IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 29, 2021, accepted April 21, 2021, date of publication April 26, 2021, date of current version May 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3075581

Effective Nonlinear Model Predictive Control Scheme Tuned by Improved NN for Robotic Manipulators

MAHMOUD ELSISI[1,2], KARAR MAHMOUD[3,4], MATTI LEHTONEN[3], AND MOHAMED M. F. DARWISH[2,3]

[1]Industry 4.0 Implementation Center, Center for Cyber-Physical System Innovation, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[2]Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University, Cairo 11629, Egypt
[3]Department of Electrical Engineering and Automation, School of Electrical Engineering, Aalto University, 00076 Espoo, Finland
[4]Department of Electrical Engineering, Faculty of Engineering, Aswan University, Aswan 81542, Egypt

Corresponding author: Mohamed M. F. Darwish (mohamed.darwish@feng.bu.ed.eg and mohamed.m.darwish@aalto.fi)

This work was supported in part by the Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland, and in part by the Center for Cyber-Physical System Innovation from the Featured Areas Research Center Program in the Agenda of the Higher Education Sprout Project, Taiwan.

ABSTRACT The nonlinearities of the robotic manipulators and the uncertainties of their parameters represent big challenges against the controller design. Moreover, the tracking of regular and irregular trajectories with fewer overshoots, short settling time, and small steady-state error is the main target for the robotic response. The model predictive control (MPC) is an efficient controller to handle the performance requirements. However, the conventional MPC requires the linearization of the system model. The linearization of the model does not cover all dynamics of the robotic system. Thus, this paper introduces the nonlinear MPC (NLMPC) as a proper control method for the nonlinear systems instead of the conventional MPC. Specifically, this work proposes the use of NLMPC for controlling robotic manipulators. However, the NLMPC gains need proper tuning to attain good performance rather than the conventional methods. The neural network algorithm (NNA) considers a sufficient adaptive intelligent technique that can be utilized for this purpose. The restriction in a local optimum reveals the main issue versus artificial intelligence techniques. This paper suggests a new improvement to reinforce the exploration behavior of the NNA to overcome the local restriction issue. This modification is carried out by utilizing the polynomial mutation as an effective method to promise the exploration manner of the intelligence techniques. The proposed system can estimate all states from only the output to reduce the cost of the required sensors to measure all states. The results confirm the superiority of the proposed systems with the estimator with negligible change in the output response. The proposed modified NNA (MNNA) is evaluated with the main NNA, genetic algorithm-based PID control scheme, besides the cuckoo search algorithm-based PID control scheme from other works. The results confirm the robustness and effectiveness of the suggested MNNA-based NLMPC to track regular and irregular trajectories compared with other techniques.

INDEX TERMS Nonlinear system, robot manipulator, nonlinear model predictive control, trajectory tracking, neural networks, signals estimation, PID controller.

I. INTRODUCTION

Recently, the robotic manipulator is utilized for diverse purposes, e.g. aiding the industry and human routine duties. Specifically, the robot manipulator can perform risky actions and track the components in a very short time effectively rather than the human. The robotic moving parts require an effective and precise control scheme for acquiring tasks, most importantly position tracking [1]. In this regard, the nonlinearity characteristics of the robot, as well as the uncertainties of parameters, are the most key challenges against the operator to adjust the managing unit of robot links [2], [3].

In the previous literature, a lot of managing strategies are employed for the robotic manipulators [4], [5]. In [6], [7],

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

a proportional-integral-derivative (PID) controller is utilized based on semi-global stability analysis for two links robotic manipulators. However, the design method is conventional, and it does not take into account the system performance which is accounted for in small steady-state error, short settling time, less maximum over-shoot. In [8], a saturated PID controller is designed based on global asymptotic stabilization for a robotic manipulator. The designed PID is constrained for special cases and it does not consider the nonlinear trajectories for the robotic manipulators. An output feedback control with fuzzy logic (FL) control is introduced for robot manipulators in [9]. Thus, the controller gains are adjusted by the try and error of the designer. In [10], a fractional-order FL based on the cuckoo search algorithm (CSA) is applied to a robot manipulator. However, the fractional-order representation increases the order of the system. Furthermore, the applied objective is traditional, and it does not take into account the system overshoot and the settling time. In [11], an adaptive sliding mode control is applied to the robot movable parts based on pole placement procedure and time delay estimation technique. Whilst, the sliding mode control suffers from the chattering problem and the pole placement is a conventional method to design the controller parameters. In [12], a fractional-order PID sliding mode controller is designed based on the bat algorithm for a bio-inspired robot. However, the applied controller is complicated in the implementation and the sliding mode suffers from a chattering issue. In [13], an H-infinity control technique is introduced for robotic manipulators. Thus, the control technique is designed based on the linearized description of the robotic dynamics. However, the linearization is created based on the approximated procedure and it does not take into account the full dynamics of the real system. Among these control techniques, the model predictive control (MPC) provided good performance for a lot of engineering applications [14].

In [15], [16], a robotic manipulator utilizes the MPC based on the linearization of the dynamic model. However, the linearization of the dynamic model is more simplified, and it does not take into account the full dynamics of the real system. In [17] a data-driven MPC is introduced for trajectory tracking by the robotic arm. Thus, the applied MPC utilizes dynamics feedback linearization. The nonlinear model predictive control (NLMPC) can be applied directly to the nonlinear systems and overcome the linearization issue [18], [19]. In [20], a PID with an NLMPC controller is utilized for a two-link robotic manipulator. However, the controller gains are tuned by the try and error method based on the designer experience. Furthermore, the applied method does not take into account the nonlinear trajectories for the robotic manipulators. A lot of conventional tuning techniques are applied for the controller parameters like Ziegler Nichols (ZN) technique [21], [22] and graphical techniques [23], [24]. However, these techniques are complicated for nonlinear systems and it fails to provide good performance in different engineering applications [25], [26]. Artificial intelligence (AI) techniques have provided good solutions for the optimization issues of the controllers in different applications with short computation [27], [28]. There are several kinds of optimization methods like the genetic algorithm (GA) optimizer [29] and particle swarm optimization [30]. Other variants are the ant colony algorithm [31] and the teaching-learning algorithm [32]. Among these algorithms, the neural network algorithm (NNA) is a new effective and adaptive optimization technique [33]–[35]. It is demonstrated to have a global search feature based on the principles of artificial neural networks. Additionally, this method does not necessitate preliminary parameters to initialize, thereby defeating the other algorithms. However, the blockade in a locally optimal solution demonstrates the major issue against AI techniques. A lot of modification strategies such as mutation operators are used to solve this issue by enhancing the exploration behavior of the methods [36]. The utilization of the adaptive mutation operators can yield promising results when integrating into the mechanisms of different optimizers [37]–[39]. Considerable types of mutations such as random mutation operator, non-uniform mutation operator, and polynomial mutation operator can be utilized to improve the exploration way of the optimizers [40]. Among these mutation strategies, the polynomial mutation operator shows enhanced performance than the other methods in a lot of optimization techniques [41].

This paper introduces an intelligent design for the NLMPC parameters based on a new modified neural network algorithm (NNA) rather than the conventional methods. The new improvement of the MNNA is created based on the polynomial mutation operator to guarantee the exploration means of this algorithm. The proposed technique is devoted to adjusting the parameters of the NLMPC according to the decreasing of a developed figure of the demerit performance index. The innovative performance index is modified to confirm the decline of the response for the settling time as well as the maximum overshoot of robot links at the same time. The performance of the recommended technique is assessed with the main NNA [33], GA-PID control scheme proposed in [42], and the cuckoo search algorithm (CSA)- PID control scheme introduced in [43]. The effectiveness of the suggested technique is verified to track regular and irregular trajectories within initial and final constraints. In turn, the uncertainties of parameters are considered to confirm the robustness of the suggested technique.

Below, the contributions of this paper are listed as follows:

- An improved MNNA is developed based on the utilization of the polynomial mutation to guarantee the exploration way of the main NNA deprived of initial parameters;
- The proposed controller can decline of the response settling time, as well as an overshoot of the robot links, is accomplished based on a developed figure of demerit fitness function at the same time;
- A novel intelligent design is introduced to tune the NLMPC parameters based on improved MNNA in order

to track regular and irregular trajectories by the robotic manipulator;
- The proposed MNNA is applied to adjust the parameters of the robot control scheme rather than the conventional methods;
- The suggested MNNA-based NLMPC method is evaluated with the main NNA [33], the GA-PID control scheme [42], and CSA-PID control [43];
- The robustness and efficiency of the suggested technique are confirmed to track regular and irregular trajectories. Moreover, the results emphasize the robustness of the suggested technique towards the system parameter variations.

The other parts of this work are listed in the following sections: Section II describes the optimal control by intelligence algorithms. Section III illustrates the procedure of NLMPC. In section IV, the modeling of the robot is given. Section V demonstrates the output results of the proposed scheme. In the final, the conclusions and the future work of the paper are concluded in Section VI.

II. OPTIMAL CONTROL BY INTELLIGENCE ALGORITHMS

The design of an effective controller has a lot of challenges due to the system's nonlinearities and uncertainties. Furthermore, conventional methods such as ZN technique [21], [22] and graphical techniques [23], [24] are no longer suitable for intelligence systems. Because these techniques are complicated for nonlinear systems and it fails to provide good performance in different engineering applications [25], [26]. In [42], the GA is applied to tune the parameters of the PID controller as an intelligent technique instead of the conventional methods. The GA imitates the natural genetics and selection to tune the optimal parameters of the controller. In every iteration, a new set of springs are created based on the best members from the last iteration. The optimization by the GA is carried out based on different stages, the first is the reproduction, the second is the crossover, and the final is the mutation. The reproduction process is done based on the fitness function value of each individual. So, the individual that has the high fitness value, will have a big chance to generate the offsprings to the next iteration. The crossover is done by probability ratio between two parents from the last population to generate new offsprings and increase the possibility of new solutions. The mutation is created in the new springs by alternate the value of the string randomly to increase the exploration manner of the GA. However, the GA requires a lot of initial parameters to start such as the number of populations, generations, mutation, and crossover operators. In [43], the CSA is utilized to tune the gains of the robotic controller. The CSA is built based on the nature brood parasitism of cuckoo along with the birds' behavior and fruit flies. All cuckoos laying their eggs and the artificial cuckoo lay only one egg. The selection is carried out to choose the high-quality eggs for the next iteration. Some host's nests are selected approximately with a certain number of probability to host a foreign egg. If the host detects the cuckoo egg, it may be cast away or the host leaves the nest for the cuckoo. This algorithm starts randomly after setting the number of nests, probability of hosts, population, and generations. To overcome the initial parameters adjusting issue, the NNA is presented below for the tuning of robotic control without adjusting initial parameters. Furthermore, a new modification for the NNA is carried out for the main NNA based on the polynomial mutation to overcome the restriction in a local optimum.

A. NEURAL NETWORKS

Typically, neural networks are novel optimization techniques inspired based on the biological behavior of nervous schemes [33]. Note that the principles of artificial neural networks are the chief procedure of the NNA. Specifically, these networks have the behavior of global research to distinguish new solutions. Still, there is no need for generating initial parameters for the starting rather than the other processes. The NNA realizes the innovative solutions by adjusting the weight values between the foreseen solutions and the target ones. Methodically, this technique has the ability to find an optimal solution throughout the exploration space. Indeed, the NNA has a dissimilar procedure than the other processes to get the optimal value. This way decreases the gap between the target solution and the different solutions. The NNA contains 4 phases described in detail below:

1) STAGE OF INITIAL POPULATION

The NNA normally begins with a random preliminary population like other techniques to produce preliminary solutions within the distinct search space where every solution is termed "pattern solution". Initially, an arbitrary pattern solution matrix "X" with a dimension of $N \times D$ is produced. In which N represents the generation number while D represents the number of variables in the problem. The pattern solutions are as follows,

$$X = [X_1, X_2, .., X_i, ...X_N]' \text{and}$$
$$X_i = [x_{i1}, x_{i2}, ..., x_{iD}].$$

where;

$$x_{ij} = L_j + rand(U_j - L_j),$$
$$i = 1, 2, \ldots\ldots, N, \quad j = 1, 2, \ldots\ldots, D \tag{1}$$

where L represents the minimum boundaries of the variables while U represents the maximum boundaries of the variables. The NNA likes the basic artificial neural network where every solution X_i has an equivalent weight vector called $W_i = [w_{i1}, w_{i2}, ..., w_{iN}]$. Where the matrix of weights for all solutions has a size of $N \times N$. Typically, the NNA begins with an arbitrary weight matrix within (0, 1). Note that the weight matrix is restructured in every iteration according to the attained network error. Finally, the sum of the weights of each computed solution is bounded, and it should not surpass 1 as follows:

$$\sum_{j=1}^{N} w_{ij} = 1, \quad i = 1, 2, ..., N \tag{2}$$

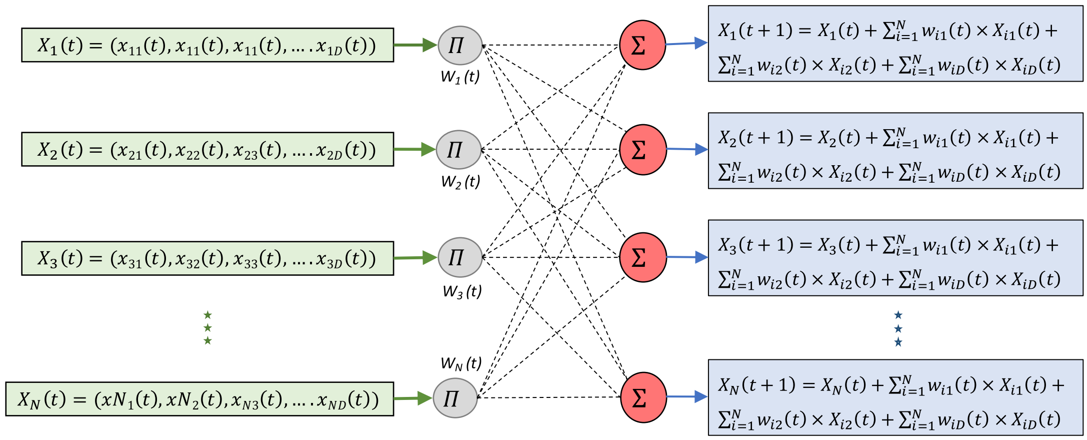


FIGURE 1. New population mechanism of NNA.

Note that the above constraint is helpful to regulate the attained bias of the searching mechanisms as well as the generation process. Specifically, it assets searching procedures from the restriction in a local best possible solution. Subsequently, the next step will be computing arbitrary solutions and the equivalent values of weights, the fitness function of each given solution is determined by the calculation of the objective function. Later, the finest comprising possible solution with particular weights are assigned to yield the new value by:

$$X_j^{new}(t+1) = \sum_{i=1}^{N} w_{ij}(t) \times X_i(t), \quad j = 1, 2, \ldots\ldots, N \tag{3}$$

$$X_i(t+1) = X_i(t) + X_i^{new}(t+1), \quad i = 1, 2, \ldots\ldots, N \tag{4}$$

In which $X_i(t)$ represent the calculated possible solution at the corresponding iteration 't' while $X_i^{new}(t+1)$ represent the weighted result at the new iteration '$t+1$'. To demonstrate this process, the updated generation mechanism is demonstrated in Fig. 1.

2) UPDATING PROCESS PF WEIGHT MATRIX

In this phase, it is required to update the weights between variables by:

$$W_i(t+1) = W_i(t) + 2 \times rand \times (W^*(t) - W_i(t)), \quad i = 1, 2, \ldots\ldots, N \tag{5}$$

in which $W^*(t)$ represents the target weight vector.

3) STAGE OF BIAS

Note that the developed NNA customs a bias operator to achieve accepted exploration. In particular, this proposed operator has been utilized to transform the ratio from produced possible solutions and the matrix of weights. Accordingly, the bias mutation operator decreases in an adaptive way with the iteration rising. For this aim, the following method can be utilized:

$$\beta(t+1) = 1 - \left(\frac{t}{t_{max}}\right), \quad t = 1, 2, \ldots\ldots, t_{max} \tag{6}$$

or as follows:

$$\beta(t+1) = 0.99\beta(t), \quad t = 1, 2, \ldots\ldots, t_{max} \tag{7}$$

in which t_{max} represents the maximum allowed iteration limit. Note that decreasing β value combined with rising the iteration can improve the exploitation feature of the solution mechanism while finding the optimal solution. An arbitrary number is formed in such stage to sense the number of populations for biasing by:

$$N_P = Round(D \times \beta) \tag{8}$$

Later, weights and the population are adapted by:

$$X_j = L + rand(U - L), \quad j = 1, 2, \ldots\ldots, N_P \tag{9}$$

Likewise, an arbitrary number is formed to detect the weight number to be changed by:

$$N_w = Round(N \times \beta) \tag{10}$$

$$W_j = m, \quad j = 1, 2, \ldots\ldots, N_w \tag{11}$$

in which m denotes an arbitrary number in the range (0, 1).

4) STAGE OF TRANSFER FUNCTION

An efficient transfer function is used here in the NNA mechanism to enhance its manipulation behavior. Specifically, this utilized operation updates the innovative possible solutions from the main positions to new ones, allowing to decline the gap between these solutions and the target ones. Note that the transfer function operation can be formulated by:

$$X_i^*(t+1) = X_i(t+1) + 2 \times rand \times (X^*(t) - X_i(t+1)), \quad i = 1, 2, \ldots\ldots, N \tag{12}$$

in which $X^*(t)$ denotes the finest solution at the corresponding iteration denoted by 't'.

B. PROPOSED NNA VARIANT

Generally, restrictions of several optimization solvers with respect to trapping in local optimal points away from the global solution are considered a big issue. This problem is happened at the initial step of the optimization operation because of the employment of arbitrary patterns. In this regard, mutation operators have the ability to overwhelm this issue with numerous single as well as multi-objective optimization procedures [37]–[39]. There are currently different mutations, such as random mutation operator, uniform mutation operator, non-uniform mutation operator, and polynomial mutation operator [40]. The polynomial mutation yields outstanding performance rather than the other methods [41] where it has a nonlinear probability. This property enables the algorithm to adapt the current possible solution to the neighboring one. Thus, the exploration behavior of the optimization algorithm can be increased that it can overcome the restriction at a local optimum. The exchange of the current solution by the neighboring solution is created by:

$$X_i(t) = X_i(t+1) + \alpha \times \delta i_{max}, \quad i = 1, 2, \ldots.., N_p \tag{13}$$

$$\alpha = \begin{cases} (2r)^{(1/(q+1))} - 1 & \text{if } r < 0.5 \\ 1 - [2(1-r)]^{(1/(q+1))} & \text{otherwise} \end{cases} \tag{14}$$

$$\delta_{max\ ij}(t) = max[X_{ij}(t) - L_j, U_j - X_{ij}(t)], \quad i = 1, 2, \ldots.., N_p, \quad j = 1, 2, \ldots.., D \tag{15}$$

in which q denotes a non-negative number and is termed a shape variable. Note that r represents an arbitrary variable in the range of (0, 1). δ_{maxij} represents the supreme permissible variation between the present solution and the mutated one. This paper suggests the mutation operation instead of the arbitrary exploration of the biasing step in (9). According to the preceding NNA steps, Figure 2 demonstrates the sequential steps of the MNNA to assign the best solution.

III. NONLINEAR MODEL PREDICTIVE CONTROL

The MPC is proved as an effective and superior control technique in most engineering applications [45]–[48]. The MPC predicts the future control signal within finite steps named the control horizon ‘M’. The MPC utilizes the prediction of the system output within finite steps named the prediction horizon ‘P’ in order to predict the proper control moves. The best control moves are selected according to the decreasing of a quadratic objective function. The MPC utilizes a linear-time-invariant (LTI) model to carry out the prediction operation for the control moves and the future system output. However, the linearization does not figure out all dynamics of the nonlinear systems. The NLMPC does not require the LTI system and it can be applied directly to the nonlinear system which is defined in the following equations,

$$\dot{x} = f(x, u, d) \tag{16}$$

$$y = h(x, d) \tag{17}$$

where x denotes the dynamic states of the system, u is the control signal, d denotes the external disturbance on the system, and y is the measured output of the system. The NLMPC

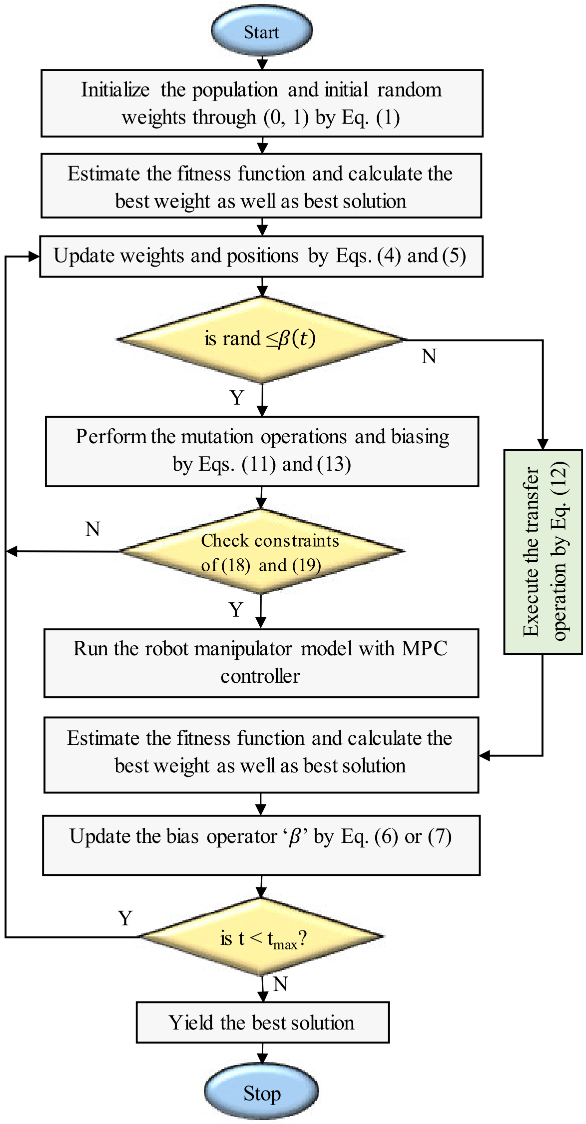


FIGURE 2. Flowchart of the modified NNA.

utilizes a continuous objective function to predict the proper control moves based on the minimization of error between the target reference and the measured output at each move ‘k’ of the sample time ‘T_s’. Where the control moves ‘Δu’ are discrete whilst the system output ‘y’ is continuous. The quadratic objective function is formulated as follows [49],

$$\varphi = \int_0^P \|r(k+t\,|k) - y(k+t\,|k)\|_Q^2 dt + \sum_{i=0}^{M-1} \|\Delta u(k+i\,|k)\|_R^2 \tag{18}$$

$$\Delta u(k) = u(k) - u(k-1) \tag{19}$$

Such as;

$$u_{\min} \le u \le u_{\max}$$
$$\Delta u_{\min} \le \Delta u \le \Delta u_{\max}$$
$$y_{\min} \le y \le y_{\max}$$

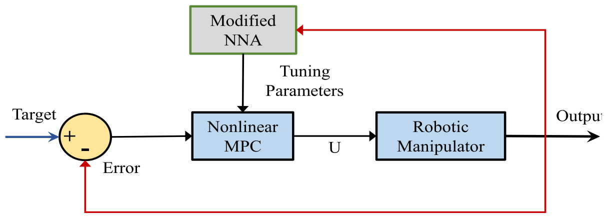


FIGURE 3. Schematic representation of the online optimization for the Nonlinear MPC based on the modified NNA.

$$P \geq 1$$
$$M \geq 1$$
$$P \geq M \geq 1$$

where

$\|.\|_Q^2, \|.\|_R^2$ Euclidian norm
Q Outputs weighting factor
R Inputs weighting factor

The NLMPC necessitates a suitable tuning for its prediction horizon 'P', control horizon 'M', outputs weighting factor 'Q', inputs weighting factor 'R', and the sample time 'T_s' in order to provide good performance. The algorithm is used to find the optimal controller not to estimate the states. So, this paper suggests the MNNA to adjust the NLMPC parameters as shown in Figure 3 rather than the conventional techniques. Note that the MPC is carried out with the system online.

IV. SYSTEM MODELING

Here, we introduce the mathematical formulation of the robotic manipulator where its dynamic model is represented by nonlinear differential formulae. These formulae have different terms such as Coriolis, load, centrifugal torques, inertia, and gravity. Note that the robot actuator in its link requires an appropriate torque that allows moving the end-effector in a certain trajectory with respect to constraint speed. The following formula rules the dynamics of the robot manipulator of different n-links [42].

$$\tau = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) \tag{20}$$

where;

τ Vector of Torque for the links with dimension of n × 1
$M(\theta)$ Non-negative matrix with dimension of n × n
$C(\theta, \dot{\theta})$ Vector of Coriolis torque with dimensions n × 1
$G(\theta)$ Vector of gravity torque with dimensions n × 1
θ The angular links position
$\dot{\theta}$ Link velocity
$\ddot{\theta}$ Acceleration of links
n link Number

In this research, the robot has 2 arms (see Figure 4). The dynamics equations of this machine can be expressed by [44]:

$$\begin{aligned}\tau_1 &= m_2 l_2^2(\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_1 l_2 c_2(2\ddot{\theta}_1 + \ddot{\theta}_2)\\ &\quad + (m_1 + m_2) l_1^2 \ddot{\theta}_1 - m_2 l_1 l_2 s_2 \ddot{\theta}_2^{\,2}\\ &\quad - 2 m_2 l_1 l_2 s_2 \ddot{\theta}_1 \ddot{\theta}_2 + m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1\end{aligned} \tag{21}$$

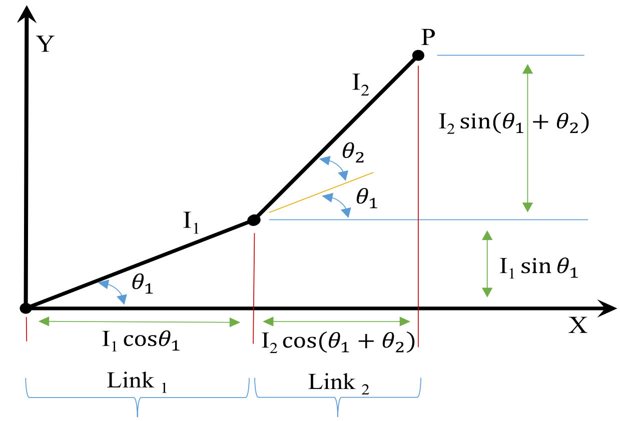


FIGURE 4. Schematic illustration of two-links robot manipulator.

$$\begin{aligned}\tau_2 &= m_2 l_2^2(\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_1 l_2 c_2 \ddot{\theta}_1\\ &\quad + m_2 l_1 l_2 c_2 \ddot{\theta}_1^{\,2} + m_2 l_1 g c_{12}\end{aligned} \tag{22}$$

In which

$$c_1 = \cos(\theta_1),\quad c_{12} = \cos(\theta_1 + \theta_2), c_2 = \cos(\theta_2),$$
$$s_1 = \sin(\theta_1),\ \text{and}\ s_2 = \sin(\theta_2).$$

The state-space model for the robot manipulator is formulated by the state-space representation in (16) and (17) by suggesting $x_1 = \theta_1, x_2 = \theta_2, x_3 = \dot{x_1} = \ddot{\theta}_1, x_4 = \dot{x_2} = \ddot{\theta}_2, u_1 = \tau_1$, $u_2 = \tau_2$. Then substitute in (21) and (22),

$$\begin{aligned}\dot{x_3} &= \frac{1}{Z_1}(-(m_2 l_2^2 + m_2 l_1 l_2 \cos(x_2)\dot{x_4} + m_2 l_1 l_2 \sin(x_2) x_4^2\\ &\quad + 2 m_2 l_1 l_2 \sin(x_2) x_3 x_4 - m_2 l_2 g \cos(x_1 + x_2)\\ &\quad - (m_1 + m_2) l_1 g \cos(x_1) + u_1)\end{aligned} \tag{23}$$

$$\begin{aligned}\dot{x_4} &= \frac{1}{Z_2}(-(m_2 l_2^2 + m_2 l_1 l_2 \cos(x_2))\dot{x_3} - m_2 l_1 l_2 \sin(x_2) x_3^2\\ &\quad - m_2 l_1 g \cos(x_1 + x_2) + u_2)\end{aligned} \tag{24}$$

where $Z_1 = m_2 l_2^2 + 2 m_2 l_1 l_2 c_2 + (m_1 + m_2) l_1^2$ and $Z_2 = m_2 l_2^2$. Then define the Jacobian of all state dynamic equations. After this step, the system is ready to apply the NLMPC in order to predict the best control moves which represents the torque of the robotic manipulator.

V. RESULTS AND DISCUSSION

Here, the proposed modified NNA is utilized to adjust the NLMPC gains in order to improve the response of the robot that is shown in Fig. 4. The key purpose of the optimization operation is the improvement of the performance of each link to achieve the target trajectory. The performance of the output response is evaluated by the decreasing settling time, the steady-state error, and the maximum overshoot. This paper suggests a developed fitness function to accomplish the declining of the response settling time as well as the overshoot for each link at the same time. This fitness function is labeled figure of demerit (FOD) where it is defined as next:

$$J = \sum_{i=1}^{2}(1 - e^{-\psi})(M_{P,i} + E_{SS,i}) + e^{-\psi}(t_{s,i} - t_{r,i}) \tag{25}$$

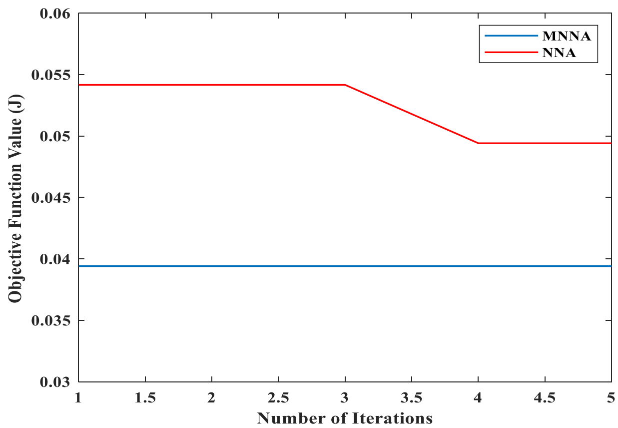


FIGURE 5. Optimization effort of the proposed modified NNA and the main NNA.

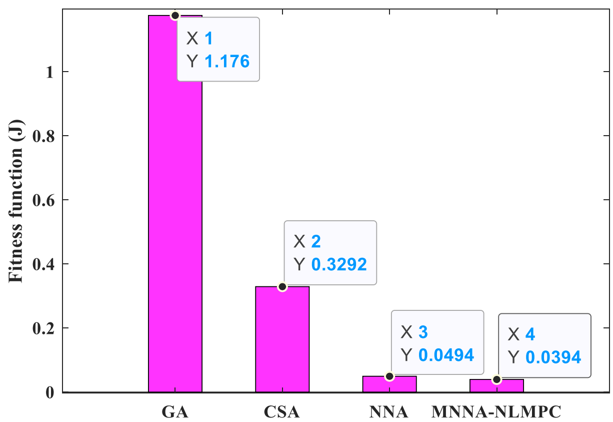


FIGURE 6. The fitness function value due to different techniques.

where;

$M_{P,i}$ maximum overshoot
$E_{SS,i}$ output response steady-state error
$t_{s,i}$ output response settling time
$t_{r,i}$ output response rise time
ψ weighting variable
i robot arm index

The previous objective function is presented as the FOD performance index in [50], however, the first formulation for it is created by Gaing [51]. The FOD imposes equal weighting for the (maximum overshoot and the system steady-state error) and equal weighting for the (rise and settling times). Regarding the necessity of exponential weights $(1 - e^{-\psi}, e^{-\psi})$ is extremely appreciated instead of the linear weights. Moreover, this objective function (J) can be extended to include many weights for amplitudes and times as follows: $J = (1 - \psi)(\mu_1 M_p + \mu_2 E_{ss}) + \psi(\lambda_1 t_s - \lambda_2 t_r)$. However, more tuning parameters $(\psi, \mu_1, \mu_2, \lambda_1, \lambda_2)$ decrease the solvability of the optimization problem due to the increased nonlinearity which leads to an increased possibility of restriction in local minima. The objective function in (25) can achieve the designer requirements by picking an appropriate value for the attained

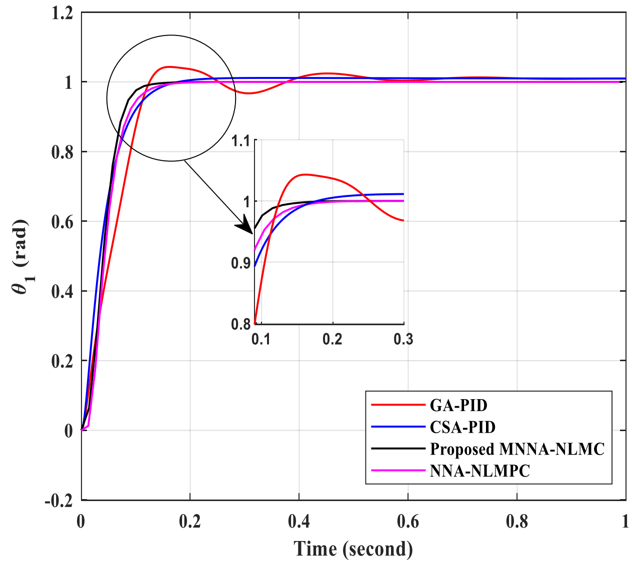


FIGURE 7. Output response of the position of robot $link_1$ in the case of unit step trajectory.

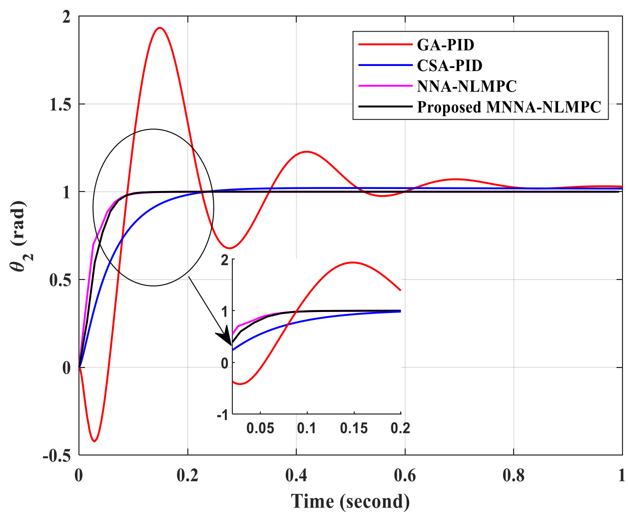
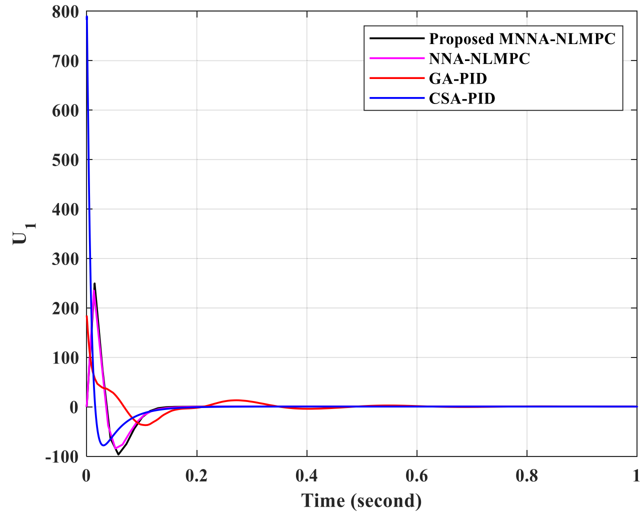
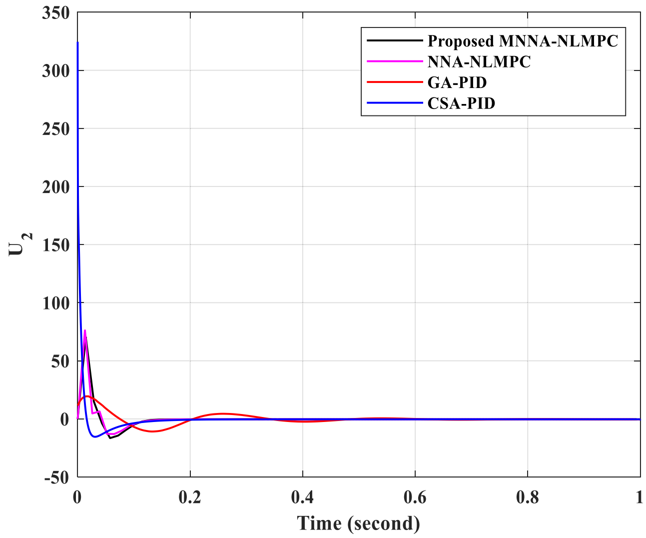


FIGURE 8. Output response of the position of robot $link_2$ in the case of unit step trajectory.

weighting factor 'ψ'. In the case that the value of $\psi < 0.7$, the settling time can be decreased. On the opposing, if the value of $\psi > 0.7$, it can decrease the overshoot. When ψ equals 0.7, the exponential weights $(1 - e^{-\psi}, e^{-\psi})$ will be $\approx (0.5, 0.5)$, so in this research, the suggested ψ equals 0.7 to accomplish the reduction of both the settling time as well as the overshoot of each link response at the same time. The introduced MNNA search about the optimal gains of the NLMPC controller by the minimization of the objective function in (25). The optimization operation is done at the system-rated parameters and a unit step reference for the position of each link. The nominal gains of the proposed system are: g = 9.81m/s^2, m_1 = 0.1, m_2 = 0.1 kg, l_1 = 0.8 m, and l_2 = 0.4 m [42]. The MNNA adopted



(a)



(b)

FIGURE 9. Control signals of each controller; (a) control signal for $link_1$, and (b) control signal for $link_2$.

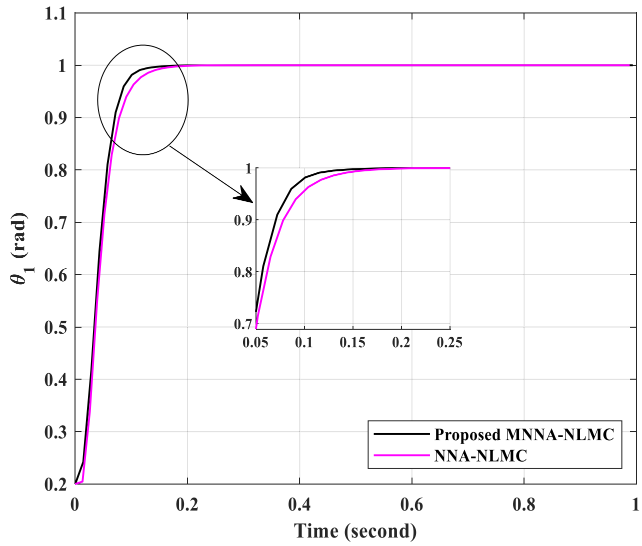


FIGURE 10. System response of the position of robot $link_1$ in the case of another initial point.

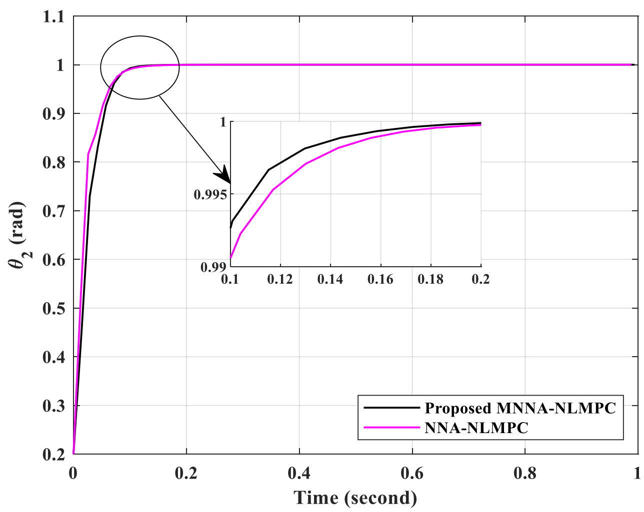


FIGURE 11. System response of the position of robot $link_2$ in the case of another initial point.

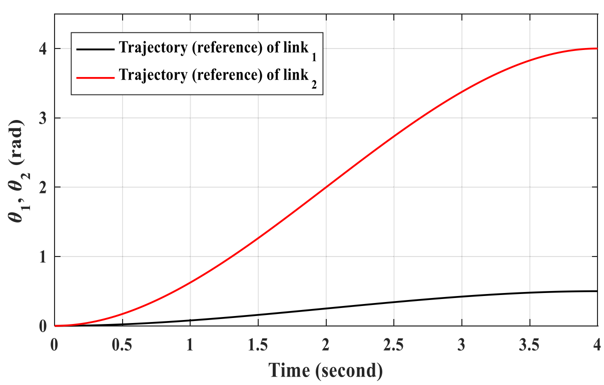


FIGURE 12. Robot links cubic position trajectories.

parameters are: the maximum number of agents is selected as 20 and the maximum iterations is equal 5. Figure 5 shows the optimization effort of the proposed MNNA and the main NNA [33]. As shown in Figure 5, the proposed MNNA can minimize the objective function fastly compared to the main NNA [33]. Furthermore, the results of the proposed NLMPC based on MNNA are affirmed by comparing them with the GA-based PID control scheme in [42], the CSA-based PID control scheme in [43]. The controller parameters due to each method with the equivalent performance index are listed in Table 1. Figure 6 shows the values of the fitness function due to different techniques for clarified comparison. It is concluded from Table 1 and Fig. 6 that the proposed MNNA has the minimum performance index compared with the other techniques. The steps of the MNNA to find the optimal parameters are summarized as follows in the pseudo-code shown in Algorithm 1.

Different simulated scenarios are created in the following subsections to confirm the effectiveness and robustness of the

TABLE 1. Comparison of controller parameters of proposed method with exist ones with the performance index.

		GA-PID [42]	CSA-PID [43]	NNA–Nonlinear MPC	Proposed Modified NNA–Nonlinear MPC
Controller Parameters	Link $_1$	$K_{P,1}$ = 184.76, $K_{I,1}$ = 49.68, $K_{D,1}$ = 8.94	$K_{P,1}$ = 782.417, $K_{I,1}$ = 225.2123, $K_{D,1}$ = 35.1995	T_s = 0.013, P = 15, M = 3, Q_1 = 10, Q_2 = 10, R_1 = 0.002, R_2 = 0.002	T_s = 0.0144, P = 19, M = 3, Q_1 = 11.6725, Q_2 = 8.7578, R_1 = 0.0198, R_2 = 0.0015
	Link $_2$	$K_{P,2}$ = 11.46, $K_{I,2}$ = 16.54, $K_{D,2}$ = 0.2	$K_{P,2}$ = 324.523, $K_{I,2}$ = 119.245, $K_{D,2}$ = 20.1025		
J		1.1758	0.3292	0.0494	0.0394

Algorithm 1 Pseudo-Code of MNNA to Find The Controller Gains

1: ***Start*** Modified NNA;
2: ***Execute*** the robot manipulator model with Nonlinear MPC;
3: ***Calculate*** the fitness function in (25);
4: ***Select*** the best weights and best solution;
5: ***While*** $(t < iterations_{max})$;
6: Do the steps of Modified NNA, as in Fig. 2;
7: Execute robot manipulator model with Nonlinear MPC;
8: Estimate the fitness function in (25);
9: Choose the best fitness value;
10: Choose the updated solution;
11: ***End While***.
12: ***Stop***.

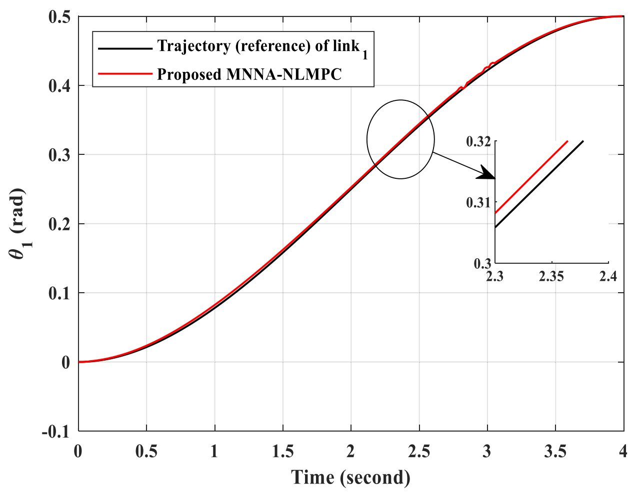


FIGURE 13. The system response of link$_1$ in case of cubic position trajectory.

proposed MNNA. In particular, these studied scenarios are the rated parameter test that involves unit step reference at different initial points and nonlinear trajectory test for the position of each link. Additionally, the robustness test of the suggested MNNA is investigated intensively against the system parameters uncertainties.

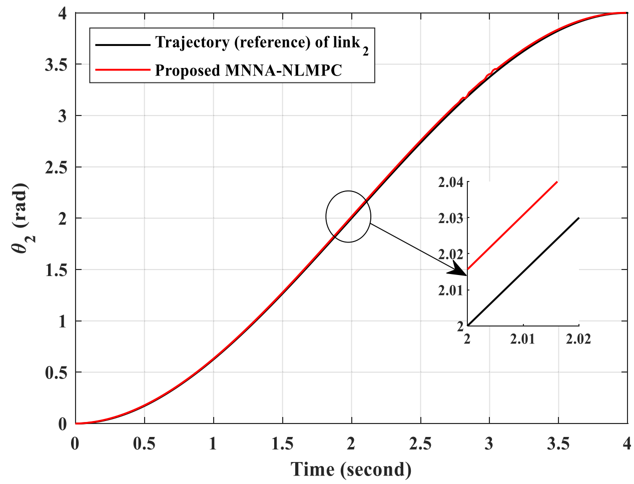


FIGURE 14. The system response of link$_2$ in case of cubic position trajectory.

A. SCENARIO 1: THE RATED PARAMETER CONDITION WITH UNIT STEP REFERENCE

Here, a unit step position reference is adopted for each link at system-rated parameters. The output response due to this test is presented in Figures 7 and 8. The robotic manipulator links starts to track a unit step trajectory at initial states x0 = [0 0 0 0]'. Figure 9 shows the control signal due to each controller for each link of the robotic manipulator. The response settling time and the overshoot due to each method are records in Table 2. It is clear from Figures 7 and 8, and Table 2 that the proposed MNNA-NLMPC control scheme beats the main NNA [33], the GA-PID control scheme [42], and the CSA-PID control scheme [43]. Besides, the introduced MNNA has the shortest settling time and overshoot rather than the other techniques. Besides, the proposed controller has a lower control signal

TABLE 2. The response overshoot and settling time based on each method.

		GA-PID [42]	CSA-PID [43]	NNA–Nonlinear MPC	Proposed Modified NNA–Nonlinear MPC
Maximum overshoot	$Link_1$	4.301%	1.1421%	0.0124%	0.0057%
	$Link_2$	93.3058%	2.1193%	0.0042%	0.0032%
Settling time	$Link_1$	0.4899	0.1404	0.1281	0.1053
	$Link_2$	1	0.694	0.0886	0.0873

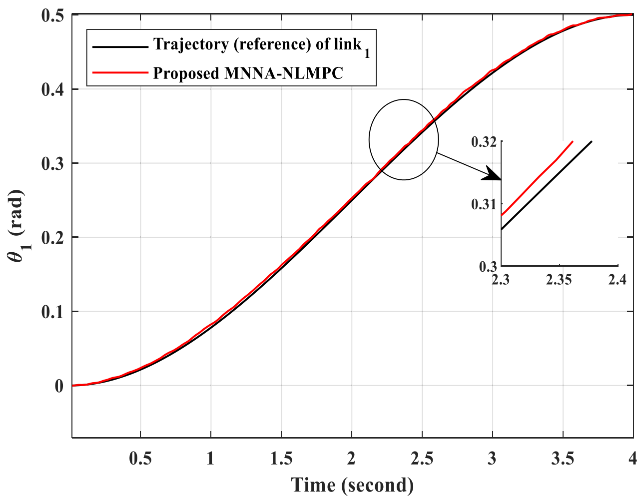


FIGURE 15. The system response of $link_1$ in case of state observer.

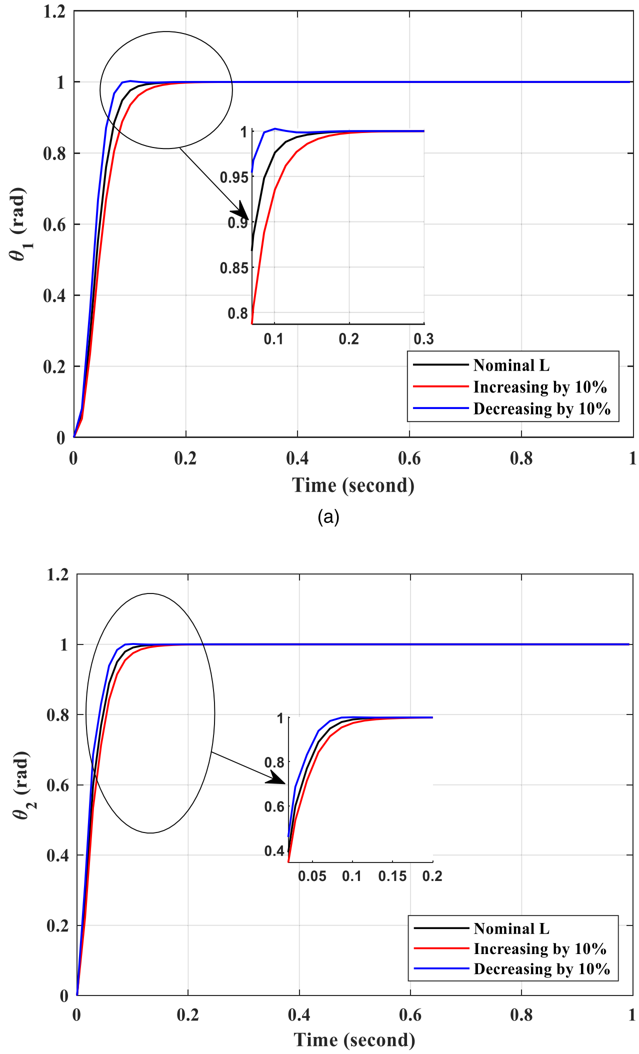


FIGURE 17. The system response in case of length uncertainty; (a) position of Link 1, and (b) position of Link 2.

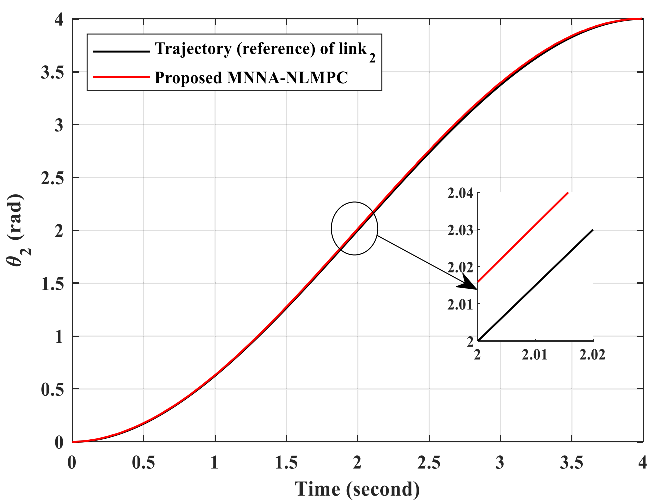


FIGURE 16. The system response of $link_2$ in case of state observer.

fluctuation than the CSA-PID controller [43] as cleared in Figure 9.

B. SCENARIO 2: TESTING OF THE PROPOSED CONTROLLER TO TRACK A UNIT STEP TRAJECTORY AT OTHER INITIAL POINTS

In this scenario, the proposed MNNA- based NLMPC is tested to track unit step trajectory at other initial points. The test is carried out by choosing initial states x0 = [0.2 0.2 0.2 0.2]' to check the ability of the proposed procedures to track the trajectory from another initial state. The system response due to this test is presented in Figures 10 and 11. As shown in these figures, the advised MNNA- based NLMPC remains a good damping characteristic to track the reference from another initial state.

TABLE 3. Final and initial cubic trajectories parameters.

	t_0	t_f (sec)	θ_{d0} (rad)	θ_{df} (rad)	$\dot{\theta}_{d0}$	c_0	c_1	c_2	c_3
Link $_1$	0	4	0	0.5	0	0	0	0.09375	- 0.015625
Link $_2$	0	4	0	4	0	0	0	0.75	- 0.125

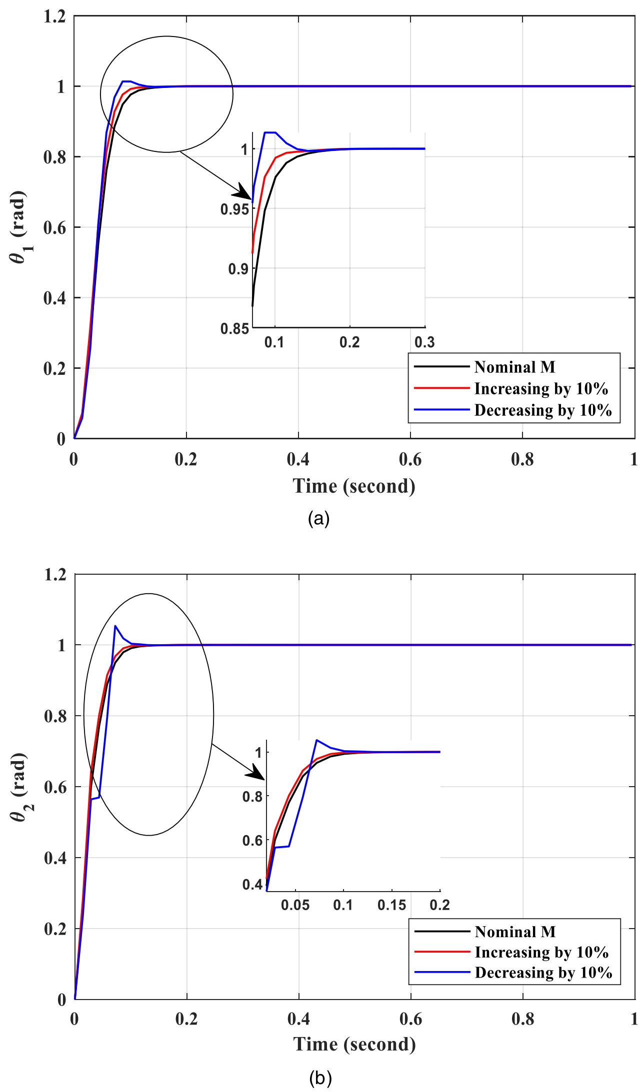


FIGURE 18. The system response in case of mass uncertainty; (a) position of Link 1, and (b) position of Link 2.

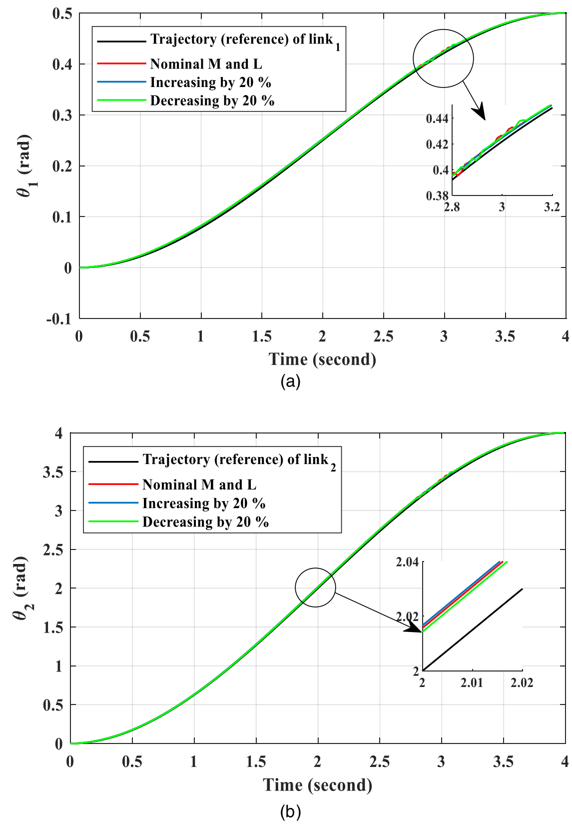


FIGURE 19. The system response in case of robustness test against parameters uncertainties and nonlinear trajectories; (a) position of Link 1, and (b) position of Link 2.

C. SCENARIO 3: EFFECTIVENESS OF THE PROPOSED METHOD AGAINST NONLINEAR TRAJECTORIES

In scenario 3, the proposed method is assessed to track nonlinear trajectories. Specifically, this scenario test is created by adopting a cubic location trajectory on each link as shown in Figure 12. It is worth noting that this nominated cubic trajectory is established based on the following formulae [44]:

$$\theta_{d,i} = c_{0,i} + c_{1,i} \times t + c_{2,i} \times t^2 + c_{3,i} \times t^3 \qquad (26)$$

with target velocity as well as acceleration constraints which are formulated as follows,

$$\dot{\theta}_{df,i} = c_{1,i} + 2c_{2,i} \times t_f + 3c_{3,i} \times t_f^2 \qquad (27)$$

$$\ddot{\theta}_{df,i} = 2c_{2,i} + 6c_{3,i} \times t_f \qquad (28)$$

where $i = 1, 2$ is the link index. t_f, $\dot{\theta}_{df}$, $\ddot{\theta}_{df}$ represent, respectively, the termination time, velocity, and acceleration. Note that, in Table 3, the initial and final gains of the cubic trajectories are recorded that represented as equality constraints. The constants $c_{0,i}, c_{1,i}, c_{2,i}, c_{3,i}$ can be determined by solving (26) -(28) collected with the starting and target position as well as velocity. Consequently, the nonlinear trajectory is sketched for every arm, as simplified in Figure 12.

The system performance based on the proposed MNNA-based NLMPC in the case of the cubic position trajectory case is shown in Figures 13 and 14. These results show that the suggested method can effectively track the irregular trajectory.

D. SCENARIO 4: EFFECTIVENESS OF THE SUGGESTED TECHNIQUE WITH STATE OBSERVER

This Scenario is carried out to confirm the efficiency of the suggested technique in the case of using a state observer. The state observer is utilized to decrease the cost of the measurement of all states. The output response due to this test is clear in Figures 15 and 16. As shown in these figures, the proposed MNNA-NLMPC can track the nonlinear trajectories effectively in the case of applying the state observer.

E. SCENARIO 5: ROBUSTNESS TEST OF SUGGESTED PROCEDURE AGAINST THE VARIATIONS OF PARAMETERS

This considered scenario has been carried out by considering $\pm 10\%$ uncertainty in the robotic masses as well as lengths of each arm from the nominal values. The system response based on the suggested MNNA-NLMPC controller is presented in Figures 17 and 18. It is clear from these figures that the proposed technique can overcome the parameter uncertainties with high damping characteristics and negligible errors.

Furthermore, the robustness test is carried out to check the effectiveness of the proposed technique against parameters uncertainties by $\pm 20\%$ and nonlinear trajectories. Figure 19 shows that the links of the robot based on the proposed controller can track the nonlinear trajectory effectively with negligible error.

VI. CONCLUSION

In this paper, we have introduced a novel developed intelligence technique named MNNA. The proposed algorithm is devoted to the tuning of NLMPC of robot manipulator control scheme instead of the traditional methods. Furthermore, a developed FOD performance index is utilized to accomplish the lessening of the response settling time as well as the overshoot of each robot link at the same time. Further experiments are done to emphasize the efficiency of the suggested technique. In addition, the proposed method is assessed with the main NNA, GA- based PID control scheme, and the CSA- based PID control scheme. The output results emphasize that the proposed method superior to the other methods and it is more effective to track regular and irregular trajectories with a mean absolute error around 0.005, short settling time around 0.11 s, and less overshoot around 0.01% for all scenarios. Moreover, the suggested technique is robust versus the system parameters uncertainties. For future work, the proposed control can be applied for other robotic manipulators include more dynamics of degree-of-freedom and joints.

REFERENCES

[1] T. N. Truong, A. T. Vo, and H.-J. Kang, "A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators," *IEEE Access*, vol. 9, pp. 31921–31931, 2021, doi: 10.1109/ACCESS.2021.3060115.

[2] K. Gkountas and A. Tzes, "Leader/follower force control of aerial manipulators," *IEEE Access*, vol. 9, pp. 17584–17595, 2021, doi: 10.1109/ACCESS.2021.3053654.

[3] B. Hu, Z.-H. Guan, F. L. Lewis, and C. L. P. Chen, "Adaptive tracking control of cooperative robot manipulators with Markovian switched couplings," *IEEE Trans. Ind. Electron.*, vol. 68, no. 3, pp. 2427–2436, Mar. 2021, doi: 10.1109/TIE.2020.2972451.

[4] M. Elsisi, K. Mahmoud, M. Lehtonen, and M. M. F. Darwish, "An improved neural network algorithm to efficiently track various trajectories of robot manipulator arms," *IEEE Access*, vol. 9, pp. 11911–11920, 2021, doi: 10.1109/ACCESS.2021.3051807.

[5] D. Feliu-Talegon and V. Feliu-Batlle, "Control of very lightweight 2-DOF single-link flexible robots robust to strain gauge sensor disturbances: A fractional-order approach," *IEEE Trans. Control Syst. Technol.*, early access, Feb. 8, 2021, doi: 10.1109/TCST.2021.3053857.

[6] I. Cervantes and J. Alvarez-Ramirez, "On the PID tracking control of robot manipulators," *Syst. Control Lett.*, vol. 42, no. 1, pp. 37–46, Jan. 2001.

[7] J. Alvarez-Ramirez, R. Kelly, and I. Cervantes, "Semiglobal stability of saturated linear PID control for robot manipulators," *Automatica*, vol. 39, no. 6, pp. 989–995, Jun. 2003.

[8] Y. Su, P. C. Müller, and C. Zheng, "Global asymptotic saturated PID control for robot manipulators," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 6, pp. 1280–1288, Nov. 2010.

[9] E. Kim, "Output feedback tracking control of robot manipulators with model uncertainty via adaptive fuzzy logic," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 3, pp. 368–378, Jun. 2004.

[10] R. Sharma, P. Gaur, and A. P. Mittal, "Design of two-layered fractional order fuzzy logic controllers applied to robotic manipulator with variable payload," *Appl. Soft Comput.*, vol. 47, pp. 565–576, Oct. 2016.

[11] J. Baek, M. Jin, and S. Han, "A new adaptive sliding-mode control scheme for application to robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3628–3637, Jun. 2016.

[12] M. Rahmani, A. Ghanbari, and M. M. Ettefagh, "Robust adaptive control of a bio-inspired robot manipulator using bat algorithm," *Expert Syst. Appl.*, vol. 56, pp. 164–176, Sep. 2016.

[13] G. Rigatos, P. Siano, and G. Raffo, "A nonlinear H-infinity control method for multi-DOF robotic manipulators," *Nonlinear Dyn.*, vol. 88, no. 1, pp. 329–348, Apr. 2017.

[14] P. D. Domański, "Performance assessment of predictive control—A survey," *Algorithms*, vol. 13, no. 4, p. 97, Apr. 2020.

[15] E.-H. Guechi, S. Bouzoualegh, L. Messikh, and S. Blazic, "Model predictive control of a two-link robot arm," in *Proc. Int. Conf. Adv. Syst. Electr. Technol. (IC_ASET)*, Mar. 2018, pp. 409–414.

[16] Q. Tang, Z. Chu, Y. Qiang, S. Wu, and Z. Zhou, "Trajectory tracking of robotic manipulators with constraints based on model predictive control," in *Proc. 17th Int. Conf. Ubiquitous Robots (UR)*, Jun. 2020, pp. 23–28.

[17] A. Carron, E. Arcari, M. Wermelinger, L. Hewing, M. Hutter, and M. N. Zeilinger, "Data-driven model predictive control for trajectory tracking with a robotic arm," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3758–3765, Oct. 2019.

[18] T. Rybus, K. Seweryn, and J. Z. Sasiadek, "Control system for free-floating space manipulator based on nonlinear model predictive control (NMPC)," *J. Intell. Robotic Syst.*, vol. 85, nos. 3–4, pp. 491–509, Mar. 2017.

[19] F. Xu, H. Chen, X. Gong, and Q. Mei, "Fast nonlinear model predictive control on FPGA using particle swarm optimization," *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 310–321, Jan. 2016.

[20] J. Wilson, M. Charest, and R. Dubay, "Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator," *Robot. Comput.-Integr. Manuf.*, vol. 41, pp. 23–30, Oct. 2016.

[21] K. Ogata, *Modern Control Engineering*. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.

[22] K. J. Åström and T. Hägglund, "Revisiting the Ziegler–Nichols step response method for PID control," *J. Process Control*, vol. 14, no. 6, pp. 635–650, Sep. 2004.

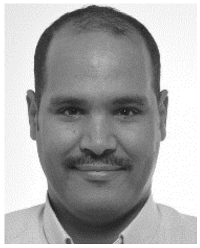
[23] Z. Shafiei and A. T. Shenton, "Tuning of PID-type controllers for stable and unstable systems with time delay," *Automatica*, vol. 30, no. 10, pp. 1609–1615, Oct. 1994.

[24] S. Srivastava and V. S. Pandit, "A PI/PID controller for time delay systems with desired closed loop time response and guaranteed gain and phase margins," *J. Process Control*, vol. 37, pp. 70–77, Jan. 2016.

[25] S. Tavakoli and M. Tavakoli, "Optimal tuning of PID controllers for first order plus time delay models using dimensional analysis," in *Proc. 4th Int. Conf. Control Autom. ICCA Final Program Book Abstr. (ICCA)*, 2003, pp. 942–946.

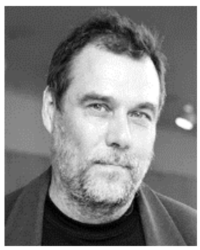
[26] H. Wu, W. Su, and Z. Liu, "PID controllers: Design and tuning methods," in *Proc. 9th IEEE Conf. Ind. Electron. Appl.*, Jun. 2014, pp. 808–813.
[27] Z. Beheshti and S. M. H. Shamsuddin, "A review of population-based meta-heuristic algorithms," *Int. J. Adv. Soft Comput. Appl*, vol. 5, no. 1, pp. 1–35, 2013.
[28] S. Nesmachnow, "An overview of metaheuristics: Accurate and efficient methods for optimisation," *Int. J. Metaheuristics*, vol. 3, no. 4, pp. 320–347, 2014.
[29] O. Kramer, *Genetic Algorithm Essentials*, vol. 679. Berlin, Germany: Springer-Verlag, 2017.
[30] Y. Zhang, S. Wang, and G. Ji, "A comprehensive survey on particle swarm optimization algorithm and its applications," *Math. Problems Eng.*, vol. 2015, pp. 1–38, Feb. 2015.
[31] M. Dorigo and T. Stützle, "Ant colony optimization: Overview and recent advances," in *Handbook of Metaheuristics*. Cham, Switzerland: Springer, 2019, pp. 311–351.
[32] R. V. Rao, "Teaching-learning-based optimization algorithm," in *Teaching Learning Based Optimization Algorithm*. Cham, Switzerland: Springer, 2016, pp. 9–39.
[33] A. Sadollah, H. Sayyaadi, and A. Yadav, "A dynamic Metaheuristic optimization model inspired by biological nervous systems: Neural network algorithm," *Appl. Soft Comput.*, vol. 71, pp. 747–782, Oct. 2018.
[34] M. S. AbouOmar, H. J. Zhang, and Y. X. Su, "Fractional order fuzzy PID control of automotive PEM fuel cell air feed system using neural network optimization algorithm," *Energies*, vol. 12, no. 8, p. 1435, 2019.
[35] Y. Zhang, Z. Jin, and Y. Chen, "Hybrid teaching–learning-based optimization and neural network algorithm for engineering design optimization problems," *Knowl.-Based Syst.*, vol. 187, Jan. 2020, Art. no. 104836.
[36] A. R. Jordehi, "Enhanced leader PSO (ELPSO): A new PSO variant for solving global optimisation problems," *Appl. Soft Comput.*, vol. 26, pp. 401–417, Jan. 2015.
[37] A. W. Mohamed and P. N. Suganthan, "Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation," *Soft Comput.*, vol. 22, no. 10, pp. 3215–3235, May 2018.
[38] A. W. Mohamed, "An improved differential evolution algorithm with triangular mutation for global numerical optimization," *Comput. Ind. Eng.*, vol. 85, pp. 359–375, Jul. 2015.
[39] R. Salgotra and U. Singh, "Application of mutation operators to flower pollination algorithm," *Expert Syst. Appl.*, vol. 79, pp. 112–129, Aug. 2017.
[40] P.-H. Tang and M.-H. Tseng, "Adaptive directed mutation for real-coded genetic algorithms," *Appl. Soft Comput.*, vol. 13, no. 1, pp. 600–614, Jan. 2013.
[41] G.-Q. Zeng, J. Chen, L.-M. Li, M.-R. Chen, L. Wu, Y.-X. Dai, and C.-W. Zheng, "An improved multi-objective population-based extremal optimization algorithm with polynomial mutation," *Inf. Sci.*, vol. 330, pp. 49–73, Feb. 2016.
[42] H. V. H. Ayala and L. dos Santos Coelho, "Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator," *Expert Syst. Appl.*, vol. 39, no. 10, pp. 8968–8974, Aug. 2012.
[43] H. Chhabra, V. Mohan, A. Rani, and V. Singh, "Multi-objective cuckoo search algorithm-based 2-DOF FOPD controller for robotic manipulator," in *Advances in Signal Processing and Communication*. Singapore: Springer, 2019, pp. 345–352.
[44] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 3rd ed. London, U.K.: Pearson, 2009.
[45] M. L. Darby and M. Nikolaou, "MPC: Current practice and challenges," *Control Eng. Pract.*, vol. 20, no. 4, pp. 328–342, Apr. 2012.
[46] M. G. Rodd, "Introduction to robotics: Mechanics and control," *Automatica*, vol. 23, no. 2, pp. 263–264, Mar. 1987.
[47] M. A. Henson, "Nonlinear model predictive control: Current status and future directions," *Comput. Chem. Eng.*, vol. 23, no. 2, pp. 187–202, Dec. 1998.
[48] F. Allgöwer and A. Zheng, Eds., *Nonlinear Model Predictive Control*, vol. 26. Basel, Switzerland: Birkhäuser, 2012.
[49] A. Zheng, "Nonlinear model predictive control of the tennessee eastman process," in *Proc. Amer. Control Conf. (ACC)*, Jun. 1998, pp. 1700–1704.
[50] S. Chatterjee and V. Mukherjee, "PID controller for automatic voltage regulator using teaching–learning based optimization technique," *Int. J. Electr. Power Energy Syst.*, vol. 77, pp. 418–429, May 2016.
[51] Z.-L. Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," *IEEE Trans. Energy Convers.*, vol. 19, no. 2, pp. 384–391, Jun. 2004.

MAHMOUD ELSISI was born in Cairo, Egypt, in 1989. He received the B.Sc., M.Sc., and Ph.D. degrees from the Department of Electrical Power and Machine, Faculty of Engineering at Shoubra, Benha University, Cairo, Egypt, in 2011, 2014, and 2017, respectively. He worked as an Assistant Professor with the Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University. He is currently a Postdoctoral Researcher with the Industry 4.0 Implementation Center, Center for Cyber-Physical System Innovation, National Taiwan University of Science and Technology, Taiwan. His research interest includes studying the power system dynamics: stability and control, artificial intelligence techniques, robotics, Industry 4.0, the IoT, and machine learning.



KARAR MAHMOUD received the B.Sc. and M.Sc. degrees in electrical engineering from Aswan University, Aswan, Egypt, in 2008 and 2012, respectively, and the Ph.D. degree from the Electric Power and Energy System Laboratory (EPESL), Graduate School of Engineering, Hiroshima University, Hiroshima, Japan, in 2016.

Since 2010, he has been with Aswan University, where he is currently an Assistant Professor with the Department of Electrical Engineering. He is also a Postdoctoral Researcher with the Prof. M. Lehtonen's Group, School of Electrical Engineering, Aalto University, Finland. He has authored or coauthored more than 70 publications in top-ranked journals, including IEEE journals, international conferences, and book chapters. His research interests include power systems, renewable energy sources, smart grids, distributed generation, optimization, applied machine learning, the IoT, Industry 4.0, and high voltage. In 2021, he became a Topic Editor in *Sensors* journal (MDPI).



MATTI LEHTONEN received the master's and Licentiate degrees in electrical engineering from the Helsinki University of Technology, Finland, in 1984 and 1989, respectively, and the Doctor of Technology degree from the Tampere University of Technology, Finland, in 1992. He was with VTT Energy, Espoo, Finland, from 1987 to 2003. Since 1999, has been a Full Professor and the Head of Power Systems and High Voltage Engineering Group's with Aalto University, Espoo. His research interests include power system planning and assets management, power system protection including earth fault problems, harmonic related issues, high voltage systems, power cable insulation, and polymer nanocomposites. He is an Associate Editor of *Electric Power Systems Research*, and *IET Generation, Transmission and Distribution*.

MOHAMED M. F. DARWISH was born in Cairo, Egypt, in 1989. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical power engineering from the Faculty of Engineering at Shoubra, Benha University, Cairo, in 2011, 2014, and 2018, respectively.

From 2016 to 2017, he joined the Prof. M. Lehtonen's Group, Department of Electrical Engineering and Automation, Aalto University, as a Ph.D. Student. He is currently a Postdoctoral Researcher with the School of Electrical Engineering, Aalto University, Finland. He is also working as an Assistant Professor with the Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University. He has authored of several international IEEE journals and conferences. His research interests include HV polymer nano-composites, nano-fluids, partial discharge detection, dissolved gas analysis, pipeline induced voltages, electromagnetic fields, renewables, control systems, optimization, applied machine learning, the IoT, Industry 4.0, and superconducting materials. He received the Best Ph.D. Thesis Prize that serves industrial life and society all over Benha University staff for the academic year from 2018 to 2019. In 2021, he became a Topic Editor in *Catalysts* journal (MDPI).