

Optimization of Multisource Sensor Exoskeleton Power-assisted Model for Power Grid Operation and Maintenance

Mingxian Liu,¹ Xinbo Zhou,^{2,3*} and Jibiao Li¹

¹Lijiang Power Supply Bureau, Yunnan Power Grid Co., Ltd., Yulong County, Lijiang 674100, China

²Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Chenggong District, Kunming 650500, China

³Kunming Holy Intelligent Technology Co., Ltd., Chenggong District, Kunming 650000, China

(Received April 18, 2023; accepted August 1, 2023)

Keywords: industrial exoskeleton, grid operation and maintenance, positive kinematics, tuna optimization algorithm, population hierarchy strategy, elite opposition-based learning

Artificial intelligence, wireless communication, heterogeneous sensors, human–machine integration, and other emerging technologies provide effective technical support for the intelligent digital transformation of grid operation and maintenance modes. Exoskeleton devices can take on larger weight loads for the human body to improve operational efficiency, safety, and security for operation scenarios that require long time assistance, such as carrying and lifting in grid operation and maintenance. In this study, for the power-assisted optimization problem of a multisource sensor industrial exoskeleton device, a workspace optimization model of a four-degree-of-freedom industrial upper limb exoskeleton is constructed on the basis of the principle of positive kinematics and the graphical solution method, and an improved tuna swarm optimization algorithm (ITSO) based on population hierarchy, elite backward learning, and genetic variation is proposed for the solution of the constructed model. The Tent chaos mapping mechanism is introduced to improve the population diversity on the basis of the traditional tuna algorithm, and the population hierarchy mechanism, elite backward learning, and genetic variation operator are introduced to further improve the global optimization capability of the algorithm. The designed algorithm is compared with the particle swarm algorithm, genetic algorithm, gray wolf algorithm, and other cutting-edge intelligent algorithms in cross-sectional simulation experiments, and the results show that the optimal search ability of ITSO is improved by 0.12, 0.16, 0.08, and 0.05% on average, respectively, compared with the other algorithms, which verify the feasibility of the model and the algorithm designed in this study for solving the exoskeleton power-assisted problem.

1. Introduction

Traditional power grid operation and maintenance involve long hours and a large number of repetitive and high-load operation scenarios, so the physical requirements of personnel are high. Moreover, maintenance operations have a high risk factor and require strong professionalism.

*Corresponding author: e-mail: zhouxinbo@stu.kust.edu.cn

<https://doi.org/10.18494/SAM4519>

Because the professionalism and safety awareness of personnel vary, any operational oversight can easily lead to safety accidents. To solve these problems, intelligent robotics has become widely used in power grid operation and maintenance in recent years.⁽¹⁾ Industrial exoskeletons, as one of them, can help in the operation and maintenance personnel complete repetitive, tedious, and dangerous tasks through a booster system and human–machine interaction technology, reduce work intensity and risk, and improve work efficiency.

Many research teams have proposed different types of power-assisted models for industrial exoskeletons, which involve key technologies such as structural design,⁽²⁾ power system design,⁽³⁾ sensors,⁽⁴⁾ and control system design.⁽⁵⁾ In a previous study,⁽⁶⁾ the camshaft acceleration and output torque accuracy of the upper limb exoskeleton cam mechanism were optimized on the basis of the radial basis function and NSGA-II. A new three-degree-of-freedom spherical mechanism was designed and applied to the exoskeleton shoulder design.⁽⁷⁾ The control strategy of the exoskeleton knee joint plunger cylinder was also simulated and optimized to solve the problem of excessive jitter during squatting.⁽⁸⁾

Although the industrial exoskeleton power-assisted model has great application prospects in power grid operation and maintenance, the current industrial exoskeleton still has some problems and deficiencies such as an unstable power system, low control accuracy, and insufficient operational flexibility.⁽⁹⁾ Therefore, in this study, we aim to improve the applicability and performance of the industrial exoskeleton power-assisted model in power grid operation and maintenance by optimizing the parameters and solving these problems and deficiencies.

Population intelligence optimization algorithms as bionic algorithms, such as genetic algorithm (GA),^(10,11) simulated annealing algorithm,⁽¹²⁾ and particle swarm algorithm (PSO),⁽¹³⁾ have a wide range of industrial applications with their advantages of efficiency,⁽¹⁴⁾ adaptability, robustness, flexibility, and scalability. In this study, we chose the tuna swarm optimization (TSO) algorithm, which has a better optimization seeking ability, to solve the established exoskeleton optimization model and introduce strategies such as population hierarchy and elite backward learning to further improve its solution performance.

According to the actual structural characteristics and performance constraints of the industrial operation and maintenance exoskeleton, we established an exoskeleton workspace optimization model on the basis of the principle of positive kinematics with the goal of improving the kinematic performance of the device and used the improved swarm intelligence algorithm to carry out simulation solutions to further optimize its structural parameters, so that it can be better applied to power strip operations and other power grid operation and maintenance tasks.

2. Operations and Maintenance Exoskeleton Power-assisted Model

2.1 Upper limb operation and maintenance exoskeleton Denavit–Hartenberg model

The physical structure of the upper limb operation and maintenance exoskeleton device is shown in Fig. 1. The structure body of the device includes a four-axis tandem upper limb wearable robot with a shoulder fixation belt, a waist fixation belt, an elbow and upper back fixation belt soft package, and a power source as the electric power drive. The left and right



Fig. 1. (Color online) Upper limb operation and maintenance exoskeleton physical diagram.

shoulder joints of the exoskeleton have independent absolute position sensors, touch force sensors, and other multisource sensors, and they also have a dual-feedback position sensor system, which can independently respond to the upward and downward positions of the left and right arms. The exoskeleton robot arm has two active degrees of freedom and two passive degrees of freedom, and both are rotational degrees of freedom.

In this study, the Denavit–Hartenberg (DH) method was used to model the linkage coordinate system of the O&M exoskeleton, i.e., the four parameters, namely, the linkage length a , linkage angle α , linkage deflection d , and joint angle θ , were used to determine the motion conversion relationship between the linkages of the mechanism.

The DH method establishes the coordinate system on the mechanism drive axis, and the joint at the end of the connecting rod is used as its solid coordinate system. Since the left and right arms of the exoskeleton device have the same structural layout, only the single arm was analyzed kinematically in this study. The DH coordinate system of the exoskeleton device was established as shown in Fig. 2, and its DH parameters are shown in Table 1.

According to the chain rule of coordinate system transformation, the transformation matrix from the $i - 1$ st coordinate system to the i -th coordinate system can be written as

$$\begin{aligned}
 {}^{i-1}T_i &= {}^{i-1}T_R R T_Q Q T_P P T_i \\
 &= Trans_{Z_{i-1}}(d_i) Rot_{Z_{i-1}}(\theta_i) Trans_{X_i}(a_i) Rot_{X_{i-1}}(\alpha_i),
 \end{aligned}
 \tag{1}$$

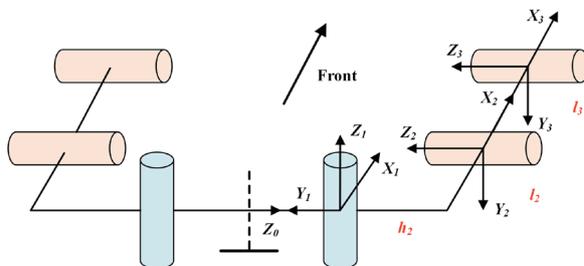


Table 1
O&M upper limb exoskeleton DH parameters.

Joint i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	θ_1
2	0	12	h_2	θ_2
3	0	13	0	θ_3

Fig. 2. (Color online) Operation and maintenance of the upper limb exoskeleton linkage coordinate system.

where *Rot* denotes the corresponding rotation matrix and *Trans* denotes the translation matrix. Then, the flush transformation matrix between the adjacent coordinate systems of this exoskeleton device is

$${}^0_1T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^1_2T = \begin{bmatrix} c_2 & -s_2 & 0 & l_2 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & h_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^2_3T = \begin{bmatrix} c_3 & -s_3 & 0 & l_3 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2)$$

The lengths of the connecting rods corresponding to h_2 , l_2 , and l_3 are shown in Fig. 2. s_i denotes $\sin\theta_i$ and c_i denotes $\cos\theta_i$ ($i = 1, 2, 3$). The total chi-square transformation matrix of the exoskeleton is obtained by multiplying the chi-square transformation matrices of each linkage coordinate system together:

$${}^0_3T = {}^0_1T {}^1_2T {}^2_3T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

where $P = (p_x, p_y, p_z)$ is the position vector of the end-effector of the exoskeleton device. The joint solution yields

$$\begin{aligned} p_x &= \cos\theta_1 l_2 + l_3(\cos\theta_1 \cos\theta_2 - \sin\theta_1 \sin\theta_2), \\ p_y &= -h_2, \\ p_z &= l_2 \sin\theta_1 + l_3(\cos\theta_1 \sin\theta_2 + \cos\theta_2 \sin\theta_1). \end{aligned} \quad (4)$$

The Robotics Toolbox for MATLAB 10.4 was used to build the visualization model of the O&M exoskeleton, thus verifying the correctness of its positive kinematic solution, in which the initial joint angle parameters θ_i ($i = 1, 2, 3$) are set to 0, as shown in Fig. 3, which is consistent with the established linkage model and verifies the correctness of the constructed coordinate system.

2.2 Exoskeleton workspace analysis

The spatial range that the workspace exoskeleton robot can control and operate includes the range that the robot can cover in three-dimensional space. From the above analysis, it can be seen that the workspace of this exoskeleton device is mainly affected by three parameters, h_2 , l_2 , and l_3 , as shown in Fig. 2.

The working space of the single arm of the O&M exoskeleton device was first solved using Matlab based on the Monte Carlo algorithm. The three parameters h_2 , l_2 , and l_3 were given initial

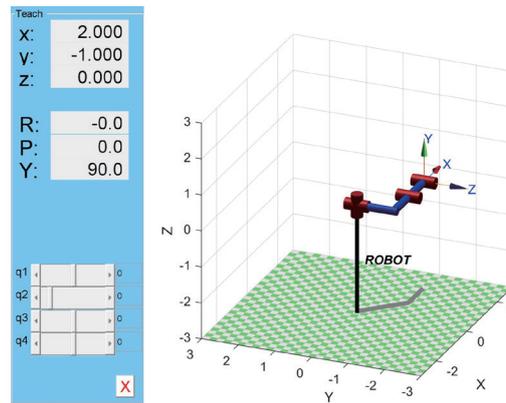


Fig. 3. (Color online) Visualization model of exoskeleton.

values; the human body has a limited range of joint rotation angles, such that $\theta_1 \in [0^\circ, 90^\circ]$, $\theta_2 \in [-90^\circ, 90^\circ]$, and $\theta_3 \in [0^\circ, 180^\circ]$. Taking the random series $N = 5000$, the working space, coverage space, and projection map of this exoskeleton device were obtained as shown in Figs. 4–5.

First, the exoskeleton workspace coverage was geometrically abstracted using the graphical method to analyze its workspace projection on the XoY plane. As shown in Fig. 6, this projection area can be divided into three different parts, where the A2 area represents the overlapping area of the workspace of the two arms.

A geometric analysis of Fig. 6 yields the following relationships:

$$R = \sqrt{h_2^2 + (l_2 + l_3)^2}, \quad (5)$$

$$r = \sqrt{h_2^2 + l_2^2}, \quad (6)$$

$$\alpha = \arctan \frac{l_1 + l_2}{h_2}, \quad (7)$$

$$\overline{o_1 c_3} = D_1 / \sin\left(\frac{\pi}{2} - \gamma\right), \quad (8)$$

$$\overline{c_2 c_3} = \sqrt{R^2 - D_1^2} - D_1 \cdot \tan\left(\frac{\pi}{2} - \gamma\right), \quad (9)$$

$$\cos \beta = \frac{\overline{o_1 c_3}^2 + \overline{o_1 c_2}^2 - \overline{c_2 c_3}^2}{2 \overline{o_1 c_3} \cdot \overline{o_1 c_2}}, \quad (10)$$

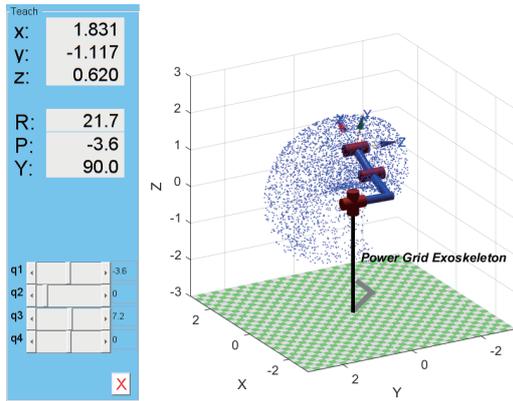


Fig. 4. (Color online) Main view of the workspace.

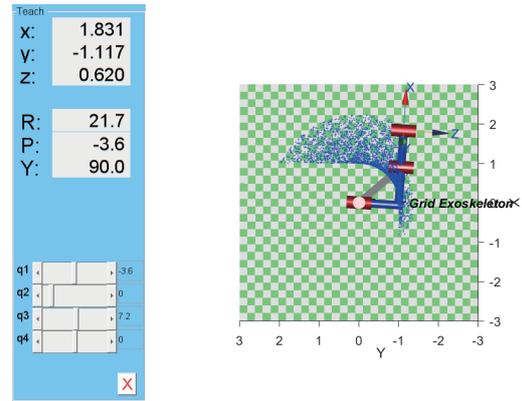


Fig. 5. (Color online) Workspace projection.

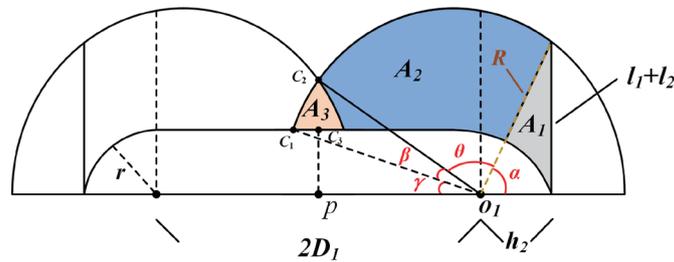


Fig. 6. (Color online) Workspace projection surface analysis.

$$\beta = \arccos \left\{ \frac{\left(\frac{D_1}{\sin \alpha} \right)^2 + R^2 - \left[\sqrt{R^2 - D_1^2} - D_1 \tan \left(\frac{\pi}{2} - \alpha \right) \right]^2}{2R \left(\frac{D_1}{\sin \alpha} \right)} \right\}, \quad (11)$$

$$\gamma = \arcsin \frac{r}{R}. \quad (12)$$

To simplify the calculations, the necessary approximations were made for the calculation of the coverage of each region, and they are

$$S(A_1) = \frac{1}{2} [h_2(l_2 + l_1) - r^2 \alpha], \quad (13)$$

$$S(A_2) = \frac{1}{2} (R^2 \theta - r^2 \theta). \quad (14)$$

The projected area of the exoskeleton bipartite overlap region can be calculated as

$$S_{\Delta o_1 c_2 c_3} = \frac{1}{2} D_1 \left[\sqrt{R^2 - D_1^2} - D_1 \tan \left(\frac{\pi}{2} - \alpha \right) \right], \quad (15)$$

$$S_{\Delta o_1 c_1 c_2} = \frac{1}{2} \beta R, \quad (16)$$

$$S(A_3) = 2[S_{\Delta o_1 c_1 c_2} - S_{\Delta o_1 c_2 c_3}]. \quad (17)$$

To optimize the workspace of the exoskeleton device, optimal solutions need to be sought for each rod length to maximize its range of boundary curves. In addition, the overlap of the working space range of both arms needs to be minimized when considering the nonessential working area. Therefore, we can define the effective workspace of the exoskeleton power-assisted model as the following objective function:

$$\max S = \omega_1 [S(A_1) + S(A_2)] - \omega_2 S(A_3). \quad (18)$$

Here, ω_1 and ω_2 are the introduced linear weights that satisfy the relation $\omega_1 + \omega_2 = 1$.

Considering the actual structure and length ratio of each joint in the human body, the length of each linkage of the exoskeleton should be limited as

$$\begin{cases} 0 \leq h_2 \leq 200 \text{ (mm)} \\ 0 \leq l_2 \leq 400 \text{ (mm)} \\ 0 \leq l_3 \leq 400 \text{ (mm)} \end{cases}. \quad (19)$$

Owing to the limited rotation angle of human upper limb joints, the joint rotation angle was limited to

$$\begin{cases} \theta_1 \in [0^\circ, 90^\circ] \\ \theta_2 \in [-90^\circ, 90^\circ] \\ \theta_3 \in [0^\circ, 180^\circ] \end{cases}. \quad (20)$$

To reduce the burden on the wearer, the self-weight of each part of the system should be as small as possible, so the total length of each linkage should be limited to

$$0 \leq h_2 + l_2 + l_3 \leq 800 \text{ (mm)}. \quad (21)$$

3. Tuna Swarm Optimization

The TSO algorithm is a new population intelligence optimization algorithm proposed by Xie *et al.* in 2021,⁽¹⁵⁾ which finds the optimal solution of a problem by simulating the prey hunting behavior of a tuna swarm.

3.1 Spiral foraging

The tuna school surrounds and pursues prey fish in a spiral formation, while the tuna will exchange information with each other. At the beginning of the algorithm search, tuna individuals generate a random position in the search space as the reference point for their own spiral update, and as the number of iterations increases, the reference point gradually transitions from the random position to the optimal tuna individual, thus completing the transition of the algorithm from global search to local optimization search. The formula of individual i position update based on the spiral foraging strategy is shown in Eq. (22).

$$X_i(t+1) = \begin{cases} \left\{ \begin{array}{l} \alpha_1 \cdot (X_{rand} + \beta \cdot |X_{rand} - X_i(t)|) + \alpha_2 \cdot X_i(t), \quad i=1 \\ \alpha_1 \cdot (X_{rand} + \beta \cdot |X_{rand} - X_i(t)|) + \alpha_2 \cdot X_{i-1}(t), \quad i=2,3,\dots,N \end{array} \right. & rand < \frac{t}{T} \\ \left\{ \begin{array}{l} \alpha_1 \cdot (X_{best} + \beta \cdot |X_{best} - X_i(t)|) + \alpha_2 \cdot X_i(t), \quad i=1 \\ \alpha_1 \cdot (X_{best} + \beta \cdot |X_{best} - X_i(t)|) + \alpha_2 \cdot X_{i-1}(t), \quad i=2,3,\dots,N \end{array} \right. & rand \geq \frac{t}{T} \end{cases} \quad (22)$$

$$\alpha_1 = a + (1-a) \cdot \frac{t}{T} \quad (23)$$

$$\alpha_2 = (1-a) - (1-a) \cdot \frac{t}{T} \quad (24)$$

$$\beta = e^{b \cdot l} \cdot \cos(2\pi b) \quad (25)$$

$$l = \exp(3 \cos\{[(T+1)/t] - 1\} \pi) \quad (26)$$

Here, t is the current number of iterations, T is the maximum number of iterations, α_1 and α_2 are the weight coefficients of individual movement trends, a is a constant indicating the degree of following in the initial phase, and b is a random number between 0 and 1.

3.2 Parabolic foraging

In addition to spiral foraging, tuna schools also have a parabolic foraging strategy, and both foraging methods have the same probability of selection. Cooperative foraging in a parabolic

formation is achieved when the tuna adopt a parabolic formation with the prey fish as a reference and also search for prey in the surrounding area. The equation for updating the position of individual i based on the parabolic foraging strategy is shown in Eq. (27).

$$X_i(t+1) = \begin{cases} X_{best} + rand \cdot (X_{best} - X_i(t)) + r \cdot p^2 \cdot (X_{best} - X_i(t)), & rand < 0.5 \\ r \cdot p^2 \cdot X_i(t), & rand \geq 0.5 \end{cases} \quad (27)$$

$$p = \left(1 - \frac{t}{T}\right)^{\frac{t}{T}}. \quad (28)$$

Here, r is a random number with the value of -1 or 1 .

4. Improved Tuna Optimization Algorithm Based on Population Hierarchy Strategy, Elite Opposition-based Learning, and Genetic Variation

In the traditional tuna optimization algorithm, the initial population is generated by random initialization, and the position update of tuna individual i is affected by the global optimal solution and individual $i + 1$, and the global search ability of the algorithm will be weakened with the increase in the number of iterations, which leads to the problems of the uneven distribution of the initial population, the weak global search ability of the algorithm, and easy to fall into the local optimum. To address these problems, we introduced Tent chaos mapping, population hierarchy, elite opposition-based learning, and a genetic variation operator to improve the algorithm in order to increase the initial population diversity, enhance the global search ability of the algorithm, and enable it to quickly jump out of the local optimum.

4.1 Tent chaos mapping

The traditional tuna algorithm uses random initialization in the search range to generate the initial population, which easily leads to low individual diversity and is prone to fall into local optimum in the iterative optimization process. In this study, the population is initialized on the basis of the Tent chaos mapping mechanism to expand the population diversity in order to obtain high-quality initial feasible solutions.⁽¹⁶⁾ The population initialization based on Tent chaos mapping is defined as

$$X_{i+1} = \begin{cases} X_i / \alpha, & X_i \in [0, \alpha] \\ (1 - X_i) / (1 - \alpha), & X_i \in [\alpha, 1]. \end{cases} \quad (29)$$

4.2 Population hierarchy strategy

At the late stage of the traditional tuna optimization iteration, each individual spirally forages to follow the global optimal solution to a considerable extent, leading to the weakening of the

global search ability at the late stage of the algorithm and becoming prone to stagnation. To address this problem, we introduced a population hierarchy mechanism, divided the tuna population into elite and inferior populations according to the size of fitness, and used different perturbation methods for different populations. According to the feature that elite individuals contain more effective information of fitness, the elite reverse learning strategy is used in the elite population to expand the development comprehensiveness of dominant individuals,⁽¹⁷⁾ and the population diversity is increased by constructing the reverse individuals of current elite individuals and retaining the individuals with greater fitness among them as the new-generation population individuals, where the reverse individuals are constructed as shown in Eq. (30).

$$X_i^{Op} = \delta(lb + ub) - X_i \tag{30}$$

For the disadvantaged population, the crossover variation operator in GA is used to perturb it,⁽¹⁸⁾ so that the algorithm is more likely to jump out of the local optimum and strengthen the search ability of the algorithm later. If each individual has the same probability of choosing crossover, the individual fitness values after the operation are compared with those before the operation based on the greedy algorithm idea, and the individuals with greater fitness are retained to join the new population generation. The schematic diagram of the population hierarchy mechanism is shown in Fig. 7.

4.3 Flow chart of improved TSO algorithm

The overall flow chart of the improved TSO algorithm (ITSO) based on population hierarchy, elite backward learning, and genetic variation is shown in Fig. 8.

5. Simulation Analysis

In this study, power grid operation and maintenance were used as the operation background, and the parameters and data used in the simulation experiments were based on the actual structural parameters of the exoskeleton and the actual business data of the project.

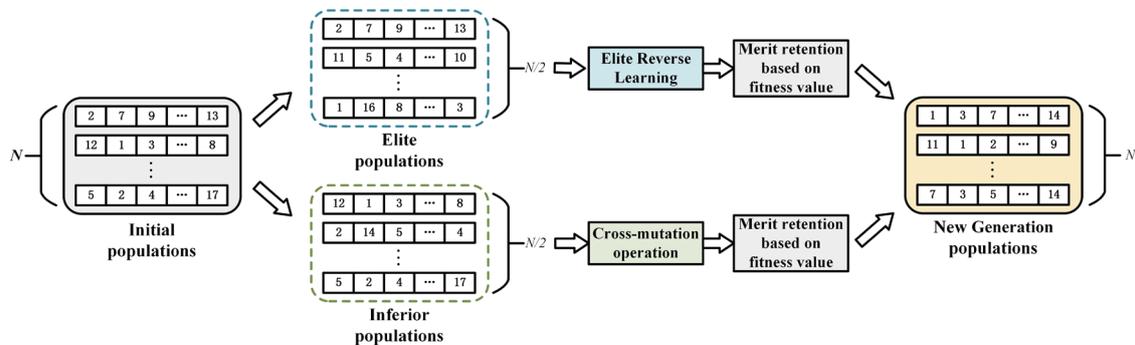


Fig. 7. (Color online) Schematic diagram of the population hierarchy strategy.

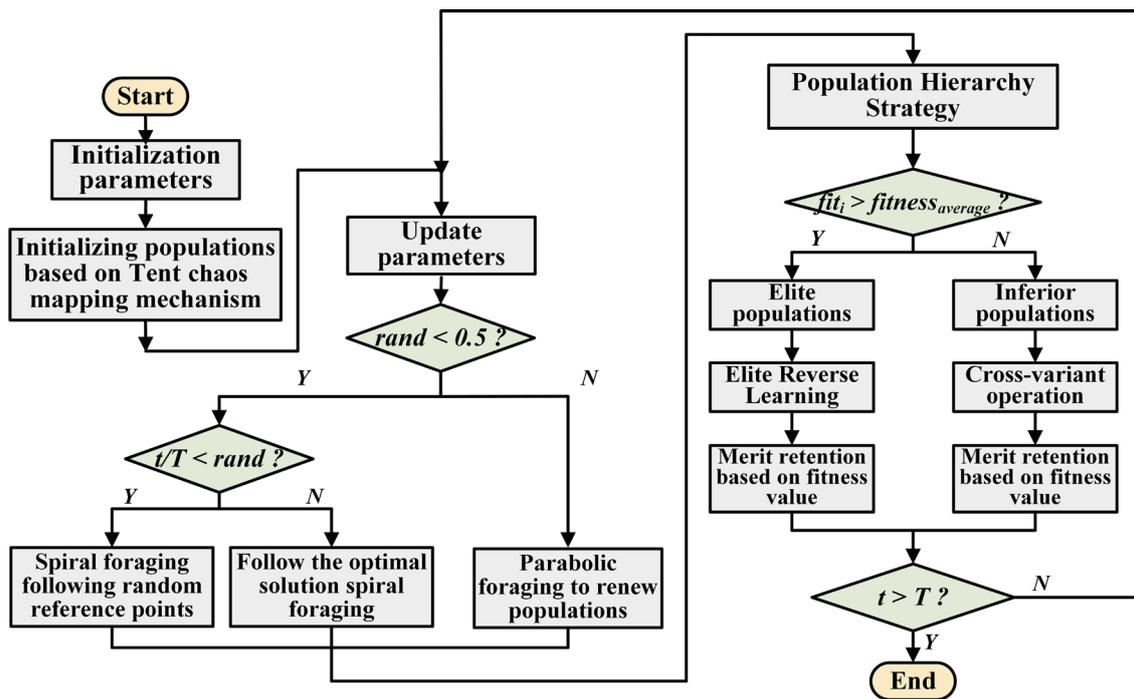


Fig. 8. (Color online) Overall flow chart of ITSO.

To verify the effectiveness and robustness of the ITSO designed in this study, PSO, GA, gray wolf algorithm (GWO), and beluga whale optimization algorithm (BWO) were also selected to solve the model built in this study.

On the basis of the actual human kinematic parameters, the simulation solution was obtained using Matlab software, where the algorithm parameters were set as follows: constant $a = 0.7$ in ITSO; particle learning rate $c1 = c2 = 1.5$ in PSO; crossover rate $pc = 0.8$ and variation rate $pm = 0.1$ in GA. Moreover, to ensure the fairness of the simulation comparison experiment, the maximum number of iterations, T , for all algorithms was set to 1000, and the population size N was set to 100.

The algorithms were run independently under four groups of conditions with weight ratios $\omega_1:\omega_2$ of 6:4, 5:5, 4:6, and 3:7.

In Figs. 9(a)–9(d), the iteration curves of each algorithm for a single run under four sets of weight ratio conditions are given, indicating that the ITSO designed in this paper outperforms similar comparative algorithms in terms of convergence ability and solution efficiency.

The operating metrics of each algorithm under each weight value condition are shown in Tables 2–5. It is seen that under the $\omega_1:\omega_2 = 6:4$ condition, the objective function values of ITSO are improved by 0.14 and 0.08% as compared with those of PSO and GA, respectively. Under the $\omega_1:\omega_2 = 5:5$ condition, they are improved by 0.11 and 0.08% as compared with those of PSO and GA, respectively. Under the $\omega_1:\omega_2 = 4:6$ condition, they are improved by 0.12, 0.20, and 0.17% as compared with those of PSO, GA, and GWO, respectively. Under the $\omega_1:\omega_2 = 3:7$ condition, they are improved by 0.11, 0.28, 0.15, and 0.19% as compared with those of PSO, GA, GWO, and

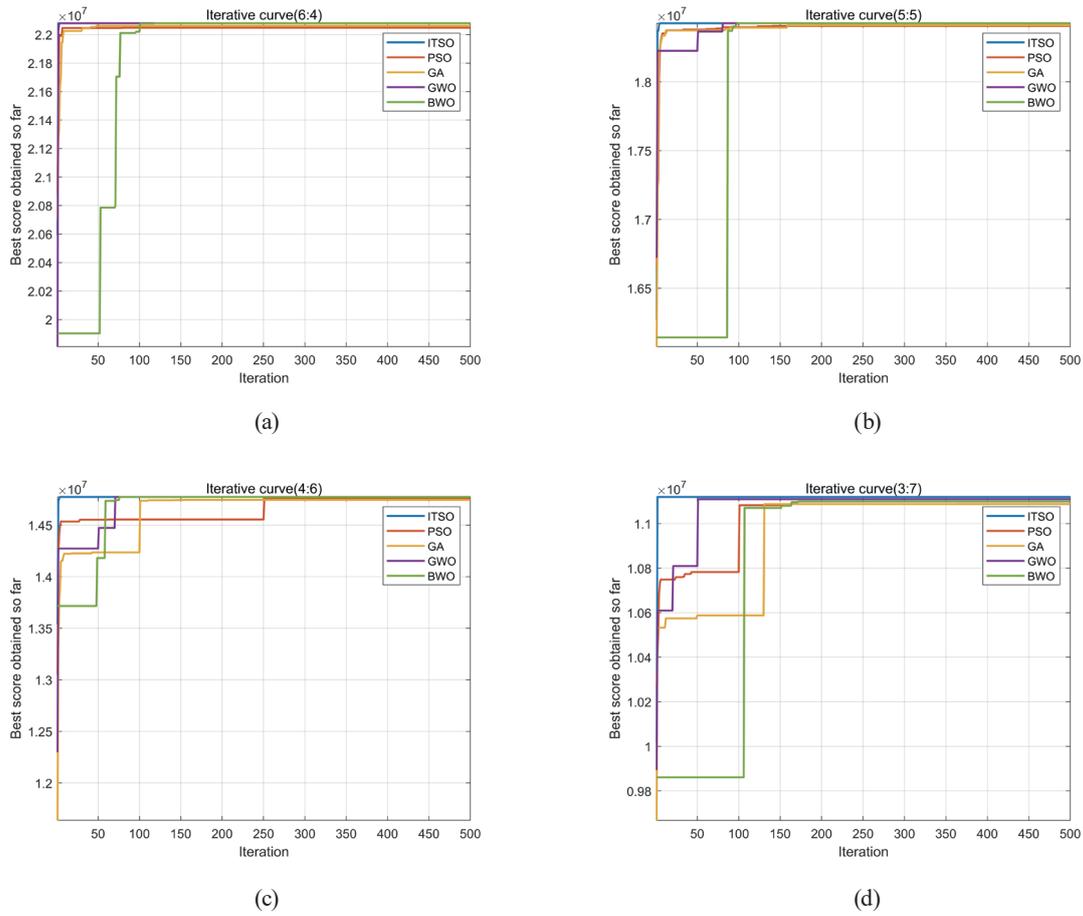


Fig. 9. (Color online) Algorithm adaptation iteration curve.

Table 2
Operating index of each algorithm (6:4).

Algorithm	h_2 (mm)	l_2 (mm)	l_3 (mm)	Objective function value
ITSO	200	400	400	2.2079×10^7
PSO	114.1805	399.8488	399.9801	2.2048×10^7
GA	163.7855	399.7948	399.9815	2.2061×10^7
GWO	200	400	400	2.2079×10^7
BWO	200	400	400	2.2079×10^7
ITSO	200	400	400	2.2079×10^7

Table 3
Operating index of each algorithm (5:5).

Algorithm	h_2 (mm)	l_2 (mm)	l_3 (mm)	Objective function value
ITSO	200	400	400	1.8426×10^7
PSO	138.3371	399.9665	399.9153	1.8406×10^7
GA	155.2759	399.9858	399.9206	1.8411×10^7
GWO	200	400	400	1.8426×10^7
BWO	200	400	400	1.8426×10^7
ITSO	200	400	400	1.8426×10^7

Table 4
Operating index of each algorithm (4:6).

Algorithm	h_2 (mm)	l_2 (mm)	l_3 (mm)	Objective function value
ITSO	200	400	400	1.4773×10^7
PSO	125.8053	399.8927	399.9515	1.4755×10^7
GA	59.8354	399.9785	399.9834	1.4744×10^7
GWO	177.9221	399.0779	400	1.4747×10^7
BWO	200	400	400	1.4773×10^7
ITSO	200	400	400	1.4773×10^7

Table 5
Operating index of each algorithm (3:7).

Algorithm	h_2 (mm)	l_2 (mm)	l_3 (mm)	Objective function value
ITSO	200	400	400	1.1120×10^7
PSO	126.6630	399.9752	399.9422	1.1108×10^7
GA	7.6704	399.8626	399.9771	1.1089×10^7
GWO	130.4746	399.5354	400	1.1103×10^7
BWO	137.3702	399.2698	400	1.1099×10^7
ITSO	200	400	400	1.1120×10^7

BWO, respectively. The results show that when the optimal parameter solutions of the model are $[h_2, l_2, l_3] = [200, 400, 400]$ for all three sets of weight ratio $\omega_1:\omega_2$ conditions, the maximum workspace projection area obtained is $7.3383 \times 10^7 \text{ mm}^2$.

It can be seen from the above results that the performance indexes of ITSO are better in the whole process of iterative search, and its solution accuracy is significantly higher than that of other similar optimization algorithms, thus verifying the feasibility and applicability of the algorithm.

6. Conclusions

In this study, we took the power-assisted model of operation and maintenance upper limb exoskeleton as the research object, and we aimed to investigate the feasibility and optimization methods of applying the power exoskeleton device to power grid operation and maintenance. From the actual structure and constraints of the four-degree-of-freedom industrial upper limb exoskeleton, we constructed its linkage coordinate system model on the basis of the principle of positive kinematics and combined the graphical solution method to construct an optimization model of the operation and maintenance exoskeleton workspace. An improved tuna algorithm based on population hierarchy, elite backward learning, and genetic variation was designed for the solution of the proposed model. The effectiveness and robustness of the algorithm designed in this study were verified by comparing it with PSO, GA, GWO, and other intelligent algorithms horizontally in simulation experiments. The results of this study provide theoretical and practical bases for the application of power exoskeleton devices in the field of power grid operation and maintenance, and they also provide new ideas and methods for the optimal design of exoskeleton devices.

Acknowledgments

This work was supported by the science and technology project of Yunnan Power Grid Co., Ltd., which provided funding under project number YNKJXM20220216.

References

- 1 A. Baba: J. King Saud Univ. Comput. Inf. Sci. **34** (2022) 2252. <https://doi.org/10.1016/j.jksuci.2020.07.009>
- 2 G. D. Lorenzo, R. Tartaglia, A. Prota, and R. Landolfo: Eng. Struct. A **275** (2023) 115252. <https://doi.org/10.1016/j.engstruct.2022.115252>
- 3 Y. Ma, Q. X. Wu, R. F. Li, H. Liu, and L. J. Zhang: Manned Spaceflight **28** (2022) 223. <https://doi.org/10.3969/j.issn.1674-5825.2022.02.012>
- 4 J. Chen, Y. P. Huang, X. B. Guo, S. T. Zhou, and L. F. Jia: Measurement **159** (2020) 107765. <https://doi.org/10.1016/j.measurement.2020.107765>
- 5 Y. He, F. Li, J. K. Li, J. H. Liu, and X. Y. Wu: Biomed. Signal Process. Control **74** (2022) 103477. <https://doi.org/10.1016/j.bspc.2021.103477>
- 6 L. Gao, C. J. Ma, N. Zhou, and L. J. Zhao: Comput. Ind. Eng. **171** (2022) 108427. <https://doi.org/10.1016/j.cie.2022.108427>
- 7 M. N. Castro, J. Rasmussen, M. S. Andersen, and S. Bai: Mech. Mach. Theory **132** (2019) 264. <https://doi.org/10.1016/j.mechmachtheory.2018.11.007>
- 8 T. K. Xiao, Y. Zhang, L. G. Qiang, L. Li, and C. Guo: J. King Saud Univ. Eng. Sci. **34** (2022) 10. <https://doi.org/10.1016/j.jksues.2021.11.010>
- 9 Y. Yang, X. X. Xiao, Z. J. Nan, N. Liu, X. M. Li, and Y. Peng: J. Mech. Eng. **55** (2019) 105. <https://doi.org/10.3901/JME.2019.11.105>
- 10 J. Wang, J. H. Liu, G. W. Zhang, and S. J. Guo: ISA Transactions **123** (2022) 87. <https://doi.org/10.1016/j.isatra.2021.05.039>
- 11 L. B. Zhang, J. C. Wang, J. L. Chen, K. Chen, B. Y. Lin, and F. Xu: Adv. Eng. Software **135** (2019) 102684. <https://doi.org/10.1016/j.advengsoft.2019.05.006>
- 12 D. B. M. M. Fontes, S. M. Homayouni, and J. F. Gonçalves: Eur. J. Oper. Res. **306** (2023) 1140. <https://doi.org/10.1016/j.ejor.2022.09.006>
- 13 Ö. Ekrem and B. Aksoy: Eng. Appl. Artif. Intell. **122** (2023) 106099. <https://doi.org/10.1016/j.engappai.2023.106099>
- 14 A. Jaafari, M. Panahi, D. Mafi-Gholami, O. Rahmati, H. Shahabi, A. Shirzadi, S. Lee, D. T. Bui, and B. Pradhan: Appl. Soft Comput. **116** (2022) 108254. <https://doi.org/10.1016/j.asoc.2021.108254>
- 15 L. Xie, T. Han, H. Zhou, Z. R. Zhang, B. Han, and A. Tang: Comput. Intell. Neurosci. **2021** (2021) 22. <https://doi.org/10.1155/2021/9210050>
- 16 M. A. El-Shorbagy, A. A. Mousa, and S. M. Nasr: Chaos, Solitons and Fractals **85** (2016) 8. <https://doi.org/10.1016/j.chaos.2016.01.007>
- 17 J. J. Jiao, J. L. Cheng, Y. B. Liu, H. Y. Yang, D. R. Tan, and P. Cheng: Comput. Geosci. **174** (2023) 105334. <https://doi.org/10.1016/j.cageo.2023.105334>
- 18 R. Norat, A. S. Wu, and X. L. Liu: Expert Syst. Appl. **218** (2023) 119529. <https://doi.org/10.1016/j.eswa.2023.119529>

About the Authors



Mingxian Liu was born in 1987 in Qujing, Yunnan Province. He graduated from Kunming University of Technology with a bachelor's degree in electrical engineering in 2017. His main research interests are distribution network operation and inspection, power strip operation, and artificial intelligence.

(liumingxian@lj.csg.cn)



Xinbo Zhou was born in 2003 in Anqing, Anhui Province. He is currently pursuing his bachelor's degree with the Faculty of Information Engineering and Automation, Kunming University of Science and Technology. His research interests cover intelligent algorithms and artificial intelligence.
(zhouxinbo@stu.kust.edu.cn)



Jibiao Li was born in 1990 in Qujing, Yunnan Province. He graduated from Chongqing University with a bachelor's degree in electrical engineering in 2017. His main research interests are distribution lines, distribution line energized operation, and distribution automation.(lijibiao@lj.yn.csg.cn)