aerospace

MDPI

Article

Aero-Engine Rotor Assembly Process Optimization Based on Improved Harris Hawk Algorithm

Bin Zhang, Hongyi Lu *, Shun Liu, Yucheng Yang and Doudou Sang

School of Aircraft Engineering, Nanchang Hangkong University, Nanchang 330063, China
* Correspondence: 13964508115@163.com

Abstract: Multi-stage disc rotor assembly is an important part of the aero-engine rotor manufacturing process. To solve the problem that excessive unbalance of assembly affects the vibration of the whole machine, this paper presents an optimization method for aero-engine rotor assembly balance based on an improved Harris Hawk algorithm. Firstly, the assembly sequence model of the single-stage disc blade and the phase assembly model of a multi-stage disc of the engine rotor is established. Secondly, by using the initial population generation based on dynamic opposing learning and the escape energy function of the non-linear logarithmic convergence factor, the search mechanism of the whale optimization algorithm is introduced in the global exploration, and the adaptive weight strategy and mutation strategy of the genetic algorithm is introduced in the development to improve the algorithm. Then, the effectiveness of the algorithm is verified by experiments and compared with particle swarm optimization, genetic algorithm, and Harris Hawk algorithm, the unbalance of the optimal blade assembly sequence is reduced by 91.75%, 99.82%, and 83.39%, respectively. The algorithm comparison and analysis are carried out for all disc-blade assembly optimization of the rotor. The optimal unbalance of the improved Harris Hawk optimization algorithm is reduced by 79.71%, 99.48%, and 54.92% on average. The unbalance of the algorithm in this paper is the best. Finally, the improved Harris Hawk algorithm is used to find the best assembly phase, and the optimized unbalanced force and moment are reduced by 84.22% and 98.05%, respectively. The results of this study prove that the improved Harris Hawk algorithm for aero-engine rotor assembly balance optimization can effectively reduce the unbalance of rotor disc blade assembly and rotor unbalance and provide a powerful solution for solving engine vibration.

Keywords: aero-engine rotor; blade sequencing optimization; Harris hawk optimization; unbalance; whale optimization

check for updates

Citation: Zhang, B.; Lu, H.; Liu, S.; Yang, Y.; Sang, D. Aero-Engine Rotor Assembly Process Optimization Based on Improved Harris Hawk Algorithm. *Aerospace* **2023**, *10*, 28. https://doi.org/10.3390/aerospace10010028

Academic Editor: Sergey Leonov

Received: 18 November 2022
Revised: 11 December 2022
Accepted: 23 December 2022
Published: 28 December 2022

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Assembly is very important in the manufacturing process of products, and its results have a significant impact on the final performance and quality of products [1]. The assembly quality of aero-engines is an important factor to ensure the overall performance of aero-engines. According to statistics, the number of engine failures caused by vibration accounts for 50% to 60% of the total number of engine failures [2]. The unbalance force caused by excessive rotor unbalance is the main factor causing vibration during the operation of an aero-engine [3]. The unbalance of the rotor will not only cause vibration but also cause airflow exciting force. Under certain conditions, it may cause rotor instability, which will adversely affect the normal operation of the entire engine. The aero-engine rotor is composed of multi-stage discs and blades. In the process of the rotor assembly, the assembly sequence of blades affects the balance quality of the assembled disc, while the assembly phase angle of multi-stage discs affects the overall balance quality of the rotor. These two factors are important factors affecting the balancing quality of the rotor [4].

The assembly sequence of the aero-engine rotor blade set not only directly determines the static balance quality of the rotor, but also indirectly affects the dynamic balance quality

of the rotor. To ensure the balanced mass of the rotor, the gravity moment of the blades must be equally distributed so that the centrifugal forces in all directions can be offset as far as possible and the residual unbalance can be kept as close as possible to the minimum. Under ideal conditions, if the moments of gravity of the rotor blades are equal, then the residual imbalance of the rotor is zero, but the actual processing conditions of the blades are unable to achieve the ideal conditions. To minimize residual imbalance, the blade assembly sequence must be optimized. To solve the engineering optimization problem of assembly sequence planning, researchers have introduced some optimization algorithms into actual engineering assembly-optimization, such as particle swarm optimization, ant colony algorithm, genetic algorithm, and simulated annealing algorithm [5]. Liu et al. [6], for NP-hard characteristics of assembly sequence planning and low efficiency in solving relatively optimal assembly sequences, used an improved simulated annealing algorithm to solve the optimization problem of the rotor blade assembly sequence. The effectiveness of the improved simulated annealing algorithm is verified by the assembly of turbine rotor blades. Zhao et al. [7] proposed a small-world network-based genetic algorithm to solve the problem of assembly sequence optimization for rotor blades. The proposed method consists of a standard genetic algorithm and NW small-world network. An experiment was carried out on the assembly of turbine rotor blades to verify the effectiveness of the algorithm. Lili Li et al. [8], while aiming at the problems of the multi-adjustable rotor blades of aero-engine blower with a low success rate of assembly, used the improved simulated annealing algorithm to optimize the assembly sequence of rotating blades, which greatly reduced the residual unbalance of blades and helped to reduce the number of assembly adjustments of blades. Li et al. [9], aiming at the optimization of rotor blade assembly sequence, adopted the improved simulated annealing algorithm to plan the assembly sequence of the blade with the restriction of gravity moment difference of diagonal blade, which provided an optimized assembly sequence for the rotor blade assembly. Zhu et al. [10], aiming at the problem of large unbalanced value in moving vane assembly, based on the brief analysis of moving vane assembly and weighing principle, put forward an intelligent optimization method for moving vane installation sequencing, by combining the global optimization and fast convergence of the genetic algorithm. Pitsoulis et al. [11] proposed a heuristic algorithm to solve NP-Hard combinatorial optimization problems in turbine engine manufacturing and maintenance. Wang et al. [12] improved the non-dominant sequencing genetic algorithm and introduced the controlling elite and dynamic congestion distance, which showed a good performance in dealing with multi-objective optimization problems of wind turbines. However, the blade matching problem is a combinatorial optimization problem, and the difficulty of solving the matching problem increases geometrically with the number of rotor stages and components. Therefore, it is difficult to find the optimal solution of the problem with traditional methods, and it is easy to fall into the local optimum.

The assembly phase angle of the multi-stage disc affects the overall balancing quality of the rotor, and the excessive initial unbalance of the multi-stage disc rotor after assembly will bring difficulties to the subsequent balancing process. At the same time, greater vibration will occur when the rotor fails, so it is very important to control the initial unevenness during an assembly. Many scholars have studied this problem. Tu et al. [13] proposed an efficient method to solve the optimal stacking assembly scheme based on geometric algebraic theory and combined it with a multi-objective optimization method. The non-dominant sequencing genetic algorithm was used to solve the multi-objective optimization problem of the rotor assembly, and the effectiveness of the method was verified by experiments. Zhang et al. [14] carried out phase optimization by predicting the assembly eccentricity of rotor parts with the intelligent algorithm, and optimized different phases with particle swarm optimization to meet the requirements of the coaxial of the aero-engine high-pressure rotor assembly, which verified the effectiveness of the method. Chen et al. [15] proposed an assembly method based on the dual-objective synchronous optimization of coaxial and imbalance, which was used to predict the coaxial and imbalance of multistage

rotors. The genetic algorithm was used to achieve the synchronous optimization of coaxial and imbalance of multistage rotors. The effectiveness of the optimization method was verified by experiments conducted by Zhang et al. [16]. Zhang et al. [17] proposed a hybrid evolutionary algorithm. This algorithm is the combination of the ant colony algorithm and SA algorithm to reduce the problem of falling into the local optimal solution. Chen et al. [18] predicted the best assembly angle of the rotor at each stage in the pre-assembly process, put forward an optimization method for multi-stage rotor assembly of the aero-engine, solved the best assembly angle of the rotor at each stage by genetic algorithm, and verified by experiments that the optimization method could achieve synchronous optimization of coaxial and imbalance. Wang et al. [19] proposed an optimization model for the coaxial error of multi-stage rotors. With coaxial error as the optimization objective, the best assembly angle of each rotor was obtained. Chen et al. [20] pointed out that the assembly datum of unbalanced optimization should not be equal to that of geometric eccentricity error optimization, and proposed an unbalanced optimization method based on the orientation change axis. Sun et al. [21] proposed an assembly optimization method for the dual objectives of the concentricity and imbalance of multi-stage rotors, which enables concentricity and imbalance to approach their optimum solutions as close as possible at the same time. Li et al. [22] further improved the optimization of rotor balance by genetic algorithm to optimize the angle installation position of discs at all levels. Chen et al. [23] proposed a multi-stage rotor assembly optimization method for aero-engines based on a genetic algorithm and constructed a dual-objective fitness function based on coaxial and imbalance. It was verified by experiments that the proposed assembly optimization method can achieve synchronous optimization of coaxial and imbalance in the pre-assembly process of the aero-engine. The relevant research at home and abroad has improved the balance of the rotor to a certain extent, but the calculation method is complex. In addition, when the number of rotor stages is large, the manufacturer faces a large workload. The balance of the rotor cannot be monitored during the assembly process, and it is easy to carry out multiple installations and tests.

The assembly planning problem is NP-hard. The complexity of searching for the optimal planning increases with the size of the possible assembly sequence space searched exhaustively, and it is difficult to obtain relatively optimal planning in a short time. This challenge has become one of the important driving forces for the study of assembly planning optimization [24]. The meta heuristic algorithm first randomly generates multiple candidate solutions in the problem-solve space, and then improves these solutions in the iterative process. Finally, the best candidate solution for improvement is introduced as the solution to the problem. Said et al. [25] established a multi-attribute optimization method for sustainable aviation fuel production process using the multi-objective response surface method. Sharma et al. [26] proposed a response surface method to optimize the performance and emission parameters of mixed-fuel diesel engines. Hendricks et al. [27] proposed an effective analysis method to optimize the gas engine cycle, and realized the design optimization of some new concepts. At present, scholars have solved assembly planning problems through different optimization algorithms, such as Ant Colony Optimization (ACO) [27], Genetic Simulated Annealing Algorithm (GSAA) [28], Genetic Algorithm (GA) [29], Immune Algorithm (IA) [30], Sparrow Search Algorithm (SSA) [31], Grey Wolf Optimization Algorithm (GWO) [32], and Flower Pollination Algorithm (IFPA) [33]. These algorithms have a common feature: the search step has two stages: exploration (diversification) and development (enhancement). In the search phase, the algorithm should use the randomization operator as much as possible to explore different regions of the feature space. The exploitation phase is usually carried out after the exploration phase. The algorithm focuses on the feature space of the solution with better adjacent quality, and strengthens the local area search-process. A good optimizer should be able to achieve a reasonable and fine balance between exploration and development trends. Otherwise, the possibility of falling into local optimal (LO) and premature convergence defects increases. Table 1 lists the performance of the optimization algorithm.

Table 1. An overview of important research in the field of optimization methods.

Reference	Algorithm	Advantages and Disadvantages of Algorithm
Literature [27]	ACO	The fast convergence speed and low diversity of the population are not conducive to improving the global optimization ability of the algorithm.
Literature [29]	GA	Fast and random search capability, with potential parallelism, so the search speed of the algorithm is relatively slow
Literature [30]	IA	Immune algorithm has the advantages of adaptability, randomness, parallelism, global convergence, population diversity
Literature [31]	SSA	The algorithm has fast convergence speed and strong global search ability.
Literature [32]	GWO	It has the characteristics of strong convergence, few parameters and easy realization.
Literature [33]	IFPA	It is easy to fall into local optimum, and the convergence speed is slow in the later stage

The above method can solve the optimization problem in the rotor assembly stage to a certain extent, but does not consider the influence of the initial unbalance of the disk on the assembly stage, and the optimization result has errors. According to the analysis of the research status of the above-mentioned optimization methods for aero-engine rotor assembly, there are problems that the vane-sequencing optimization method does not consider the initial static imbalance of the disc, and the influence of the vane-sequencing optimization of the multi-disc is not considered in the rotor multi-disc assembly optimization. In this paper, an assembly optimization method for the multi-stage rotor of an aero-engine based on an improved Harris Hawk algorithm is proposed. The improved Harris Hawk algorithm uses initial population generation based on dynamic opposing learning and improved escape from energy function. In the global exploration, the exploration mechanism of the whale optimization algorithm is introduced to expand the search space, and the adaptive weight strategy is introduced in the development to improve the optimization efficiency and convergence speed of the algorithm. Before entering the next iteration, the mutation in the genetic algorithm is used to avoid a premature algorithm. For the assembly optimization of the aero-engine rotor, the assembly process is as follows: Firstly, with the restriction of diagonal mass-moment difference, the single-stage blade, and the disc assembly model is established, and the optimal assembly sequence is found by using an improved Harris Hawk algorithm. Then, the assembly sequence of the remaining single-stage blade and the corresponding disc is completed by repeated operations. Finally, in multi-stage disc assembly, based on the imbalance of multiple discs, the multi-stage disc rotor assembly model is established, and the optimal assembly phase is found by using an improved Harris Hawk algorithm, and the optimal imbalance force and the corresponding assembly phase of the multi-stage disc are obtained. This paper takes an aero-engine rotor as an example to verify the effectiveness of the algorithm and obtain the optimal assembly sequence. At the same time, the performance of the algorithm is analyzed and compared with particle swarm optimization, the genetic algorithm, and the Harris Hawk algorithm. It is found that the improved Harris Hawk optimization algorithm is superior to other algorithms in convergence speed and optimization. The flow chart is shown in Figure 1, which briefly describes the process.

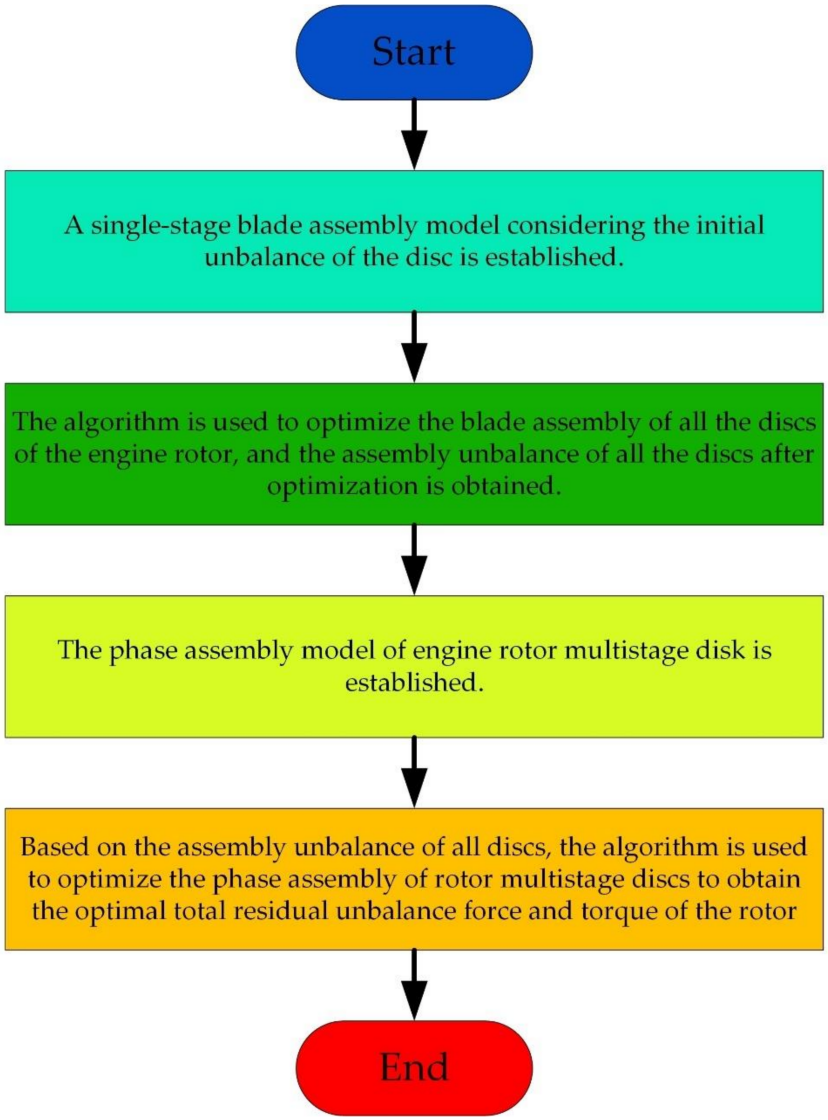


Figure 1. Flow chart of engine rotor assembly optimization.

2. Single Stage Disc and Blade Optimization Model

The objective of single-stage blade assembly sequence-planning is to ensure that the design values are not exceeded thereby ensuring that the rotor blades are assembled in the planned assembly sequence and that the rotor static balance meets the design requirements. Because all blades rotate at the same angular velocity, the minimum residual imbalance is a vector and the minimum of the mass moment of all blades relative to the axis of rotation. The residual imbalance is a calculated method shown in Equations (1)–(4):

$$M_x = \sum_{i=1}^{n} M_i \cos\theta_i \tag{1}$$

$$M_y = \sum_{i=1}^{n} M_i \sin\theta_i \tag{2}$$

$$M_{Blade} = \sqrt{M_x^2 + M_y^2} \tag{3}$$

$$\alpha = \arctan\left(\frac{M_y}{M_x}\right) \tag{4}$$

where, M_x, M_y is the component of gravity moment in x direction and y direction respectively; M_i is the moment of gravity of blade i; θ_i is the angle between the gravity moment

vector of the blade i and the positive direction of axis x, M_{left} is the residual unbalance, α is the angle between the residual unbalance vector and the positive direction of axis x.

Figure 2 shows the position representation and initial unbalance vector distance $\left(\left|\vec{d}\right|\right)$, showing the solution method of residual unbalance M_{left}.

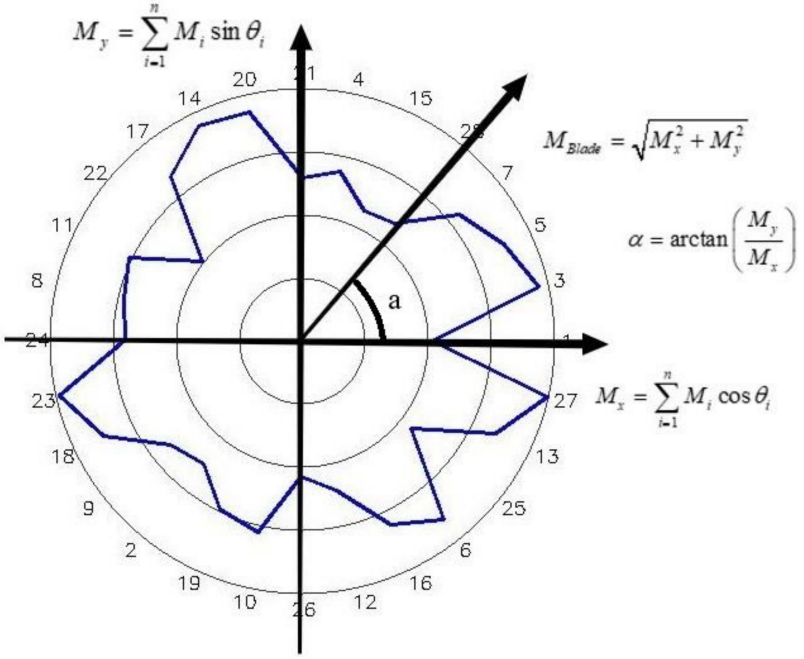


Figure 2. Remaining unbalance of blade.

To make the rotor run more stably, the deviation range of the blade groups should be controlled to avoid too much difference of mass moment between the same group of blades. According to the enterprise design regulations, the difference in the mass moment between the diagonal position of the blades on both blades must not exceed the specified value. Therefore, the constraint conditions for blade assembly in this paper are shown in Equation (5):

$$\begin{cases} |M_i r_r - M_j r_j| \le \delta \\ |i - j| = \frac{n}{2} \end{cases} \tag{5}$$

where δ is the upper limit of the mass moment difference at the diagonal position of the blade at a certain level.

Ideally, the discs are symmetrical. In fact, the discs are asymmetric due to machining errors. Before optimizing the assembly sequence of the blades, we should consider both the unbalance of the mass moment of the blades and the initial unbalance of the disc. The objective of our assembly sequence optimization is to achieve the minimum unbalance when the blades are assembled on the disc. Figure 3 shows the effect of the initial disc imbalance on the assembled disc imbalance. The disc mass moment affects the minimum imbalance of the optimum assembled blade sequence. How to obtain the minimum imbalance of the entire blade disc is the next problem to be solved. As can be seen from Figure 3, considering the eccentric mass moment of the disc, the minimum imbalance is shown in Equations (6)–(8):

$$\vec{M}_f = \vec{M}_{balde} + \vec{M}_e \tag{6}$$

$$M_f = \sqrt{M_{fx}^2 + M_{fy}^2} = \sqrt{\left(M_e \cos\theta_a + \sum_{i=1}^{n} M_i \cos\theta_i\right)^2 + \left(M_e \cos\theta_a + \sum_{i=1}^{n} M_i \cos\theta_i\right)^2} \tag{7}$$

$$\beta = \arctan\left(\frac{M_{fy}}{M_{fx}}\right) \tag{8}$$

where M_e is the mass moment of the disc, θ_a is the angle between the M_e vector and the positive direction of the x axis, and β is the angle between the residual unbalance vector and the positive direction of the x axis.

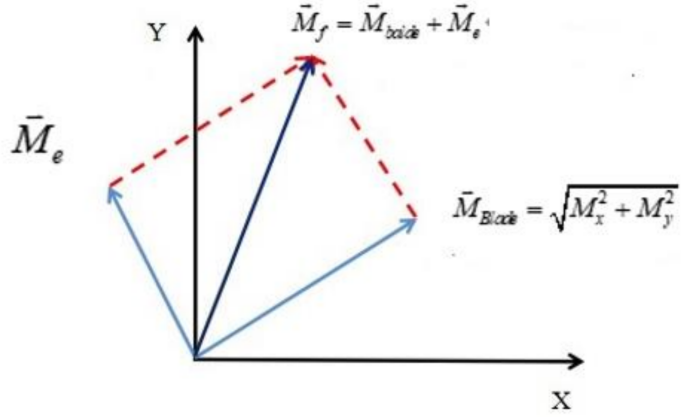


Figure 3. The initial unbalance of the disc influences optimum unbalance in the blade assembly.

Then the optimization objective function for the assembly sequence optimization of discs and blades are shown in Equation (9).

$$\begin{aligned} &\min\left\{f = M_f\right\} \\ S.T.&\begin{cases} |M_i r_r - M_j r_j| \le \delta \\ |i-j| = \frac{n}{2} \end{cases} \end{aligned} \tag{9}$$

3. Optimum Assembly Model of Multi-Stage Disc Rotor

The single-stage blade and disc optimization model established can calculate the residual imbalance after each stage of disc and blade assembly, while the total imbalance generated by the aero-engine rotor is the result of the accumulation of residual imbalance after all blade assembly. When the rotor rotates at high speed, these residual imbalances produce large imbalance forces and moments, which can seriously affect engine performance, as shown in Figure 4. Take the five-stage disc blade as an example. After each stage of the blade assembly, unbalanced blade groups are superimposed on the xy plane. In the actual assembly process, changing the assembly phase angle of the blades can effectively adjust the overall imbalance of multi-stage blades after assembly, so it is necessary to consider the influence of the blade assembly phase on the overall imbalance of the rotor.

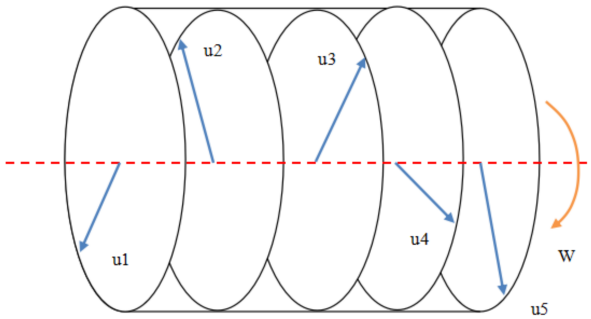


Figure 4. Five-stage disc rotor assembly imbalance.

The sum of the initial unbalance of each stage disc and the unbalance after blade assembly is $u_j(j = 1, 2, 3, \ldots, m)$. Due to the unbalance moment, a deformation force is

generated at the journal position of the engine rotor, thus the position of the journal force in the rotor is recorded.

In addition, the distance coordinates of the force points between the other discs and blade groups are defined as $L_j(j = 1, 2, 3, \ldots, m)$. The direction $\theta_i(i = 1, 2, 3, \ldots, m)$ of unbalance of each stage disc and blade is taken as a design variable, and the component of unbalance force on x axis and y axis is expressed as Equations (10) and (11).

$$F_x = \sum_{j=1}^{m} F_{jx} = \sum_{j=1}^{m} u_j \cos\theta_j \times w^2 \tag{10}$$

$$F_y = \sum_{j=1}^{m} u_j \sin\theta_j \times w^2 \tag{11}$$

where F_x is the component of the unbalanced force on the x axis; F_y is the component of the unbalanced force on the y axis. m is the number of fan blade stages.

The moment of the unbalanced force around the x and y axes is expressed as Equations (12) and (13).

$$T_x = \sum_{j=1}^{m} F_{jy} \times L_j = \sum_{j=1}^{m} u_j \sin\theta_j \times w^2 \times L_j \tag{12}$$

$$T_y = \sum_{j=1}^{m} F_{jx} \times L_j = \sum_{j=1}^{m} u_j \cos\theta_j \times w^2 \times L_j \tag{13}$$

where T_x is the moment of the unbalanced force around the x axis; T_y is the moment when an unbalanced force is applied around the y axis.

Multi-stage phase optimization aims to minimize unbalance after the assembly of multi-stage blades, which means that the unbalance forces and moments are minimized. Therefore, the objective function of the multi-stage vane phase optimization model is Equations (14) and (15).

$$\begin{cases} \min F = \sqrt{F_x^2 + F_y^2} \\ \min T = \sqrt{T_x^2 + T_y^2} \end{cases} \tag{14}$$

$$\min F(X) = \mu F + (1 - \mu) T \tag{15}$$

where μ is the weight coefficient.

4. Harris Hawk Optimization Algorithm

Heidari et al. [34] proposed the HHO model method based on the Harris Hawk's hunting strategy and operator structure of grey wolf optimization and divided the Harris Hawk algorithm into two stages of exploration and development.

4.1. Exploration Phase

When $|E| \geq 1$, the Harris Hawk randomly inhabits the search space and searches for prey through two strategies, balancing the probability of using the strategy with a random number r_1. The mathematical expression is Equation (16).

$$X(t+1) = \begin{cases} X_{rand}(t) - r_2 |X_{rand}(t) - 2r_3 X(t)| & r_1 \geq 0.5 \\ X_{best}(t) - X_M(t) - r_4(LB + r_5(UB - LB)) & r_1 \leq 0.5 \end{cases} \tag{16}$$

where $X(t+1)$ and $X(t)$ are Harris Hawk locations at iterations $t+1$ and t, $X_{rand}(t)$ are random Harris Hawk locations at iterations t,r_1, r_2, r_3, r_4, r_5 are random numbers evenly distributed in [0, 1] intervals, X_{best} and X_M are prey rabbit locations (current iteration optimum), and Harris Hawk center locations, LB and UB are upper and lower bounds of the range of values. Formula (16) describes the exploratory behavior of the HHO algorithm when the prey escapes energy $|E| \geq 1$, where E decreases with iteration, the formula is as Equations (17) and (18).

$$E = 2E_0\delta \tag{17}$$

$$\delta = 1 - \frac{t}{T} \tag{18}$$

where T is the maximum number of iterations, t is the current iteration step, E_0 is the initial energy value, and take the random number that is evenly distributed between [1, 1].

4.2. Development Phase

When $|E| < 1$ indicates that the Harris Eagle has found the direction of the prey, the HHO algorithm performs the development behavior. The "surprise" hunting behavior of the Haris hawk can be divided into four strategies by comparing its absolute energy value with its random number of 0.5.

4.2.1. Soft Surrounding

When $|E| \geq 0.5$ and $rand \geq 0.5$, The prey still has enough energy and tries to escape through some random misleading jumps, but it cannot escape in the end. the Harris Hawk performs a hard encirclement strategy. Harris Hawk implemented a hard encirclement strategy rule such as Equation (19).

$$X(t+1) = (X_{best}(t) - X(t)) - E|JX_{best}(t) - X(t)| \tag{19}$$

Whereas r_6 is a random number within [0, 1], $J = 2(1 - r_6)$ represents the jumping ability of the prey. The J value varies randomly during each iteration to simulate the nature of the prey's motion.

4.2.2. Hard Surrounding

When $|E| < 0.5$ and $rand \geq 0.5$, The prey is very tired and its escape energy is very low. In addition, Harris's eagle almost did not encircle the target, and finally carried out a surprise attack. Harris Hawk implemented a hard encirclement strategy rule such as Equation (20).

$$X(t+1) = X_{rabbit}(t) - E|X_{best}(t) - X(t)| \tag{20}$$

4.2.3. Progressive Rapid Subduction Soft Surrounding

When $|E| \geq 0.5$ and $rand < 0.5$, because the prey still has a large jumping ability J, the Harris Hawk initiates a fast dive soft surprise. If the attack fails (Y fitness does not improve), a random walk Z is performed and if it fails, it returns to its original location. The formula is as Equations (21)–(23).

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \tag{21}$$

$$Y = X_{best}(t) - E|JX_{best}(t) - X(t)| \tag{22}$$

$$Z = Y + S \times LF(n) \tag{23}$$

In the formula, F is the fitness function, n is the problem dimension, S is the n dimensional random vector and $LF(n)$ is the Levy flight function expression. LF is a Levy flight function and is calculated using the formula in [33] as Equation (24).

$$LF(n) = s \times \frac{u \times \sigma}{|\nu|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right), \beta = 1.5, s = 0.01 \tag{24}$$

where u and v are random values within (0, 1).

4.2.4. Progressive Rapid Subduction Hard Surrounding

When $|E| < 0.5$ and $rand < 0.5$, because the prey still has a large jumping ability J, the Harris Hawk initiates a fast dive hard surprise. If the attack fails (Y fitness does not improve), a random walk Z is performed and if it fails, it returns to its original location. The formula is as Equations (25)–(27).

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \tag{25}$$

$$Y = X_{best}(t) - E|JX_{best}(t) - X_M(t)| \tag{26}$$

$$Z = Y + S \times LF(n) \tag{27}$$

5. Improved Discrete Harris Hawk Algorithm

HHO achieves the transition from global search to local search based on the decay of the prey energy. The exploration process mainly relies on the best individual information and does not communicate with other individuals, which leads to the decline in population diversity and a slow convergence rate. As the number of iterations increases, the energy of the prey decreases and enters the local development stage. Four different strategies are adopted according to the energy of the prey and the probability of escape.

Figure 5 shows all stages of the improved HHO algorithm, which are described in the following sections.

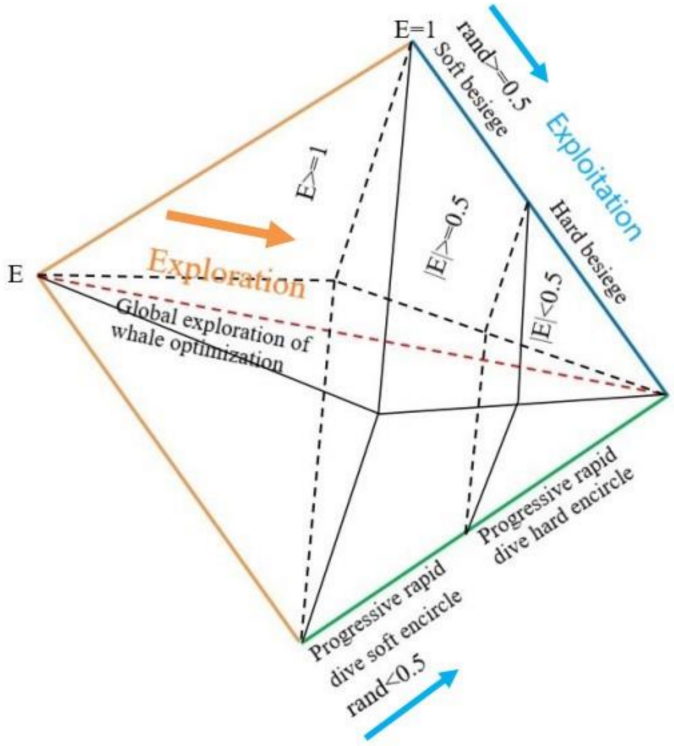


Figure 5. Different stages of improving the Harris Hawk optimization algorithm.

5.1. Initial Population Generation of Dynamic Opposing Learning

In the meta-heuristic optimization algorithms, population initialization is often generated randomly in the search space, which can only guarantee the dispersion of the population in the search space but cannot guarantee the quality of the initial solution. However, the research shows that the quality of initialization will directly affect the convergence speed and accuracy of the algorithm. The randomly generated solutions do not consider whether these solutions satisfy the constraints. An anti-inverse learning method is introduced, which can significantly improve the performance of the algorithm.

Assuming $p(x_1, x_2, \cdots x_n)$ is the solution vector of dimension, the inverse solution $p'(x_1', x_2', \cdots, x_n')$ is calculated as Equation (28).

$$x_i' = x^L + x^U - x_i \tag{28}$$

where x_i represents the initialization population generated randomly; x^L and x^U is the lower and upper limits of variable x_i, respectively.

First, the algorithm generates the original initial population x_i and the reverse initial population x_i' respectively, and then combines the two populations into a new population $x_{new} = \{x_i \cup x_i'\}$. Calculate the fitness value of the new population, and make full competition within the population by using greedy strategy, select the best n individuals as the initialization population. Through this method, the population can approach the optimal solution faster, thus improving the convergence speed of the algorithm.

5.2. The Improved Escape Energy Function

For a group intelligence optimization algorithm, how to better balance its exploration and development ability is particularly important. At the early stage of algorithm evolution, strong global exploration capability is required to maintain the diversity of the population, while at the later stage of algorithm iteration, better development capability is required to ensure that the algorithm can be developed locally. For the HHO algorithm, the escape energy E of prey is an important parameter in the transition from the exploration stage to the development stage of the algorithm. It is used to control the balance between the exploration and development stages. Based on the improved method of convergence factor in reference [35]. In this paper, a nonlinear logarithmic convergence factor escape energy function such as Equation (28).

$$E = 2E_0\left(1 - \ln\left(1 + \frac{t(e-1)}{T}\right)\right) \tag{29}$$

where e is the natural constant, t is the current iteration, T is the maximum iteration number, and E_0 is the initial escape energy. The rate of reduction is faster in the early stage and slower in the later stage. Due to the excellent population diversity of the HHO algorithm in the early iteration period, strong local development capability is required in the later iteration period.

5.3. Discrete Exploration for Whale Optimization

The introduction of a whale optimization algorithm [36] to search for a prey strategy increases diversity and solves the problem that the algorithm has poor global exploration capability. The improved global pollination method is shown by Equations (30)–(32):

$$X(t+1) = X_{rand}(t) - A|2R_3X_{best}(t) - X(t)| \tag{30}$$

$$A = 2ar - a \tag{31}$$

$$a = 2 - \left(\frac{2 \times t}{T}\right) \tag{32}$$

where α is a weight vector, α decreases from 2 linear to 0; R_3 is a random number in [0, 1]. X_{rand} is a randomly selected individual from the t iteration population and r is a random number of $[-1, 1]$ distributions.

To enable exploratory development to deal with discrete problems, the following discrete rules are set.

(1) A new assembly sequence $p_k(x_1, x_2, \cdots x_n)$ is obtained by subtracting assembly sequence $p_i(x_1, x_2, \cdots x_n)$ from $p_j(x_1, x_2, \cdots x_n)$. Considering that the result will be negative, the absolute value is adopted. The calculation formula is as Equation (33).

$$x_k = |x_i - x_j| \tag{33}$$

(2) Array multiplication, corresponding value multiplication, the rules are as follows:

The constraint conditions for blade assembly in this paper are as Equation (34).

$$x_k = x_i \times x_j \tag{34}$$

(3) Arrays are added and corresponding values are added according to the following rules as Equation (35).

$$x_k = x_i + x_j \tag{35}$$

(4) $LF(n)$ is an n-dimensional integer array randomly generated by Levy distribution. The values of the array are generated by the Formula (24). The characteristics of LF are that it generates a small number in most of the time and suddenly jumps to a very large number randomly, thus mimicking the Levy flight behavior of birds. r_i is an n-dimensional array of numbers 0 and 1 that are generated.

The following example shows the modified discrete global exploration rule and illustrates the above discretization process:

$$\begin{aligned}
&X(t) = (16, 17, 23, 9, 7, 6, 4, 10, 12, 13, 15, 27, 14, 11, 8, 18, 19, 24, 3, 22, 28, 1, 5, 25, 20, 21, 2, 26)\\
&X_{best}(t) = (11, 21, 23, 14, 17, 28, 18, 7, 5, 6, 9, 13, 25, 4, 15, 12, 10, 27, 19, 16, 3, 26, 22, 20, 24, 8, 1, 2)\\
&R_3 = (0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0)\\
&|2R_3X_{best}(t) - X(t)| = (16, 25, 23, 19, 27, 6, 32, 4, 2, 1, 3, 27, 36, 3, 8, 6, 19, 24, 3, 10, 28, 51, 39, 25, 20, 21, 0, 26)\\
&A = (0, -1, -1, 0, 0, 0, -1, -1, 0, -1, -1, -1, 0, 0, 1, -1, 1, -1, 0, 0, -1, 0, -1, -1, 1, 0, -1, 1)\\
&X_{rand}(t) = (6, 32, 32, 22, 8, 10, 51, 25, 16, 26, 21, 51, 23, 20, 19, 20, 7, 50, 15, 17, 39, 28, 44, 28, 16, 1, 13, 2)\\
&X(t+1) = X_{rand}(t) - A|2R_3X_{best}(t) - X(t)|\\
&\qquad = (6, 32, 32, 22, 8, 10, 51, 25, 16, 26, 21, 51, 23, 20, 19, 20, 7, 50, 15, 17, 39, 28, 44, 28, 16, 1, 13, 24)
\end{aligned}$$

In the obtained assembly sequence, there are more than the maximum number of blades, duplicate values and missing values, and the assembly sequence needs to meet the constraints, so the final assembly sequence needs to be repaired, the repair steps are as follows:

(1) Statistically identify the values in the array that exceed the maximum number of blades and repeat the values, and change these values to 0. At the same time, record the missing values and the number of values in the sequence.
(2) The missing value is randomly filled into the position of 0 in the array to get a new sequence.
(3) To satisfy the constraints and generate the optimal sequence which meets the conditions, the assembly sequence is updated. The update rule is to exchange the blades which do not meet the conditions randomly until the conditional output is met.

The new assembly sequence after all repairs is as follows:

$$X_i^{t+1} = (6, 5, 16, 22, 8, 10, 2, 25, 27, 26, 21, 18, 23, 11, 19, 9, 7, 20, 15, 17, 4, 28, 3, 12, 14, 1, 13, 24)$$

5.4. Discrete Development Method of Adaptive Weighting

Inspired by particle swarm optimization (PSO), an inertia weight varying with the number of iterations is added to the position update of the Harris Hawk. In the early stage of the algorithm search, the effect of the optimal Harris Hawk position on current individual position adjustment is weakened and the search area of the algorithm in the early stage is enlarged. As the number of iterations increases, when the inertia weight is small, the later search ability of the algorithm is strong, and a fine search can be carried out around the optimal solution. According to the change of update times in the Harris Hawk optimization algorithm, the adaptive inertia weight composed of iterations is as Equations (36) and (37):

$$w(t) = \sin\left(\frac{\pi \times t}{2 \times T} + \pi\right) + 1 \tag{36}$$

$$X'_{best}(t) = w(t) \times X_{best}(t) \tag{37}$$

The adaptive inertia weight has a property of non-linear change between $[0, 1]$. Because of the change characteristics of cos function between $[0, \frac{\pi}{2}]$, the weight is small at the beginning of the algorithm, but the change speed is slightly faster. At the later stage of the algorithm, the value is large, but the change speed will be slow down, which can fully guarantee the convergence of the algorithm.

5.4.1. Discrete Soft Surrounding

When $|E| \geq 0.5$ and $rand \geq 0.5$, the Harris Hawk performs a soft encirclement strategy. Such as Equation (38).

$$X(t+1) = \left(X'_{best}(t) - X(t)\right) - E|JX'_{best}(t) - X(t)| \tag{38}$$

Whereas r_6 is a random number within [0, 1], $J = 2(1 - r_6)$ represents the jumping ability of the prey. The J value varies randomly during each iteration to simulate the nature of the prey's motion.

Discrete processing in global exploration rules is used to apply it to local development rules. The following example illustrates the modified local development rules and procedure flow.

$$\begin{aligned}
&X(t) = (5, 9, 22, 27, 12, 23, 10, 3, 19, 1, 18, 26, 16, 17, 20, 15, 8, 21, 24, 28, 7, 4, 11, 25, 13, 2, 6, 14)\\
&X'_{best}(t) = (16, 2, 25, 7, 19, 27, 4, 17, 10, 15, 26, 28, 5, 6, 8, 21, 20, 23, 24, 22, 18, 3, 1, 13, 9, 12, 14, 11)\\
&J = (2, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 0, 1, 2, 1, 0, 2, 2, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1)\\
&|JX_{best}(t) - X(t)| = (26, 7, 2, 13, 6, 3, 6, 13, 0, 13, 7, 26, 11, 5, 12, 15, 31, 24, 0, 6, 10, 4, 10, 12, 13, 9, 7, 3)\\
&X'_{best}(t) - X(t) = (10, 7, 2, 20, 6, 3, 6, 13, 9, 13, 7, 1, 11, 11, 12, 5, 11, 1, 0, 6, 10, 1, 10, 12, 4, 9, 7, 3)\\
&X_{rand}(t) = (6, 32, 32, 22, 8, 10, 51, 25, 16, 26, 21, 51, 23, 20, 19, 20, 7, 50, 15, 17, 39, 28, 44, 28, 16, 1, 13, 2)\\
&X(t+1) = X'_{best}(t) - X(t) - E|JX'_{best}(t) - X(t)|\\
&\qquad = (10, 1, 0, 9, 1, 0, 1, 2, 9, 2, 1, 19, 2, 7, 2, 7, 13, 18, 0, 1, 2, 2, 2, 2, 6, 1, 1, 0)
\end{aligned}$$

The repair strategy in the improved global exploration rule is adopted to repair the defects. The new assembly sequence after all the repairs is as follows:

$$X_i^{t+1} = (10, 4, 5, 11, 22, 9, 28, 7, 2, 17, 1, 19, 12, 27, 14, 3, 13, 18, 16, 26, 8, 25, 21, 20, 6, 24, 23, 15)$$

5.4.2. Discrete Hard Surrounding

When $|E| < 0.5$ and $rand \geq 0.5$ B, the Harris Hawk performs a hard encirclement strategy. Such as Equation (39).

$$X(t+1) = \left(X'_{best}(t) - X(t)\right) - E|JX'_{best}(t) - X(t)| \tag{39}$$

Discrete processing in global exploration rule is used to apply it to discrete hard surrounding rule. The following example illustrates the modified local pollination rules and procedure.

$$\begin{aligned}
&X(t) = (17, 3, 16, 18, 6, 8, 1, 9, 15, 11, 5, 20, 27, 28, 7, 21, 10, 14, 4, 23, 12, 19, 2, 22, 25, 24, 26, 13)\\
&X'_{best}(t) = (16, 25, 19, 3, 13, 10, 1, 15, 14, 9, 12, 11, 22, 28, 27, 6, 7, 5, 23, 21, 8, 17, 26, 24, 4, 20, 18, 2)\\
&|X'_{best}(t) - X(t)| = (1, 21, 2, 15, 6, 1, 0, 5, 1, 2, 6, 9, 5, 0, 19, 15, 3, 9, 18, 2, 4, 2, 23, 1, 21, 4, 8, 11)\\
&X(t+1) = X'_{best}(t) - E|X'_{best}(t) - X(t)|\\
&\qquad = (15, 18, 18, 1, 11, 9, 0, 13, 13, 8, 10, 8, 20, 27, 21, 1, 6, 2, 17, 20, 6, 16, 19, 23, 2, 18, 15, 1)
\end{aligned}$$

The repair strategy in the improved global exploration rule is adopted to repair the defects. The new assembly sequence after all the repairs is as follows:

$$X_i^{t+1} = (20, 15, 2, 6, 11, 9, 3, 13, 5, 18, 10, 14, 28, 27, 21, 25, 26, 4, 17, 8, 1, 16, 19, 23, 22, 24, 7, 12)$$

5.4.3. Discrete Progressive Rapid Subduction Soft Surrounding

When $|E| \geq 0.5$ and $rand < 0.5$, because the prey still has a large jumping ability J, the Harris Hawk initiates a fast dive soft surprise. If the attack fails (Y fitness does not improve), a random walk Z is performed and if it fails, it returns to its original location. The formula is as Equations (40)–(42).

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \tag{40}$$

$$Y = X'_{best}(t) - E|JX'_{best}(t) - X(t)| \tag{41}$$

$$Z = Y + S \times LF(n) \tag{42}$$

In the formula, F is the fitness function, n is the problem dimension, S is the n dimensional random vector and $LF(n)$ is the Levy flight function expression.

The discretization treatment in the global exploration rule is applied to the local pollination rule. The following example illustrates the modified local pollination rule and procedure flow.

$X(t) = (10, 16, 1, 15, 4, 3, 20, 2, 22, 5, 14, 27, 7, 9, 19, 8, 6, 24, 26, 11, 13, 12, 25, 21, 28, 18, 23, 17)$
$X'_{best}(t) = (12, 14, 19, 3, 17, 6, 18, 11, 24, 1, 8, 13, 5, 16, 15, 21, 9, 20, 4, 7, 28, 27, 2, 22, 25, 26, 10, 23)$
$J = (1, 1, 1, 1, 2, 2, 1, 0, 1, 1, 0, 2, 0, 1, 0, 1, 0, 0, 0, 2, 2, 1, 1, 1, 1, 2, 0, 1)$
$|JX'_{best}(t) - X(t)| = (1, 2, 17, 12, 29, 8, 2, 2, 1, 4, 14, 1, 7, 6, 19, 12, 6, 24, 26, 2, 42, 14, 23, 0, 3, 33, 23, 5)$
$Y = X'_{best}(t) - E|JX'_{best}(t) - X(t)| =$
$(10, 11, 1, 9, 12, 2, 15, 8, 22, 3, 6, 11, 2, 9, 4, 8, 2, 4, 22, 4, 14, 12, 21, 21, 21, 7, 13, 17)$
$F(X(t)) = 2241.460, F(Y) = 8097.92, F(X(t)) < F(Y)$
$S = (0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0)$
$LF(n) = (0, 1, 4, 8, 0, 5, 8, 2, 7, 7, 0, 4, 0, 8, 0, 8, 0, 0, 7, 8, 2, 8, 1, 6, 2, 2, 0, 8)$
$Z = Y + S \times LF(n) = (10, 3, 5, 9, 11, 23, 23, 19, 25, 3, 6, 24, 12, 21, 18, 35, 5, 16, 8, 24, 14, 4, 22, 28, 28, 9, 13, 17)$
$F(Z) = 1406.79, F(X(t)) = 2241.46, F(Z) < F(X(t))$
$X(t+1) = Z = Y + S \times \mathrm{LF} =$
$(10, 3, 5, 9, 11, 23, 23, 19, 25, 3, 6, 24, 12, 21, 18, 35, 5, 16, 8, 24, 14, 4, 22, 28, 28, 9, 13, 17)$

The repair strategy in the improved global exploration rule is adopted to repair the defects. The new assembly sequence after all the repairs is as follows:

$X_i^{t+1} = (10, 2, 9, 27, 11, 24, 15, 19, 25, 5, 6, 28, 12, 21, 18, 20, 23, 16, 8, 26, 14, 4, 22, 3, 7, 1, 13, 17)$

5.4.4. Discrete Progressive Rapid Subduction Hard Surrounding

When $|E| < 0.5$ and $rand < 0.5$, because the prey still has a large jumping ability J, the Harris Hawk initiates a fast dive hard surprise. If the attack fails (Y fitness does not improve), a random walk Z is performed and if it fails, it returns to its original location. The formula is as Equations (43)–(45).

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \tag{43}$$

$$Y = X'_{best}(t) - E|JX'_{best}(t) - X_r(t)| \tag{44}$$

$$Z = Y + S \times LF(n) \tag{45}$$

The discretization treatment in the global exploration rule is applied to the local pollination rule. The process is the same as that in Section 5.4.3, where X_r is the randomly selected individual in the t iteration.

5.5. Genetic Operator Variation

It is known from the construction of HHO that the HHO algorithm will produce the phenomenon that the updated individual will fall into the local optimum during the iteration process, which will affect the optimization ability of HHO. In this paper, the idea of mutation in the genetic algorithm is used to expand the population search space by mutation variation before the algorithm enters the next iteration, which can make the algorithm jump out of local optimum, maintain the diversity of the population, and increase the ability of optimization.

The adaptive variation probability makes the variation probability P_m increase with the number of iterations, which helps the Harris Hawk individual avoid being trapped by the local optimum at the later stage of iteration and improves the convergence accuracy of the algorithm. such as Equation (46).

$$P_m = 0.03 + 0.5\left(1 - \frac{T-t}{T}\right)^2 \tag{46}$$

The probability of variation varies nonlinearly during the iteration process and slows down the increase of the probability of variation at the later stage of iteration to avoid large oscillations and converge the population. The rules for improvement are as follows:

Calculate the variation probability of the variation operator. If the probability is less than the variation probability, randomly select two different locations, exchange their values, and repeat the exchange five times to complete the variation. Otherwise, skip.

5.6. Improving the Procedure of Harris Hawk Algorithm Implementation

Based on the above improvements, an improved Harris Hawk optimization algorithm is proposed, which has the following steps:

Step 1: Initialization of parameters, determination of population M, dimension (part number) n, the maximum number of iterations N_iter;

Step 2: Initial population generation is carried out based on dynamic opposing learning and greedy strategy, and fitness value of individual population is calculated, optimum fitness value and corresponding assembly sequence are saved;

Step 3: Calculate the escape energy E by Formula (29). If $|E| \geq 1$ goes to step 4, no, go to step 5;

Step 4: Global exploration, updating individual population according to Formula (30);

Step 5: Local development, if $rand < 0.5$ goes to Step 6, no, then goes to Step 11;

Step 6: If $|E| < 0.5$ goes to step 7, no, go to step 9;

Step 7: Calculate Y according to the Formula (44). If $F(Y) < F(X)$ is updated to Y, turn to Step 12. No, turn to Step 8;

Step 8: Calculate Z according to the Formula (45). If $F(Z) < F(X)$ is updated to Step 12 of Z, no, do not update Step 12 of Z;

Step 9: Calculate Y according to the Formula (41). If $F(Y) < F(X)$ is updated to Y, turn to Step 12. No, turn to Step 10;

Step 10: Calculate Z according to the Formula (42). If $F(Z) < F(X)$ is updated to Step 12 of Z, no, do not update Step 12 of Z;

Step 11: If $|E| < 0.5$ is updated according to Formula (39), no, according to Formula (38);

Step 12: If $rand < p_b$ carries out genetic variation, no, skip;

Step 13: Are all population individuals updated? No, go to Step 3;

Step 14: Calculate the fitness value of individuals who update the population, find the optimum fitness value and corresponding solution in the updated population, and update the optimum fitness value and corresponding is assembly sequence;

Step 15: To determine whether the end condition is met, yes, output the optimum value and the corresponding solution and exit the program, no, go to Step 3.

The flow chart for improving the Harris Hawk optimization algorithm is shown in Figure 6.

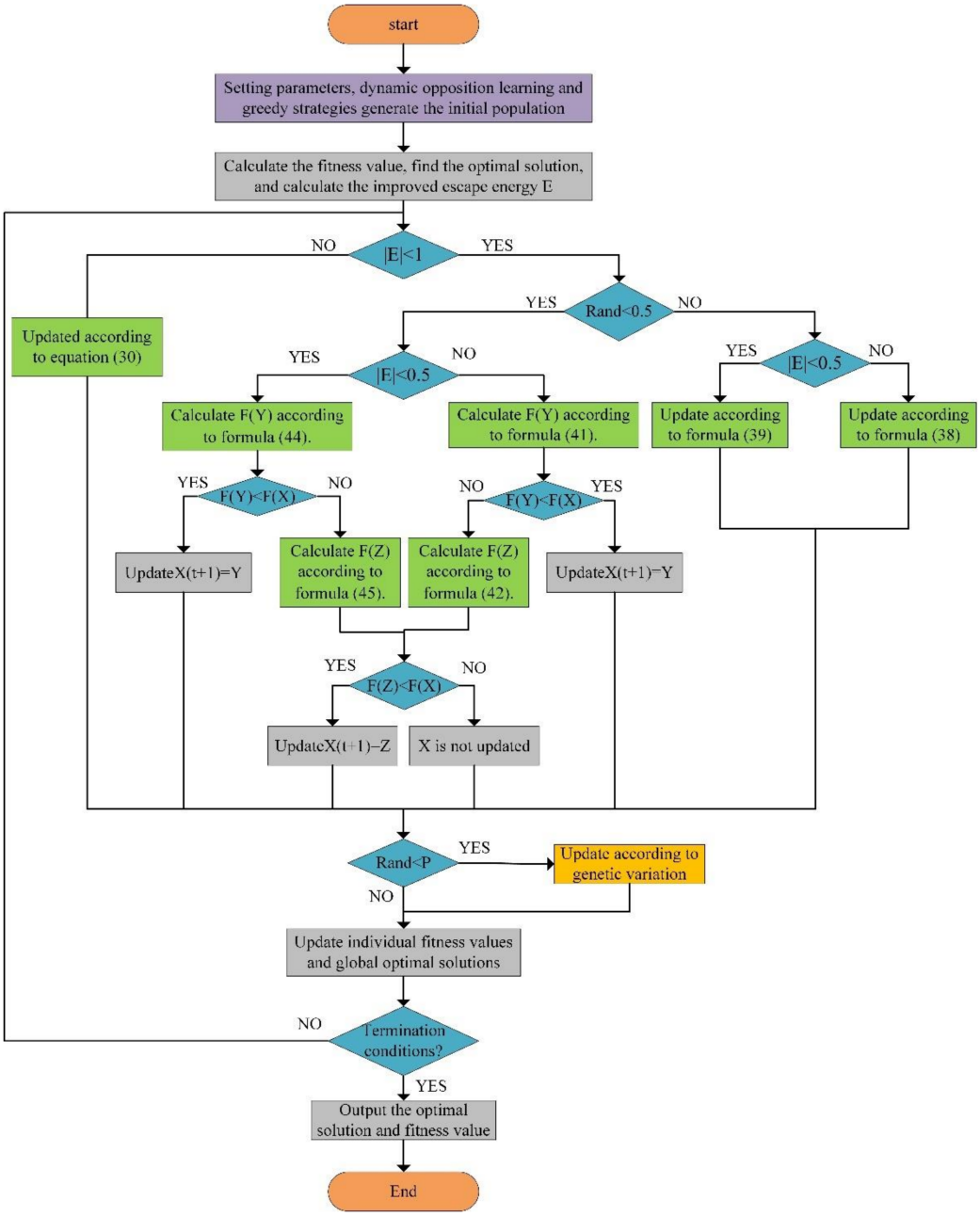


Figure 6. Improved flow chart for Harris Hawk algorithm.

6. Instance Validation

In this paper, the assembly of rotor blades of a certain stage fan of an aero-engine is taken as an example for verification. To verify the effectiveness of the algorithm, we use C++ programming on Visual Studio 2017 software to implement the algorithm in this paper. The operating environment is CPU frequency 2.9 GHz, memory 16 GB, Windows10 64 bit operating system.

6.1. Single Stage Disc and Single Stage Blade Assembly

The assembly sequence of a stage blade of an aero-engine is optimized. The blades are numbered from minimum to maximum according to the mass moment data. The blade data are shown in Table 2.

Table 2. Blade data.

Blade Number	Mass Moment (g·mm)	Blade Number	Mass Moment (g·mm)
1	273,380	15	274,100
2	274,780	16	275,760
3	276,260	17	275,760
4	275,040	18	275,880
5	276,100	19	275,180
6	276,120	20	276,120
7	275,660	21	274,840
8	275,140	22	272,860
9	274,960	23	276,460
10	275,420	24	275,080
11	275,300	25	273,760
12	274,300	26	273,580
13	275,840	27	276,460
14	276,180	28	274,120

For the simulation run of the algorithm in this paper, the parameters are set as follows: initial population m = 100, 500 iterations. Figure 7 shows the convergence diagram of the improved pollination algorithm in 500 iterations.

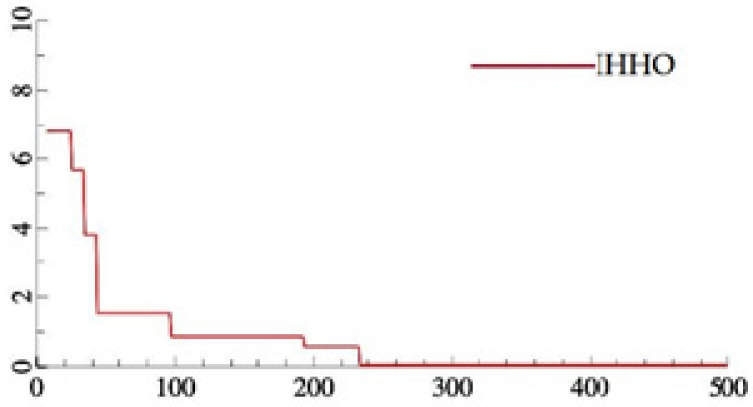


Figure 7. Iterative Graph of Improved Harris Hawk Optimization Algorithm.

For general or unreasonable sorting, the rotor unbalance can reach 6000 g·mm or more. According to Figure 6, we can see that the imbalance becomes smaller and smaller as the number of iterations of the algorithm increases, and the minimum imbalance is 0.0517 g·mm when the number of iterations reaches 234. The corresponding sorting results are shown in Table 3, and the corresponding assembly sequence diagram is shown in Figure 8.

The improved Harris Hawk algorithm is compared with the particle swarm optimization algorithm, genetic algorithm, and basic Harris Hawk algorithm. The above algorithms have been independently simulated for 200 iterations. Table 4 lists the optimal sequence of aero-engine single-stage blade assembly obtained by the four algorithms, and gives the unbalance calculated by the optimal assembly sequence. The convergence graphs of average fitness and global optimal fitness of the four algorithms are shown in Figures 9 and 10, and the assembly sequence is shown in Figure 11.

Through the comparative analysis of the data in Table 3, it can be found that the minimum unbalance of the optimal assembly sequence given by the improved Harris Hawk algorithm proposed in this paper is 0.024, which is 91.75% lower than the particle swarm optimization algorithm, 99.82% lower than the genetic algorithm, and 83.39% lower than the Harris Hawk algorithm. By comparing the global optimal fitness convergence graph of the algorithm, it is found that the improved eagle algorithm is obviously superior

to the other three algorithms. First, the optimal assembly sequence found by the improved Harris Hawk algorithm has the lowest unbalance and the fastest convergence speed in the early stage of the algorithm. Compared with PSO and GA algorithms, IHHO and HHO iterations converge faster, and IHHO finds a better fitness value than HHO, that is, IHHO finds a better assembly sequence. This shows that the performance of the IHHO algorithm proposed in this paper is better than the other three algorithms.

Table 3. Optimum blade assembly sequence found by IHHO.

Assembly Position Number	Blade Number	Mass Moment (g·mm)	Assembly Position Number	Blade Number	Mass Moment (g·mm)
1	1	273,380	15	24	275,080
2	3	276,260	16	23	276,460
3	5	276,100	17	18	275,880
4	7	275,660	18	9	274,960
5	28	274,120	19	2	274,780
6	15	274,100	20	19	275,180
7	4	275,040	21	10	275,420
8	21	274,840	22	26	273,580
9	20	276,120	23	12	274,300
10	14	276,180	24	16	275,760
11	17	275,760	25	6	276,120
12	22	272,860	26	25	273,760
13	11	275,300	27	13	275,840
14	8	275,140	28	27	276,460

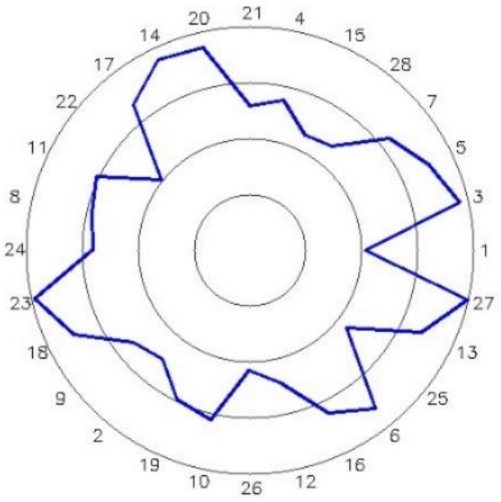


Figure 8. Installation drawing of optimal blade assembly sequence searched by IHHO.

Table 4. Optimum blade assembly sequence and unbalance data for each algorithm.

Algorithm	Assembly Sequence	Unbalance (g·mm)
PSO	2-17-11-19-9-27-23-13-18-7-14-22-10-20-25-6-24-12-5-26-3-1-4-16-21-28-15-8	0.2473
CA	1-25-23-3-13-21-22-14-16-9-6-27-26-28-10-19-17-18-4-2-15-12-7-5-8-11-24-20	11.1504
HHO	20-4-12-27-22-17-3-19-2-23-8-7-5-26-15-18-28-11-6-24-9-14-1-10-13-25-21-16	0.1228
IHHO	23-9-2-26-1-10-12-28-4-16-27-20-8-7-15-21-22-24-19-14-13-6-11-18-5-17-3-25	0.0204

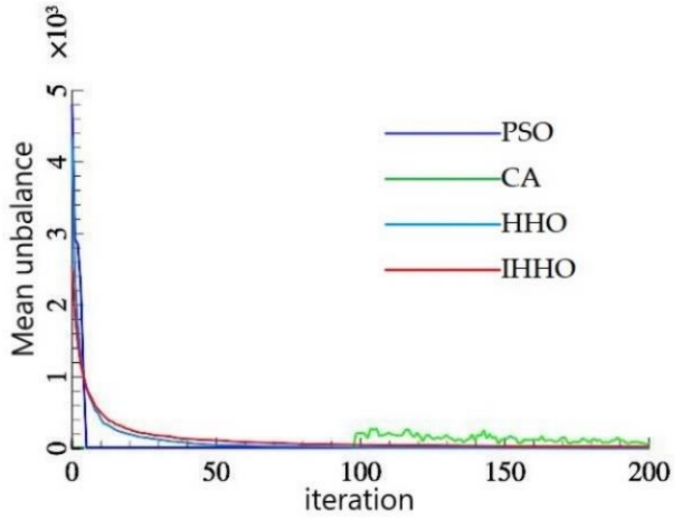


Figure 9. IHHO average unbalance iteration chart.

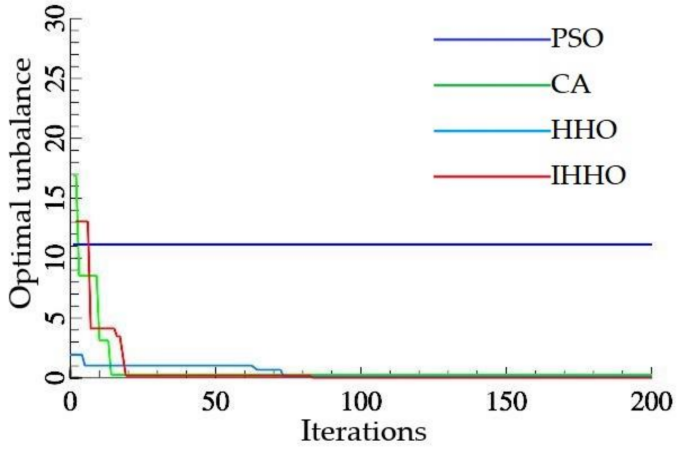


Figure 10. IHHO optimal unbalance iteration chart.

6.2. Rotor Multistage Disc Phase Assembly

The blade data of the first, second, third, fourth, and fifth stages of an aero-engine rotor were used as an example to verify the optimization effect of this algorithm applied to the phase assembly of rotor multistage discs. The blade data of each stage are shown in Table 5.

Particle swarm optimization, genetic algorithm, HHO optimization algorithm, and IHHO optimization algorithm are used for blade assembly optimization of each stage of rotor. The results are shown in Table 6.

Analysis

Table 5 compares the results of the improved Harris Hawk optimization algorithm with particle swarm optimization, genetic algorithm, and the original Harris Hawk optimization algorithm. The optimal imbalance of improved Harris Hawk optimization algorithm is reduced by 79.71%, 99.48%, and 54.92%, respectively. The IHHO proposed in this paper is much lower than other algorithms in finding the optimal sequence and is obviously superior to the other three algorithms in finding the optimal solution.

The assembly sequence table (counterclockwise) of the first stage, second stage, third stage, fourth stage, and fifth stage disc blades optimized by the improved Harris Hawk algorithm is shown in Table 7. The phase angle of the multistage blade assembly is optimized by improving Harris Hawk algorithm, and the total residual unbalance force,

The optimized blade layout obtained by improving Harris Hawk algorithm is shown in Figure 12.

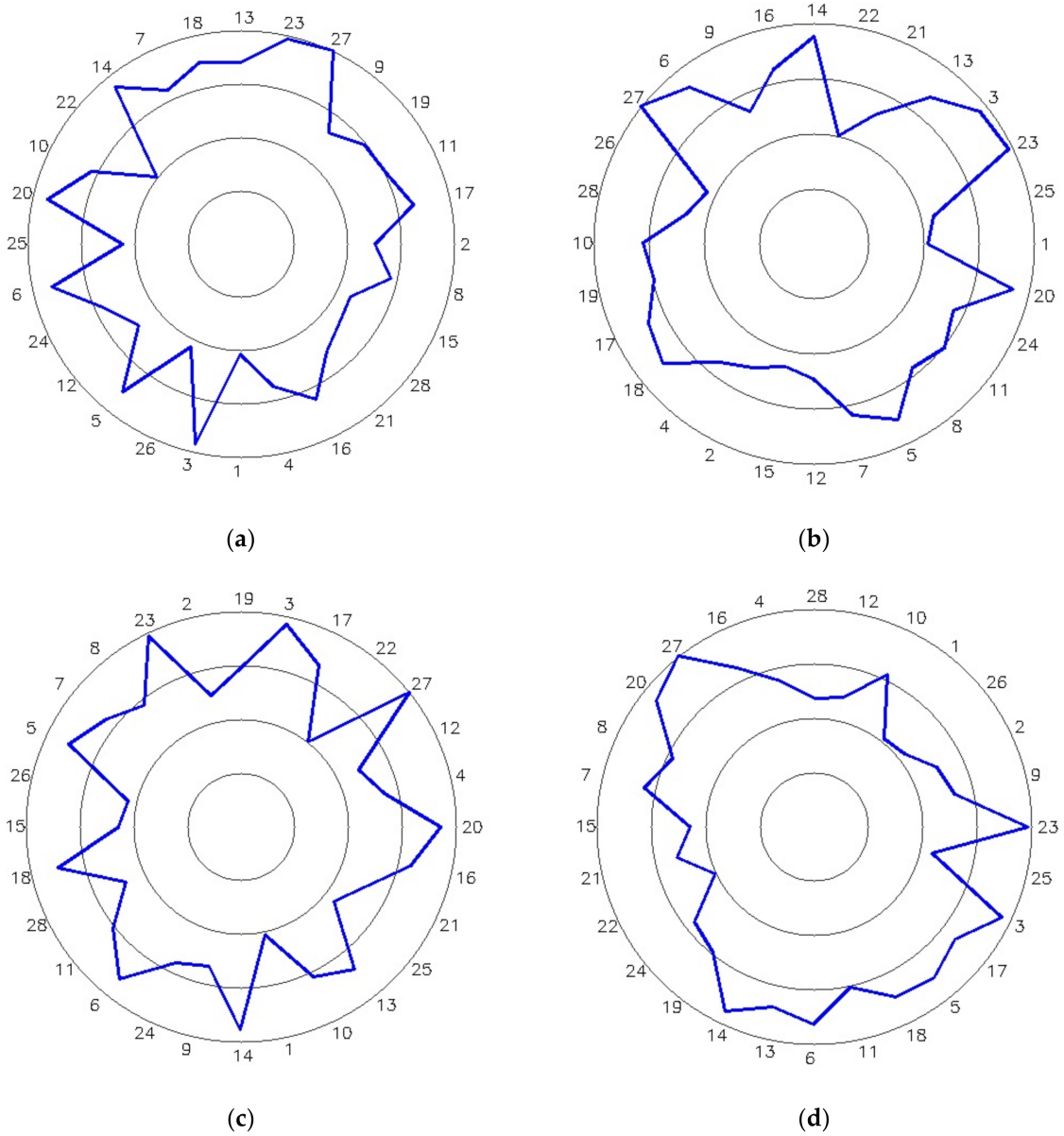


Figure 11. Blade mass moment arrangement under 4 optimization algorithms: (**a**) PSO; (**b**) CA; (**c**) HHO; (**d**) IHHO.

After the blade layout of each stage is completed, the assembly phase angle between discs of each stage is optimized by improving the Harris Hawk algorithm. The residual unbalanced force vectors of discs at all levels are superposed to make the residual unbalanced force and torque of the whole rotor meet the requirements. Unbalanced moment and blade assembly phase angle before and after optimization are compared, as shown in Table 8.

By analyzing the data in Table 8, before optimization, the greatest residual unbalance force calculated by using the phase angle of the first-, second-, third-, and fifth-stage blade sets is 0.1058, and the unbalance moment is 0.6348. The optimal installation angle of the first stage blade group after IHHO optimization is 247° relative to the disk, the second stage blade group is 92° relative to the disk, the third stage blade group is 261° relative to the disk, the fourth stage blade group is 0° relative to the disk, the greatest residual unbalanced force is 0.0167, which is 84.22% lower than that before optimization, and the unbalanced

moment is 0.0124, which is 98.05% lower than that before optimization. As shown above, it is workable to optimize the phase angle of multistage disk blade assembly by using the improved Harris Hawk algorithm.

Table 5. Blade data of all rotors.

Number of Blade Stage	Mass Moment of Disc Blades Arranged by Number (g·mm)
First-stage blade	12341,12767,13234,13600,13469,13624,12978,12858,14062,13364, 12605,13445,13981,12927,13261,12791,13495,13442,13527,12336, 14091,12504,12602,12453,12592,13882,13521,13016,14018,12395, 12347,12426
Second-stage blade	8941,9367,9234,9000,10069,9024,9578,9458,9462,9364,9805,9445, 9381,8927,9261,9391,9495,10042,8927,9536,10091,9104,9202,9053, 9192,9282,9521,9616,9418,9595,9547,9026,9271,9638,9569,9612, 9367,9999,9135,9194
Third-stage blade	6741,6767,7434,6800,7469,7224,6978,7258,7262,7164,6805,6845, 6781,6727,7061,7191,7295,7442,6727,7336,7091,6904,7402,6853, 6992,7082,7321,7016,7218,7395,7347,6826,7071,7038,6969,7412, 6767,7399,6935,6994,7403,7311,6822,7433,6773,7364
Fourth-stage blade	5241,5667,5534,5200,5369,5424,5678,5558,5662,5664,5405,5345, 5481,5527,5661,5691,5695,5642,5527,5636,5591,5304,5602,5353, 5492,5582,5621,5416,5418,5595,5647,5426,5471,5238,5569,5612, 5367,5499,5235,5594,5403,5511,5522,5533,5373,5364,5341,5411, 5453,5568,5247,5344
Fifth-stage blade	4041,4167,4034,4100,4269,4124,4078,4258,4262,4164,4005,4245, 4181,4027,4061,4191,4295,4242,4027,4036,4291,4204,4002,4153, 4292,4082,4021,4116,4218,4095,4047,4126,4071,4138,4069,4112, 4167,4199,4235,4294,4203,4111,4122,4033,4273,4164,4141,4211, 4053,4268,4047,4044,4262,4057,4237,4259

Table 6. Comparisons of optimization algorithms for rotor blade assembly at various stages.

Number of Blade Stage	Optimum Unbalance of Various Algorithms (g·mm)			
	PSO	CA	HHO	IHHO
First-stage blade	0.3760	11.6661	0.0496	0.0356
Second-stage blade	0.1620	7.4786	0.0634	0.0158
Third-stage blade	0.1685	12.2214	0.0462	0.0239
Fourth-stage blade	0.0572	2.6407	0.0342	0.0210
Fifth-stage blade	0.0303	2.8651	0.0611	0.0095

Table 7. Improve the IHHO assembly sequence of each stage of disc blades.

Number of Blade Stage	Assembly Sequence	Unbalance
First-stage blade	28-2-4-22-23-5-11-17-27-3-24-21-18-12-14-13-20-8-25-31-6-30-10-9-16-29-32-1-26-19-15-7	0.0356
Second-stage blade	21-38-30-14-18-4-6-29-23-7-13-24-26-10-39-11-31-17-25-3-5-9-27-16-35-32-40-2-12-15-33-34-20-22-36-28-37-1-19-8	0.0158
Third-stage blade	17-27-30-15-9-46-12-21-45-7-4-25-5-2-36-10-42-28-29-11-31-20-3-39-44-35-32-14-19-38-41-6-43-13-26-22-33-18-40-34-23-37-1-24-16-8	0.0239
Fourth-stage blade	8-26-36-6-4-9-49-46-52-14-2-37-44-10-29-45-38-21-41-40-25-39-34-16-20-1-18-33-51-15-23-22-31-17-42-43-32-28-19-13-3-50-12-11-47-27-5-30-7-48-24-35	0.0210
Fifth-stage blade	45-49-50-32-10-41-25-22-18-37-29-35-38-42-44-28-19-17-2-47-16-48-4-39-53-7-3-21-14-9-43-5-30-8-36-40-1-34-54-23-46-33-52-51-6-27-24-12-31-11-26-55-20-15-13-56	0.0095

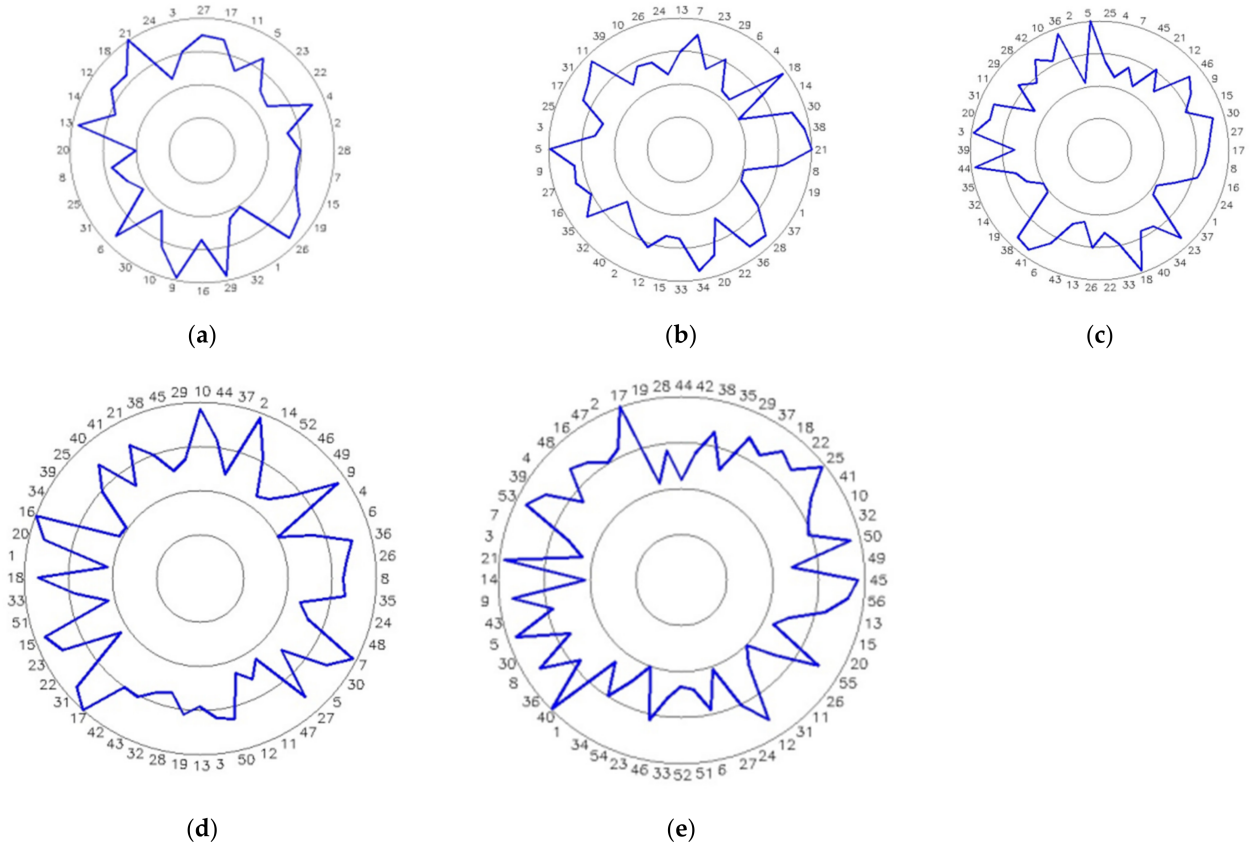


Figure 12. Assembly sequence of engine rotor disc blades at each stage: (**a**) First stage disc blade assembly sequence; (**b**) Assembly sequence of the second stage disc blade; (**c**) Assembly sequence of the third stage disc blade; (**d**) Assembly sequence of fourth stage disc blade. (**e**) Assembly sequence of fifth stage disc blade.

Table 8. IHHO-optimized aero-engine rotor assembly phase angle.

		Before Optimization	After Optimization
Input	Remaining unbalance force of the first-stage blade	0.0356	0.0356
	Remaining unbalance force of the second-stage blade	0.0158	0.0158
	Remaining unbalance force of the third-stage blade	0.0239	0.0239
	Remaining unbalance force of the fourth-stage blade	0.0210	0.0210
	Remaining unbalance force of the fifth-stage blade	0.0095	0.0095
Output	Overall residual unbalance force	0.1058	0.0167
	Overall residual unbalance moment	0.6348	0.0124
	Installation phase angle	0/0/0/0/0	247/92/261/0/115

7. Conclusions

In this paper, we propose a discrete improved Harris Hawk optimization algorithm to optimize the imbalance of engine rotor assembly. The improved Harris Hawk optimization algorithm adopts the initial population generation based on dynamic anti-vertical learning and the improved escape from energy function and introduces the exploration mechanism of the whale optimization algorithm in the global exploration to expand the search space. The adaptive weight strategy is introduced in the development to improve the optimization efficiency and convergence speed of the algorithm. Before entering the next iteration, the mutation in the genetic algorithm is used to avoid premature algorithm. First, based on the mass moment of a single blade, the diagonal mass moment difference is introduced as a constraint to establishing a single-stage blade and disk assembly model. The improved Harris Hawk algorithm is used to find the optimal assembly sequence, and repeated operations

are performed to complete the assembly sequence of the remaining single-stage blades and corresponding disks. For the assembly of the engine rotor, based on the imbalance amount of multiple disc blades after assembly obtained by the improved algorithm in this paper, a multi-stage disc rotor assembly model is established. The improved Harris Eagle optimization algorithm is used to find the optimal assembly phase, and the optimal unbalanced force and the assembly phase of the multi-stage disc corresponding to the unbalanced force distance are obtained. The main results of this study are as follows:

1. For the optimization of single stage disc and blade assembly unbalance, the improved Harris Hawk optimization algorithm is compared with the particle swarm optimization, the genetic algorithm, and the Harris Hawk optimization algorithm, and the unbalance of the optimal blade assembly sequence is reduced by 91.75%, 99.82%, and 83.39%, respectively;
2. For the assembly of each stage disc blade of the engine rotor, the results of the improved Harris Hawk optimization algorithm, particle swarm optimization algorithm, genetic algorithm, and Harris Hawk optimization algorithm are compared and analyzed. The optimal unbalance of the improved Harris Hawk optimization algorithm is reduced by 79.71%, 99.48%, and 54.92%, respectively. The proposed IHHO algorithm is obviously superior to the other three algorithms in finding the optimal solution;
3. The phase angle of the multistage blade assembly is optimized by improving the Harris Hawk optimization algorithm. The total residual unbalanced force, unbalanced torque, and blade assembly phase angle of each stage before and after optimization are compared. The optimized assembly phase angles are 247°, 92°, 261°, 0°, 115°. The total residual unbalanced force and moment of assembly are 0.0167 and 0.0124, and the total residual unbalanced force and moment of assembly after optimization are 84.22% and 98.05% lower than those before optimization.

This research is effective in developing an improved Harris Hawk optimization algorithm to improve the assembly quality of aero-engine rotor and blade. In order to improve the assembly quality in the future work, the assembly optimization model will be improved in the future, and more realistic factors will be considered. In the future, it is planned to test the assembly optimization method developed in this study in the actual aero-engine assembly, and compare it with the method in engineering, so as to adapt to the aero-engine rotor assembly in the enterprise in the future.

Author Contributions: Conceptualization, B.Z. and H.L.; methodology, B.Z.; software, D.S.; validation, B.Z., Y.Y. and H.L.; formal analysis, S.L.; investigation, H.L. and B.Z.; resources, H.L.; data curation, S.L. and B.Z.; writing—original draft preparation, B.Z.; writing—review and editing, H.L.; visualization, B.Z.; supervision, H.L.; project administration, H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by Jiangxi Natural Science Foundation (No. 20201BBE51002).

Data Availability Statement: Not applicable.

Acknowledgments: Thanks to Liu Shun processed the experimental data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, Y.M.; Zhang, M.W.; Sun, C.Z.; Hu, M.; Chen, D.Y.; Liu, Z.W.; Tan, J.B. A method to minimize stage-by-stage initial unbalance in the aero engine assembly of multistage rotors. *Aerosp. Sci. Technol.* **2019**, *85*, 270–276. [CrossRef]
2. Chen, D. Research on Vibration Suppression Method of Aero-Engine Rotor Based on Stack Assembly Optimization. Master's Thesis, Harbin Institute of Technology, Harbin, China, 2019.
3. Lin, Z. *Research on Rotor Assembly Method Based on Axial Compression*; Harbin Institute of Technology: Harbin, China, 2018.
4. Zhang, H.; Wang, M.; Li, Z.; Zhou, J.; Zhang, K.; Ma, X.; Wang, M. Semi-Physical Simulation of Fan Rotor Assembly Process Optimization for Unbalance Based on Reinforcement Learning. *Aerospace* **2022**, *9*, 342. [CrossRef]
5. Abdullah, M.A.; Ab, R.M.F.F.; Ghazalli, Z. Optimization of assembly sequence planning using soft computing approaches: A review. *Arch. Comput. Methods. Eng.* **2019**, *26*, 461–474. [CrossRef]

6. Liu, L.K.; LI, L.L. Optimization of Assembly Sequence Planning of Turbine Low-pressure Rotor Blades Based on the Improved Simulated Annealing Algorithm. *J. Phys. Conf. Ser.* **2021**, *1939*, 012098.
7. Zhao, L.P.; Li, B.H.; Chen, H.R.; Zhao, L.P. An assembly sequence optimization oriented small world networks genetic algorithm and case study. *Assem. Autom.* **2018**, *38*, 387–397. [CrossRef]
8. Li, L.L.; Chen, K.; Gao, J.M.; Liu, J.K. Research on Optimizing Selection and Optimizing Matching Technologies of Aeroengine Fan Rotor Blades. *Shock Vib.* **2021**, *2021*, 5595535. [CrossRef]
9. Li, L.L.; Chen, K.; Gao, J.M.; Liu, J.K.; Gao, Z.Y.; Dai, H.W. Research on optimizing-assembly and optimizing-adjustment technologies of aero-engine fan rotor blades. *Adv. Eng. Inform.* **2022**, *51*, 101506. [CrossRef]
10. Zhu, M.Y.; Li, M.Q.; Wen, X.; Li, D.Y. Intelligent optimization of turbine rotor blade assembly sequence. *J. Aerosp. Power.* **2017**, *32*, 2536–2543.
11. Pitsoulis, L.S.; Pardalos, P.M.; Hearn, D.W. Approximate solutions to the turbine balancing problem. *Eur. J. Oper. Res.* **2001**, *130*, 147–155. [CrossRef]
12. Wang, L.; Wang, T.G.; Luo, W.Y. Improved non-dominated sorting genetic algorithm (NSGA)-II in multi-objective optimization studies of wind turbine blades. *Appl. Math. Mech. Engl. Ed.* **2011**, *32*, 739–748. [CrossRef]
13. Tu, J.B.; Li, Z.; Ge, H.T.; Liu, L.; Liu, H.H. Multi-objective optimization of rotor-stack assembly based on geometric algebra theory. *Acta Aeronaut. Astronaut. Sin.* **2021**, *42*, 395–405.
14. Zhang, Z.H.; Guo, J.K.; Hong, J.; Sun, Y.H. Application study intelligent algorithms for prediction and phase optimization of assembly eccentricity of aero-engine high pressure rotor. *J. Xi'an Jiaotong Univ.* **2021**, *55*, 47–54.
15. Chen, Y.; Cui, J.; Sun, X. A Vibration Suppression Method for the Multistage Rotor of an Aero-Engine Based on Assembly Optimization. *Machines* **2021**, *9*, 189. [CrossRef]
16. Zhang, Y.; He, L.; Yang, J.; Wang, J.; Zhu, G. Vibration control of tie rod rotors with optimization of unbalanced force and unbalanced moment. *IEEE Access* **2020**, *8*, 66578–66587. [CrossRef]
17. Zhao, B.; Yuan, Q.; Li, P. Improvement of the vibration performance of rod-fastened rotor by multi optimization on the distribution of original bending and unbalance. *J. Mech. Sci. Technol.* **2020**, *34*, 83–95. [CrossRef]
18. Chen, Y.; Cui, J.; Sun, X. An Assembly Method for the Multistage Rotor of An Aero-Engine Based on the Dual Objective Synchronous Optimization for the Coaxality and Unbalance. *Aerospace* **2021**, *8*, 94. [CrossRef]
19. Wang, L.; Sun, C.; Tan, J.; Zhao, B.; Wan, G. Improvement of location and orientation tolerances propagation control in cylindrical components assembly using stack-build assembly technique. *Assem. Autom.* **2015**, *35*, 358–366. [CrossRef]
20. Chen, Y.; Cui, J.; Sun, X. An unbalance optimization method for a multi-Stage rotor based on an assembly error propagation model. *Appl. Sci.* **2021**, *11*, 887. [CrossRef]
21. Sun, C.Z.; Liu, Z.W.; Liu, Y.M.; Wang, X.M.; Tan, J.B. An adjustment method of geometry and mass centers for precision rotors assembly. *IEEE Access* **2019**, *7*, 169992–170002. [CrossRef]
22. Zhi, L.L.; Ying, W. Optimum Design for Balance in Multi-Disk Rotor Installation Based on Genetic Algorithm. *J. Vib. Meas. Diagn.* **2008**, *28*, 4.
23. Chen, Y.; Cui, J.W.; Sun, X.; Cui, S.H. Research on multistage rotor assembly optimization methods for aeroengine based on the genetic algorithm. *Complexity* **2021**, *2021*, 8847690. [CrossRef]
24. Wang, H.; Rong, Y.; Xiang, D. Mechanical assembly planning using ant colony optimization. *Comput. Aided Des.* **2014**, *47*, 59–71. [CrossRef]
25. Said, Z.; Nguyen, T.H.; Sharma, P.; Li, C.; Ali, H.M.; Ahmed, S.F.; Van, D.N.; Truong, T.H. Multi-attribute optimization of sustainable aviation fuel production-process from microalgae source. *Fuel* **2022**, *324*, 124759. [CrossRef]
26. Sharma, P.; Le, M.P.; Chhillar, A.; Said, Z.; Deepanraj, B.; Cao, D.N.; Bandh, S.A.; Hoang, A.T. Using response surface methodology approach for optimizing performance and emission parameters of diesel engine powered with ternary blend of Solketal-biodiesel-diesel. *Sustain. Energy Technol. Assess.* **2022**, *52*, 102343. [CrossRef]
27. Hendricks, E.S.; Gray, J.S. pyCycle: A Tool for Efficient Optimization of Gas Turbine Engine Cycles. *Aerospace* **2019**, *6*, 87. [CrossRef]
28. Shan, H.B.; Zhou, S.H.; Sun, Z.H. Research on assembly sequence planning based on genetic simulated annealing algorithm and ant colony optimization algorithm. *Assem. Autom.* **2009**, *29*, 249–256. [CrossRef]
29. Chen, S.F.; Liu, Y.J. An adaptive genetic assembly-sequence planner. *Int. J. Comput. Integr. Manuf.* **2001**, *14*, 489–500. [CrossRef]
30. Zhang, H.Y.; Liu, H.J.; Li, L.Y. Research on a kind of assembly sequence planning based on immune algorithm and particle swarm optimization algorithm. *Int. J. Adv. Manuf. Technol.* **2014**, *71*, 795–808. [CrossRef]
31. Zhang, Z.; Han, Y. Discrete sparrow search algorithm for symmetric traveling salesman problem. *Appl. Soft Comput.* **2022**, *118*, 108469. [CrossRef]
32. Ab Rashid, M.F.F. A hybrid Ant-Wolf Algorithm to optimize assembly sequence planning problem. *Assem. Autom.* **2017**, *37*, 238–248. [CrossRef]
33. Zhang, B.; Lu, H.Y.; Song, H.Q.; Liu, S.; Yang, Y.C.; Sang, D.D. Overall planning of aero-engine assembly based on improved pollination Algorithm. *J. Aerosp. Power.* **2022**, *37*, 1–12.
34. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [CrossRef]

35. Guo, W.; Xu, P.; Dai, F.; Zhao, F.; Wu, M. Improved Harris hawks optimization algorithm based on random unscented sigma point mutation strategy. *Appl. Soft Comput.* **2021**, *113*, 108012. [CrossRef]
36. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.