IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2019, accepted March 5, 2019, date of publication April 4, 2019, date of current version May 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2909087

Optimization Design and Application of Active Disturbance Rejection Controller Based on Intelligent Algorithm

CHAOHAI KANG[1,2], SIQI WANG[1,2], WEIJIAN REN[1,2], YANG LU[3,4], AND BOYU WANG[1,2]

[1]College of Electrical and Information Engineering, Northeast Petroleum University, Daqing 163318, China
[2]Heilongjiang Provincial Key Laboratory of Networking and Intelligent Control, Northeast Petroleum University, Daqing 163318, China
[3]College of Information Technology, Heilongjiang Bayi Agricultural University, Daqing 163319, China
[4]Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou 350108, China

Corresponding author: Weijian Ren (renwj@126.com)

This work was supported in part by the Provincial Science Foundation of Heilongjiang under Grant F2018004, and in part by the Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University) Opening Fund under Grant MJUKF201729.

ABSTRACT The parameter tuning optimization design is realized for an active disturbance rejection controller (ADRC) in combination with the improvement of the existing swarm intelligence algorithm. Taking the optimization design and application of ADRC as an example, this paper is focused on investigating the improvement of the hybrid algorithm composed of fish swarm algorithm and particle swarm optimization algorithm and its application in parameter tuning of ADRC. The main contents are as follows. First, the parameters that need to be tuned are determined based on the composition and principle of the ADRC. The module building technology of S-function is adopted to create the module library of ARDC in terms of the modular construction idea and a complete simulation example of ADRC is built in Simulink. Second, the parameters are improved according to the proposed hybrid algorithm composed of the artificial fish swarm algorithm and the standard particle swarm optimization algorithm, and the control performance is tested by the MATLAB simulation of the ADRC whose parameters are optimized by using the algorithm. Finally, the flight attitude control of the unmanned aerial vehicle (UAV) is taken as an application example, and the fixed-wing UAV is selected as the research object. Through the analysis of the experimental results, the effectiveness of the optimized design is verified for the ADRC in the attitude control of the UAV.

INDEX TERMS Active disturbance rejection controller, attitude control, hybrid algorithm of the fish swarm and particle swarm, parameter optimization tuning, simulated flight simulation.

I. PREFACE

The active disturbance rejection controller (ADRC) was proposed by researcher Han Jingqing in 1998 [1]. Nowadays, the ADRC technology [2] has been investigated and applied in many areas such as motor control [3], precision machining [4], chemical process [5], spacecraft control [6], aircraft flight control [7], robot control [8], and a lot of results have been achieved. However, the application effect of ADRC is not ideal in practical applications due to its disadvantages of many parameters and tuning difficulties.

Group intelligence is an intelligent behavior that is performed by a single intelligent individual in any form and participates in coordinated behavior. It is also a computing technology based on the laws of biological group habits.The group intelligence algorithm is based on mathematics and is developed by simulating these natural processes, such as artificial fish swarm algorithm, particle swarm algorithm, and ant colony algorithm.Nowadays, the group intelligence algorithm developed by group intelligence has covered multiple target optimization, data screening and clustering, robot behavior control, simulation and system identification.Therefore, the exploration and improvement of intelligent intelligence algorithms with intelligent characteristics has become a hot research direction.

In addition to solving the function optimization problem, the artificial fish swarm algorithm has also been applied to many other fields and certain results have been achieved

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

on fault diagnosis [9], image detection and recognition [10], path planning [11], network coverage optimization [12], etc. However, there is big blindness of global search in the early stage and a slow convergence rate in the later stage of the artificial fish swarm algorithm. Although it can find the approximate region where the optimal value is located, it is difficult to find the optimal value with high precision. In this regard, scholars have proposed some improved methods, for example, the conjugate gradient method has been introduced in the literature [13] to reduce randomness and enhance individual local optimization ability and so on. Despite that the overall performance of the algorithm has been improved by some proposed improvements, there is still a problem of poor balance between global search and local search, which results in unstable optimization performance.

The particle swarm optimization algorithm is to solve the optimization problem of the objective function, whose idea is changed from something desirable in the foraging behavior of the flock. This algorithm has the advantages of less parameters, easy implementation, a fast calculation speed and a strong ability of local search, etc. It has been applied in many fields, such as power system optimization [14], pattern recognition [15], target tracking [16]. Nevertheless, the algorithm also has the disadvantages of poor performance of global search, being easy to fall into local optimum, and frequent stagnation when it iterates. The concepts of commutator and exchange order have been proposed in the literature [17], and a particle swarm optimization algorithm has been constructed with a special arithmetic form. These improvements have perfected the optimization results of the particle swarm optimization algorithm, but have not given an effective solution to the problem that the algorithm is easy to fall into the local optimum.

In this paper, the parameter tuning optimization problem [18] in the optimization design of the active disturbance rejection controller is taken as the research background. For the defects of ADRC that there are many parameters to be tuned and it is not easy to adjust the parameters, the active disturbance rejection control technology is used as the theoretical basis. The influence from parameters of each part of the ADRC on the control performance is analyzed, and the optimization performance of the artificial fish swarm algorithm [19] and the standard particle swarm optimization algorithm is investigated. Then, a kind of improved hybrid algorithm of fish swarm and particle swarm optimization is proposed based on elite Gaussian learning. Thus, the parameter tuning problem is solved in the optimization design for the active disturbance rejection controller. It is realized that the parameter adjustment is fast and accurate, and the control performance of the active disturbance rejection controller is improved. Finally, the optimization design of the ADRC is applied to the flight attitude control of the UAV, and a simulation of a test flight is carried on for the UAV in conjunction with MATLAB/Simulink. This proves that the improved active disturbance rejection controller can solve the practical problem of control application well.

The main work and contributions of this paper are stated as follows:

1. Aiming at the advantages and disadvantages of basic artificial fish swarm algorithm and particle swarm optimization algorithm, an improved hybrid algorithm of fish population and particle swarm is proposed. Three improvements are made to the hybrid algorithm: 1) the distribution of the initial population is optimized through uniform initialization; 2) grouping by frog jumping algorithm, and using different search strategies for excellent individuals and general individuals in the group, in order to improve the purpose and efficiency of search; and 3) the improved elite Gaussian learning is introduced to improve the accuracy of the final result. Then through the standard function test comparison and the independent impact analysis of each improved part, it is verified that the improved hybrid algorithm is feasible and effective.

2. Taking the application of the active disturbance rejection controller as an example, the improved hybrid algorithm is used to optimize the parameters, and the improved active disturbance rejection controller is applied to the attitude control of the UAV. Take the control problem of the drone's flight attitude as an example. The ADRC-based UAV attitude control simulation platform is built in Simulink, which verifies that the improved active disturbance rejection controller has better control effect in UAV attitude control.

Therefore, in combination with the improvement by the existing swarm intelligence algorithm, the parameter tuning optimization design is realized for active disturbance rejection controller, which will lay a good foundation for the active disturbance rejection controller to solve practical problems. At the same time, it has great theoretical research significance and application reference value.

II. COMPOSITION PRINCIPLE OF THE ACTIVE DISTURBANCE REJECTION CONTROLLER

The active disturbance rejection controller is composed of a tracking differentiator (TD), an extended state observer (ESO) and a nonlinear state error feedback (NLSEF) control law. The structure diagram of the active disturbance rejection controller is shown in Fig. 1.

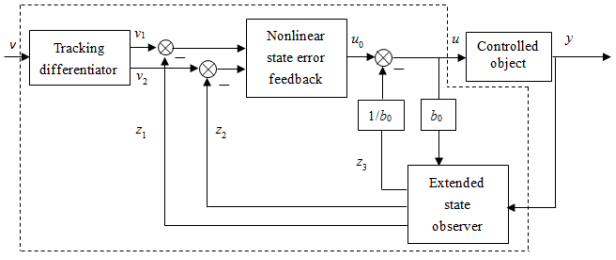


FIGURE 1. Structure diagram of the active disturbance rejection controller.

The expression of the tracking differentiator is as follows:

$$\begin{cases} e = x_1 - v \\ fh = fhan(e, x_2, r, h_0) \\ x_1 = x_1 + Tx_2 \\ x_2 = x_2 + Tfh \end{cases} \tag{1}$$

where v is the input of the system. In order to prevent the occurrence of chattering in digital calculation, $fhan(x_1, x_2, r, h)$ is used as the time optimal control function of the discrete system. T is the sampling period, r and h_0 are variable parameters, the tracking speed is determined by the speed factor r, and h_0 is the filtering factor which filters the noise. According to the introduction of the differential tracker, it can be seen that the parameters r and h_0 are to be tuned. The speed factor r and the filtering factor h_0 are tuned to find the best values of them, thus the fast tracking function as well as the differential action of noise reduction and extraction can be realized of the differential tracker.

The expression of extended state observer is:

$$\begin{cases} \dot{z}_1 = z_2 - \beta_{01}\varepsilon_1 \\ \dot{z}_2 = z_3 - \beta_{02} fal(\varepsilon_1, \alpha_1, \delta) + bu \\ \dot{z}_3 = -\beta_{03} fal(\varepsilon_1, \alpha_2, \delta) \end{cases} \tag{2}$$

where $\varepsilon_1 = z_1 - y$. $fal(\varepsilon_1, \alpha_i, \delta)(i = 1, 2)$ is the continuous power function. It can be seen from the analysis of the above equation that there are mainly three parameters β_{01}, β_{02} and β_{03} that need to be tuned. Considering the gains of the error feedback, the larger the values of them, the smaller the hysteresis of the disturbance estimation and the faster the convergence process. However, if the parameter value is selected to be too large, then it will cause serious oscillation of the observer, and the ability to suppress noise will also decrease along with it, which has a great influence on the convergence speed of the extended state observer. Also, there exist mutual restrictions among these three parameters.

The expression of nonlinear state error feedback control law is:

$$\begin{cases} e_1 = & v_1 - z_1 \\ e_2 = & v_2 - z_2 \\ u_0 = & \beta_1 fal(e_1, \alpha_1, \delta) + \beta_2 fal(e_2, \alpha_2, \delta) \end{cases} \tag{3}$$

where β_1 and β_2 are variable parameters that need to be adjusted to satisfy the control requirements. In order to suppress high frequency oscillation, it is generally selected that $g_i(\varepsilon_1) = fal(\varepsilon_1, \alpha_i, \delta)$, where $fal(\varepsilon_1, \alpha_i, \delta)$ is a continuous power function with the following form:

$$fal(\varepsilon_1, \alpha_i, \delta) = \begin{cases} |\varepsilon_1|^{\alpha} sgn(\varepsilon_1) & |\varepsilon_1| > \delta, \\ \varepsilon_1 / \delta^{1-\alpha} & \varepsilon_1 \le \delta, \end{cases} \delta > 0. \tag{4}$$

It can be seen from the equation that the three parameters β_1, β_2 and b_0 need to be tuned.

In summary,eight parameters are required for parameter tuning.

III. IMPROVED HYBRID ALGORITHM OF FISH POPULATION AND PARTICLE SWARM OPTIMIZATION BASED ON ELITE GAUSSIAN LEARNING

A. THE IMPROVED PART OF THE ALGORITHM

Firstly, for the random initialization part of the fish swarm algorithm, a method of uniform initialization of dimensional space is used in this paper to initialize the population. Uniform initialization reduces unnecessary constraints and the number of loop operations, and improves the initial search efficiency while ensuring the uniform distribution of the initial population in space. Its operation method is as follows:

(1) According to the total number of population individuals, the population density coefficient ρ in the subspace and the upper bound X_{MAX} and the lower bound X_{MIN} of the population individuals in each dimension component, the space where the entire population is located is uniformly divided into K subspaces.

(2) In each subspace, M population individuals are randomly generated, and the Euclidean distance matrix is established for the M individuals by equation (5). According to the matrix and the population density coefficient ρ in the subspace, if $D_{ij} < \rho * \|X_{MAX} - X_{MIN}\|$, which means that the two individuals x_i and x_j are too close, then one of the individuals is removed according to the Euclidean distance matrix and a new individual is regenerated to replace the old one.

$$D_{ij} = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \tag{5}$$

where D_{ij} represents the spatial Euclidean distance between any two individuals $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ and $x_j = (x_{j1}, x_{j2}, \ldots, x_{jn})$ of the M population individuals, and n is the dimension of the population individuals.

The $K \times M$ population individuals constitute the initial population, and the population initialization is completed.

Secondly, in this paper, guiding towards the direction with a certain probability [20] being utilized in combination with dynamic population collaboration [21], [22] in order to strengthen the global optimization ability of the fish swarm algorithm part, which shows good feedback on each iteration result. The adopted grouping mode follows the leapfrog algorithm. At the same time, different position update equations are used for excellent individuals and general individuals in the group, as well as dynamic adjustment strategies for visual field and step size. Adaptive functions of the power function type are used for excellent individuals to adjust the visual field and step size, and the perturbation variable of global optimal value is introduced to the position update equation [23]. Linear function type adjustment is utilized for general individuals, and the perturbation variable of the optimal value in the group is introduced to the position update equation. The specific implementation method is as follows.

(1) Grouping mode

For all N artificial fish in all groups of the initial fish population, they are sorted in terms of fitness value from small

to large. The ordered artificial fish individuals are divided into m groups, and each group contains p artificial fish satisfying $N = m \times p$. The grouping method is: The first artificial fish is placed in the first group, the second artificial fish is placed in the second group, the mth artificial fish is placed in the mth group by analogy, and the $m + 1$th artificial fish is placed in the first group, and so on until the division is completed. Sort and group according to this method after each iteration.

(2) Dynamic adjustment of visual field and step size

For all individuals in each group, the top 20% of the individuals are defined as excellent individuals, and the rest are general individuals. For excellent individuals, the adjustments of their visual field and step size are shown in equations (6) and (7):

$$visual = visual_{\max} \times iter^{\frac{\log(visual_{\min}/visual_{\max})}{\log(gen_{\max})}} \quad (6)$$

$$step = visual \times A \quad (7)$$

For the remaining individuals in the group, the adjustments of their visual field and step size are shown in equation (8) and equation (7).

$$visual = visual_{\max} - \frac{iter \times (visual_{\max} - visual_{\min})}{gen_{\max}} \quad (8)$$

where $visual_{\max}$ represents the initial value of the visual field, $visual_{\min}$ represents the end value of the visual field, $iter$ represents the current iteration number, $gen_{\max}$ represents the maximum number of iterations, and $A \in [0.5, 1]$ is a random number.

(3) Position update equation of individuals in the group

Improvements are made to the location update equation with foraging behavior and random behavior. Taking the foraging behavior as an example, its position update equation for excellent individuals and the update equation for other individuals are shown in equation (9).

$$X_{next} = X_i + Step \times \frac{X_j - X_i}{\|X_j - X_i\|} + R \times Step \times \frac{X_G - X_i}{\|X_G - X_i\|} \quad (9)$$

where X_i is the current position state of the artificial fish, X_j is a randomly selected state position within its visual field, $Step$ is the step size, X_G is the position state of the optimal artificial fish, and R is the disturbance influence factor used to adjust the influence from optimal value of the global or the group on the moving direction. For excellent individuals, X_G is the position state of the global optimal artificial fish. For other individuals, X_G is the positional state of the optimal artificial fish in the group. The equation for random behavior is similar to it.

Finally, in order to solve the problem of optimal value stagnation and low precision of the final result when the particle swarm optimization algorithm is iterative, a kind of improved elite Gaussian learning is adopted in this paper to jump out of the state of stagnation, thus the accuracy of the final result is improved [24]. The Gaussian learning mode is shown in equation (10) and equation (11):

$$P^d = P^d + (X_{\max} - X_{\min}) \times Gaussian(\mu, \sigma^2) \quad (10)$$

$$d = random(1, D) \quad (11)$$

where P represents the position state of the current optimal individual, d represents a random dimension of the individual, $X_{\max}$ and $X_{\min}$ are the upper and lower bounds of the dimensional component, μ is the mean of the Gaussian distribution, σ^2 is the variance of the Gaussian distribution, and D is the total dimension.

However, all of the above usage modes have shortages, and the main representations are that the large randomness of the variation result causes the low success rate of variation; and when learning is done for the optimal individual, too much blind learning is unstable to the increase of convergence accuracy and affects the running speed of the algorithm.

In view of the above deficiencies, the following improvements from two aspects are made in this paper on the basis of the existing elite Gaussian learning:

(1) Adjusting the timing of the call and the applicable individuals of the elite Gaussian learning. When the global optimal value has no change for 3 consecutive generations, the elite Gaussian learning is performed on the global optimal individual and the top 10% of the best individuals in this iteration. The impact from multiple learning is reduced on the running speed of the overall algorithm while expanding the learning object.

(2) Conducting the directional learning for elite groups. First, Gaussian learning is performed one by one on all the dimensions of the individual, the results of each dimension are summarized and sorted, and the first d_{best} dimensions are selected as the learning directions. Then, the T-dimensional Gaussian learning is conducted for the d_{best} dimensions of the individual to take the optimal value. If the optimal value is better than the global optimal one, then the optimal value is made to replace the global optimal one, otherwise it is not retained, and this method is also adopted to deal with other elite individuals. Through the determination of the learning direction, the uncertainty in the random learning process is reduced, and the execution efficiency of the elite Gaussian learning is improved.

B. ALGORITHM ANALYSIS AND HYBRID ALGORITHM STEPS

In summary, the steps of the improved hybrid algorithm of fish swarm and particle swarm optimization based on elite Gaussian learning are as follows:

Step 1: Set the parameters in the hybrid algorithm: The initial population number N, the number of groupings m, the initial value of the visual field $visual_{\max}$, the end value of the visual field $visual_{\min}$, the number of attempts $try\ number$, the maximum number of iterations gen_{afsa} of the fish swarm part, and the maximum number of iterations gen_{pso} of the particle swarm optimization part, and so on.

Step 2: Perform the population location initialization according to the improvements of Section III-A.

Step 3: Calculate the fitness of the population and sort the groups.

Step 4: For the outstanding individuals occupying the top 20% of each group and the general individuals, conduct behaviors such as clustering, rear-end and foraging in accordance with the corresponding dynamic adjustment strategy of the visual field and step size and the improved position update equation in Section III-A.

Step 5: Determine whether the global optimal value is less than 10^{-4} or the number of iterations reaches the end condition of the maximum number of iterations. If it is satisfied, the global optimal value and the first 20% individuals of each group in the bulletin board are used as the initial particles of the particle swarm optimization algorithm part, and Step 6 is executed. Otherwise, continue with Step 3.

Step 6: Update the particles according to the update equation of particle swarm optimization to find the global optimal value.

Step 7: Determine whether the global optimal value has no change for 3 consecutive generations. If it is satisfied, then the improved elite Gaussian learning is conducted for the global optimal value and the top 10% of the best individuals in this iteration according to Section III-A. Otherwise, continue with Step 6.

Step 8: Judge whether the end condition of the maximum number of iterations is reached. If it is satisfied, then the final global optimal value is output. Otherwise, continue with Step 6.

The algorithm flow chart is shown in Fig. 2.

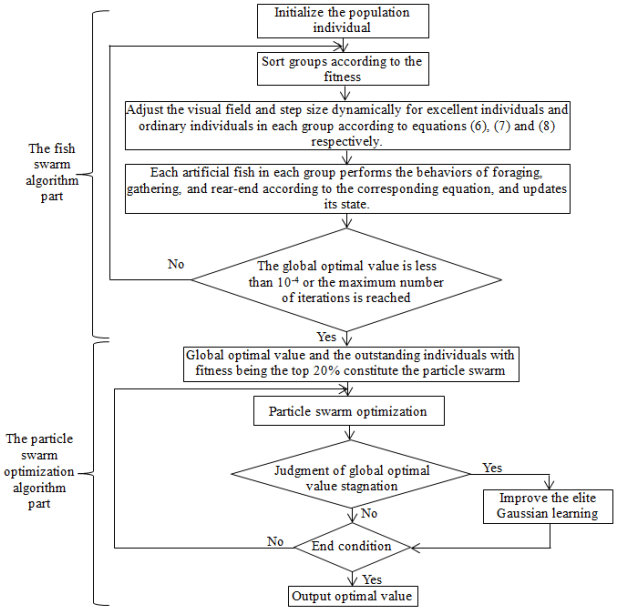


FIGURE 2. Hybrid algorithm flow chart.

IV. OPTIMIZATION DESIGN OF ACTIVE DISTURBANCE REJECTION CONTROLLER BASED ON IMPROVED HYBRID ALGORITHM

A. OPTIMIZATION DESIGN FLOW OF ACTIVE DISTURBANCE REJECTION CONTROLLER BASED ON THE HYBRID ALGORITHM

Combined with Section 1, it is known that there are eight parameters of the ADRC that need to be tuned, i.e. r, h_0, β_{01}, β_{02}, β_{03}, β_1, β_2, b_0. By consulting a large number of literatures and conducting simulation analysis, it is found that the parameters β_{01}, β_{02}, β_{03}, β_1, β_2 play a decisive role in the performance of the active disturbance rejection controller. Therefore, these five parameters are selected to be tuned by the usage of an optimized intelligent algorithm, which can avoid excessive computational optimization and enable ADRC to obtain better control effects. The system design for its parameter optimization is shown in Fig. 3.

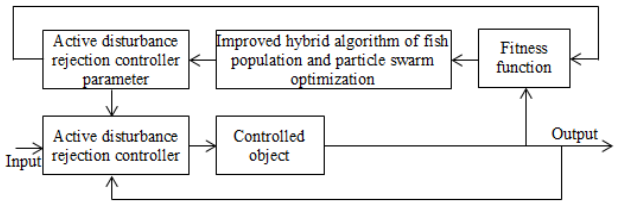


FIGURE 3. Parameter optimization design of active disturbance rejection controller based on improved hybrid algorithm.

The specific steps of parameter optimization for the hybrid algorithm are:

Step 1: Initialize the parameters to be tuned. Determine the parameters such as the value range of the parameters $\{\beta_{01}, \beta_{02}, \beta_{03}, \beta_1, \beta_2\}$ of the active disturbance rejection controller which are to be tuned, the size of the initial population, the maximum number of iterations, and the initial value and the end value of the visual field.

Step 2: Initialize the population according to Section III-A.

Step 3: Using the created active disturbance rejection control technology module library, selecting the corresponding module to build the simulation platform of the active disturbance rejection controller, determining the fitness value corresponding to each initial individual, and sorting and grouping the population.

Step 4: For the outstanding individuals occupying the top 20% of each group and the general individuals, conducting behaviors such as clustering, rear-end and foraging in accordance with the corresponding dynamic adjustment strategy of the visual field and step size and the improved position update equation in Section III-A.

Step 5: Determine whether the performance index of global optimal value is satisfied or the number of iterations reaches the end condition of the maximum number of iterations. If it is satisfied, the global optimal value and the first 20% individuals of each group in the bulletin board are used as the initial particles of the particle swarm optimization algorithm part, and Step 6 is executed. Otherwise, continue with Step 3.

Step 6: Update the particles according to the particle swarm optimization update equation to find the global optimal value.

Step 7: Judge whether the global optimal value has no change for 3 consecutive generations. If it is satisfied, then the improved elite Gaussian learning is conducted for the global optimal value and the top 10% of the best individuals in this iteration according to Section III-A. Otherwise, continue with Step 6.

Step 8: Decide whether the end condition of the maximum number of iterations is reached. If it is not satisfied, continue with Step 6 until the maximum number of iterations is satisfied. If it is satisfied, then the control parameter is output which corresponds to the final global optimal value.

B. SIMULATION AND SIMULATION ANALYSIS

In the Simulink simulation environment of ADRC, a step signal with a set value of 3 is used as the input of the system. During the setting of the algorithm parameters, the maximum number of iterations of all algorithms is adjusted to 50 times, where it is selected that $gen_{afsa} = 15$ and $gen_{pso} = 35$ in the hybrid algorithm of this paper. The population size is 50, and the dimension of population individuals is 5. That is, five parameters $\{\beta_{01}, \beta_{02}, \beta_{03}, \beta_1, \beta_2\}$ are optimized. According to the debugging experience, the value ranges of β_{01}, β_{02}, β_{03}, β_1, β_2 are set respectively as [0, 200], [0, 500], [0, 4000], [0, 500], [0, 100]. For other parameters of the active disturbance rejection controller, the compensation coefficient b_0, the velocity factor r of the TD part and the filtering factor h_0 are taken as 1, 1 and 0.01.

The parameters of the ADRC optimized by the algorithm of this paper compared with the other three algorithms are summarized in Table 1. The response curves to the input step signal are shown in Fig. 4 of the ADRC whose parameters are optimized by the four algorithms. The corresponding performance indicators are shown in Table 2, and the fitness iteration curves are shown in Fig. 5.

TABLE 1. Summary table of ADRC parameters optimized by four algorithms.

Algorithm name	β_{01}	β_{02}	β_{03}	β_1	β_2
PSO	200.62	415.58	3000	420.11	7.78
AFSA	130.23	174.79	2700	435.23	44.18
PSO-FSA	150.73	392.52	3000	500	97.35
Algorithm of this paper	61.11	348.30	969.55	464.81	89.33

TABLE 2. Comparison of step response performance indicators of ADRC after optimization of four algorithms.

Algorithm name	$\sigma(\%)$	$t_p(s)$	$t_s(s)$	$t_r(s)$	$e(\infty)(\%)$
PSO	4.90	1.13	2.38	0.91	2.31
AFSA	1.52	1.71	1.71	1.59	0.51
PSO-FSA	1.21	0.64	1.34	0.60	0.23
Algorithm of this paper	0.78	0.62	0.96	0.59	0.11

By comparing the simulation results of the four algorithms, it can be seen that under the same step input signal

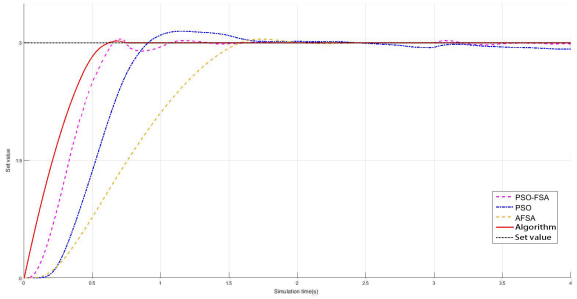


FIGURE 4. Comparison curves of step response.

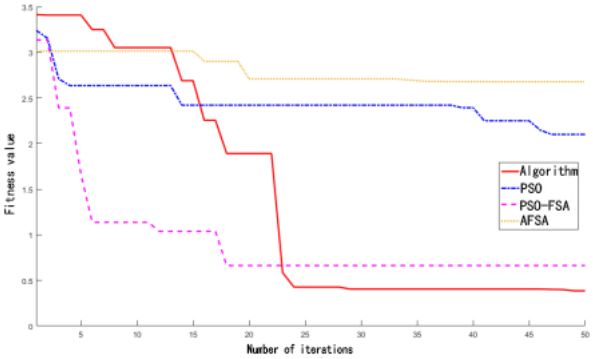


FIGURE 5. Comparison curves of fitness iteration.

and external disturbance, the ADRC can respond to system instructions quickly whose parameters are optimized and tuned by using the hybrid algorithm of this paper and the PSO-FSA algorithm. However, compared with the PSO-FSA algorithm, the ADRC, whose parameters are tuned by adopting the algorithm of this paper, has shorter rise time and smaller overshoot, and this indicates that it has better dynamic performance. At the same time, the steady state can also be reached more quickly of the ADRC with the algorithm of this paper, and the oscillation phenomenon does not occur after the stability with the disturbance from the outside, which indicates that it has good anti-interference ability and steady state performance. The controllers that use PSO and AFSA algorithms for parameter tuning have relatively long rise time, relatively long time to reach the steady state and relatively large overshoots. The oscillation phenomenon occurs in the ADRC after it reaches the steady state which is influenced by the external disturbance, and it is known that its dynamic performance and steady state performance are relatively poor compared with the former controllers. It can also be seen from the fitness iteration curve of Fig. 5 that the smaller the fitness function value is, the better the control effect of the corresponding controller is, whose parameters are tuned by using it. The convergence speed and final precision of the proposed algorithm in this paper are obviously better than the other three algorithms, and there exists no phenomenon of optimal value iteration stagnation. It can be seen that the

improvement of the algorithm in Section 2 still has a good effect in practical applications.

V. UAV ATTITUDE SIMULATION TEST AND ANALYSIS USING OPTIMIZED DESIGN OF ADRC

The structure diagram of the UAV attitude control system is shown in Fig. 6.

Refer to [25] to select the transfer function and gain of the corresponding steering gear and compensation network. The Simulink simulation design of the three attitude angles is depicted in Figs. 7-10.Fig. 7 and Fig. 8 are to verify the applicability of the optimization parameters and the robustness of the algorithm when the simulation conditions change. In the absence of basic difficulties, the simulations of the remaining two attitude angles will obtain similar results, which are hence omitted.

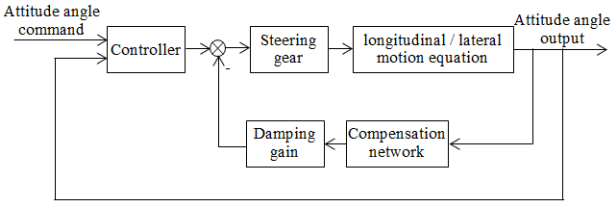


FIGURE 6. Structure chart of the attitude control system.

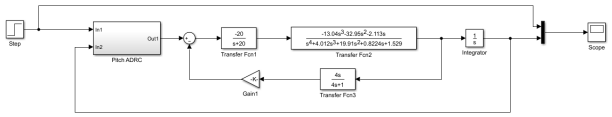


FIGURE 7. Simulink simulation of pitch angle control.

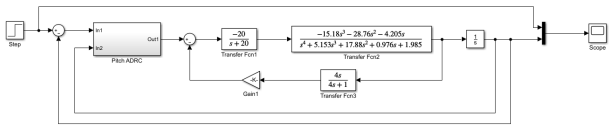


FIGURE 8. Simulink simulation of pitch angle control.

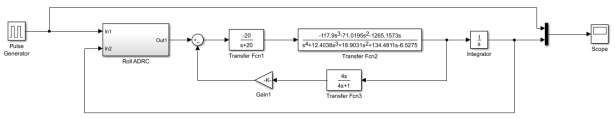


FIGURE 9. Simulink simulation of roll angle control.

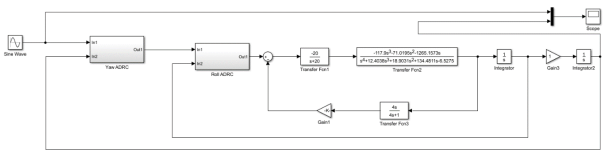


FIGURE 10. Simulink simulation of yaw angle control.

Firstly, based on the built-up simulation example of Simulink UAV attitude, the ADRC parameters to be tuned for the three attitude angle control channels are optimized respectively according to the method in Section IV-A. The parameter tuning of the PID controller participating in the comparison is in a similar way. The relevant parameter settings of the hybrid algorithm are consistent with the experiments in Section IV-B. According to the debugging experience, some parameters are given as follows. The value ranges of β_{01}, β_{02} and β_{03} of the ESO part of ADRC are set as [0, 3000]. The value ranges of β_1 and β_2 in NSEFL part are set as [0, 500]. The compensation coefficient b_0, speed factor r of the TD part and the filtering factor h_0 are respectively 1, 1 and 0.01. The value ranges of K_P, K_I and K_D of the PID controller part are [0, 20], [0, 1] and [0, 1]. The parameters optimized by the hybrid algorithm are shown in Tables 3-4.

TABLE 3. ADRC parameter table optimized by the hybrid algorithm.

Structure name	Parameter	Yaw angle control	Roll angle control	Pitch angle control
ESO	β_{01}	1661.11	1942.86	967.17
	β_{02}	248.30	404.28	313.99
	β_{03}	2969.55	924.53	1655.23
NSEFL	β_1	264.81	84.50	87.78
	β_2	389.33	487.18	445.42

TABLE 4. PID controller parameter table optimized by the hybrid algorithm.

Structure name	Parameter	Yaw angle control	Roll angle control	Pitch angle control
PID	K_P	4.13	1.93	4.97
	K_I	0.43	0.33	0.84
	K_D	0.85	0.08	0.06

Using the optimizing tuning results in Tables 3-4 for the three attitude ADRC parameters and PID parameters, the ADRC and PID are set respectively, and then independent simulation tests are conducted on the pitch angle, roll angle and yaw angle of the UAV. During the test, it is assumed that the speed of the UAV is constant, and three kinds of typical signals, i.e. step signal, square wave signal and sinusoidal signal, are respectively selected as the input signals. The step signal is selected in the pitch angle test to simulate the action of maintaining the climbing or descending posture of the UAV. The square wave signal is selected in the roll angle test to simulate the actions of horizontal attitude switching and returning of the UAV. The sinusoidal signal is selected in the yaw angle test to simulate the action of swinging the rudder to balance the airflow interference and maintain the flight path during the level flight of the UAV. At the same time, sinusoidal interference with amplitude and frequency of 1 is added in each channel to simulate external disturbance. After the simulation test, the respective responses of the pitch angle, the roll angle and the yaw angle are shown in Figs. 11-14.

It can be seen from the comparative experiments of simulationsfigures (Figs. 10-12) that compared with the

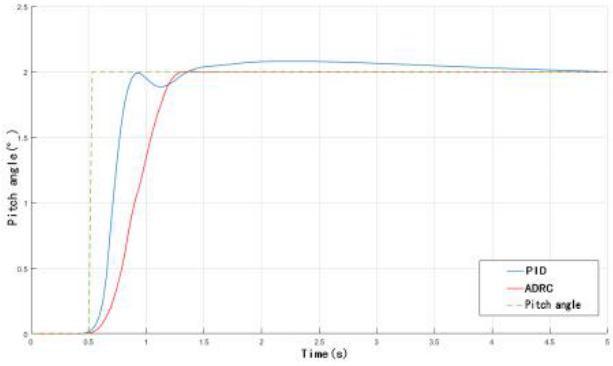


FIGURE 11. Comparison of the pitch angle response curves.

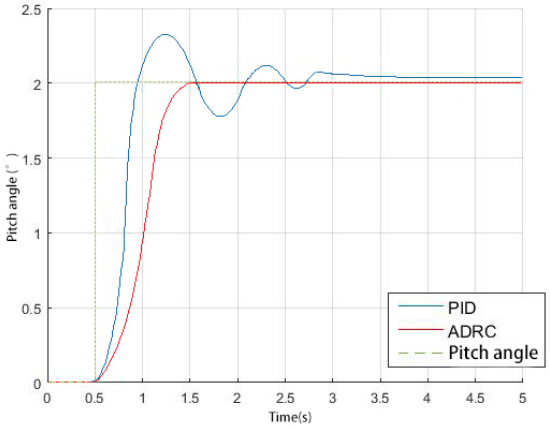


FIGURE 12. Comparison of the pitch angle response curves.

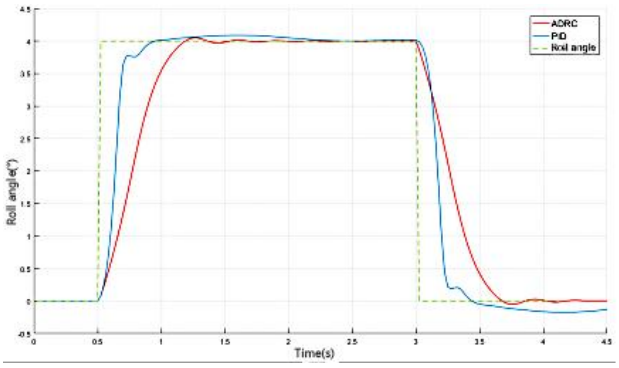


FIGURE 13. Comparison of roll angle response curves.

TABLE 5. Comparison of the performance index in pitch angle response curve.

Controller	$\sigma(\%)$	$t_p(s)$	$t_s(s)$	$t_r(s)$
PID	4.05	2.26	1.27	1.46
ADRC	0.28	1.46	1.26	1.40
Symbolic note: $\sigma(\%)$–Overshoot, $t_p(s)$–Peak time, $t_s(s)$–Adjust time, $t_r(s)$–Rise Time.				

PID controller, the response curves of the pitch angle, the roll angle and the yaw angle using the ADRC can track the changes of the respective command signals well. Although

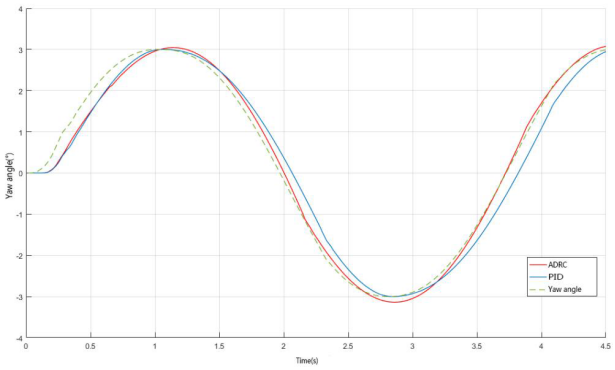


FIGURE 14. Comparison of yaw angle response curves.

there is a shorter rise time when using the PID controller, the overshoot is larger. In contrast, the dynamic response process using the ADRC is about 1.3 seconds and the overshoot is much smaller than the PID controller. At the same time, for the disturbance existing in each channel, when conducting the attitude control by adopting the ADRC, the steady state can be quickly got into after reaching the corresponding command, whereas the attitude control using PID controller needs a relatively longer time to get into the steady state. Summing up the comparative experiments above, for the attitude control of the UAV, the ADRC has better control effect and anti-interference ability than the PID controller which also uses the hybrid algorithm of this paper for parameter tuning.

It is known from the simulation and comparison test of the UAV flight command with the above three typical input signals that compared with the PID controller, the active disturbance rejection controller whose parameters are optimized by the hybrid algorithm can obtain the ideal tracking effect on the attitude control of the UAV, can well complete the basic attitude control action of the UAV and has good dynamic performance and the robustness.

VI. CONCLUSIONS

This paper has been focused on the improvement of the hybrid algorithm of fish swarm and particle swarm optimization and its application problem on the parameter tuning of the active disturbance rejection controller. Firstly, the principle of the active disturbance rejection controller has been described. Through the analysis of the parameter tuning problem of the active disturbance rejection controller and the research on the optimization performance of the artificial fish swarm algorithm and the standard particle swarm optimization algorithm, an improved hybrid algorithm of fish swarm and particle swarm optimization has been proposed based on the elite Gaussian learning, and the parameter tuning problem has been solved in the optimization design of active disturbance rejection controller. Finally, the optimized and designed active disturbance rejection controller has been applied to the flight attitude control of the UAV. The ADRC-based UAV attitude control simulation platform has been built in

Simulink, and it has been verified that the improved active disturbance rejection controller has better control effect in the UAV attitude control.

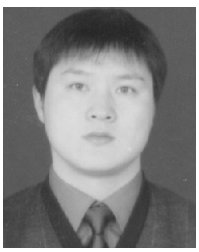
REFERENCES

[1] J. Han, "From PID technology to 'auto-disturbance control' technology," *Control Eng.*, no. 19, pp. 13–18, 2002.
[2] L. Qiang, L. Donghai, and T. Wen, "Performance robustness comparison of ADRC and GPC," in *Proc. IEEE 31st Chin. Control Conf.*, Jul. 2012, pp. 4586–4590.
[3] C. Du, Z. Yin, Y. Li, X. Sun, and Y. Zhong, "Vector control method of induction motor based on repetitive active disturbance rejection control," *Trans. China Electrotech. Soc.*, vol. 32, no. 19, pp. 81–89, 2017.
[4] L. Xinghua and C. Wenlei, "Application of active disturbance rejection controller for high precision servo system," in *Proc. Int. Conf. Mechtron. Sci., Electr. Eng. Comput. (MEC)*, Jilin, China, Aug. 2011, pp. 2467–2470.
[5] W. Cheng, Q. Chen, M. Sun, and Q. Sun, "Multivariable inverse decoupling active disturbance rejection control and its application in distillation column process," *Acta Automatica Sinica*, vol. 43, no. 6, pp. 1080–1088, 2017.
[6] W. Xu, Y. Chen, N. Qi, and Y. Yang, "Auto disturbance rejection control of spacecraft rendezvous and docking simulation system approximation process," *Acta Aeronautica Sinica*, vol. 37, no. 5, pp. 1552–1562, 2016.
[7] H. Hu, Y. Wen, L. Yang, and M. Xu, "Design of attitude controller for winged aircraft based on active disturbance rejection control," *Flight Dyn.*, vol. 31, no. 3, pp. 239–243, 2013.
[8] L. Fei, F. Lou, and G. Yan, "Design and analysis of active disturbance rejection control system for underactuated walking robot," *J. Jiangsu Univ. (Natural Sci. Ed.)*, vol. 37, no. 5, pp. 541–547, 2016.
[9] H. Shao, B. Xia, M. Hu, and G. Fu, "Fault diagnosis of pumping unit based on artificial fish swarm neural network," *Elect. Automat.*, vol. 39, no. 2, pp. 26–28, 2017.
[10] M. Jiang, Y. Wang, S. Pfletschinger, M. A. Lagunas, and D. Yuan, "Optimal multiuser detection with artificial fish swarm algorithm," *Commun. Comput. Inf. Sci.*, vol. 2, pp. 1084–1093, 2007.
[11] W. Zhang, Z. Lin, W. Liu, and Y. Zhang, "Robot path planning based on improved artificial fish swarm algorithm," *Comput. Simul.*, vol. 33, no. 12, pp. 374–379 and 448, 2016.
[12] X. Wang, X. Feng, and H. Zheng, "Research on WMSN coverage optimization based on artificial fish group and particle swarm fusion algorithm," *Microelectron. Comput.*, vol. 33, no. 10, pp. 106–110, 2016.
[13] J. Li and X. Liang, "Improved artificial fish swarm algorithm based on conjugate gradient method," *Appl. Res. Comput.*, vol. 34, no. 12, pp. 3589–3593, 2017.
[14] A. A. Abido, "Particle swarm optimization for multimachine power system stabilizer design," in *Proc. Conf. Power Eng. Soc. Summer Meeting*, Vancouver, BC, Canada, Jul. 2001, pp. 1346–1351.
[15] T. Xia, X. Wang, S. Liang, Z. Dang, and J. Wang, "Particle swarm neural network with adaptive genetic operator and its application," *J. PLA Univ. Sci. Technol. (Natural Sci. Ed.)*, vol. 12, no. 1, pp. 70–74, 2011.
[16] C. Zhang, G. Lei, and D. Han, "Research on particle swarm particle filter algorithm based on target tracking," *Comput. Simul.*, vol. 31, no. 8, pp. 392–396, 2014.
[17] L. Huang, K. Wang, C. Zhou, W. Pang, L. Dong, and L. Peng, "Particle swarm optimization for traveling salesman problem," *J. Jilin Univ. Sci. Ed.*, vol. 41, no. 4, pp. 447–480, 2003.
[18] C. Ma, W. Hao, R. He, and B. Moghimi, "A multiobjective route robust optimization model and algorithm for hazmat transportation," *Discrete Dyn. Nature Soc.*, vol. 2018, Oct. 2018, Art. no. 2916391.
[19] L. Dexiang, Z. Yongquan, and H. Huajuan, "Hybrid optimization algorithm based on particle swarm and artificial fish swarm algorithm," *Comput. Appl. Chem.*, vol. 15, no. 2, pp. 763–777, 2009.
[20] J. Li, X. Chen, and X. Chen, "Application of active disturbance rejection control in fast tool servo system control," *Mach. Tool Hydraul.*, vol. 43, no. 8, pp. 124–127, 2015.
[21] H.-C. Tsai and Y.-H. Lin, "Modification of the fish swarm algorithm with particle swarm optimization formulation and communication behavior," *Appl. Soft Comput.*, vol. 11, no. 8, pp. 5367–5374, 2011.
[22] J. J. Liang and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer," in *Proc. IEEE Swarm Intell. Symp.*, Jun. 2010, pp. 124–129.
[23] C. Ma and R. He, "Green wave traffic control system optimization based on adaptive genetic-artificial fish swarm algorithm," *Neural Comput. Appl.*, vol. 26, no. 5, pp. 1–11, 2015.
[24] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009.
[25] X. Wang, "Design and simulation of attitude control law for small UAVs," Shenyang Aerosp. Univ., Liaoning, China, Tech. Rep., 2016.

CHAOHAI KANG received the M.S. degree in control theory and control engineering from Northeast Petroleum University, Daqing, China, in 2005, where he is currently an Associate Professor. His research interests include intelligent control and pattern recognition.

SIQI WANG received the B.S. degree in automation from Northeast Petroleum University, in 2016, where she is currently pursuing the master's degree in control science and engineering. Her major research directions are control theory and control engineering, and intelligent control theory and application.

WEIJIAN REN received the M.S. degree in control theory and control engineering and the Ph.D. degree in oil storage and transport engineering from Northeast Petroleum University, Daqing, China, in 1990 and 2006, respectively, where she is currently a Professor. Her research interests include modeling and control of complex systems, fault diagnosis, and simulation.



YANG LU received the M.S. degree in computer application and the Ph.D. degree in oil and gas engineering from Northeast Petroleum University, in 2005 and 2013, respectively. He is currently a Professor with Heilongjiang Bayi Agricultural University. His research interests include machine learning, pattern recognition, computer vision, and neural networks.

BOYU WANG received the B.S. degree in automation and the M.S. degree in control science and engineering from Northeast Petroleum University, Daqing, China, in 2015 and 2018, respectively. His major research direction is control theory and control engineering.