(k)$ is also bounded. The following shows that $u(k)$ is bounded.

From (14), since B is a constant and w_1, w_2 are bounded, therefore the control input is bounded if the CFMFAC output, v is bounded. According to $\left(\sqrt{\lambda}\right)^2 + \left|\hat{\phi}(k)\right|^2 \ge 2\sqrt{\lambda}\hat{\phi}(k)$, and (6) the following inequality in (29) can be derived.

$$\begin{aligned}|\Delta v(k)| &= \left|\frac{\rho\hat{\phi}(k)}{\lambda + \left|\hat{\phi}(k)\right|^2}\right| |e(k)|\\ &\le \left|\frac{\rho\hat{\phi}(k)}{2\sqrt{\lambda}\hat{\phi}(k)}\right| |e(k)|\\ &\le \left|\frac{\rho}{2\sqrt{\lambda_{\min}}}\right| |e(k)|\\ &\le M_1\, |e(k)|\end{aligned} \tag{29}$$

where $M_1 = \rho/2\sqrt{\lambda_{min}}$ is a bounded constant. From (28) and (29) we have

$$\begin{aligned}|v(k)| &\le |v(k-1) + \Delta v(k)|\\ &\le |v(k) - v(k-1)| + |v(k-1)|\end{aligned}$$

$$\begin{aligned}
&\leq |v(k)-v(k-1)|+|v(k-1)-v(k-2)|+|v(k-2| \\
&\leq |\Delta v(k)|+|\Delta v(k-1)|+\cdots+|v(1)| \\
&\leq M_1\left(|e(k)|+|e(k-1)|+\cdots+|e(2)|\right)+|v(1)| \\
&\leq M_1\left(|d_3^{k-1}|\,|e(1)|+\cdots+|d_3|\,|e(1)|\right)+|v(1)| \\
&\leq M_1\frac{|d_3|}{1-|d_3|}|e(1)|+|v(1)| \qquad (30)
\end{aligned}$$

The following can be derived from (22)

$$\begin{aligned}
|\Delta u(k)| &\leq \eta_1 B^2|e(k)| \\
&\quad+\eta_2\left(M_1\frac{|d_3|}{1-|d_3|}|e(1)|+|v(1)|\right)^2|e(k)| \\
&\quad+w_2(k-1)M_1|e(k)| \\
&\leq M_2|e(k)| \qquad (31)
\end{aligned}$$

where

$$\begin{aligned}
M_2 = \eta_1 B^2+\eta_2\left(M_1\frac{|d_3|}{1-|d_3|}|e(1)|+|v(1)|\right)^2 \\
+w_2(k-1)M_1 \qquad (32)
\end{aligned}$$

therefore

$$\begin{aligned}
|u(k)| &\leq |u(k)-u(k-1)|+|u(k-1)| \\
&\leq |u(k)-u(k-1)|+|u(k-1)-u(k-2)|+|u(k-2| \\
&\leq |\Delta u(k)|+|\Delta u(k-1)|+\cdots+|u(1)| \\
&\leq M_2\left(|e(k)|+|e(k-1)|+\cdots+|e(2)|\right)+|u(1)| \\
&\leq M_2\left(|d_3^{k-1}|\,|e(1)|+\cdots+|d_3|\,|e(1)|\right)+|u(1)| \\
&\leq M_2\frac{|d_3|}{1-|d_3|}|e(1)|+|u(1)| \qquad (33)
\end{aligned}$$

Hence, this proves that the control input is bounded. For the proposed NNCFMFAC, the adaptive weights from the designed NN works together with the PPD estimation algorithm in (10) to reduce the system error. From (28), the adaptive parameters can be observed to help reduce d_3 hence obtaining faster convergence. This will also be verified by simulations and experiments in the following sections.

IV. SIMULATION RESULTS

This section aims to evaluate the effectiveness of the proposed controller on a nonlinear system. Simulations are conducted on a SISO nonlinear discrete model, followed by the control on actual TMM. The system in (34), is a dynamic nonlinear system, which has certain nonlinear characteristics in a TMM system [4]. The desired set-point (35) is a time-varying output, which reflects applications of a TMM driven system to manipulate objects. Furthermore, the model is used to evaluate other neuro based improved MFACs on nonlinear systems in previous research [15], [19]. The system and setpoint are as follow:

$$\begin{aligned}
y(k+1) &= \frac{2.5y(k)\,y(k-1)}{1+y^2(k)+y^2(k-1)}+1.2u(k) \\
&\quad+0.09u(k)\,u(k-1)+1.6u(k-2) \\
&\quad+0.7\sin(0.5(y(k)+y(k-1))) \\
&\quad\times\cos(0.5(y(k)+y(k-1))) \qquad (34)
\end{aligned}$$

$$y^*(k+1) = 5\sin\left(\frac{k\pi}{50}\right)+2\cos\left(\frac{k\pi}{20}\right) \qquad (35)$$

where y denote the actual output of the system, u is the control input, k is the iteration, and y^* is the desired output of the system. All values in this section are dimensionless. The initial conditions are set as: $y(0) = y(1) = 0$ and $u(0) = 0$. In this simulation, the proposed NNCFMFAC method is compared with other data-driven controllers such as the particle swarm optimization proportional integral derivative (PSOPID) controller and the conventional CFMFAC. Besides this, to evaluate the effectiveness of the introduced bias, a neural network compact form model free adaptive controller without bias (NNCFMFAC-WB) is temporarily introduced for ablation analysis. The NNCFMFAC-WB represents the NNCFMFAC without the advantage of the bias and additional adaptive weight, hence it is designed to be a "1-1" neural network structure with one neuron in the input layer and output layer. Furthermore, an advanced adaptive controller combining a back propagation neural network architecture and the CFMFAC (BPNNCFMFAC) is designed according to [18] and [19] and is used to evaluate the performance of the proposed controller against other intelligent parameter tuning algorithms. The structure of the back propagation neural network is "5-8-2" which means the number of neurons in the input layer, hidden layer and output layer are 5, 8, and 2, respectively. This neural network consists of 56 adaptive weights.

The gains of the PID used in this work is obtained by using particle swarm optimization (PSO) technique [31] with 50 swarm particles and 50 iterations. The inertia weight chosen are from 0.9 to 0.4 and the acceleration coefficients are set at 1.48. The control parameter values obtained from PSO are $K_p = 0.0500$, $K_i = 0.2012$, and $K_d = 0.0500$. The root mean squared error (RMSE) is used as a cost function and is defined as (36).

$$RMSE = \sqrt{\frac{\sum_k^K e(k)^2}{K}} \qquad (36)$$

where K is the total number of sampling data and $e(k)$ is the difference between the desired output and the actual output at instance k. For all CFMFAC strategies, the controller parameters are set to $\eta = 0.1$, $\mu = 0.1$, $\varepsilon = 0.1$, and $\hat{\phi}_0 = 8$. The controller parameters ρ for CFMFAC and NNCFMFAC are 0.5 and 1 respectively, while the BPNNCFMFAC uses NN to tune both ρ and λ. The values of λ in CFMFAC, NNCFMFAC-WB, and NNCFMFAC are set to 0.3, 2.8, and 0.3, respectively. The learning rate, η_1 and inertia coefficient α, used for BPNNCFMAC are $\eta_1 = 4.0\times10^{-2}$ and $\alpha = 0.1$, while values set for NNCFMFAC are $\eta_1 = 3.5\times10^{-6}$, $\eta_2 = 1.5\times10^{-2}$, and $B = 343$. Tuning of control parameters are obtained by the grid search method [32] and further fine-tuned with smaller increments. The consistency of the results compared to [15] and [19] validates that the chosen

parameters are significant and are suitable to be used for this simulation.

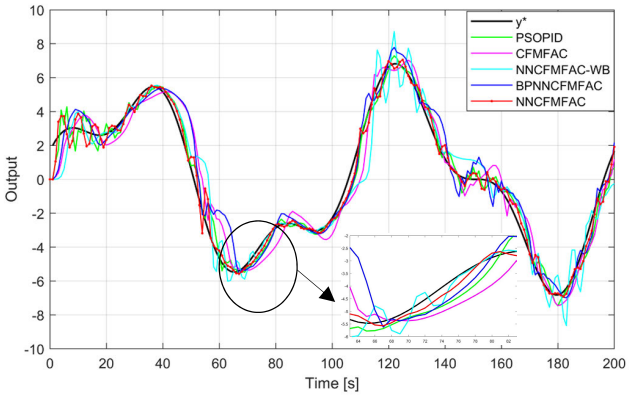

FIGURE 4. Tracking curves on nonlinear system.

The tracking performance of the designed controllers is displayed in Fig. 4 and summarized in Fig. 5. In the first 20 seconds from Fig. 4, it can be observed that NNCFMFAC has the fastest response followed by the PSOPID and BPNNCFMFAC. Although all control strategies can be seen to have decent tracking performance, the proposed NNCFMFAC shows the closest tracking to the desired set point. This can be observed in the detailed view in Fig. 4. Results from Fig. 5 show the tracking error of the NNCFMFAC is significantly reduced by approximately 45%, 47%, 50%, and 53% in terms of RMSE when compared to PSOPID, BPNNCFMAC, CFMFAC, and NNCFMFAC-WB methods, respectively. The BPNNCFMFAC also shows slight improvement of 5% compared to CFMFAC in terms of RMSE.

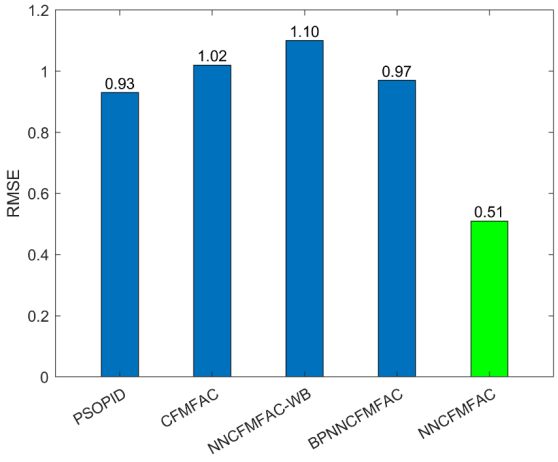

FIGURE 5. Evaluation results of each controller for nonlinear system.

From Fig. 6, it can be observed that the PSOPID and NNCFMFAC are able to sustain errors below 2. Alternately, the BPNNCFMFAC and CFMFAC controllers had a range of errors below 3 and NNCFMAC-WB below 4, due to large fluctuations. These fluctuations can be observed from the control signals in Fig. 7. From graphs in Fig. 8 and 9, the NNCFMFAC adaptive ability to tune the adaptive weights and pseudo-partial derivative (PPD) sensitively and coherently at each operating point is shown. On the other hand, Fig. 10 shows the parameter self-tuning results of the BPNNCFMFAC. Although the tuning parameters can be observed to tune parameters sensitively at each moment, the curves have a certain degree of similarity and can be explained by theoretical formulation in (37), where λ and ρ play roles in preventing input from excessive deviation. This may also explain the smoother control signal when using the BPNNCFMFAC in Fig. 7. Furthermore, the results are also in line with previous studies conducted by [15] and [19]. This strengthens the cause for developing the NNCFMFAC according to the FFAC to optimize the NN by combining the self-tuned parameters.

$$\Delta v(k) = \frac{\rho\phi(k)}{\left(\lambda + |\phi(k)|^2\right)}\left(y^*(k+1) - y(k)\right) \tag{37}$$

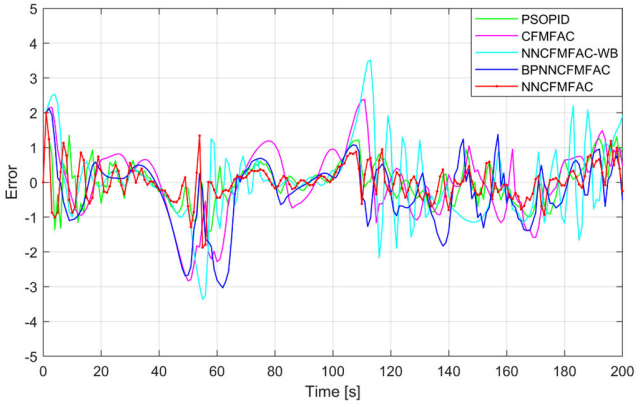

FIGURE 6. Tracking errors of nonlinear system.

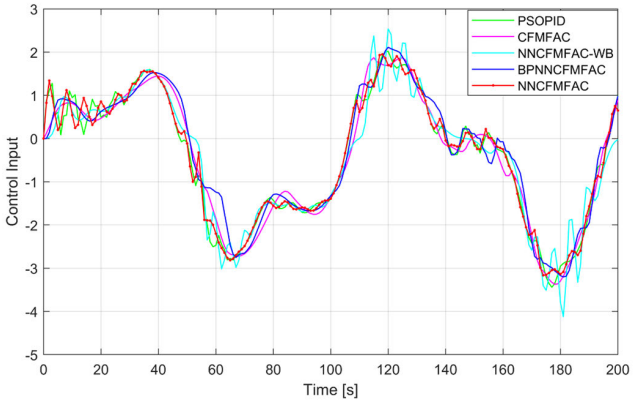

FIGURE 7. Control input of nonlinear system.

The performance of the NNCFMFAC can also be compared in terms of the complexity of the neural network. The BPNNCFMAC consists of 56 adaptive weights while the neural network of the proposed controller comprises of

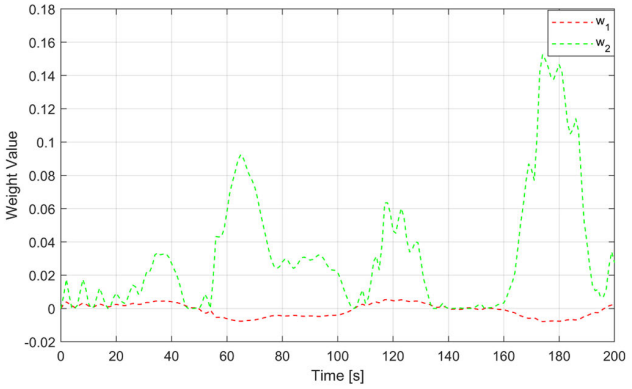

FIGURE 8. Adaption of neural network weights.

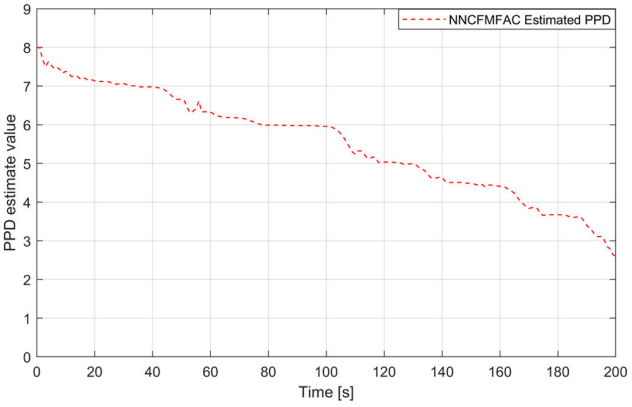

FIGURE 9. PPD estimated values.

only 2 adaptive weights. The proposed controller shows a reduction of 96.42% in total adaptive weights, hence a major reduction in complexity is achieved while attaining lower RMSE. This can also be compared with other NN-based CFMFAC strategies in recent scientific literatures. The proposed NNCFMFAC showed a RMSE reduction of 27.23% and a reduction of 99.55% in terms of total adaptive weights against a parameter self-tuning compact form model-free adaptive controller based on long short-term memory neural network proposed by [19].

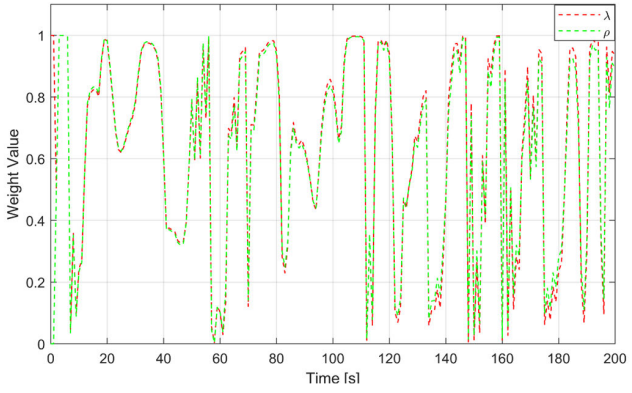

FIGURE 10. Parameter self-tuning results of λ and ρ for BPNNCFMFAC.

Based on the evidence presented so far, simulation results on a nonlinear system showed the proposed NNCFMFAC is superior in terms of tracking accuracy compared to other data-driven control strategies. This indicates that combining a two layered neural network with two adaptive weights is adequate to significantly improve the tracking performance of the CFMFAC. From a practical point of view, the proposed controller can achieve satisfactory tracking performance for a discrete nonlinear system such as TMM driven systems with time-varying set-points by utilizing only the input-output data of the system. Despite the reduction in adaptive weights, the NNCFMFAC-WB still has a lower complexity than the proposed controller with a reduction of about one adaptive weight. However, according to the FFAC in (12), the applied penalty is insignificant at low total number of weights [21]. Furthermore, the proposed controller showed a 53% improvement in accuracy compared to the simpler NNCFMFAC-WB. This shows that designing a NN architecture with a lower number of neurons affect the approximation performance of the NN, hence the proposed architecture is a better optimized solution. Despite the prominent performance of the NNCFMFAC from simulation results, the performance can be further validated by real-time experiments to ensure the capability of the controller on TMM driven systems.

V. EXPERIMENTAL RESULTS

As discussed in the introduction, the main objective of this study is to evaluate the performance of a neural network compact form model-free adaptive controller (NNCFMFAC) with reduced number of weights on a highly nonlinear thin McKibben muscle (TMM) system as set up in Fig. 1. This TMM system closely mimics the human finger from the metacarpal bone (link 1) to the proximal bone (link 2) in terms of muscle actuation, ligaments, tendons, and motion. Due to the softness of the materials used, this system can be used for applications such as gripping and manipulating objects while reducing hazardous impacts.

The platform is driven by a TMM attached to both links by artificial tendons made by Dyneema (high-density polyethylene). Two inertial measurement units (IMUs) are placed on the links to obtain the joint angle. The joint angle is read by a microcontroller (Arduino Mega 2560), where the proposed controller uses the input and output signals to obtain the control signal for the system. The microcontroller produces pulse width modulation (PWM) output that is converted to 0-10 VDC by a PWM to voltage converter which is then fed to an electro-pneumatic regulator (SMC ITV2050-312BL). This regulator controls the pressure input to the TMM which contracts the muscle, hence producing the force to pull link 2 for flexion motion as shown in Fig. 11.

Similar to human joints, ligaments are attached to the bones to ensure joints return to its resting position when muscles are not actuated. These ligaments are artificially replaced by using 1 mm thick silicone sheets to ensure stability at joint. The highly nonlinear characteristics of the TMM system such as hysteresis at muscles and valves, large dead-zones due to

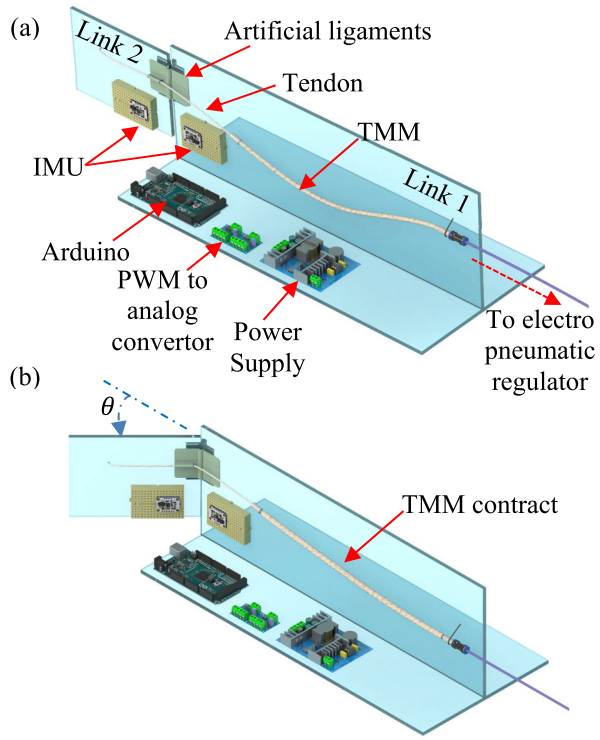

FIGURE 11. TMM system (a) TMM system at rest position (b) TMM system during flexion motion.

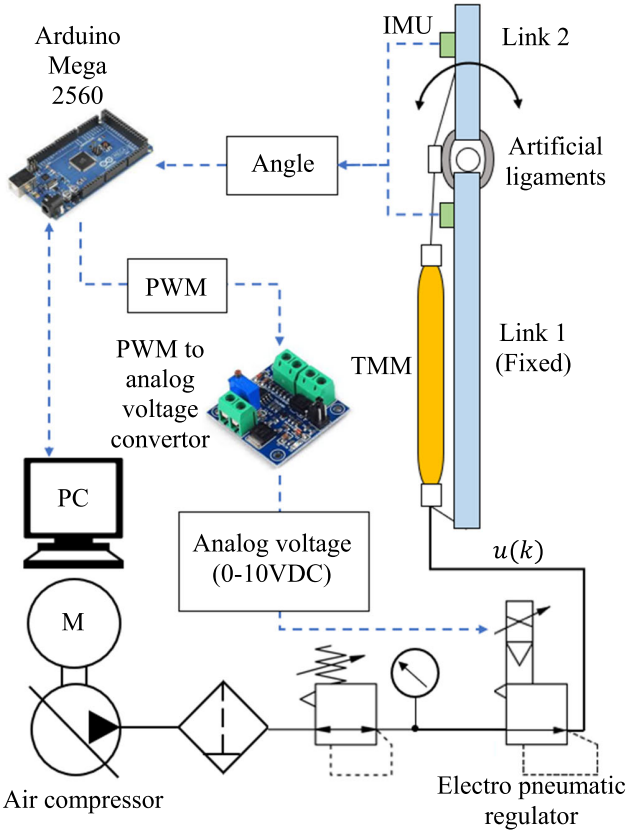

FIGURE 12. Schematic diagram for TMM system.

slack regions at tendons, and time-varying properties due to complicated braiding structure of muscles makes it a particularly challenging system for motion control. A schematic diagram of the control structure is presented in Fig. 12.

The performance of the controller is accessed by a single degree-of-freedom angular position tracking at joint according to given desired trajectories. The proposed controller is compared with other data-driven methods such as the well-known and established PID controller and CFMFAC controller. An improved back propagation neural network compact form model-free adaptive controller (BPNNCFMFAC) with self-tuning capabilities [18], [19] is also used as a benchmark.

For analysis, two different set-points were used to assess the controllers according to applications of TMM such as gripping and manipulating objects. Case I and Case II are as (38) and (39) respectively.

$$y(t) = \begin{cases} 0, & 0 \le t < 1 \\ 5, & 1 \le t \le 50 \end{cases} \tag{38}$$

$$y(t) = \begin{cases} 0, & 0 \le t < 1 \\ 6, & 1 \le t < 10 \\ \frac{1}{2}t + 2, & 10 \le t < 12 \\ 8, & 12 \le t \le 20 \end{cases} \tag{39}$$

The values used for the proportional gain, integral gain, and derivative gain of the PID are 0.5, 2.0, and 0.3 respectively.

For all CFMFAC strategies, the controller parameters are set to $\eta = 0.1$, and $\varepsilon = 0.1$. For CFMFAC, $\hat{\phi}_0 = 12$ is used, while BPNNCFMFAC and NNCFMFAC use $\hat{\phi}_0 = 8$. Both CFMFAC and NNCFMFAC use $\mu = 0.1$ while BPNNCFMFAC is set to $\mu = 7$. Learning rate and inertia coefficient for BPNNCFMFAC is 3.85×10^{-3} and 0.1, respectively. The learning rates and bias set for NNCFMFAC are $\eta_1 = 5 \times 10^{-4}$, $\eta_2 = 5 \times 10^{-2}$ and $B = 35$. Tuning of control parameters are obtained by the grid search method [32] and further fine-tuned by smaller increments. The experiments were run on MATLAB Simulink with sampling time of 0.05 s. The settling time, with 2% settling time threshold, percentage of overshoot and steady state error were measured for comparative analysis.

From Fig. 13, it can be observed that the proposed controller reaches steady state faster with no overshoot and smooth tracking response at transient state. It can also be observed that the controller is better at eliminating the effects of the dead-zones of TMM system. From Table 1, the response shows a significant improvement in terms of settling time by close to 86%, 76%, and 74% faster compared to the PID controller, CFMFAC and BPNNCFMFAC, respectively. The proposed controller also shows notable reduction in overshoot as opposed to PID controller, CFMFAC and BPNNCFMFAC by 19%, 33%, and 57%, respectively. Findings

from the error signal in Fig. 14 shows that the error at joint angle for the NNCFMFAC converges to zero with improved speed compared to the others. The improvement in convergence speed appears to agree with the analytical proofs formulated in the previous section.

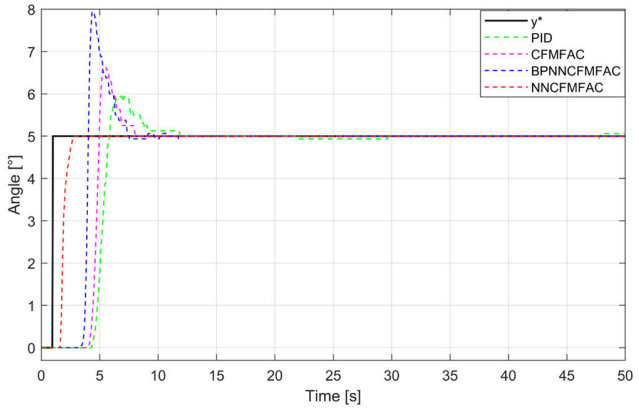

FIGURE 13. Angle tracking results of TMM system experiment for Case I.

TABLE 1. Step response characteristics of TMM system experiment for Case I.

	t_s (s)	$OS(\%)$	$e_{ss}(°)$
PID	11.8	19	0.06
CFMFAC	7.05	33	0.00
BPNNCFMFAC	6.40	57	0.00
NNCFMFAC	1.70	0	0.00

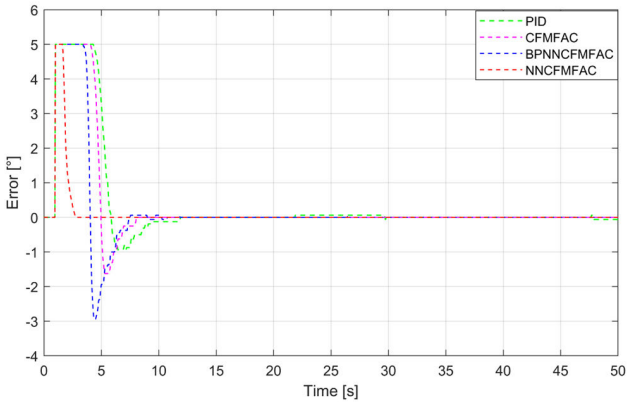

FIGURE 14. Angle tracking error signal of TMM system experiment for Case I.

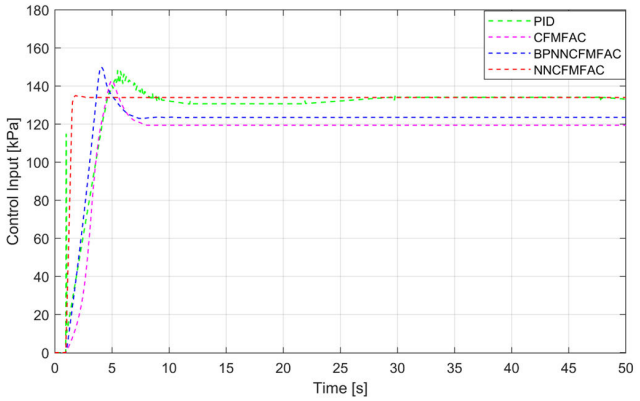

FIGURE 15. Control input of TMM system experiment for Case I.

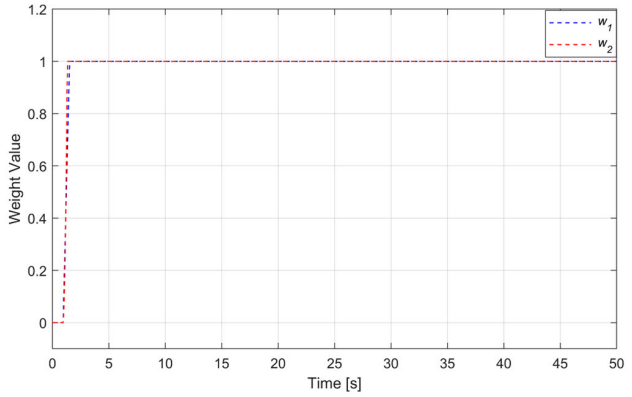

FIGURE 16. Adaptive weights of NNCFMFAC for Case I.

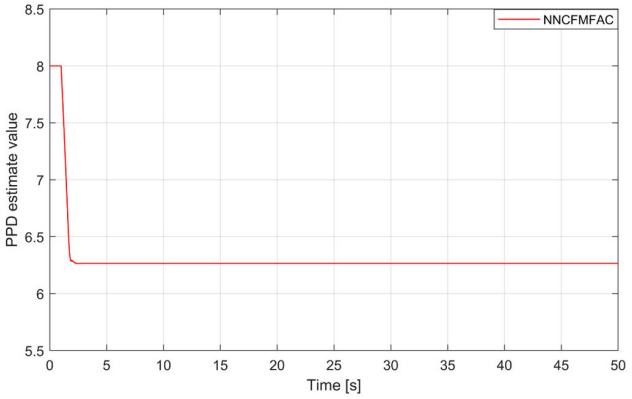

FIGURE 17. PPD estimated value of NNCFMFAC for Case I.

Fig. 15 presents the control signal for Case I. Differences in control signals are observed during the steady state response of the controllers. These discrepancies might be due to the dependence of the state of the system on the previous states or commonly known as hysteresis which is a well-known nonlinear characteristic in TMM [4]. From Fig. 16 and 17, the adaptive weights and PPD estimation are presented, respectively. The adaptive elements can be seen to coherently adapt together at each sample time. Overall, these findings suggest that the NNCFMFAC offers a fast response, ability to suppress the overshoot and handle dead-zones which are critical for TMM systems especially when TMMs are used in an antagonistic configuration with large dead-zones for high-speed gripping applications.

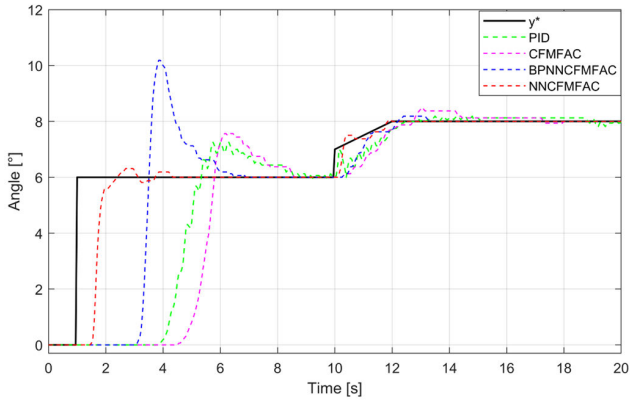

FIGURE 18. Angle tracking of TMM system experiment for Case II.

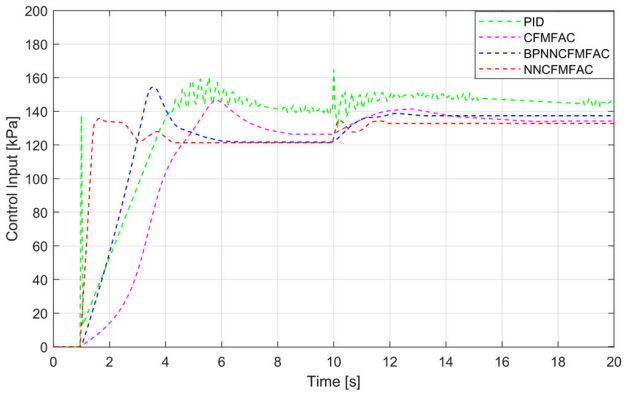

FIGURE 21. Control input of TMM system experiment for Case II.

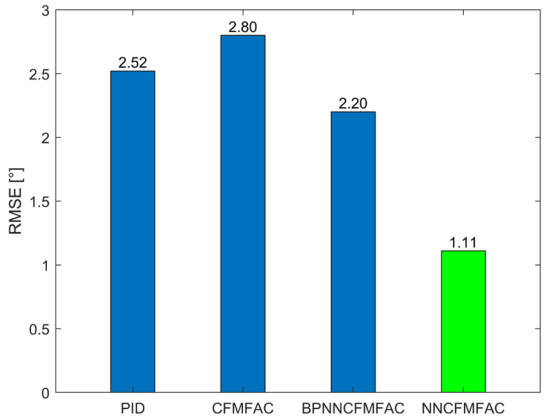

FIGURE 19. Evaluation results of each controller for Case II.

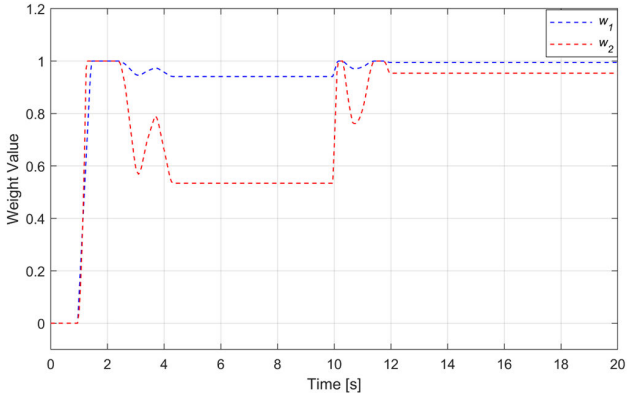

FIGURE 22. Adaptive weights of NNCFMFAC for Case II.

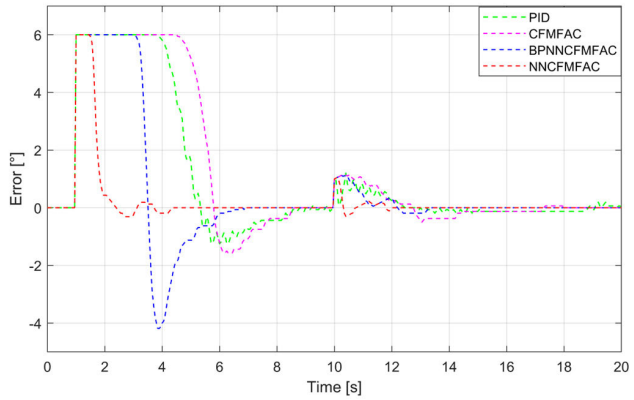

FIGURE 20. Angle tracking error signal of TMM system experiment for Case II.

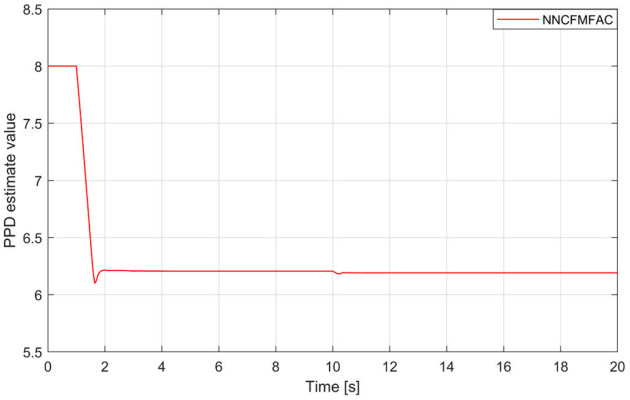

FIGURE 23. PPD estimated value of NNCFMFAC for Case II.

The second case is coordinated to evaluate the performance of the proposed controller in tracking the angle for a time-varying set-point. Fig. 18 presents the tracking response of the controllers for Case II. From the figure, it is observed that the proposed controller produces a faster and smoother tracking response in comparison with the other controllers. Further analysis in Fig. 19 supports this observation as the proposed controller demonstrates having the lowest RMSE with reductions of approximately 56%, 60%, and 50% compared to the PID controller, CFMFAC, and BPNNCFMFAC, respectively.

The error signal graph in Fig. 20 consistently points out the faster convergence of error for the proposed controller. Fig. 21 shows the control signal for Case II, while Fig. 22 and 23 show the adaptive parameters in synchronization with each other. From a practical implementation point of view, the findings imply that the enhanced response of the NNCFMFAC has improved tracking accuracy which is an

important attribute in TMM driven system for manipulating objects sensitively.

VI. CONCLUSION

The aim of this work is to develop a neural network compact form model-free adaptive control (NNCFMFAC) with reduced complexity and to evaluate its performance. One of the significant key findings in this study is the ability of a neural network with minimum adaptive weights to enhance the performance of the CFMFAC in terms of convergence speed, which is proven analytically, by simulation on a nonlinear system and experiments on a TMM driven system. Furthermore, the NNCFMFAC can overcome the dead-zone issue of TMM driven systems, while suppressing overshoots compared to other data-driven controllers. Evidence from the study suggests that, although smaller NN architectures are known to have limitations in solving problems such as the XOR problem [26], when combined with the PPD estimation ability of the CFMFAC, the controller has crucial merits to be applied in TMM applications such as fast-paced gripping operations and fine-precision object manipulation. Additionally, the simple and standard architecture of the NN allows control engineers to easily retrofit the designed NN into the CFMFAC for any SISO discrete system. From a sustainable perspective, the reduced complexity of the neural network is more convenient for maintenance and troubleshoot in real world engineering processes. Despite the merits of the proposed solution, selection of the control parameters for improved CFMFACs is still an open problem. Although the grid search method is considered a systematic approach, extensive amounts of experiments need to be carried out for this method. As an extension, it would be interesting to explore other model-free tuning methods for the proposed controller.

ACKNOWLEDGMENT

The authors would like to express their gratitude to Universiti Teknologi Malaysia, vote no (5F137) which makes this research viable and effective.

REFERENCES

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, May 2015, doi: 10.1038/nature14543.

[2] C. Zhang, P. Zhu, Y. Lin, W. Tang, Z. Jiao, H. Yang, and J. Zou, "Fluid-driven artificial muscles: Bio-design, manufacturing, sensing, control, and applications," *Bio-Design Manuf.*, vol. 4, no. 1, pp. 123–145, Mar. 2021, doi: 10.1007/s42242-020-00099-z.

[3] Y. Yuan, Y. Yu, and L. Guo, "Nonlinear active disturbance rejection control for the pneumatic muscle actuators with discrete-time measurements," *IEEE Trans. Ind. Electron.*, vol. 66, no. 3, pp. 2044–2053, Mar. 2019, doi: 10.1109/TIE.2018.2838061.

[4] S. Kurumaya, H. Nabae, G. Endo, and K. Suzumori, "Design of thin Mckibben muscle and multifilament structure," *Sens. Actuators A, Phys.*, vol. 261, pp. 66–74, Jul. 2017, doi: 10.1016/j.sna.2017.04.047.

[5] T. Abe, S. Koizumi, H. Nabae, G. Endo, K. Suzumori, N. Sato, M. Adachi, and F. Takamizawa, "Fabrication of '18 weave' muscles and their application to soft power support suit for upper limbs using thin Mckibben muscle," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2532–2538, Jul. 2019, doi: 10.1109/LRA.2019.2907433.

[6] Y. Peng, Y. Liu, Y. Yang, N. Liu, Y. Sun, Y. Liu, H. Pu, S. Xie, and J. Luo, "Development of continuum manipulator actuated by thin Mckibben pneumatic artificial muscle," *Mechatronics*, vol. 60, pp. 56–65, Jun. 2019, doi: 10.1016/j.mechatronics.2019.05.001.

[7] A. A. M. Faudzi, M. R. M. Razif, G. Endo, H. Nabae, and K. Suzumori, "Soft-amphibious robot using thin and soft Mckibben actuator," in *Proc. IEEE Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2017, pp. 981–986, doi: 10.1109/AIM.2017.8014146.

[8] Y. Chen, N. Sun, D. Liang, Y. Qin, and Y. Fang, "A neuroadaptive control method for pneumatic artificial muscle systems with hardware experiments," *Mech. Syst. Signal Process.*, vol. 146, Jan. 2021, Art. no. 106976, doi: 10.1016/j.ymssp.2020.106976.

[9] C.-T. Chen, W.-Y. Lien, C.-T. Chen, M.-J. Twu, and Y.-C. Wu, "Dynamic modeling and motion control of a cable-driven robotic exoskeleton with pneumatic artificial muscle actuators," *IEEE Access*, vol. 8, pp. 149796–149807, 2020, doi: 10.1109/ACCESS.2020.3016726.

[10] X. Guan, Z. He, M. Zhang, and H. Xia, "Filtering-error constrained angle tracking adaptive learning fuzzy control for pneumatic artificial muscle systems under nonzero initial errors," *IEEE Access*, vol. 10, pp. 41828–41838, 2022, doi: 10.1109/ACCESS.2022.3168564.

[11] L. Zhao, H. Cheng, J. Zhang, and Y. Xia, "Adaptive control for a motion mechanism with pneumatic artificial muscles subject to dead-zones," *Mech. Syst. Signal Process.*, vol. 148, Feb. 2021, Art. no. 107155, doi: 10.1016/j.ymssp.2020.107155.

[12] L. Zhao, H. Cheng, Y. Xia, and B. Liu, "Angle tracking adaptive backstepping control for a mechanism of pneumatic muscle actuators via an AESO," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4566–4576, Jun. 2019, doi: 10.1109/TIE.2018.2860527.

[13] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013, doi: 10.1016/j.ins.2012.07.014.

[14] Q. Ai, D. Ke, J. Zuo, W. Meng, Q. Liu, Z. Zhang, and S. Q. Xie, "High-order model-free adaptive iterative learning control of pneumatic artificial muscle with enhanced convergence," *IEEE Trans. Ind. Electron.*, vol. 67, no. 11, pp. 9548–9559, Nov. 2020, doi: 10.1109/TIE.2019.2952810.

[15] Y. Yang, C. Chen, and J. Lu, "An improved partial-form MFAC design for discrete-time nonlinear systems with neural networks," *IEEE Access*, vol. 9, pp. 41441–41455, 2021, doi: 10.1109/ACCESS.2021.3065311.

[16] Y. Li, Q. Liu, W. Meng, Y. Xie, Q. Ai, and S. Q. Xie, "MISO model free adaptive control of single joint rehabilitation robot driven by pneumatic artificial muscles," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Boston, MA, USA, Jul. 2020, pp. 1700–1705, doi: 10.1109/AIM43001.2020.9158805.

[17] N. Dong, W. Lv, S. Zhu, Z. Gao, and C. Grebogi, "Model-free adaptive nonlinear control of the absorption refrigeration system," *Nonlinear Dyn.*, vol. 107, no. 2, pp. 1623–1635, Jan. 2022, doi: 10.1007/s11071-021-06964-5.

[18] C. Chen and J. Lu, "Design of self-tuning SISO partial-form model-free adaptive controller for vapor-compression refrigeration system," *IEEE Access*, vol. 7, pp. 125771–125782, 2019, doi: 10.1109/ACCESS.2019.2939261.

[19] Y. Yang, C. Chen, and J. Lu, "Parameter self-tuning of SISO compact-form model-free adaptive controller based on long short-term memory neural network," *IEEE Access*, vol. 8, pp. 151926–151937, 2020, doi: 10.1109/ACCESS.2020.3017532.

[20] Y. Wang, S. Li, and B. Zhang, "General regression neural network-based data-driven model-free predictive functional control for a class of discrete-time nonlinear systems," *Nonlinear Dyn.*, vol. 107, no. 1, pp. 953–966, Jan. 2022, doi: 10.1007/s11071-021-06991-2.

[21] P. G. Benardos and G.-C. Vosniakos, "Optimizing feedforward artificial neural network architecture," *Eng. Appl. Artif. Intell.*, vol. 20, no. 3, pp. 365–382, Apr. 2007, doi: 10.1016/j.engappai.2006.06.005.

[22] L. K. Tan, Y. M. Liew, E. Lim, and R. A. McLaughlin, "Convolutional neural network regression for short-axis left ventricle segmentation in cardiac cine MR sequences," *Med. Image Anal.*, vol. 39, pp. 78–86, Jul. 2017, doi: 10.1016/j.media.2017.04.002.

[23] S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures," *IEEE Access*, vol. 6, pp. 64270–64277, 2018, doi: 10.1109/ACCESS.2018.2877890.

[24] M. Y. Rafiq, G. Bugmann, and D. J. Easterbrook, "Neural network design for engineering applications," *Comput. Struct.*, vol. 79, no. 17, pp. 1541–1552, 2001, doi: 10.1016/S0045-7949(01)00039-6.

[25] A. A. Kulaksiz and R. Akkaya, "A genetic algorithm optimized ANN-based MPPT algorithm for a stand-alone PV system with induction motor drive," *Sol. Energy*, vol. 86, no. 9, pp. 2366–2375, Sep. 2012, doi: 10.1016/j.solener.2012.05.006.
[26] J. Zou, Y. Han, and S.-S. So, "Overview of artificial neural networks," *Artif. Neural Netw.*, vol. 458, pp. 14–22, Sep. 2009, doi: 10.1007/978-1-60327-101-1_2.
[27] P. Netrapalli, "Stochastic gradient descent and its variants in machine learning," *J. Indian Inst. Sci.*, vol. 99, no. 2, pp. 201–213, Jun. 2019, doi: 10.1007/s41745-019-0098-4.
[28] Y. Li, C. Fan, Y. Li, Q. Wu, and Y. Ming, "Improving deep neural network with multiple parametric exponential linear units," *Neurocomputing*, vol. 301, pp. 11–24, Aug. 2018, doi: 10.1016/j.neucom.2018.01.084.
[29] X. Wang and E. K. Blum, "Discrete-time versus continuous-time models of neural networks," *J. Comp. Sys. Sci.*, vol. 45, pp. 1–19, Aug. 1992, doi: 10.1016/0022-0000(92)90038-K.
[30] H. Yang, C. Xiang, L. Hao, L. Zhao, and B. Xue, "Research on PSA-MFAC for a novel bionic elbow joint system actuated by pneumatic artificial muscles," *J. Mech. Sci. Technol.*, vol. 31, no. 7, pp. 3519–3529, Jul. 2017, doi: 10.1007/s12206-017-0640-0.
[31] N. M. H. Norsahperi and K. A. Danapalasingam, "Particle swarm-based and neuro-based FOPID controllers for a twin rotor system with improved tracking performance and energy reduction," *ISA Trans.*, vol. 102, pp. 230–244, Jul. 2020, doi: 10.1016/j.isatra.2020.03.001.
[32] S. Yuanyuan, W. Yongming, G. Lili, M. Zhongsong, and J. Shan, "The comparison of optimizing SVM by GA and grid search," in *Proc. 13th IEEE Int. Conf. Electron. Meas. Instrum. (ICEMI)*, Yangzhou, China, Oct. 2017, pp. 354–360.

MUHAMAD HAZWAN ABDUL HAFIDZ received the B.Eng. degree in mechanical engineering, the M.Eng. degree in mechatronics and automatic control from Universiti Teknologi Malaysia, Malaysia, in 2015 and 2019, respectively, where he is currently pursuing the Ph.D. degree. His current research interests include artificial intelligence and bioinspired robotics.

AHMAD ATHIF MOHD FAUDZI (Senior Member, IEEE) received the B.Eng. degree in computer engineering, the M.Eng. degree in mechatronics and automatic control from Universiti Teknologi Malaysia, Malaysia, and the Dr.Eng. degree in system integration from Okayama University, Japan, in 2004, 2006, and 2010, respectively. He is the Director at the Centre for Artificial Intelligence and Robotics (CAIRO) and a Professor at the Faculty of Engineering, Universiti Teknologi Malaysia. He is mainly engaged in the research fields of soft actuators, inspection robotics, and bioinspired robotics.

NOR MOHD HAZIQ NORSAHPERI (Member, IEEE) received the B.Eng. and M.Sc. degrees in mechatronics engineering from the International Islamic University of Malaysia (IIUM), in 2015 and 2017, respectively, and the Ph.D. degree in electrical engineering from Universiti Teknologi Malaysia, in 2020. He is currently a Senior Lecturer at the Department of Electrical and Electronic Engineering, Universiti Putra Malaysia (UPM). His research interests include nonlinear control, robotics, and artificial intelligence.

MOHD NAJEB JAMALUDIN (Member, IEEE) received the B.Eng. degree in electrical and electronics, the M.Eng. degree in electrical engineering, and the Ph.D. degree from Universiti Teknologi Malaysia, Malaysia, in 2003, 2007, and 2015, respectively. He is a Senior Lecturer at the Faculty of Engineering, School of Biomedical Engineering, Universiti Teknolgi Malayisa. His research interests include biomedical instrumentation, signal processing, engineering of prosthetic hand, and applications of artificial intelligence.

DAYANG TIAWA AWANG HAMID received the B.Ed. degree in educational technology from Universiti Pertanian Malaysia, Malaysia, in 1990, and the M.Ed. and Ph.D. degrees in educational technology from Universiti Teknologi Malaysia, Malaysia, in 1997 and 2006, respectively. She is currently the Dean at the Faculty of Technology, Universiti Sultan Sharif Ali, Brunei Darussalam. She is engaged in the research fields of data analytics, artificial intelligence, and technology-based innovations.

SHAHROL MOHAMADDAN received the B.Eng. and M.Eng. degrees in mechanical engineering from the Shibaura Institute of Technology, Japan, in 2005 and 2007, respectively, and the Ph.D. degree from Loughborough University, U.K. He is currently an Associate Professor at the Department of Bioscience and Engineering, Shibaura Institute of Technology. His research interests include biomedical engineering and ergonomics.