Sensors and Materials, Vol. 36, No. 1 (2024) 305–322
MYU Tokyo

S & M 3520

Power-assisted Optimization Model of Heterogeneous Sensor Exoskeleton Devices Based on Swarm Intelligence Algorithm and Dynamics Optimization

Mingxian Liu,[1] Xinbo Zhou,[2,4*] and Jing Bao[3,4]

[1]Lijiang Power Supply Bureau, Yunnan Power Grid Co., Ltd., Lijiang 674100, China
[2]Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China
[3]Faculty of Civil Aviation and Aeronautics, Kunming University of Science and Technology, Kunming 650500, China
[4]Kunming Holy Intelligent Technology Co., Ltd., Kunming 650500, China

(Received July 11, 2023; accepted January 4, 2024)

Keywords: heterogeneous sensor exoskeleton, power-assisted model, workspace optimization, improved Harris Hawks algorithm, sine cosine operator

With the ongoing integration of bionic technology and mechatronics, exoskeleton devices are finding applications in industry, healthcare, and logistics. This study is centered on enhancing the performance of heterogeneous sensor exoskeleton devices by presenting a six-degree-of-freedom upper limb exoskeleton robot model based on the Denavit–Hartenberg (MDH) approach. The model's accuracy is verified using MATLAB. We construct a multi-objective optimization model that prioritizes workspace expansion. To realize this model, we propose an improved Harris Hawks algorithm (SCA-HHO) based on the sine cosine algorithm. The algorithm's effectiveness is compared with popular swarm intelligence methods (PSO, AOA, WOA) through cross-sectional simulations. SCA-HHO achieves average improvements of 6.30, 1.48, and 0.88% in objective function values compared with the swarm intelligence algorithms, respectively. This indicates SCA-HHO's superior suitability for solving the model proposed in this paper.

1. Introduction

An exoskeleton device is a wearable electromechanical intelligent robot. Compared with military and medical rehabilitation fields, in industry, exoskeletons are still in the early stage of research, development, and commercialization. Heterogeneous sensor exoskeletons, with their sensors, power drivers, and rigid and flexible mechanical structures, can provide assistance during physical labor, increase the strength and endurance of the operator, and assist the wearer in completing specific operational actions, thus improving the productivity or operational efficiency of the user. Linnenberg and Weidner[(1)] studied four overhead exoskeletons to investigate the stresses that occur within the human-machine interface (HMI) of the arm during

*Corresponding author: e-mail: zhouxinbo@stu.kust.edu.cn
https://doi.org/10.18494/SAM4649

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

overhead work and the effect of the HMI on the neurovascular supply to the upper extremity of the user. Antwi-Afari *et al.*[2] tested the effect of exoskeletal systems on the biomechanics and subjective response of the spine during repetitive lifting tasks performed by construction personnel to verify their utility and reliability. In recent years, with the continuous development of industrial automation, industrial exoskeleton devices have received widespread industry attention for their portability and their ability to effectively enhance the efficiency and capacity of operators.[3]

To achieve effective assistance, the self-weight of each part of the exoskeleton system should be as small as possible while the power of the drive system should be sufficiently large. The choice of drive degrees of freedom and the range of joint degrees of freedom also affect the actual operational capability of the exoskeleton, so lightness, power-assisted capability, and effective operational space should be the main performance optimization directions for industrial exoskeleton devices.[4] Because of constraints such as the size, structure, and performance of industrial exoskeleton devices, such optimization problems can be called dynamic optimization problems.

The exoskeleton workspace refers to the working area that can be reached by the end of the exoskeleton and is one of the criteria for judging its performance. At present, research on exoskeleton dynamics models has focused on optimizing the direction of driving joint kinematic parameters, control strategies, and trajectory planning.[5–8] For example, in Ref. 9, an adaptive-interactive-torque-based assist-on-demand (AITAAN) control method for a lower limb rehabilitation exoskeleton was proposed to provide effective support for the gait rehabilitation of patients. As reported in Ref. 10, the control strategy of the knee plunger cylinder was optimized to solve the knee joint jitter that may occur during the squatting process of a weight-assisted exoskeleton system. However, most of the existing studies on workspace research are focused on the robotic arm, and there is less involvement in the analysis of the workspace of industrial exoskeleton devices. The three objective functions of trajectory minimization, power consumption reduction, and multicriteria optimization were used to optimize the work cycle of the anthropomorphic robotic arm in the workspace.[11] The robotic arm work area of a two-arm rescue robot has also been simulated to achieve trajectory planning and control of both arms.[12]

The Denavt–Hartenberg (DH) and modified Denavit–Hartenberg (MDH) methods are used conventionally in the kinematic analysis of models such as those of exoskeletons and robotic arms, i.e., the kinematic equations of a tandem robot composed of connecting rods are obtained by coordinate transformation. In addition, for the optimization or simulation analysis of exoskeleton models, software programs such as MATLAB and Adams[13] are widely used in dynamics and kinematics studies. The application of advanced swarm intelligence algorithms such as particle swarm algorithms,[14,15] genetic algorithms,[16,17] the whale optimization algorithm,[18] and nondominated ranking genetic algorithms (NSGA-II)[19] in industrial and intelligent manufacturing[20] has also become one of the popular research directions for domestic and foreign scholars.

In summary, research on exoskeleton workspaces remains relatively rare in existing domestic and international studies. Moreover, the analysis and optimization of power-assisted workspaces for industrial exoskeletons are even more unexplored. The primary challenge of our task is to

establish a mathematical model for the exoskeleton device. Such a model must take into account various factors including mechanical structure and kinematic principles. It is also necessary to consider multiple factors in order to determine the optimization target and establish the optimization objectives. Simultaneously, the design of an efficient optimization algorithm is essential to solving the proposed model.

To address the above issue, we use the MDH method to establish an exoskeleton tandem linkage model based on actual project business data following the positive kinematic principle. Furthermore, in reference to the actual structural characteristics and performance constraints of the industrial exoskeleton, the structural parameters of the exoskeleton are taken as the optimization variables, and the exoskeleton power-assisted optimization model is established on the basis of analytical and graphical methods. Meanwhile, for the problem that the traditional Harris Hawks algorithm tends to fall into iterative stagnation at a later stage, an improved Harris Hawks algorithm (SCA-HHO) based on the positive cosine algorithm is designed for simulation, so that an optimal structural parameter scheme of the industrial exoskeleton device can be obtained.

2. Analysis of Power-assisted Models for Industrial Exoskeleton Devices

2.1 Establishment and analysis of MDH models for industrial upper limb exoskeleton

The model of the upper limb exoskeleton device is shown in Fig. 1. The device body is a 6-axis tandem wearable robotic device with two powered degrees of freedom with servo motors and four underdriven degrees of freedom; all six degrees of freedom are rotational degrees of freedom.

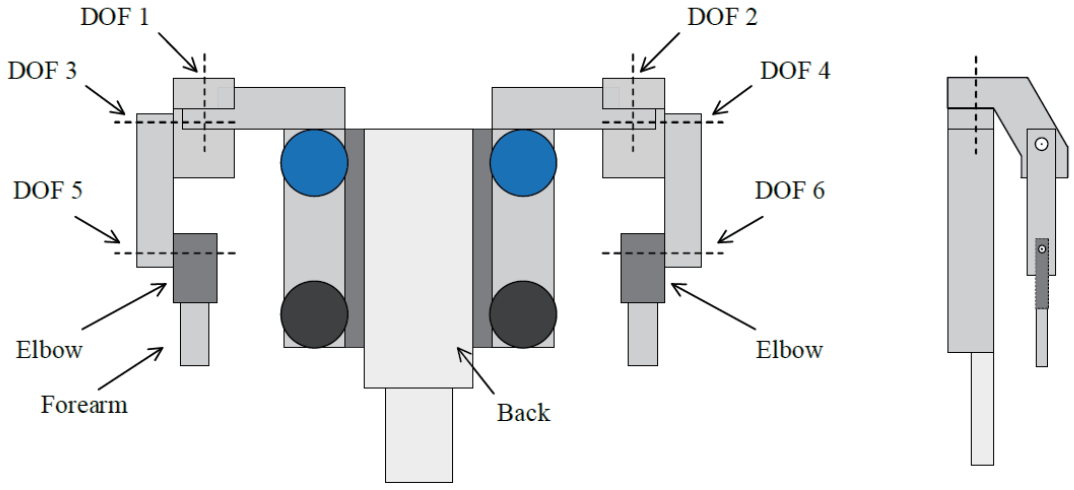


Fig. 1. (Color online) Model of upper limb exoskeleton. (a) Back view and (b) side view.

In this paper, the positive kinematics method is used for calculation. The MDH method is a modified version of the DH linkage coordinate system method proposed by Craig, which uses the front-end joint of the linkage as the fixation coordinate system.

Since the left and right arms of the exoskeleton device are symmetrical, only one side of the robotic arm needs to be analyzed. The MDH method is used to establish its linkage coordinate system, as shown in Fig. 2, in which the positions of each linkage coordinate system are marked, where the X1 positive direction points out-of-plane and Y2 and Y3 point in-plane. The corresponding DH parameters obtained are shown in Table 1.

According to the chain rule of coordinate system transformation, the transformation matrix from the i-1st coordinate system to the i-th coordinate system (i = 1, 2, 3) can be written as follows.

$$\begin{aligned} {}^{i-1}_{i}T &= {}^{i-1}_{R}T\,{}^{R}_{Q}T\,{}^{Q}_{P}T\,{}^{P}_{i}T \\ &= Rot_{X_{i-1}}(\alpha_{i-1})Trans_{X_{i-1}}(a_{i-1})Rot_{Z_i}(\theta_i)Trans_{Z_i}(d_i) \end{aligned} \tag{1}$$

Then the homogeneous transformation matrix between adjacent coordinate systems of this exoskeleton device is

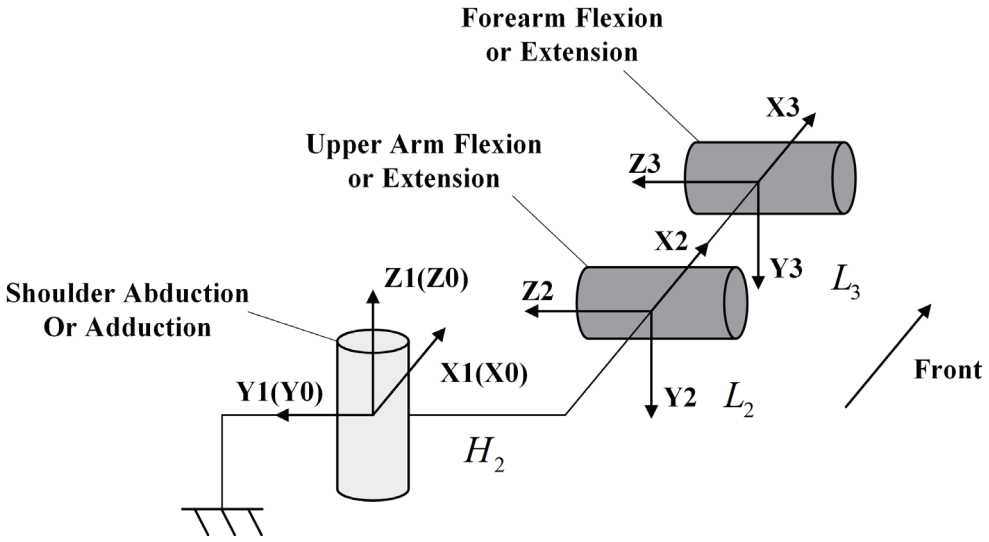


Fig. 2. DH coordinate system of industrial upper limb exoskeleton.

Table 1
DH parameters of industrial upper limb exoskeleton.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	$-\pi/2$	L_2	$-H_2$	θ_2
3	0	L_3	0	θ_3

$$ {}^{0}_{1}T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{1}_{2}T = \begin{bmatrix} c_2 & -s_2 & 0 & H_2 * c_2 \\ 0 & 0 & 1 & L_2 \\ -s_2 & -c_2 & 0 & -H_2 * s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{2}_{3}T = \begin{bmatrix} c_3 & -s_3 & 0 & 0 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & L_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2} $$

where s_i represents $\sin\theta_i$ and c_i represents $\cos\theta_i$ (i = 1, 2, 3). Thus, the total homogeneous transformation matrix of the device is

$$ {}^{0}_{3}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T = \begin{bmatrix} n_{11} & n_{12} & n_{13} & p_x \\ n_{21} & n_{22} & n_{23} & p_y \\ n_{31} & n_{32} & n_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3} $$

where $P = (p_x, p_y, p_z)$ is the position vector of the end of the upper limb exoskeleton device. The simultaneous solution leads to the following.

$$ \begin{aligned} p_x &= H_2 \cos\theta_1 \cos\theta_2 - L_3 \sin\theta_1 - L_2 \sin\theta_1 \\ p_y &= L_2 \cos\theta_1 + L_3 \cos\theta_1 + H_2 \cos\theta_2 \sin\theta_1 \\ p_z &= -H_2 \sin\theta_2 \end{aligned} \tag{4} $$

Robotics Toolbox for MATLAB 10.4 was used to establish a visual model of the upper extremity exoskeleton, in which the initial joint angle parameter θ_i was set as (0, 0, 0). As shown in Fig. 3, the upper extremity exoskeleton was in a forward flat lifting state, which was consistent with the established linkage coordinate system, verifying the correctness of the established model.

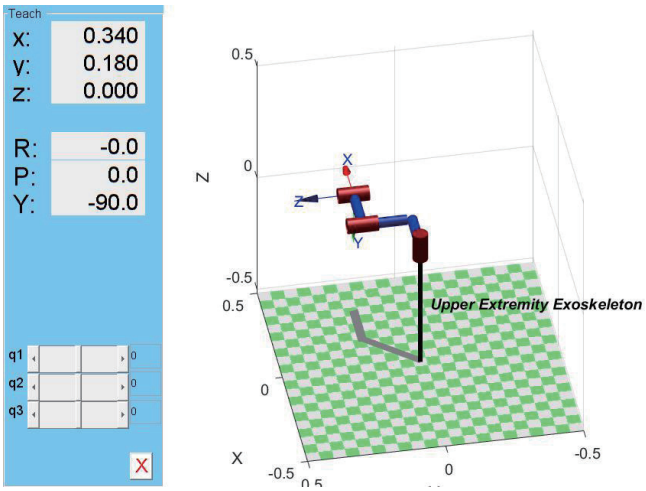


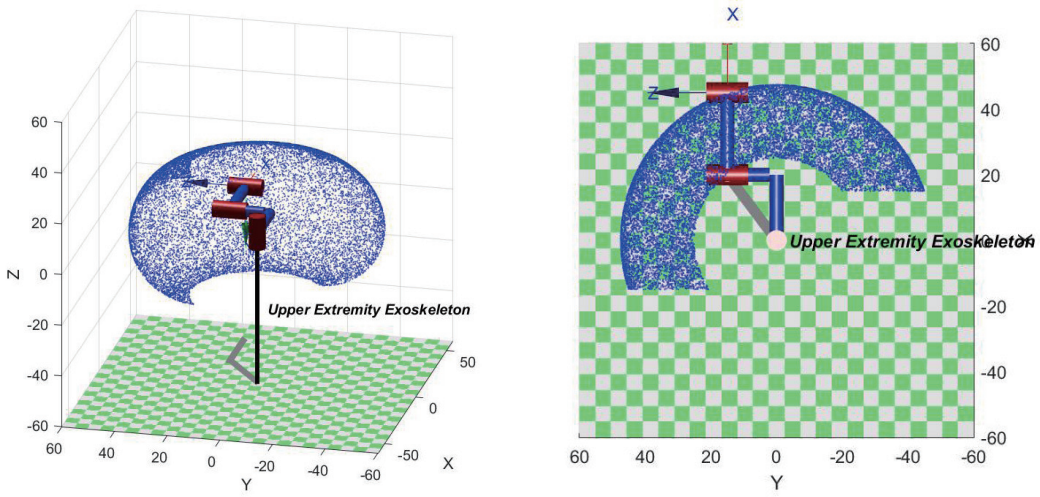
Fig. 3. (Color online) Visualization of exoskeleton model.

2.2 Workspace analysis of exoskeleton device

The workspace refers to the set of locations that the end of the exoskeleton can reach. According to the results of the above analysis, as shown in Fig. 2, the workspace of the exoskeleton device is mainly affected by three parameters: H_2, L_2, and L_3. Therefore, the following modeling analysis of the workspace is carried out by simulation and a graphical method, and the above three variables are used as decision variables to establish a power-assisted optimization model in order to obtain the maximum effective workspace for the exoskeleton device.

2.2.1 Workspace analysis

First, the workspace of the single arm of the industrial upper limb exoskeleton device is established by the Monte Carlo algorithm based on MATLAB. The Monte Carlo method is a method of solving mathematical problems by random sampling and yields a sample that well reflects the area accessible at the end of the device. The initial values of parameters H_2, L_2, and L_3 are respectively set as 150, 200, and 250 mm. Since the range of the joint rotation angle is limited, let $\theta_1 \in [-90°, 90°]$, $\theta_2 \in [-90°, 90°]$, and $\theta_3 \in [0°, 180°]$. Take the random number sequence N = 20000 to obtain the workspace of the exoskeleton device and its projection on each coordinate plane, as shown in Fig. 4.



(a) (b)

Fig. 4. (Color online) (a) Main view of the workspace and (b) projection of the workspace on the *XY* plane.

2.2.2 Optimization model of exoskeleton power assist

Second, the graphic method is used to analyze the projection of the exoskeleton workspace on the XoY plane, as shown in Fig. 5. The projection area is divided into three parts, among which A_3 is the overlapping part of the two arm workspaces. To optimize the workspace of the exoskeleton device, the optimal parameters of each rod length should be determined to maximize the range of the workspace boundary curve l_2l_3. Also, considering the nonessential working area, the overlapping part of the working space range of the two arms should be minimized, so the objective function of the effective working space of the exoskeleton power-assisted model can be defined as the following multi-objective optimization function.

$$\max F(H_2, L_2, L_3) = F[f_1(H_2, L_2, L_3), f_2(H_2, L_2, L_3)] \tag{5}$$

$$f_1(H_2, L_2, L_3) = 2[S(A_1) + S(A_2)] \tag{6}$$

$$f_2(H_2, L_2, L_3) = S(A_3) \tag{7}$$

This mathematical model is a multi-objective optimization model, where S is the distance between the two shoulders of the upper limb exoskeleton, thus

$$R = \sqrt{H_2{}^2 + (L_2 + L_3)^2}, \tag{8}$$

$$r = \sqrt{H_2^2 + L_2^2}, \tag{9}$$

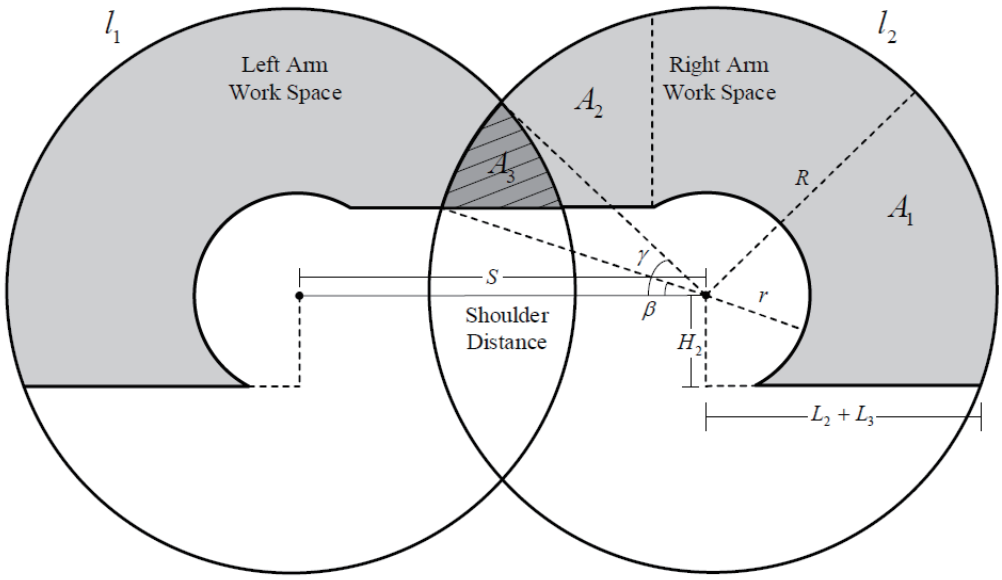


Fig. 5. Analysis of workspace projection surface.

$$\tan\beta = \frac{H_2}{L_2 + L_3}, \tag{10}$$

$$\cos\gamma = \frac{S}{2R}. \tag{11}$$

All the parameters in the equation can be seen in Fig. 5. As it is difficult to directly calculate the area of some figures in the working space, necessary simplification can be performed for some figures so as to have

$$S(A_1) = \frac{\pi(R^2 - r^2) - S(A_2) - L_3(R - r)\dfrac{H_2}{\sqrt{L_2^2 + H_2^2}}}{2}, \tag{12}$$

$$S(A_2) = \pi(R - r)^2 / 4, \tag{13}$$

$$S(A_3) = (2R - S) \cdot (R\sin\gamma - R\sin\beta) / 2. \tag{14}$$

To simplify the calculation, Eq. (5) is transformed from a multi-objective optimization function to a single-objective optimization through a linear weighting method by introducing the weight values ω_1 and ω_2 ($\omega_1 + \omega_2 = 1$) so that the total objective function of the optimization model is

$$\max Z = 2\omega_1[S(A_1) + S(A_2)] - \omega_2 S(A_3). \tag{15}$$

2.2.3 Analysis of constraints

2.2.3.1 Rod length limit

Considering the actual structure and length ratio of each joint of the human body, the length of each link of the exoskeleton should be limited, to a certain extent, so that

$$0 \le H_2 \le 200 \text{ (mm)}, \tag{16}$$

$$0 \le L_2 \le 400 \text{ (mm)}, \tag{17}$$

$$0 \le L_3 \le 400 \text{ (mm)}. \tag{18}$$

2.2.3.2 Joint angle limitation

Because of the limited rotation angle of the human upper limb joint, the arms of the exoskeleton naturally droop. Therefore, when analyzing the exoskeleton workspace, the joint angle is limited to

$$\theta_1 \in [-90°, 90°], \theta_2 \in [-90°, 90°]. \tag{19}$$

2.2.3.3 Exoskeleton weight constraint

Considering the power assist role of the industrial exoskeleton, the dead weight of each part of the system should be as small as possible to achieve effective power assist; thus, the total length of each connecting rod should be limited to a certain extent.

$$0 \le H_2 + L_2 + L_3 \le 800 \text{ (mm)} \tag{20}$$

3. Improved Harris Hawks Algorithm Based on the Sine Cosine Algorithm

For the exoskeleton power-assisted optimization model described, it is difficult to obtain an optimal solution within a feasible time by traditional convex optimization methods because of the existence of multiple variables and the complexity of the mathematical relationships between the variables. In contrast, metaheuristic algorithms represented by, for example, particle swarm algorithms[21] and simulated annealing algorithms[22] can better solve such problems and can give near-optimal feasible solutions in an acceptable time and space. Therefore, in this paper, we choose the more novel Harris Hawks algorithm for solving the problem, and optimize and improve it to lessen its shortcomings to further improve its convergence accuracy and efficiency.

3.1 Principle and steps of Harris Hawks algorithm

The Harris Hawks optimization (HHO) algorithm is a new swarm intelligence optimization algorithm proposed by Heidari *et al.*[23] in 2019. The algorithm is based on the cooperative predation behavior of Harris eagles and has the following stages.

3.1.1 Exploration phase

This phase is the global search phase. The Harris hawks fly in the air to detect and track the location of prey. In the exploration stage, the eagle group updates its position through random number q, as shown in Eq. (22).

$$X_m = \frac{1}{N}\sum_{i=1}^{N} X_i(t) \tag{21}$$

$$X(t+1)=\begin{cases} X_{rand}(t)-r_1\left|X_{rand}(t)-2r_2X(t)\right|, & q\geq 0.5 \\ [X_{rabbit}(t)-X_m(t)]-r_3[B_1+r_4(ub-lb)], & q<0.5 \end{cases} \tag{22}$$

Here, the upper and lower limits of parameters r_i and q are random numbers between 0 and 1, X_{rand} is a random individual in the population, and X_{rabbit} is the location of prey.

3.1.2 Transition phase

This stage is the transition of the eagle group from the exploration stage to the development stage and is mainly determined by the prey escape factor E, whose iteration formula is

$$E=2E_0\left(1-\frac{t}{T}\right), \tag{23}$$

where E_0 is the initial escape factor, t is the number of current iterations, and T is the maximum number of iterations. When the absolute value of E is less than 1, the population changes from the global search stage (exploration phase) to the local search stage (development phase).

3.1.3 Development stage

3.1.3.1 Hard encirclement attack

When $|E| < 0.5$ and $r \geq 0.5$, the population is updated by the following hard enveloping strategy.

$$X(t+1)=X_{rabbit}(t)-E\times\left|X_{rabbit}(t)-X(t)\right| \tag{24}$$

3.1.3.2 Soft encirclement attack

When $|E| \geq 0.5$ and $r \geq 0.5$, the population is updated by the following soft enveloping strategy.

$$X(t+1)=[X_{rabbit}(t)-X(t)]-E\times\left|2\times(1-rand)\times X_{rabbit}(t)-X(t)\right| \tag{25}$$

3.1.3.3 Fast dive hard encirclement

When $|E| < 0.5$ and $r < 0.5$, the population is updated by the hard encircling strategy of a fast dive, as shown by

$$Y=X_{rabbit}(t)-E\times\left|2\times(1-rand)\times X_{rabbit}(t)-X_m(t)\right|, \tag{26}$$

$$Z = X_{rabbit}(t) - E \times \left|2 \times (1 - rand) \times X_{rabbit}(t) - X_m(t)\right| + rand \times LF(D), \tag{27}$$

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ X, & F(Z) < F(X(t)) \end{cases} \tag{28}$$

$$LF(x) = 0.01 \times \frac{ru \times \sigma}{\left|rv\right|^{1/\beta}}, \tag{29}$$

$$\sigma = \left[\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\frac{\beta-1}{2}}} \right]^{1/\beta}. \tag{30}$$

3.1.3.4 Fast dive soft encirclement

When $|E| \geq 0.5$ and $r < 0.5$, the population is updated by a soft encircling strategy of a fast dive, as shown by

$$Y = [X_{rabbit}(t) - X(t)] - E \times \left|2 \times (1 - rand) \times X_{rabbit}(t) - X(t)\right|, \tag{31}$$

$$Z = [X_{rabbit}(t) - X(t)] - E \times \left|2 \times (1 - rand) \times X_{rabbit}(t) - X(t)\right| + rand \times LF(D), \tag{32}$$

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ X. & F(Z) < F(X(t)) \end{cases} \tag{33}$$

3.2 Improvement and optimization of Harris Hawks Optimizer

3.2.1 Population initialization based on Chebyshev chaotic mapping

Harris Hawks Optimizer (HHO), like most swarm intelligence optimization algorithms, generates the initial population in a randomized manner. However, this approach tends to make the initial population unevenly distributed, which affects the optimization performance of the algorithm and leads to a slow convergence rate.

This problem can be solved by using chaotic sequences to generate the initial population, which can generate states without repetition within a certain range, i.e., it is ergodic and can make the individuals in the population more uniformly distributed in the solution space. In existing literature, logistic mapping[24] and cubic mappings[25] are often used to initialize populations. In comparison, Chebyshev mapping[26] can have a larger distribution and better ergodicity. The Chebyshev chaos mapping formula is

$$x_{n+1} = \cos(k \arccos x_n),\ x_n = [-1, 1]. \tag{34}$$

When $k \geq 2$, the generated sequences are all uncorrelated, so $k = 3$ is taken in the simulation experiment of this study to generate the initial iterative population by this chaotic mapping. This can improve the population diversity and enhance the convergence ability of the algorithm.

3.2.2 Sine cosine search operator

To address the shortcomings of HHO, such as weak global search ability in the late stage and ease of falling into iterative stagnation, we introduce the sine cosine algorithm to enhance its search ability in the late stage of algorithm iteration.

The sine cosine algorithm (SCA) is a new natural-like optimization algorithm proposed by Seyedali Mirjalili, an Australian scholar, in 2016,[27] and has the advantages of few parameters and easy implementation. The sine cosine search operator is shown in Eq. (35). When the number of iterations of the algorithm $t > T$ and the algorithm falls into stagnation, i.e., the optimal value remains unchanged for five consecutive generations, each individual in the population is perturbed and updated as

$$X' = \begin{cases} X(t+1) + r_1 \times \sin(r_2) \times \left| r_3 gbest - X(t+1) \right|, & r_4 \leq 0.5 \\ X(t+1) + r_1 \times \cos(r_2) \times \left| r_3 gbest - X(t+1) \right|, & r_4 > 0.5 \end{cases} \tag{35}$$

$$r_1 = a - t\frac{a}{T}, \tag{36}$$

where a is the algorithm constant, r_2 is a random number within $[0,2\pi]$, r_3 is a random number within [0,2], r_4 is a random number within [0,1], and *gbest* denotes the position of the globally optimal individual.

3.2.3 Merit retention

After the sine cosine perturbation operation, to retain the optimal individuals and prevent the elite individuals from being destroyed during the updating process, we use a merit retention mechanism to determine whether to retain the updated individuals.

The positions of the new individuals after the perturbation are compared with the positions before the perturbation, and the positions of the individuals of the new generation are retained depending on their objective function values, determined as

$$X(t+1) = \begin{cases} X', & fit(X') > fit(X(t+1)) \\ X(t+1), & \text{else} \end{cases} \tag{37}$$

where $X(t+1)$ on the left side of the equation denotes the final retained new generation of individuals, and *fit*() denotes the fitness function. The above method enables individuals to update their positions within a certain range to find whether there is a more optimal solution, while retaining the superiority of the original solution. The method has a significant effect on

improving the later development capability of the HHO algorithm, and can effectively prevent the algorithm from falling into a local optimum.

3.2.4 Flow chart of improved Harris Hawks algorithm

The algorithmic flowchart of SCA-HHO based on the sine cosine algorithm proposed in this paper is shown in Fig. 6.

4. Simulation Analysis

In this study, MATLAB R2022a is used as the simulation experiment tool with Windows 11 as the operating system with 16 GB of RAM on board, AMD Ryzen 7 5800U as the processor, and NVIDIA GeForce RTX 3050 as the graphics card.

To verify the effectiveness and superiority of SCA-HHO designed in this study, the particle swarm algorithm (PSO), arithmetic optimization algorithm (AOA), whale optimization algorithm (WOA), and other cutting-edge population intelligence optimization algorithms are also selected to conduct cross-sectional simulation comparison experiments on the proposed model.

AOA is a novel metaheuristic algorithm proposed by Abualigah *et al.* in 2021[28] and the algorithm is optimized by introducing combinations of the four arithmetic operations. Among them, multiplication (M) and division (D) are used to achieve global exploration to enhance the solution dispersion; addition (A) and subtraction (S) are used to complete local exploitation to improve the local density of the solution, in addition to determining the search pattern of the algorithm using an iteration parameter of AOA. Finally, the global optimization search process of AOA is completed.

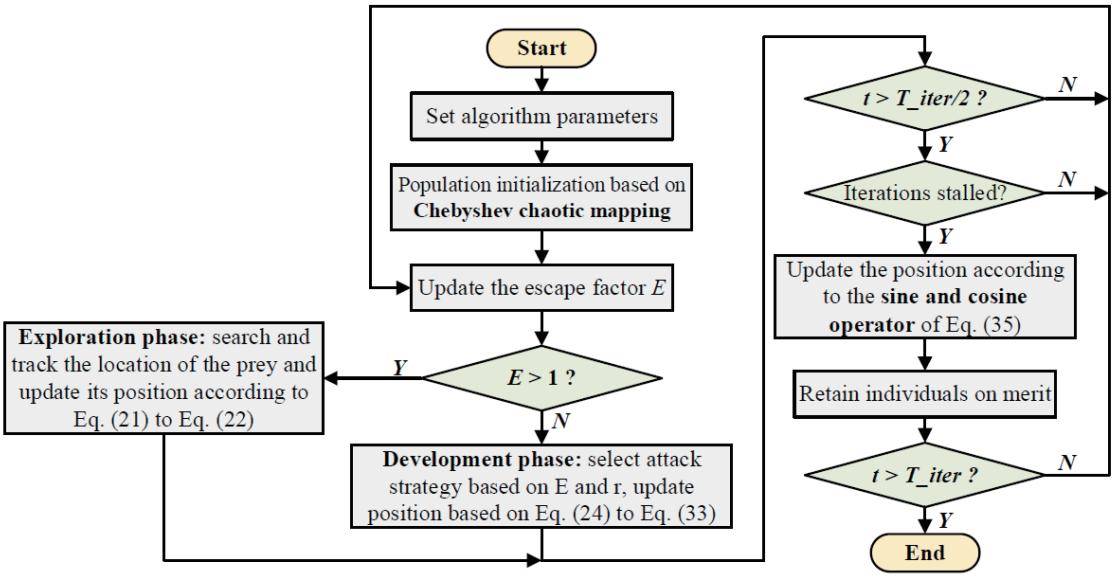


Fig. 6. (Color online) Flow chart of improved Harris Hawk algorithm.

WOA is a new population intelligence optimization algorithm proposed by Mirjalili and Lewis in 2016[29] and mimics the bubble-net hunting behavior of humpback whales to complete the optimization search through three stages: prey seizure, bubble-net predation, and prey search. The global optimal solution is ultimately obtained. WOA has the advantages of few control parameters, easy implementation, and strong optimization-seeking ability.

In the model parameters, the shoulder width S of the exoskeleton device is set to 500 mm. The parameters of each algorithm are set as follows: in SCA-HHO, the positive cosine parameter $a = 2$; in PSO, the maximum particle velocity $vmax = 3$, the minimum velocity $vmin = -3$, the learning factor $c_1 = c_2 = 1.5$, the maximum value of inertia weight wmax = 0.8, and the minimum value $wmin = 0.4$; in AOA, the maximum value of the acceleration function $MOP\_Max = 1$, the minimum value $MOP\_Min = 0.2$, the control parameter $\mu = 0.499$, and the sensitivity parameter $\alpha = 5$.

To ensure the fairness of the simulation experiments, the population size NP of each algorithm is set to 100 and the maximum number of iterations T is set to 300. Each algorithm is run independently 50 times in three groups with the weight ratios ω_1:ω_2 of 6:4, 5:5, and 4:6.

The average iteration curves of each algorithm for 50 runs under the three sets of weights are given in Fig. 7, the optimal solutions and the corresponding objective function values of each

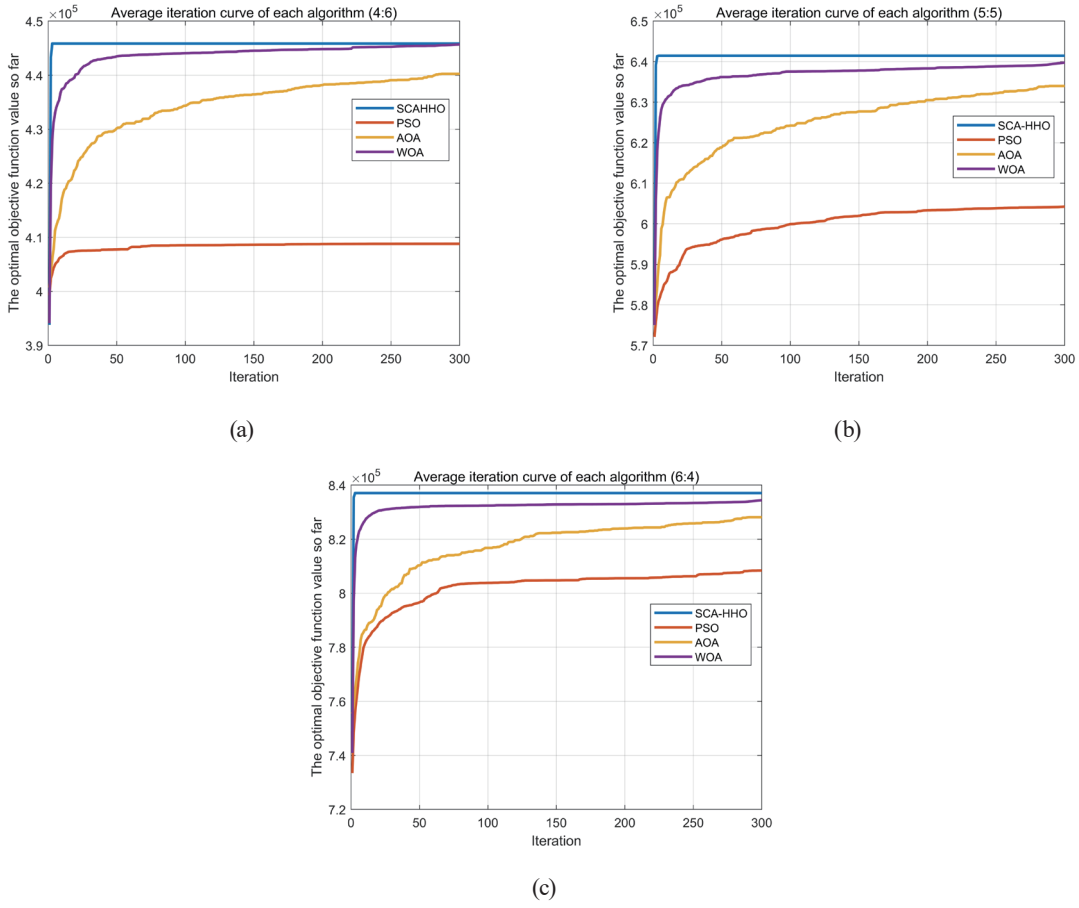


Fig. 7. (Color online) Iterative curves of four algorithms with weight ratios of (a) 4:6, (b) 5:5, and (c) 6:4.

algorithm for a single run are given in Tables 2–4, and the operation indexes of each algorithm for a single run are given in Tables 5–7.

Table 2
Optimal solutions obtained from a single run of each algorithm (6:4).

Optimal solution	H_2 /(mm)	L_2 /(mm)	L_3 /(mm)	Objective function value
SCA-HHO	200	400	400	8.3707×10^5
PSO	0.2111	398.9965	399.9882	8.1179×10^5
AOA	132.0746	393.9469	399.0402	8.2364×10^5
WOA	131	400	400	8.3259×10^5

Table 3
Optimal solutions obtained from a single run of each algorithm (5:5).

Optimal solution	H_2 /(mm)	L_2 /(mm)	L_3 /(mm)	Objective function value
SCA-HHO	200	400	400	6.4146×10^5
PSO	0.1544	399.8365	399.9973	6.0757×10^5
AOA	187.7649	392.1069	399.0343	6.3274×10^5
WOA	200	400	400	6.4146×10^5

Table 4
Optimal solutions obtained from a single run of each algorithm (4:6).

Optimal solution	H_2 /(mm)	L_2 /(mm)	L_3 /(mm)	Objective function value
SCA-HHO	200	400	400	4.4584×10^5
PSO	27.8262	372.0970	399.9862	4.0458×10^5
AOA	194.6146	386.4409	399.3021	4.3953×10^5
WOA	196	400	400	4.4534×10^5

Table 5
Operating metrics of each algorithm in a single run (6:4).

Optimal solution	$S(A_1)$ /(mm^2)	$S(A_2)$ /(mm^2)	$S(A_3)$ /(mm^2)	Workspace area /(mm^2)
SCA-HHO	6.9790×10^5	1.1187×10^5	3.3662×10^5	1.6195×10^6
PSO	6.8966×10^5	1.2566×10^5	4.1649×10^5	1.6306×10^6
AOA	6.8456×10^5	1.1849×10^5	3.5006×10^5	1.6061×10^6
WOA	6.9415×10^5	1.1931×10^5	3.5890×10^5	1.6269×10^6

Table 6
Operating metrics of each algorithm in a single run (5:5).

Optimal solution	$S(A_1)$ /(mm^2)	$S(A_2)$ /(mm^2)	$S(A_3)$ /(mm^2)	Workspace area /(mm^2)
SCA-HHO	6.9790×10^5	1.1187×10^5	3.3662×10^5	1.6195×10^6
PSO	6.9074×10^5	1.2566×10^5	4.1766×10^5	1.6328×10^6
AOA	6.8528×10^5	1.1244×10^5	3.2997×10^5	1.5955×10^6
WOA	6.9790×10^5	1.1187×10^5	3.3662×10^5	1.6195×10^6

Table 7
Operating metrics of each algorithm in a single run (4:6).

Optimal solution	$S(A_1)$ /(mm^2)	$S(A_2)$ /(mm^2)	$S(A_3)$ /(mm^2)	Workspace area /(mm^2)
SCA-HHO	6.9790×10^5	1.1187×10^5	3.3662×10^5	1.6195×10^6
PSO	6.5601×10^5	1.2532×10^5	3.6748×10^5	1.5627×10^6
AOA	6.7931×10^5	1.1151×10^5	3.2187×10^5	1.5816×10^6
WOA	6.9765×10^5	1.1235×10^5	3.3777×10^5	1.6200×10^6

As seen from the average iteration curves in Fig. 7, SCA-HHO has a significantly higher convergence speed and better search ability than the other algorithms. The results in Tables 2–7 show that SCA-HHO has the highest solution accuracy and the highest quality of feasible solutions among the simulation algorithms, and its solution performance maintains good stability in multiple runs. Under the ω_1:ω_2 = 6:4 condition, the objective function values improve by 3.11, 1.63, and 0.53% in a single run compared with the PSO, AOA, and WOA, respectively; under the ω_1:ω_2 = 5:5 condition, they improve by 5.58, 1.38, and 0%, respectively; under the ω_1:ω_2 = 4:6 condition, they improve by 10.20, 1.43, and 0.11%, respectively. The objective function values of SCA-HHO improved by 6.30, 1.48, and 0.88% on average under the three weight ratios.

The simulation results show that the optimal solutions of the model for all three weight ratios ω_1:ω_2 are H_2 = 200 mm, L_2 = 400 mm, and L_3 = 400 mm, when the area $S(A_1) = 6.9790 \times 10^5$ mm^2, $S(A_2) = 1.1187 \times 10^5$ mm^2, and $S(A_3) = 3.3662 \times 10^5$ mm^2, and the exoskeleton working space projection area is 1.6195×10^6 mm^2.

In summary, SCA-HHO proposed in this paper performs well in terms of search ability and convergence speed compared with other popular swarm intelligence algorithms. Its search stability is also significantly higher than those of other similar algorithms, thus verifying the feasibility and applicability of the algorithm.

5. Conclusions

The application of power exoskeleton devices provides higher operational efficiency and more reliable operational safety than traditional operation methods, and can effectively compensate for the lack of human function in such situations as electrically charged, high-altitude, and high-load operations, thus effectively assisting personnel to complete their tasks. In this paper, we proposed an optimization method for the upper limb exoskeleton workspace based on the industrial exoskeleton power-assisted model. By referring to an actual physical structure and constraints of an upper limb exoskeleton device, the linkage model was derived on the basis of the positive kinematic method, and the multi-objective optimization model of the workspace was further established. Finally, an improved Harris Hawks algorithm (SCA-HHO) based on the sine and cosine algorithm was designed to solve the optimal parameters of the model. Chebyshev chaos mapping, sine cosine search, and merit retention were introduced to improve the convergence and post-optimal search capability of the algorithm. The results showed that the SCA-HHO algorithm is comparable to PSO, AOA, and WOA.

The results showed that the objective function of SCA-HHO was improved by 6.30%, 1.48%, and 0.88% on average compared with PSO, AOA, and WOA, respectively. In addition, the optimal solution of SCA-HHO was $[H_2, L_2, L_3] = [200, 400, 400]$ mm for each condition, which shows that it has good solution performance and accuracy, thus verifying the effectiveness and robustness of the algorithm designed in this study. In future research work, more attention will be paid to the improvement of the intelligence and human–machine synergy of exoskeleton technology to realize more flexible and intelligent exoskeleton devices to adapt to different operating environments and operational tasks. Exoskeleton technology will also be combined with robotics, artificial intelligence, and other cutting-edge technologies to achieve more efficient and intelligent industrial production.

Acknowledgments

This work was supported by a Science and Technology Project of China Southern Power Grid Co., Ltd. under Grant No. YNKJXM20220216.

References

1 C. Linnenberg and R. Weidner: Appl. Ergon. **101** (2022) 103706. https://doi.org/10.1016/j.apergo.2022.103706
2 M. F. Antwi-Afari, H. Li, S. Anwer, D. W Li, Y. Yu, H. Y. Mi, and I. Y. Wuni: Saf. Sci. **142** (2021) 105382. https://doi.org/10.1016/j.ssci.2021.105382
3 S. A. Elprama, B. Vanderborght, and A. Jacobs: Appl. Ergon. **100** (2022) 103615. https://doi.org/10.1016/j.apergo.2021.103615
4 M. Pollák, M. Kočiško, D. Paulišin, and P. Baron: Adv. Mech. Eng. **12** (2020). https://doi.org/10.1177/1687814020972893
5 S. Ma, J. H. Yao, X. L. Wei, and Y. H. Zhu: 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conf. (IMCEC) (2016) 1705. https://doi.org/10.1109/IMCEC.2016.7867509.
6 L. Roveda, L. Savani, S. Arlati, T. Dinon, G. Legnani, and L. M. Tosatti. Int. J. Ind. Ergon. **79** (2020) 102991. https://doi.org/10.1016/j.ergon.2020.102991
7 A. Foroutannia, M. R. Akbarzadeh-T, and A. Akbarzadeh: Biomed. Signal Process. Control **75** (2022) 103557. https://doi.org/10.1016/j.bspc.2022.103557
8 R. Sharma, P. Gaur, S. Bhatt, and D. Joshi: Appl. Soft Comput. **105** (2021) 107226. https://doi.org/10.1016/j.asoc.2021.107226
9 Y. Wang, H. P. Wang, and Y. Tian: ISA Trans. **128** (2021) 184. https://doi.org/10.1016/j.isatra.2021.10.009
10 T. K. Xiao, Y. Zhang, L. G. Qiang, 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
11 D. Skrobek, D. Cekus: Eng. Optim. **51** (2019) 1997.
12 D-H. Lee, H. Park, J-H. Park, M-H. Baeg, and J-H. Bae: Intell. Serv. Rob. **10** (2017) 137. https://doi.org/10.1007/s11370-017-0215-z
13 S. K. Hasan and A. K. Dhingra: Results Control Optim. **7** (2022) 100107. https://doi.org/10.1016/j.rico.2022.100107
14 Y. Zheng, Y. Q. Wang, and J. X. Liu: Future Gener. Comput. Syst. **129** (2022) 187. https://doi.org/10.1016/j.future.2021.11.021
15 A. F. Sallama, M. F. Abbod, S. Mahmood, and Khan: Int. J. Eng. Technol. Innov. **4** (2014) 223.
16 M. S. Amiri, R. Ramli, and M. F. Ibrahim: Rob. Auton. Syst. **125** (2020) 103425. https://doi.org/10.1016/j.robot.2020.103425
17 W. A. Azar and P. S. Nazar: Biomed. Signal Process. Control **69** (2021) 102864. https://doi.org/10.1016/j.bspc.2021.102864
18 I. M. Chao, S. C. Hsiung, and J. L. Liu: Adv. Technol. Innov. **5** (2020) 147.
19 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

20 B. Velusamy and S. Pushpan: Int. J. Eng. Technol. Innov. **9** (2019) 182.
21 A. Madani, A. Engelbrecht, and B. Ombuki-Berman: Swarm Evol. Comput. **101262** (2023). https://doi.org/10.1016/j.swevo.2023.101262
22 H. L. Lv, X. Y. Chen, and X. W. Zeng: Chaos, Solitons Fractals **148** (2021) 111048. https://doi.org/10.1016/j.chaos.2021.111048
23 A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. L. Chen: Future Gener. Comput. Syst. **97** (2019) 849. https://doi.org/10.1016/j.future.2019.02.028
24 A. Daoui, H. Karmouni, O. E. Ogri, M. Sayyouri, and H. Qjidaa. Expert Syst. Appl. **190** (2022) 116193. https://doi.org/10.1016/j.eswa.2021.116193
25 X. Z. Zhang, Z. Y. Wang, and Z. Y. Lu: Appl. Energy **306(A)** (2022) 118018.
26 S. H. Yu and S. B. Su: J. Frontiers of Computer Science and Technology (2014).
27 S. Mirjalili: Knowledge-Based Syst. **96** (2016) 120. https://doi.org/10.1016/j.knosys.2015.12.022
28 L. Abualigah, A. Diabat, S. Mirjalili, M. A. Elaziz, and A. H. Gandomi: Comput. Methods Appl. Mech. Eng. **376** (2021) 113609. https://doi.org/10.1016/j.cma.2020.113609
29 S. Mirjalili and A. Lewis: Adv. Eng. Software **95** (2016) 51. https://doi.org/10.1016/j.advengsoft.2016.01.008

About the Authors

Mingxian Liu graduated from Kunming University of Technology with a bachelor's degree in electrical engineering and its automation. His main research interests are in distribution network operation and inspection, power strip operation, and artificial intelligence. (liumingxian@lj.csg.cn)

Xinbo Zhou is currently pursuing his bachelor’s degree in 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)

Jing Bao is currently pursuing his bachelor’s degree in the Faculty of Civil Aviation and Aeronautics, Kunming University of Science and Technology. His research interests include artificial intelligence and intelligent systems. (baojing@stu.kust.edu.cn)