Mathematical Biosciences and Engineering

MBE, 19(9): 9371–9387.
DOI: 10.3934/mbe.2022436
Received: 28 April 2022
Revised: 16 June 2022
Accepted: 23 June 2022
Published: 27 June 2022

http://www.aimspress.com/journal/MBE

Research article

Optimal design of model predictive controller based on transient search optimization applied to robotic manipulators

Xingjia Li[1], Jinan Gu[1,*], Zedong Huang[1], Wenbo Wang[1] and Jing Li[1,2]

[1] School of Mechanical Engineering, Jiangsu University, Zhenjiang 212013, China
[2] School of Electronic Information and Electrical Engineering, Anyang Institute of Technology, Anyang 455000, China

* **Correspondence:** Email: gujinan@tsinghua.org.cn.

Abstract: Due to nonlinearity and uncertainty of the robotic manipulator, the design of the robot controller has a crucial impact on its performance of motion and trajectory tracking. In this paper, the linear parameter varying (LPV) - model predictive controller (MPC) of a two-link robot manipulator is established and then the controller's optimal parameters are determined via a newly developed meta-heuristic algorithm, transient search optimization (TSO). The proposed control method is verified by set point and nonlinear trajectory tracking. In the test of set-point tracking, the LPV-MPC scheme optimized by TSO has better performance compared to the computed torque controller (CTC) schemes tuned by TSO or other metaheuristic algorithms. In addition, good performances can also be observed in the tests of nonlinear trajectory tracking via the LPV-MPC scheme by TSO. Moreover, the robustness of the method to structural uncertainty is verified by setting a large system parameter deviation. Results reveal that we achieved some improvements in the optimization of MPC of the robot manipulator by employing the proposed method.

Keywords: model predictive controller (MPC); linear parameter-varying (LPV); metaheuristic algorithm; transient search optimization (TSO); computed torque controller (CTC)

1. Introduction

In the past decade, robot manipulators have been extensively concerned in industrial assembly, agricultural picking, medical surgery, and other fields [1]. Robot manipulators usually face uncertainties such as parameter perturbations, external interferences, frictions, and noises during the service process. Meanwhile, it is expected of high working accuracy such as in position reaching and trajectory tracking. Therefore, the control techniques of robot manipulators are fundamental.

PID controller has been widely used due to its simple structure and acceptable performance during the early stages of the robot industry [2]. However, it is quite challenging to obtain the optimum PID parameters because robot manipulators are complex systems with nonlinearity, strong coupling and time varying characteristics. Even worse, for occasions with high precision requirements and complex dynamic constraints, PID is difficult to achieve an ideal performance. To improve the control precision of robot manipulators, sustained efforts have been made to implement robust and optimal control. Many novel control techniques have since emerged, for example, artificial neural network controller [3], fuzzy logical controller [4–6], adaptive nonlinear controller [7], sliding mode controller [8], linear matrix inequality scheme [9], linear quadratic controller [10], $H\infty$ controller [11,12], reinforcement learning-based controller [13], model predictive control (MPC) [14,15] and other optimal control strategies [16–18].

Optimal control techniques always have a distinctive feature, which can simultaneously consider the constraints of input variables, output variables and state variables. Due to robustness, MPC is considered to be a promising controller among the optimal control techniques for the applications in industrial robots [19]. Satoh et al. proposed a disturbance observer-based MPC for the tracking control of manipulators [20]. Wilson et al. introduced a simplified nonlinear model predictive control (NMPC) for a 2-DoF (degree of freedom) vertical robot manipulator [21]. Best et al. put forward a control scheme based on MPC for a 5-DoF soft humanoid robot [22]. Based on MPC, Incremona et al. presented a hierarchical multiloop control scheme integrated with a sliding mode controller for a 3-DoF robot manipulator [19]. Guechi et al. compared MPC with a linear quadratic controller of a 2-DoF robot manipulator [23]. Carron et al. developed a Gaussian process-based MPC for the offset-free tracking of a robot manipulator [24]. MPC and NMPC always encounter some parameters to be determined, and intuitively chosen parameters mostly bring mediocre performance. The aforementioned MPC/NMPC parameters are determined by the trial and error method or the Ziegler Nichols method. Nevertheless, these traditional optimization methods are usually brutal or cumbersome to determine the parameters of MPC/NMPC for robot manipulators due to the nonlinearity and complexity.

The determination of MPC parameters can also be transformed into an engineering optimization problem. In this way, the problem can be extended to take advantage of the benefits from the meta heuristics, which is an efficient strategy for solving complex engineering optimization problems based on computational intelligence. Growing numbers of metaheuristic algorithms have emerged in recent years, such as Particle swarm optimization (PSO) [25], Grey wolf optimizer (GWO) [26], Monarch butterfly optimization (MBO) [27], Sparrow search algorithm (SSA) [28], Slime mould algorithm (SMA) [29], Moth search algorithm (MSA) [30], Hunger games search (HGS) [31], RUNge Kutta optimizer (RUN) [32], Colony predation algorithm (CPA) [33], Multi-tracker optimization (MTO) [34], weIghted meaN oF vectOrs (INFO) [35], Harris hawks optimization (HHO) [36], Mayfly optimization (MO) [37], and Satin bowerbird optimizer (SBO) [38]. Lately, a new metaheuristic algorithm, Transient search optimization (TSO) [39], inspired by the transient physical behavior of switched electrical circuits with inductance and capacitance included, was initiated and it has shown considerable competitiveness for solving engineering optimization problems.

To address the problem of the determination of MPC parameters, Elsisi recently provided an optimal design scheme of NMPC on the basis of MTO, by using the packaged NMPC toolbox in Matlab [40]. However, it is noted that NMPC generally requires higher computational time or cost than MPC does. In addition, to our best knowledge, only seldom literature discussed the optimization of MPC/NMPC by metaheuristic algorithm and no existing applications of TSO in MPC parameter optimization have been mentioned yet. This motivated our attempt to propose a new scheme for optimizing MPC parameters of the robotic manipulator by using TSO.

In this paper, inspired by [39] and [40], aiming of reducing the computational complexity due to nonlinearity of NMPC and extending the application of TSO algorithm, an alternative TSO-based MPC (TSO-MPC for short) controller is proposed. Specifically, the nonlinear dynamics equation of a two-link robot manipulator is transformed into linear parametric-varying (LPV) forms and then the corresponding MPC scheme is established. Afterwards, the TSO algorithm is employed to optimize the objective function, figure of demerit (FoD), of the MPC for the robot manipulator. As will be seen in a later section, the proposed TSO-MPC suggests state-of-the-art performance in solving the tracking problem of the robot manipulator.

The remainder of this paper is organized as follows. Section 2 reviews the dynamic model of the robotic system and establishes the linear parametric-varying MPC (LPV-MPC, also written by MPC for short below). In Section 3, the TSO is described and the objective function and error indices to be optimized are given. Section 4 implements the proposed MPC scheme and optimization process, and discusses the optimized performance of the MPC scheme in set-point tracking, trajectory tracking, and robustness on parameters uncertainties. Section 5 summarizes the full text.

2. LPV-MPC for robot manipulators

2.1. Dynamics equation of robot manipulators

When a robot manipulator working in a horizontal plane, the dynamics equation can be denoted by [41]

$$\boldsymbol{D}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{u} \tag{1}$$

where $\mathbf{q}$ is the generalized coordinates, $\mathbf{D}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is the centrifugal and Coriolis torque, and $\mathbf{u}$ is the input torque of the links. Let $\mathbf{x} = [\mathbf{q},\dot{\mathbf{q}}]$ and $\mathbf{y} = \mathbf{q}$, then the nonlinear dynamics equation can be reformed as a linear parameter-varying (LPV) model:

$$\begin{cases}\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(p(t))\boldsymbol{x}(t) + \boldsymbol{B}(p(t))\boldsymbol{u}(t)\\ \boldsymbol{y}(t) = \boldsymbol{C}(p(t))\boldsymbol{x}(t)\end{cases} \tag{2}$$

where $\boldsymbol{A} = \begin{bmatrix}\mathbf{0} & \boldsymbol{I}\\ \mathbf{0} & \boldsymbol{D}^{-1}(-\boldsymbol{C})\end{bmatrix}$, $\boldsymbol{B} = \begin{bmatrix}\mathbf{0}\\ \boldsymbol{D}^{-1}\end{bmatrix}$, and $\boldsymbol{C} = \begin{bmatrix}\boldsymbol{I}\\ \mathbf{0}\end{bmatrix}$. $\boldsymbol{A}$ and $\boldsymbol{B}$ depend on the parameter $p(t)$.

To facilitate digital control, Eq (2) needs to be discretized. Set the sampling period as T_s, then

$$\begin{aligned}\boldsymbol{x}(k+1) &= \boldsymbol{A}_d(p(k))\boldsymbol{x}(k) + \boldsymbol{B}_d(p(k))\boldsymbol{u}(k)\\ \boldsymbol{y}(k) &= \boldsymbol{C}_d(p(k))\boldsymbol{x}(k)\end{aligned} \tag{3}$$

where $\boldsymbol{A}_d = e^{\boldsymbol{TsA}}$, $\boldsymbol{B}_d = \boldsymbol{A}^{+}(e^{\boldsymbol{TsA}} - \boldsymbol{I})\boldsymbol{B}$. And $\boldsymbol{A}^{+}$ is the pseudo inverse matrix of $\boldsymbol{A}_{\mathrm{d}}$.

2.2. LPV-MPC for robot manipulators

In the LPV-MPC, the objective function needs to minimize the error between the output trajectory and the desired one. Most of the time, extra penalties concerning the input torques are also needed. Thus the objective function can be defined by

$$J = \sum_{i=0}^{N} \left\| \boldsymbol{w}_y (\boldsymbol{y}(k+i|k) - \boldsymbol{r}(k+i|k)) \right\|_2^2 + \sum_{i=0}^{N-1} \| \boldsymbol{w}_u (\boldsymbol{u}(k+i|k)) \|_2^2 \tag{4}$$

where w_y and w_u are diagonal matrices concerning the output and input weighting values. $\boldsymbol{y}(k+i\,|\,k)$, $\boldsymbol{r}(k+i\,|\,k)$, and $\boldsymbol{u}(k+i\,|\,k)$ respectively represent the prediction output, reference output and prediction input for the (k + i) times during the k^{th} sampling. $\|\cdot\|_2$ is the Euclidean norm.

Let $\boldsymbol{Q} = \boldsymbol{w}_y^T \boldsymbol{w}_y$, $\boldsymbol{R} = \boldsymbol{w}_u^T \boldsymbol{w}_u$ and $\boldsymbol{e}(k+i\,|\,k) = \boldsymbol{y}(k+i\,|\,k) - \boldsymbol{r}(k+i\,|\,k)$, then

$$\begin{aligned} J &= \sum_{i=0}^{N} \boldsymbol{e}(k+i\,|\,k)^T \boldsymbol{Q} \boldsymbol{e}(k+i\,|\,k) + \sum_{i=0}^{N-1} \boldsymbol{u}(k+i\,|\,k)^T \boldsymbol{R} \boldsymbol{u}(k+i\,|\,k) \\ &= \boldsymbol{e}(k)^T \boldsymbol{Q} \boldsymbol{e}(k) + \sum_{i=1}^{N} \boldsymbol{e}(k+i\,|\,k)^T \boldsymbol{Q} \boldsymbol{e}(k+i\,|\,k) + \sum_{i=0}^{N-1} \boldsymbol{u}(k+i\,|\,k)^T \boldsymbol{R} \boldsymbol{u}(k+i\,|\,k) \end{aligned} \tag{5}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are the error weight matrix and the input torque weight matrix, respectively. They are also the parameter to be optimized in MPC in this article.

Eventually, the objective function at sampling step k can be formulated as:

$$\min_{u_k} J = \sum_{i=1}^{N} \boldsymbol{e}(k+i\,|\,k)^T \boldsymbol{Q} \boldsymbol{e}(k+i\,|\,k) + \sum_{i=0}^{N-1} \boldsymbol{u}(k+i\,|\,k)^T \boldsymbol{R} \boldsymbol{u}(k+i\,|\,k) \tag{6}$$

$$s.\,t. \quad \begin{aligned} &\boldsymbol{x}(k+1) = \boldsymbol{A}_d(k)\boldsymbol{x}(k) + \boldsymbol{B}_d(k)\boldsymbol{u}(k) \\ &\boldsymbol{u}_{\min} \le \boldsymbol{u}_k \le \boldsymbol{u}_{\max} \end{aligned}$$

where $\boldsymbol{u}_{\min}$ and $\boldsymbol{u}_{\max}$ represent the lower and upper limits of the input torque.

3. Transient search optimization and objective function design

TSO is inspired by the transient physical behavior of the switched electrical circuits rather than by swarm animals' behavior. It has been testified the superiority against other algorithms, simple yet powerful.

3.1. Transient search optimization

The TSO algorithm includes three main procedures: i) Initialization, the procedure to generate the search agents among the search space; ii) Exploration, the procedure to explore the entire search space to seek diverse solutions; iii) Exploitation, the procedure to search for better solutions close to the local optimal solutions.

Like other metaheuristic algorithms, the initialization of the search agent is generated randomly, as denoted by Eq (7)

$$X=\begin{bmatrix} X_{11} & \cdots & X_{1d} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nd} \end{bmatrix} \tag{7}$$

where d and n describe the variables' dimension and the total number of search agents therein. For instance, X_{nd} is the coordinate of the n^{th} search agent in d^{th} dimension.

$$X_{ij}^{t+1}=\begin{cases} X_{best}^{t}+\sin(2\pi T+\dfrac{\pi}{4})\cdot\left|X_{ij}^{t+1}-c_1 X_{best}^{t}\right|\cdot e^{-\Gamma}, & r_1\geq 0.5 \\ X_{best}^{t}+\left(X_{ij}^{t+1}-c_1 X_{best}^{t}\right)\cdot e^{-\Gamma}, & r_1<0.5 \end{cases} \tag{8}$$

where $c_1=r_3\times k_{\mathrm{c}}\times c_2+1$, $c_2=2\left(1-i/t_{\max}\right)$ and $\Gamma=\left(2r_2-1\right)\times c_2$; r_1, r_2 and r_3 are random numbers in the interval [0,1], t is the number of iterations, $t_{\max}$ is the maximum number of iterations; c_1 and Γ are random coefficients, c_2 is an attenuation variable gradually decaying from 2 to 0, and k_{c} is a constant. X_{ij}^{t} is the search location of the agent X_{ij} in iteration t; X_{best}^{t} is the current best solution in iteration t. When r_1 is greater than or equal to 0.5, TSO implements the exploration process; otherwise, TSO implements the exploitation process. Besides, by using the big-oh notation O(n), the computational complexity of TSO is $O\left(n\left(t_{\max}d+t_{\max}+1\right)\right)$.

3.2. Objective function and error indices of the TSO-MPC

Many factors can influence the performance of the robot manipulator response. An ideal answer of robot manipulators must have slight maximum overshoot, settling time, and steady-state error in the meantime. In this work, Figure of demerit (FoD) is selected as the objective function to evaluate the control performance of the robot system to minimize the maximum overshoot, steady-state error, and settling time synchronously. The objective function FoD is represented by

$$\mathrm{FoD}=\sum_{i=1}^{2}\left[\left(1-\mathrm{e}^{-\eta}\right)\left(M_{os}^{i}+e_{ss}^{i}\right)+\mathrm{e}^{-\eta}\left(t_{st}^{i}-t_{r}^{i}\right)\right] \tag{9}$$

where M_{os}^{i} is the maximum overshoot (%), t_{st}^{i} is the settling time (sec), t_{r}^{i} is the rise time (sec), e_{ss}^{i} is the steady-state error, η is an exponential factor to balance the weights of the term $\left(M_{os}^{i}+e_{ss}^{i}\right)$ and the term $\left(t_{st}^{i}-t_{r}^{i}\right)$, and i is the index of robot links.

By setting the value of ψ, one can adjust the penalty degree of the items. Specifically, when η = 0.6932, the weight factor $\left(1-\mathrm{e}^{-\eta}\right)$ and $\mathrm{e}^{-\eta}$ are equal, which means the term of the maximum overshoot, the term of the steady-state error, and the term of the setting time have fair influences on the objective function FoD.

In addition, for quantitative comparative analysis, four popular error indices are employed to evaluate the performance of the controller approaches [42]. The error indices can be denoted as follows.

$$\begin{aligned} \text{IAE}: &\quad J=\int_0^{\infty}\left|e(t)\right|dt \\ \text{ITAE}: &\quad J=\int_0^{\infty}t\left|e(t)\right|dt \\ \text{ISE}: &\quad J=\int_0^{\infty}e^2(t)dt \\ \text{ITSE}: &\quad J=\int_0^{\infty}te^2(t)dt \end{aligned} \tag{10}$$

where ISE is the abbreviation of the integral of the squared error, ITSE is the abbreviation of the integral of the product of time and the square error, IAE is the abbreviation of the integral of the absolute error, ITAE is the abbreviation of the integral time absolute error.

4. Results and discussion

Without loss of generality, a two-link robot manipulator working in a horizontal plane is suggested in Figure 1.

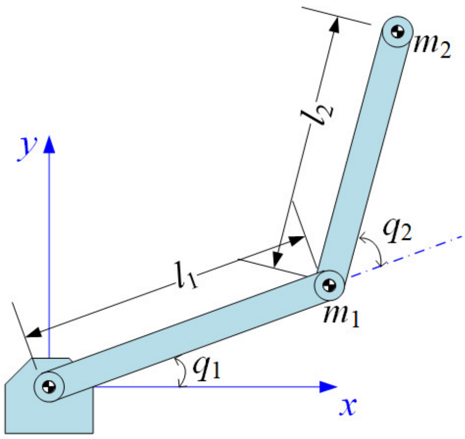


Figure 1. Diagram of two-link planar robot manipulator.

Thus the terms $\mathbf{D}$ and $\mathbf{C}$ of the dynamics equation (1) can be presented as

$$\mathbf{D}(\mathbf{q}) = \begin{pmatrix} (m_1+m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos q_2 & m_2 l_2^2 + m_2 l_1 l_2 \cos q_2 \\ m_2 l_2^2 + m_2 l_1 l_2 \cos q_2 & m_2 l_2^2 \end{pmatrix}$$
$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{pmatrix} -m_2 l_1 l_2 \dot{q}_2 \sin q_2 & -m_2 l_1 l_2 (\dot{q}_1 + \dot{q}_2) \sin q_2 \\ m_2 l_1 l_2 \dot{q}_1 \sin q_2 & 0 \end{pmatrix} \quad (11)$$

where the structural parameters in this work are set to $l_1 = 0.8$, $l_2 = 0.4$, $m_1 = 0.1$, and $m_2 = 0.1$; g is the gravity acceleration.

4.1. Optimization of TSO-MPC of the robot manipulator

The optimization progress of MPC and CTC based on the TSO of robot manipulators was implemented via a unit step reference, as shown in Figure 2.

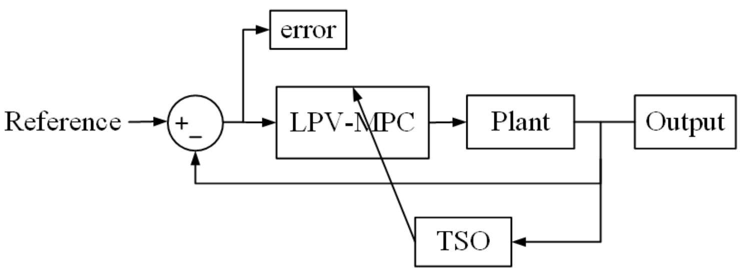


Figure 2. Tuning process of LPV-MPC using TSO.

In the optimization of MPC and CTC, the sampling period is set to T_s = 0.01 s, and a prediction horizon H_p = 10 is employed. In the MPC control, the parameters to be optimized are **Q** and **R**; We set $\boldsymbol{Q}=\begin{bmatrix} Q_1\cdot\boldsymbol{I}_{2\times2} & \boldsymbol{0}_{2\times2} \\ \boldsymbol{0}_{2\times2} & Q_2\cdot\boldsymbol{I}_{2\times2}\end{bmatrix}$, $\boldsymbol{R}=\begin{bmatrix} R_1 \\ R_2\end{bmatrix}$; $0.1\le Q_1,Q_2\le 50$, $0\le R_1,R_2\le 50$, and $u_{\min}=-500$, $u_{\max}=500$. In the CTC control, the parameters to be optimized are $\boldsymbol{k}_{\mathrm{p}}$ and $\boldsymbol{k}_{\mathrm{d}}$.

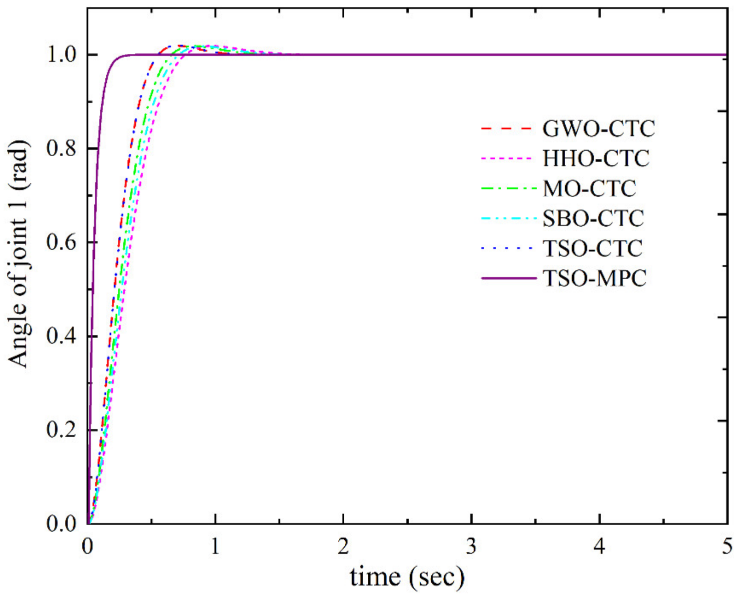
During the procedure of the simulation experiment, the search agents of these metaheuristic algorithms are 30, and the maximum iterations of the optimization are 50 times. Expressly, in the TSO method, the constant k_c is set to 1; In GWO, HHO, MO, and SBO, the required parameters are consistent with the original literature. For instance, in GWO, a decreases linearly from 2 to 0; in HHO, β is set to 1.5; in MO, g = 0.8, a_1 = 1.0, a_2 = 1.5, a_3 = 1.5, β = 2, and mutation rate is set to 0.01; in SBO, α = 0.94, mutation rate is set to 0.05, and the percent of the difference between the upper and lower limit is set to 0.02. Afterwards, each algorithm takes the mean value of six independent trials as the final result. After the optimization process, we get the cost function value FoD of these different methods, the standard deviation, the key parameters of the controllers, the settling time, and the maximum overshoot, as tabulated in Table 1. Then the set-point tracking performance on the unit step reference for the robot manipulator using these controllers can be illustrated in Figure 3.

As can be seen from Table 1 and Figure 3, the proposed TSO-MPC has the minimum settling time, overshoot, as well as the performance indices compared with the other methods. Particularly, by employing TSO-MPC, the overshoot of joints 1 and 2 are both zero. To compare with the trial and error method, we set Q_1 = 0.1, Q_2 = 50, R_1 = R_2 = 0; however, the FoD value obtained is 0.06952. It is slightly larger than the optimal value of 0.06455.

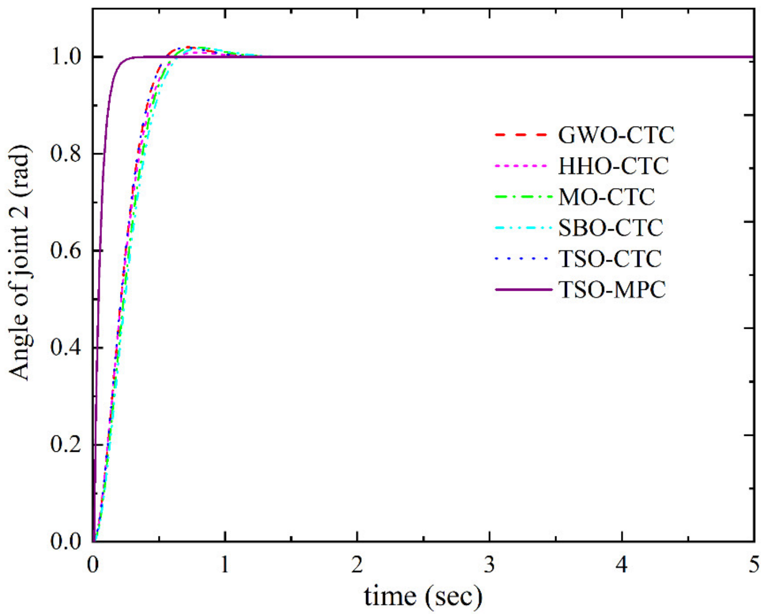
In conclusion, the larger the weight coefficient of the angle, the smaller the weight coefficient of the angular speed, and the smaller the weight coefficient of the input torque, the better the performance of the objective function FoD. Although the optimal parameters are not all at the boundaries, they are very close to them.

Table 1. The cost function value, optimal parameters and performances of the metaheuristic-algorithm-based CTC and MPC.

		GWO-CTC	HHO-CTC	MO-CTC	SBO-CTC	TSO-CTC	TSO-MPC
FoD	Ave.	0.10401	0.12794	0.11343	0.1222	0.10386	0.06455
	St.d.	0.00675	0.00379	0.00711	0.00703	0.00383	0.00332
Optimal parameters	Link1 and Link2	kp1 = 11.2297, kd1 = 49.5653, kp2 = 11.0395, kd2 = 48.3801.	kp1 = 8.1915, kd1 = 26.5169, kp2 = 11.9300, kd2 = 49.8765.	kp1 = 9.5640, kd1 = 35.9166, kp2 = 10.0813, kd2 = 40.3256.	kp1 = 8.88092, kd1 = 30.9013, kp2 = 9.80207, kd2 = 37.4771.	kp1 = 11.1218, kd1 = 48.9795, kp2 = 11.164, kd2 = 48.9795.	Q1 = 0.1010, R1 = 0, Q2 = 50, R2 = 0.
Settling time(s)	Link1	0.5	0.69	0.59	0.64	0.5	0.19
	Link2	0.5	0.55	0.55	0.59	0.5	0.19
Maximum overshoot(%)	Link1	1.89468	1.85569	1.83179	1.78839	1.97104	0
	Link2	1.99610	0.86986	1.95880	1.77093	1.89138	0



(a)



(b)

Figure 3. Set-point tracking performance for (a) joint 1 and (b) joint 2 using MPC and CTC based on the TSO and other metaheuristic algorithms.

4.2. Performance of set-point tracking

Set-point tracking control is an important performance index for the control technique of robot manipulators. Unit step tracking is a typical set-point tracking. In the previous section, MPC and CTC are optimized using a unit step reference trajectory, and key parameters in these control methods are obtained. For quantitative comparative purposes, different error indices are employed to evaluate the performance of different control algorithms in terms of set-point tracking. The implementations of these algorithms under the error indexes are listed in Table 2.

Table 2. The error indices of the control schemes by the different metaheuristic algorithms.

	GWO-CTC	HHO-CTC	MO-CTC	SBO-CTC	TSO-CTC	TSO-MPC
IAE	0.45338	0.45337	0.45337	0.50303	0.45561	0.11608
ITAE	0.07620	0.07666	0.076198	0.09532	0.08517	0.00561
ISE	0.28802	0.29004	0.28802	0.31247	0.29497	0.06850
ITSE	0.00486	0.00492	0.00486	0.00700	0.00500	8.38E-05

As can be seen from Table 2, all the error indicators of the TSO-MPC algorithm are the smallest. Again, it is proved that TSO-MPC has the best performance among the control schemes.

4.3. Performance of nonlinear trajectory tracking

In addition to set-point tracking, nonlinear trajectory tracking is also essential in various applications of robot manipulators. Besides, proper nonlinear excitation trajectories for the robot manipulator are required to test the effectiveness of the MPC methods.

For this purpose, the periodic trajectories by a sum of finite Fourier series are utilized hereinafter due to the ascendancy in terms of signal processing and target reaching. The finite-term periodic Fourier series trajectories for joint *i* of the robot manipulator can be formulated as

$$q_i(t) = \sum_{i=1}^{N} a_i \cos\left(\omega_f it\right) + \sum_{i=1}^{N} b_i \sin\left(\omega_f it\right) + q_{0i} \tag{12}$$

where ω_f is the fundamental frequency, a_i and b_i are amplitudes of the terms, and q_{io} is the offsets of the i^{th} joint angles. The fundamental frequencies are all set to 0.5, and thus, all of the period of the trajectories is 4π second.

In order to facilitate analysis, we let a_1 = [0.156, -0.478, 0.078, -0.388, -0.070], b_1 = [0.088, 0.253, -0.207, 0.549, 0.150], a_2 = [0.064, -0.335, 0.451, 0.292, 0.746], b_2 = [-0.125, 0.292, -0.369, 0.557, 0.564], q_0 = [0.168, 0.193]; and the initial state of the system is set to $q_1(0) = \dot{q}_1(0) = q_2(0) = \dot{q}_2(0) = 0$. The final generated nonlinear reference trajectory is presented in Figure 4.

After simulation, the output response of the proposed MPC for tracking nonlinear trajectory is demonstrated in Figure 5. It can be seen that the performance of TSO-MPC still follows the nonlinear trajectory well, and the steady-state error is insignificant.

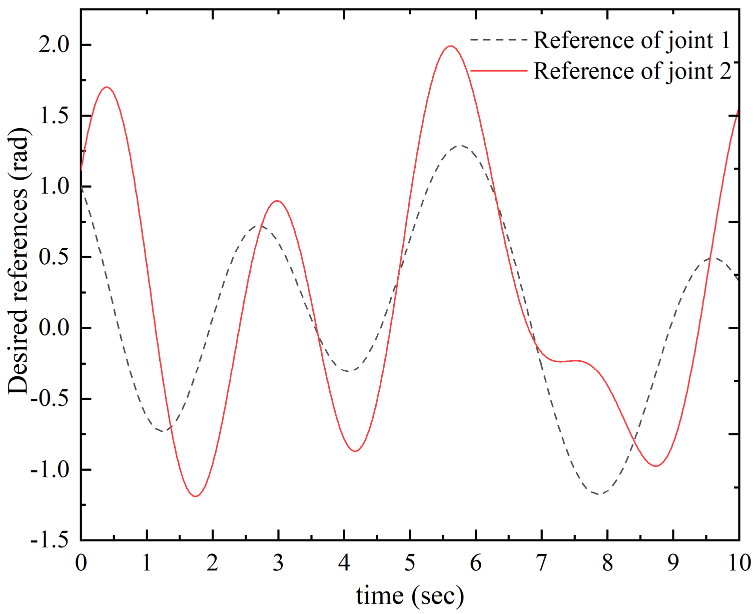


Figure 4. The design of the nonlinear reference trajectories.

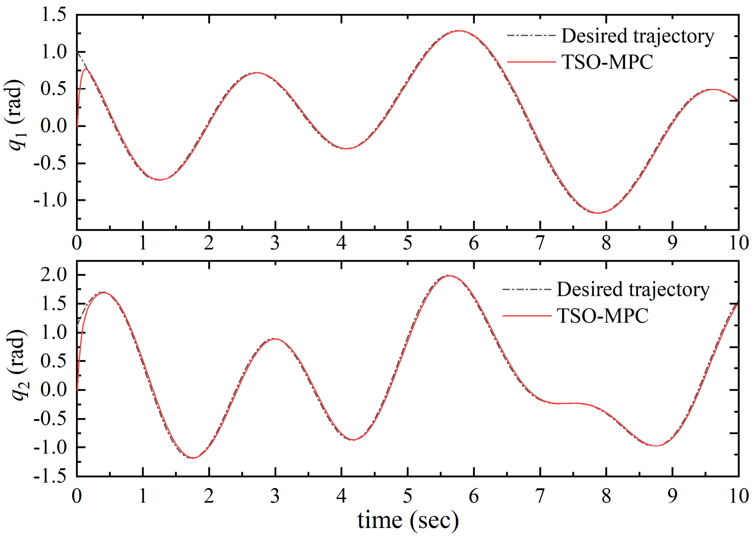


Figure 5. Tracking performance of (a) joint 1 and (b) joint 2 on the nonlinear reference using TSO-MPC.

4.4. Performance of robustness regarding parameter uncertainties

Robot manipulators always encounter varieties of structural and non-structural uncertainties when in service. Uncertainty factors adversely affect the tracking implementation and performance of the robotic system. Therefore, the controllers of robot manipulators are required of good robustness to uncertainty factors. In this section, a pair of trails are conducted to validate the robust performance of the proposed TSO-MPC scheme. Specifically, ± 50% measurement errors in terms of the link masses and the link lengths of the robot manipulator were established to test and verify the robustness of the proposed TSO-MPC.

As demonstrated in Figures 6 and 7, the TSO-MPC can tackle the uncertainties of the system parameters with an ignorable steady-state error in the system response. It should be pointed out that there are many other excellent algorithms that can be used to optimize MPC of the robot manipulator, but they were not yet considered in this work.

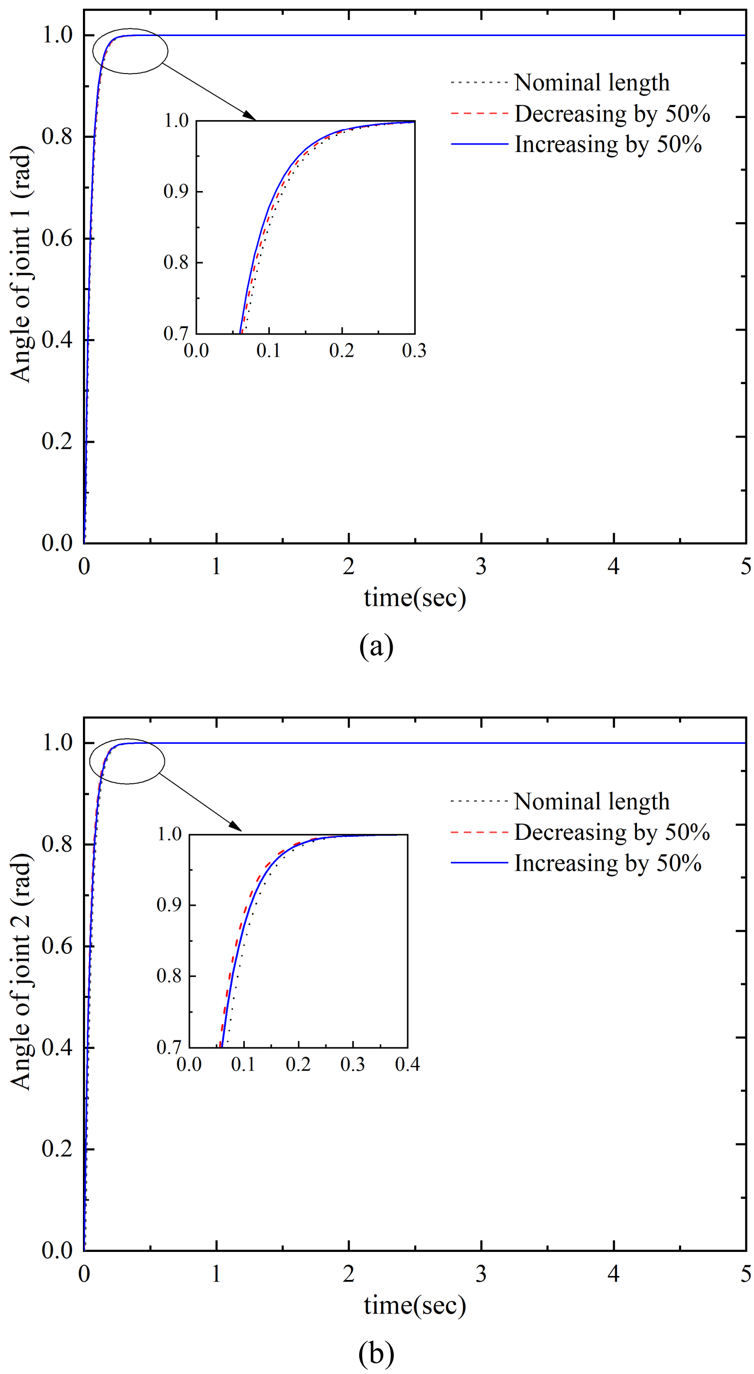
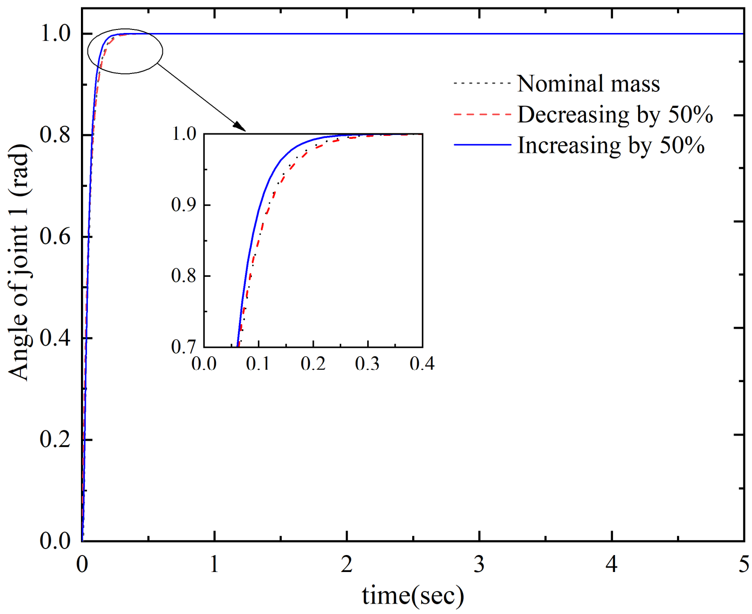
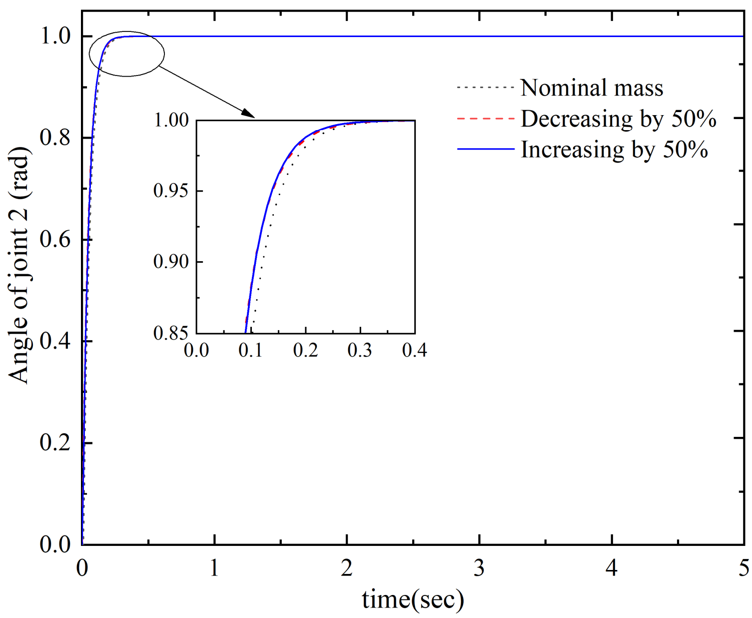


Figure 6. Robust performance against length uncertainty for (a) joint 1 and (b) joint 2 using TSO-MPC.



(a)



(b)

Figure 7. Robust performance against mass uncertainty for (a) joint 1 and (b) joint 2 using TSO-MPC.

5. Conclusions

A newly proposed metaheuristic, TSO algorithm, was employed for tuning the LPV-MPC parameters of robot manipulators rather than using the trial and error method from experts' experience. The parameters of LPV-MPC were turned using TSO by the minimization of an objective function FoD, which can simultaneously minimize overshoot, steady-state error, and settling time. It is suggested that the larger the weight coefficient of the angle, the smaller the weight coefficient of the angular speed, and the smaller the weight coefficient of the input torque, the better the performance of the TSO-MPC. The control performances achieved with TSO-MPC controllers are then compared with the performances obtained using GWO-CTC, HHO-CTC, MO-CTC, SBO-CTC, and TSO-CTC controllers. Although all the algorithms can achieve small steady-state error, TSO-MPC has better performance in terms of maximum overshoot and settling time than the metaheuristic-algorithm-based CTC schemes. Moreover, TSO-MPC can also effectively track nonlinear trajectories and handle the uncertainty of the robot manipulator parameters. The results can indicate that the proposed TSO-MPC method in some areas is more efficient than other control schemes of robot manipulators.

Acknowledgments

This project is financially supported by the National Natural Science Foundation of China (No. 51875266).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. P. Quan, Y. Lou, H. Lin, Z. Liang, S. Di, Research on fast identification and location of contour features of electric vehicle charging port in complex scenes, *IEEE Access*, **10** (2022), 26702–26714. https://doi.org/10.1109/ACCESS.2021.3092210
2. S. Skogestad, Simple analytic rules for model reduction and PID controller tuning, *J. Process Control*, **13** (2003), 291–309. https://doi.org/10.1016/S0959-1524(02)00062-8
3. M. K. Singh, D. R. Parhi, Path optimisation of a mobile robot using an artificial neural network controller, *Int. J. Syst. Sci.*, **42** (2011), 107–120. https://doi.org/10.1080/00207720903470155
4. A. Kosari, H. Jahanshahi, S. A. Razavi, Optimal FPID control approach for a docking maneuver of two spacecraft: translational motion, *J. Aerosp. Eng.*, **30** (2017), 04017011. https://doi.org/10.1061/(ASCE)AS.1943-5525.0000720
5. S. S. Haq, D. Lenine, S. Lalitha, Performance enhancement of UPQC using Takagi–Sugeno fuzzy logic controller, *Int. J. Fuzzy Syst.*, **23** (2021), 1765–1774. https://doi.org/10.1007/s40815-021-01095-w
6. A. Kosari, H. Jahanshahi, S. A. Razavi, An optimal fuzzy PID control approach for docking maneuver of two spacecraft: Orientational motion, *Eng. Sci. Technol., Int. J.*, **20** (2017), 293–309. https://doi.org/10.1016/j.jestch.2016.07.018
7. L. L. Whitcomb, A. A. Rizzi, D. E. Koditschek, Comparative experiments with a new adaptive controller for robot arms, *IEEE Trans. Rob. Autom.*, **9** (1993), 59–70. https://doi.org/10.1109/70.210795
8. H. Benbouhenni, N. Bizon, A synergetic sliding mode controller applied to direct field-oriented control of induction generator-based variable speed dual-rotor wind turbines, *Energies*, **14** (2021), 4437–4453. https://doi.org/10.3390/en14154437

9. M. Veysi, J. Aghaei, M. Shasadeghi, R. Razzaghi, B. Bahrani, D. J. Ryan, Energy-efficient speed control of electric vehicles: linear matrix inequality approach, *IEEE Trans. Veh. Technol.*, **69** (2020), 10469–10483. https://doi: 10.1109/ tvt.2020.3008500
10. M. Moradi, F. Bayat, M. Charmi, A salient object detection framework using linear quadratic regulator controller, *J. Visual Commun. Image Represent.*, **79** (2021), 103259. https://doi.org/10.1016/j.jvcir.2021.103259
11. F. W. Alsaade, H. Jahanshahi, Q. Yao, M. S. Al-zahrani, A. S. Alzahrani, A new neural network-based optimal mixed H2/H∞ control for a modified unmanned aerial vehicle subject to control input constraints, *Adv. Space Res.*, 2022. https://doi.org/10.1016/j.asr.2022.02.012
12. S. Bekiros, H. Jahanshahi, F. Bezzina, A. A. Aly, A novel fuzzy mixed H2/H∞ optimal controller for hyperchaotic financial systems, *Chaos, Solitons Fractals*, **146** (2021), 110878. https://doi.org/10.1016/j.chaos.2021.110878
13. Q. Ding, H. Jahanshahi, Y. Wang, S. Bekiros, M. O. Alassafi, Optimal reinforcement learning-based control algorithm for a class of nonlinear macroeconomic systems, *Mathematics*, **10** (2022), 499–512. https://doi.org/10.3390/math10030499
14. H. Jahanshahi, S. S. Sajjadi, S. Bekiros, A. A. Aly, On the development of variable-order fractional hyperchaotic economic system with a nonlinear model predictive controller, *Chaos, Solitons Fractals*, **144** (2021), 110698. https://doi.org/10.1016/j.chaos.2021.110698
15. A. Hakimzadeh, V. Ghaffari, Designing of non-fragile robust model predictive control for constrained uncertain systems and its application in process control, *J. Process Control*, **95** (2020), 86–97. https://doi.org/10.1016/j.jprocont.2020.10.004
16. S. B. Chen, S. Soradi-Zeid, H. Jahanshahi, R. Alcaraz, J. F. Gómez-Aguilar, S. Bekiros, et al., Optimal control of time-delay fractional equations via a joint application of radial basis functions and collocation method, *Entropy*, **22** (2020), 1213. https://doi.org/10.3390/e22111213
17. S. B. Chen, S. Soradi-Zeid, M. Alipour, Y. Chu, J. F. Gomez-Aguilar, H. Jahanshahi, Optimal control of nonlinear time-delay fractional differential equations with Dickson polynomials, *Fractals*, **29** (2021), 2150079. https://doi.org/10.1142/S0218348X21500791
18. H. Wang, H. Jahanshahi, M. K. Wang, S. Bekiros, J. Liu, A. A. Aly, A Caputo–Fabrizio fractional-order model of HIV/AIDS with a treatment compartment: sensitivity analysis and optimal control strategies, *Entropy*, **23** (2021), 610. https://doi.org/10.3390/e23050610
19. G. P. Incremona, A. Ferrara, L. Magni, MPC for robot manipulators with integral sliding modes generation, *IEEE/ASME Trans. Mechatron.*, **22** (2017), 1299–1307. https://doi.org/10.1109/TMECH.2017.2674701
20. T. Satoh, R. Abe, N. Saito, Control of a two-link manipulator using disturbance observer-based model predictive control, *Trans. Japan Soc. Mech. Eng.*, **81** (2015), 15-00084. https://doi.org/10.1299/transjsme.15-00084
21. J. Wilson, M. Charest, R. Dubay, Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator, *Rob. Comput. Integr. Manuf.*, **41** (2016), 23–30. http://doi.org/10.1016/j.rcim.2016.02.003
22. C. M. Best, M. T. Gillespie, P. Hyatt, L. Rupert, V. Sherrod, M. D. Killpack, A new soft robot control method: using model predictive control for a pneumatically actuated humanoid, *IEEE Rob. Autom. Mag.*, **23** (2016), 75–84. http://doi.org/10.1109/mra.2016.2580591

23. E. H. Guechi, S. Bouzoualegh, Y. Zennir, Sašo Blažič, MPC control and LQ optimal control of a two-link robot arm: a comparative study, *Machines*, **6** (2018), 37. http://doi.org/10.3390/machines6030037
24. A. Carron, E. Arcari, M. Wermelinger, L. Hewing, M. Hutter, M. N. Zeilinger, Data-driven model predictive control for trajectory tracking with a robotic arm, *IEEE Rob. Autom. Lett.*, **4** (2019), 3758–3765. http://doi.org/10.1109/lra.2019.2929987
25. F. Liu, H. Huang, B. Li, F. Xi, A parallel learning particle swarm optimizer for inverse kinematics of robotic manipulator, *Int. J. Intell. Syst.*, **36** (2021), 6101–6132. https://doi.org/10.1002/int.22543
26. S. Mirjalili, S. M. Mirjalili, A. Lewis, Grey Wolf optimizer, *Adv. Eng. Software*, **69** (2014), 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007
27. G. G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization, *Neural Comput. Appl.*, **31** (2019), 1995–2014. https://doi.org/10.1007/s00521-015-1923-y
28. X. Li, J. Gu, X. Sun, J. Li, S. Tang, Parameter identification of robot manipulators with unknown payloads using an improved chaotic sparrow search algorithm, *Appl. Intell.*, **52** (2022), 10341–10351. https://doi.org/10.1007/s10489-021-02972-5
29. S. Li, H. Chen, M. Wang, A. A. Heidari, S. Mirjalili, Slime mould algorithm: a new method for stochastic optimization, *Future Gener. Comput. Syst.*, **111** (2020), 300–323. https://doi.org/10.1016/j.future.2020.03.055
30. G. G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, *Memetic Comput.*, **10** (2018), 151–164. https://doi.org/10.1007/s12293-016-0212-3
31. Y. Yang, H. Chen, A. A. Heidari, A. H. Gandomi, Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts, *Expert Syst. Appl.*, **177** (2021), 114864. https://doi.org/10.1016/j.eswa.2021.114864
32. I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, H. Chen, RUN beyond the metaphor: an efficient optimization algorithm based on Runge Kutta method, *Expert Syst. Appl.*, **181** (2021), 115079. https://doi.org/10.1016/j.eswa.2021.115079
33. J. Tu, H. Chen, M. Wang, A. H. Gandomi, The colony predation algorithm, *J. Bionic Eng.*, **18** (2021), 674–710. https://doi.org/10.1007/s42235-021-0050-y
34. E. Zakeri, S. A. Moezi, Y. Bazargan-Lari, A. Zare, Multi-tracker optimization algorithm: a general algorithm for solving engineering optimization problems, *Iran. J. Sci. Technol., Trans. Mech. Eng.*, **41** (2017), 315–341. https://doi.org/10.1007/s40997-016-0066-9
35. I. Ahmadianfar, A. A. Heidari, S. Noshadian, H. Chen, A. H. Gandomi, INFO: an efficient optimization algorithm based on weighted mean of vectors, *Expert Syst. Appl.*, **195** (2022), 116516. https://doi.org/10.1016/j.eswa.2022.116516
36. A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: algorithm and applications, *Future Gener. Comput. Syst.*, **97** (2019), 849–872. https://doi.org/10.1016/j.future.2019.02.028
37. K. Zervoudakis, S. Tsafarakis, A mayfly optimization algorithm, *Comput. Ind. Eng.*, **145** (2020), 106559. https://doi.org/10.1016/j.cie.2020.106559
38. S. Moosavi, V. K. Bardsiri, Satin bowerbird optimizer: a new optimization algorithm to optimize anfis for software development effort estimation, *Eng. Appl. Artif. Intell.*, **60** (2017), 1–15. https://doi.org/10.1016/j.engappai.2017.01.006

39. M. H. Qais, H. M. Hasanien, S. Alghuwainem, Transient search optimization: a new meta-heuristic optimization algorithm, *Appl. Intell.*, **50** (2020), 3926–3941. https://doi.org/10.1007/s10489-020-01727-y
40. M. Elsisi, Optimal design of nonlinear model predictive controller based on new modified multitracker optimization algorithm, *Int. J. Intell. Syst.*, **35** (2020), 1857–1878. https://doi.org/10.1002/int.22275
41. J. Jin, N. Gans, Parameter identification for industrial robots with a fast and robust trajectory design approach, *Rob. Comput. Integr. Manuf.*, **31** (2015), 21–29. https://doi.org/10.1016/j.rcim.2014.06.004
42. V. Bhatia, R. Ram, V. Kalaichelvi, R. Karthikeyan, Application of model predictive controller for 2-DOF robot manipulator, in *2015 10th International Symposium on Mechatronics and its Applications (ISMA)*, (2015), 1–5. https://doi.org/10.1109/ISMA.2015.7373478

Appendix

Table A. Abbreviations.

Abbreviations	Full name
CTC	Computed Torque Controller
CPA	Colony Predation Algorithm
DoF	Degree of Freedom
FoD	Figure of Demerit
GWO	Grey Wolf Optimizer
HGS	Hunger Games Search
HHO	Harris Hawks Optimization
ISE	Integral of the Squared Error
ITSE	Integral of the product of Time and the Square Error
IAE	Integral of the Absolute Error
ITAE	Integral of the product of Time and the Absolute Error
LPV	Linear Parameter Varying
MBO	Monarch Butterfly Optimization
MO	Mayfly Optimization
MPC	Model Predictive Controller
MSA	Moth Search Algorithm
MTO	Multi-Tracker Optimization
NMPC	Nonlinear Model Predictive Control
PID	Proportion Integration Differentiation

Continued on next page

PSO	Particle Swarm Optimization
RUN	RUNge Kutta optimizer
SSA	Sparrow Search Algorithm
SMA	Slime Mould Algorithm
SBO	Satin Bowerbird Optimizer
TSO	Transient Search Optimization
INFO	WeIghted meaN oF vectOrs