the rime optimization algorithm (RIME), inspired by the growth behavior of rime-ice in nature. The RIME consists of three main processes:

1) Simulating the motion of soft-rime particles in rime-ice and proposing a soft-rime search strategy, mainly used to explore the algorithm. This strategy has a unique stepwise exploration and exploitation method, and the algorithm can continuously switch between large-range exploration and small-range exploitation to achieve high efficiency and high accuracy.

2) Simulate the crossover behavior between hard-rime agents, and propose a hard-rime puncture mechanism, mainly used to exploit the algorithm. This mechanism achieves effective information exchange between agents through dimensional cross-swapping between ordinary agents and optimal agents.

3) Enhance the greedy selection mechanism of the algorithm, and propose the positive greedy selection mechanism. This mechanism filters the inferior solutions in the population and actively introduces suboptimal solutions by changing the selection of optimal solutions. On the premise of ensuring the quality of the population, the diversity of the population is increased, and the algorithm is avoided to fall into local optimum as much as possible.

In the experiments, to elucidate the characteristics and adaptability of the RIME algorithm, qualitative analysis experiments and parameter sensitivity experiments are designed in this paper. Further, to verify the comprehensive performance of the algorithm, this paper tests the RIME algorithm with 10 highly-cited original algorithms and 10 recent high-performance improved algorithms on the test sets of CEC2017 [48] and CEC2022 [49], respectively. In addition, this paper designs experiments for the parametric analysis of RIME to discuss the algorithm's potential when running with different parameters and dealing with different problems. Finally, to verify the ability of the algorithm to solve real-world problems, the RIME algorithm is used in this paper for five classical engineering optimization problems, including pressure vessel design (PVD) problem, welded beam design (WBD) problem, speed reducer design (SRD) problem, I-beam design (IBD) problem, and multiple disk clutch brake design (MDCBD) problem.

In summary, the contributions of this paper are as follows:

1) A novel meta-heuristic algorithm based on natural phenomena, called the rime optimization algorithm, is inspired by the growth of rime-ice.

2) A new exploration strategy, exploitation mechanism, and selection mechanism are constructed in the RIME algorithm, and each strategy is portable and can be used to improve peer algorithms.

3) Through qualitative analysis experiments and parameter sensitivity experiments, the algorithmic characteristics of RIME are detailed for more relevant application to various optimization problems.

4) A comparison experiment between RIME and 20 peer algorithms is designed based on the complete data set, and the experimental results confirm that the RIME has a tremendous advantage over peer algorithms in terms of optimal performance in various types of problems.

5) The RIME algorithm is applied to five practical engineering optimization problems, which initially demonstrates the algorithm's potential for application to practical optimization problems and can be subsequently used on other optimization problems.

The rest of this paper is structured as follows. In Section 2, the mechanism of rime-ice formation is described and the mechanism inspired is illustrated. In Section 3, rime formation is modeled, and the RIME algorithm is proposed. Section 4 describes the experiments involved in this work, including qualitative analysis, performance experiments, parametric analysis, and practical applications. Section 5 concludes the whole paper and clarifies future research directions.

2. Inspiration from the formation of rime-ice

Rime-ice is resulted from accumulated water vapor in the air that has not yet condensed. It freezes and sticks to objects such as tree branches at low temperatures. Due to their unique climatic characteristics and topography, some regions form a unique landscape like rime-ice every year, as shown in Fig. 1[7].

The growth process of rime ice is determined by the temperature, wind speed, humidity, air, and other factors, and rime formation varies under different conditions. At the same time, due to the influence of environmental factors and the growth pattern, rime-ice cannot grow indefinitely, and it will stop growing when it reaches a relatively stable state. The growth pattern of rime is generally divided into two types: soft-rime and hard-rime, which is mainly determined by the wind speed during the formation process, as shown in Fig. 2, where ΔABC represents the growth plane of rime, and D1, D2, D3, D4 represent the birth points of rime. Usually, soft rime is generated in a breeze environment, and hard rime is formed in a high-wind environment. The breeze is characterized by small wind speed and variable wind direction, and the wind exists in all directions simultaneously and in the same height plane, as shown in Fig. 2(a). Therefore, the soft rime formed by the breeze grows slowly and randomly. On the other hand, a gale is characterized by high wind speed and roughly the same wind direction in the same height plane, as shown in Fig. 2(b). Therefore, the hard rime formed by the gale is fast and grows in approximately the same direction.

In summary, this study is inspired by the growth mechanism of rime-ice and proposes a soft-rime search strategy for algorithm search by simulating the motion of soft-rime particles. Also, a hard-rime puncture mechanism is proposed to exploit the algorithm by simulating the crossover behavior between hard rime agents. Finally, the selection mechanism of the metaheuristic algorithm is improved, and the positive greedy selection mechanism is proposed. This paper proposes the RIME algorithm with better performance by combining the above three mechanisms.

3. Mathematical model of the RIME

In this section, the growth process of each rime strip is simulated by analyzing the effects of wind speed, freezing coefficient, the cross-sectional area of the attached material, and growth time. On the other hand, inspired by the diffusion-limited aggregation [50] method of simulating metal particle aggregation, the motion process of each rime particle coalescing into a rime agent is simulated by modeling the motion behavior of each rime particle, and the final generated rime-agent is in the form of a strip crystal. The RIME consists of four stages: the initialization of rime clusters, the proposed soft-rime search strategy, the proposed hard-rime puncture mechanism, and the improvement of the greedy selection mechanism.

3.1. Rime cluster initialization

Inspired by reality, this paper treats each agent rime as the searched agent of the algorithm and the rime-population formed by all agents as the population of the algorithm. Firstly, the whole rime-population R is initialized. The rime population consists of n rime agents S_i and each rime- agent consists of d rime-particles

[7] Pictures obtained from https://pixabay.com/ as copy right free images (a) https://pixabay.com/photos/barbed-wire-frost-frozen-cold-ice-1938842/ (b) https://pixabay.com/photos/thuja-ice-winter-cold-frozen-6015613/.

Fig. 1. Rime-ice real scene.

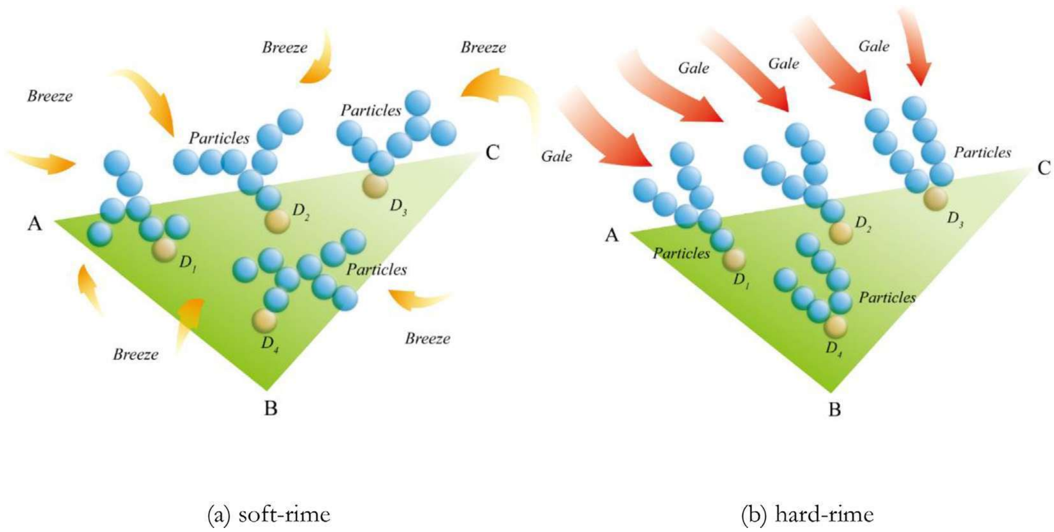

Fig. 2. The formation process of soft rime and hard rime under different environments.

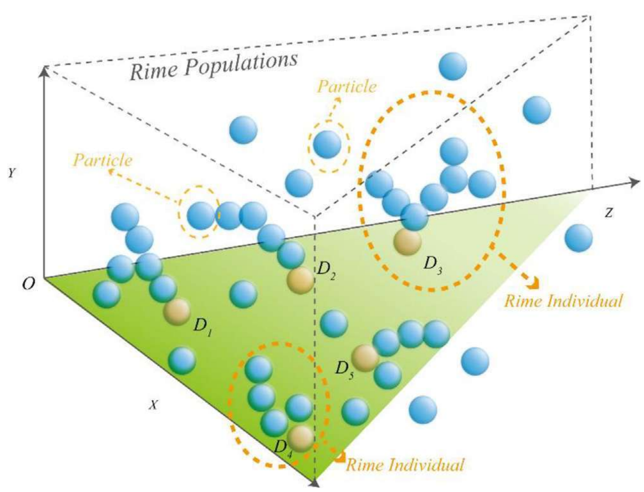

Fig. 3. Initialization of the rime space.

x_{ij}, as shown in Fig. 3 and Eq. (1). Thus, the rime-population R can be directly represented by the rime-particles x_{ij}, as shown in Eq. (2).

$$R = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_i \end{bmatrix}; S_i = \begin{bmatrix} x_{i1} x_{i2} \cdots x_{ij} \end{bmatrix} \tag{1}$$

$$R = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1j} \\ x_{21} & x_{22} & \cdots & x_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i1} & x_{i2} & \cdots & x_{ij} \end{bmatrix} \tag{2}$$

where i is the ordinal number of the rime agent and j is the ordinal number of the rime particle. In addition, $F(S_i)$ is used to denote the growth state of each rime-agent, i.e., the fitness value of the agent in the meta-heuristic algorithm.

3.2. Soft-rime search strategy

In a breezy environment, soft-rime growth is strongly random, and the rime particles can freely cover most of the surface of the attached object but grow slowly in the same direction. Inspired by the growth of soft-rime, this study proposes a soft-rime search strategy using the strong randomness and coverage of rime particles, which enables the algorithm to cover the entire search space in the early iteration quickly and does not easily fall into the local optimum.

When the rime particles condense into soft-rime agents, there are the following characteristics:

1) Before the particles condense to form a soft rime agent, each particle x_{ij} will wander according to a certain law, and the efficiency of the wandering is affected by environmental factors.

2) If the free-state rime particles move to the vicinity of a soft-rime agent, they will condense with the particles in the agent so that the stability of the soft-rime agent will change.

3) The distance between the centers of the two particles adhering to each other is not fixed, as the degree of condensation varies between each particle.

4) If the particles move directly outside the escape radius, no interparticle condensation occurs.

5) During the formation of a soft rime, the random condensation of each particle increases the area to which the agent is attached, resulting in a greater probability of free particle condensation. However, the agent will not grow indefinitely and will eventually reach a stable state due to environmental factors.

In this paper, corresponding to the five motion characteristics of the rime particles, the process of condensation of each particle is concisely simulated, as shown in Fig. 4, and the position of the rime- particles is calculated as shown in Eq. (3).

$$R_{ij}^{new} = R_{best,j} + r_1 \cdot \cos\theta \cdot \beta \cdot \left(h \cdot \left(Ub_{ij} - Lb_{ij}\right) + Lb_{ij}\right), r_2 < E \tag{3}$$

where, R_{ij}^{new} is the new position of the updated particle, and i and j denote the j-th particle of the i-th rime-agent. $R_{best,j}$ is the j-th particle of the best rime-agent in the rime-population R. The parameter r_1 is a random number in the range (-1,1) and r_1 controls the direction of particle movement together with $\cos\theta$ will change following the number of iterations, as shown in Eq. (4). β is the environmental factor, which follows the number of iterations to simulate the influence of the external environment and is used to ensure the convergence of the algorithm, as shown in Eq. (5). h is the degree of adhesion, which is a random number in the range (0,1), and is used to control the distance between the centers of two rime-particles.

$$\theta = \pi \cdot \frac{t}{10 \cdot T} \tag{4}$$

where t is the current number of iterations and T is the maximum number of iterations of the algorithm.

$$\beta = 1 - \left[\frac{w \cdot t}{T}\right] / w \tag{5}$$

where the mathematical model of β is the step function, $[\cdot]$ denotes rounding; the default value of w is 5, which is used to control the number of segments of the step function. Returning to Eq. (3), Ub_{ij} and Lb_{ij} are the upper and lower bounds of the escape space, respectively, which limit the effective region of particle motion. E is the coefficient of being attached, which affects the condensation probability of an agent and increases with the number of iterations, as shown in Eq. (6).

$$E = \sqrt{(t/T)} \tag{6}$$

r_2 is a random number in the range (0,1) which, together with E, controls whether the particles condense, i.e., whether the particle positions are updated. The pseudo-code for the soft-rime search strategy is shown in **Algorithm 1**.

Algorithm 1 Pseudo-code of the soft-rime search strategy

```
Initialize the rime-population R
Get the current optimal agent and optimal fitness
While t ≤ T
  Coefficient of adherence E = √(t/T)
    For i = 1 : n
      For j = 1 : d
        If r2 < E
          Position update according to the characteristics of
          the rime particles by Eq. (3)
        End If
      End For
    End For
  Update the current optimal agent and optimal fitness
  t = t + 1
End While
```

3.3. Hard-rime puncture mechanism

In strong gale conditions, hard-rime growth is simpler and more regular than soft-rime growth. When the rime particle condenses into a hard rime, there are the following characteristics: 1) The gale is so strong that other influences are negligible, resulting in different hard-rime agents snowballing in the same direction. 2) Due to the growth direction being the same, each rime agent can easily cross over, a phenomenon called rime puncture. 3) Like soft-rime agents, hard-rime agents increase in size as they grow, resulting in a greater probability of puncturing between agents in better growing conditions.

Therefore, this paper is inspired by the puncturing phenomenon and proposes a hard-rime puncture mechanism, which can be used

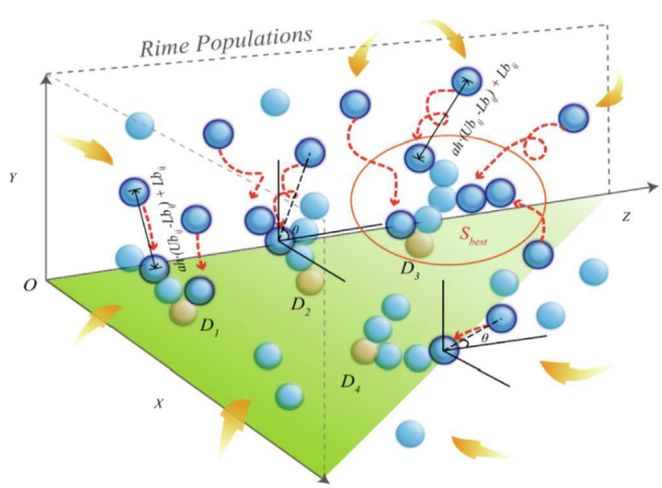

Fig. 4. Soft-rime particles motion.

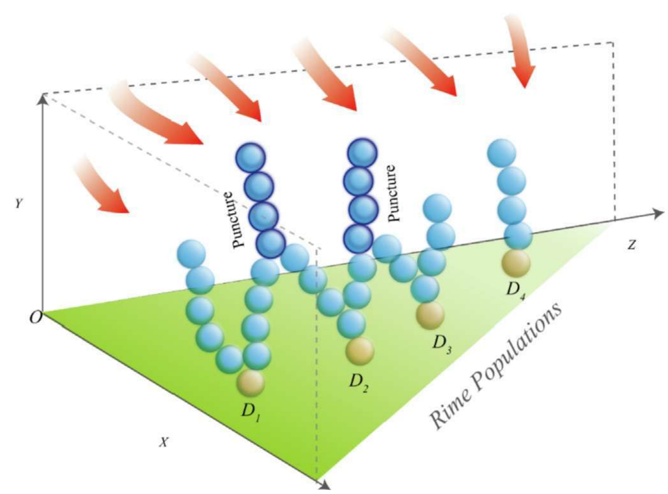

Fig. 5. Hard-rime puncturing.

to update the algorithm between agents, so that the particles of the algorithm can be exchanged and the convergence of the algorithm and the ability to jump out of the local optimum can be improved. The puncture phenomenon is shown in Fig. 5, and the formula for replacement between particles is shown in Eq. (7).

$$R_{ij}^{new} = R_{best,j}, r_3 < F^{normr}(S_i) \quad (7)$$

where R_{ij}^{new} is the new position of the updated particle and $R_{best,j}$ is the j-th particle of the best rime-agent in the rime-population R. $F^{normr}(S_i)$ denotes the normalized value of the current agent fitness value, indicating the chance of the i-th rime-agent being selected. r_3 is a random number in the range)-1,1(.

The pseudo-code for the hard-rime puncture mechanism is shown in **Algorithm 2**.

Algorithm 2 Pseudo-code of the hard-rime puncture mechanism

Initialize the rime-population R
Get the current optimal agent and optimal fitness
While $t \le T$
  For $i = 1 : n$
  For $j = 1 : d$
  If $r_3 < NormalizefitnessofS_i$
   Position update according to the characteristics of the rime-particles by Eq. (7)
   End If
   End For
   End For
Update the current optimal agent and optimal fitness
 $t = t + 1$
End While

3.4. Positive greedy selection mechanism

Typically, metaheuristic optimization algorithms have a greedy selection mechanism that replaces and records the best fitness value and the best agent after each update. The typical idea is to compare the updated fitness value of an agent with the global optimum, and if the updated value is better than the current global optimum, then the optimum fitness value is replaced, and the agent is recorded as the optimum. The advantage of such an operation is that it is simple and fast, but it does not help in the exploration and exploitation of the population and only serves as a record.

Therefore, the paper proposes an aggressive greedy selection mechanism for participating in population updates to improve global exploration efficiency. The specific idea is to compare the updated fitness value of an agent with the fitness value of an agent before the update, and if the updated fitness value is better than the value before the update, a replacement occurs, and also, the solution of both agents is replaced. On the one hand, this mechanism allows the population to continuously have good agents through active agent replacement, which improves the quality of the global solution. On the other hand, as the position of the agents of the population changes significantly with each iteration, there will inevitably be agents that are worse than the population before the update and are detrimental to the next iteration. Therefore, this operation can be used to ensure that the population evolves in a more optimal direction at each iteration.

In this paper, the pseudo-code of the positive greedy selection mechanism for solving the minimum value problem, as an example, is shown in **Algorithm 3**.

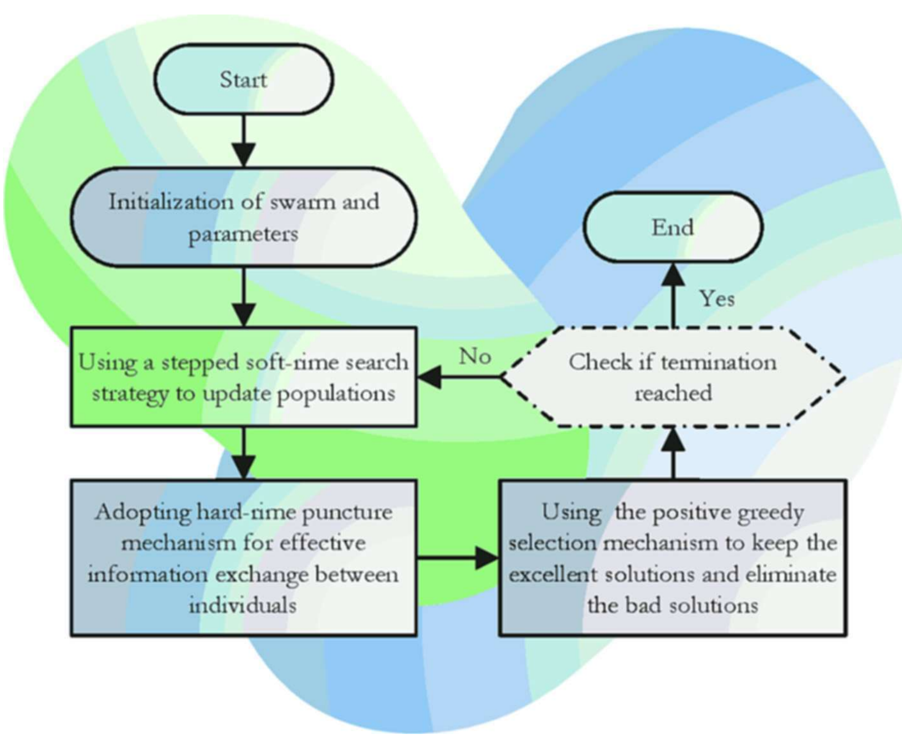

Fig. 6. Flowchart of RIME.

Algorithm 3 Pseudo-code of the positive greedy selection mechanism

Initialize the rime population R
Get the current optimal agent and optimal fitness
While $t \leq T$
For $i = 1 : n$
If $F(R_i^{new}) < F(R_i)$ // Compare fitness values
  $F(R_i) = F(R_i^{new})$ // Replace fitness values
  $R_i = R_i^{new}$ // Replace the current agent
  If $F(R_i^{new}) < F(R_{best})$ // Compare optimal fitness values
  $F(R_{best}) = F(R_i^{new})$ // Record optimal fitness values
  $R_{best} = R_i^{new}$ // Record the current optimal agent
  End If
 End If
 End For
 $t = t + 1$
End While

3.5. Proposed RIME algorithm

In summary, firstly, inspired by the motion of soft-rime particles in this section, a unique stepwise search and exploitation approach is designed to propose a soft-rime search strategy as the core optimization-seeking method of the algorithm. Immediately afterward, inspired by the crossover of hard-rime agents, a hard-rime puncture mechanism is proposed to achieve dimensional crossover interchange between ordinary and optimal agents, which is conducive to improving the solution accuracy of the algorithm. Finally, based on the greedy selection mechanism, an improved positive greedy selection mechanism is proposed to increase the diversity of the population and prevent the algorithm from falling into the local optimum as far as possible by changing the selection of optimal solutions. The overall structure of the algorithm in terms of pseudo-code and flow chart is shown in Algorithm 4 and Fig. 6.

Algorithm 4 Pseudo-code of RIME

Initialize the rime population R
Get the current optimal agent and optimal fitness
While $t \leq T$
 Coefficient of adherence $E = (t/T)^0.5$
 If $r_2 < E$
  Update rime agent location by ***the soft-rime search strategy***
 End If
 If $r_3 < Normalizefitnessof S_i$
  Cross updating between agents by ***the hard-rime puncture mechanism***
End If
If $F(R_i^{new}) < F(R_i)$
  Select the optimal solution and replace the suboptimal solution using ***the positive greedy selection mechanism***
 End If
 $t = t + 1$
End While

The complexity of RIME mainly includes the soft-rime search strategy, the hard-rime puncture mechanism, the positive greedy selection mechanism, and the calculation of the fitness value. First, the complexity level of the soft-rime search mechanism is $O(n^2)$. Then, the complexity level of the hard-rime puncture mechanism in the two extreme cases is $O(n)$ and $O(n^2)$. The complexity of the positive greedy selection mechanism is $O(n)$. Finally, the complexity level of the fitness value calculation is $O(n * logn)$. Therefore, the overall complexity level of the RIME algorithm is $O(RIME) = O((n + logn) * n)$.

4. Experiments and results

This section demonstrates the RIME algorithm's advantages and characteristics through experiments. Firstly, a qualitative analysis of the RIME algorithm demonstrates the algorithm's characteristics in finding the optimal solution. Then, the performance advantages of the algorithm are demonstrated experimentally by comparing the RIME algorithm with peer algorithms. Further, the parameter sensitivity analysis of the RIME algorithm is used to determine the parameters of RIME for different optimization problems to ensure maximum performance. Finally, the RIME algorithm is applied to five engineering optimization problems to demonstrate the algorithm's potential for application to practical optimization problems.

To ensure fairness and reproducibility of the experiments, all experiments in this paper were run in a unified environment where the software used was MATLAB 2017b and the core hardware was an Intel(R) Xeon(R) CPUE5-2660v3 (2.60 GHz).

4.1. Qualitative analysis of RIME

This subsection uses the classical 23 benchmark functions [24] to design four experiments for qualitatively analyzing the RIME algorithm concerning agents, dimensional particles, fitness values and iteration curves.

First, to analyze the distribution of optimal agents in the solution space of the RIME algorithm in the optimization problem, the search characteristics of the agents of the algorithm are visualized. In this paper, experiments on the historical position of agents are designed by recording the position of the optimal agent for each iteration. Further, while analyzing the agent updates, the paper analyses the particles in the agents, i.e., the dimensionality of the agent row vectors, to demonstrate the pattern and magnitude of change of the particles. In this paper, particle change experiments are designed by recording the first particle of the optimally solved agent at each iteration. Then, to analyze the changing trend of the fitness value of the algorithm after each iteration, this paper designs the fitness value change experiment by recording the optimal fitness value after each update. Finally, to analyze the algorithm's overall iteration trend, this paper records the fitness value of the optimal solution after each iteration and designs an iteration curve experiment. In the experiments, the population size of RIME is set to 30, the number of iterations is 2000, and 30 iterations are independently parallel.

Fig. 7 shows the results of RIME for the four qualitative experiments described above. In particular, the plots in the column of Fig. 7(a) represent the 3D location distribution of all solutions for each benchmark function, and it is within this solution space that RIME searches for the optimal solution. Fig. 7(b) represents the two-dimensional location distribution of RIME search histories. It can be seen that a small number of historical optimal solutions are scattered within the solution space, and most of the historical optimal solutions are clustered around the global optimal solution. This indicates that the RIME algorithm can find the approximate optimal solution within the solution space quickly and can enter the exploitation stage earlier to improve the accuracy of the solution.

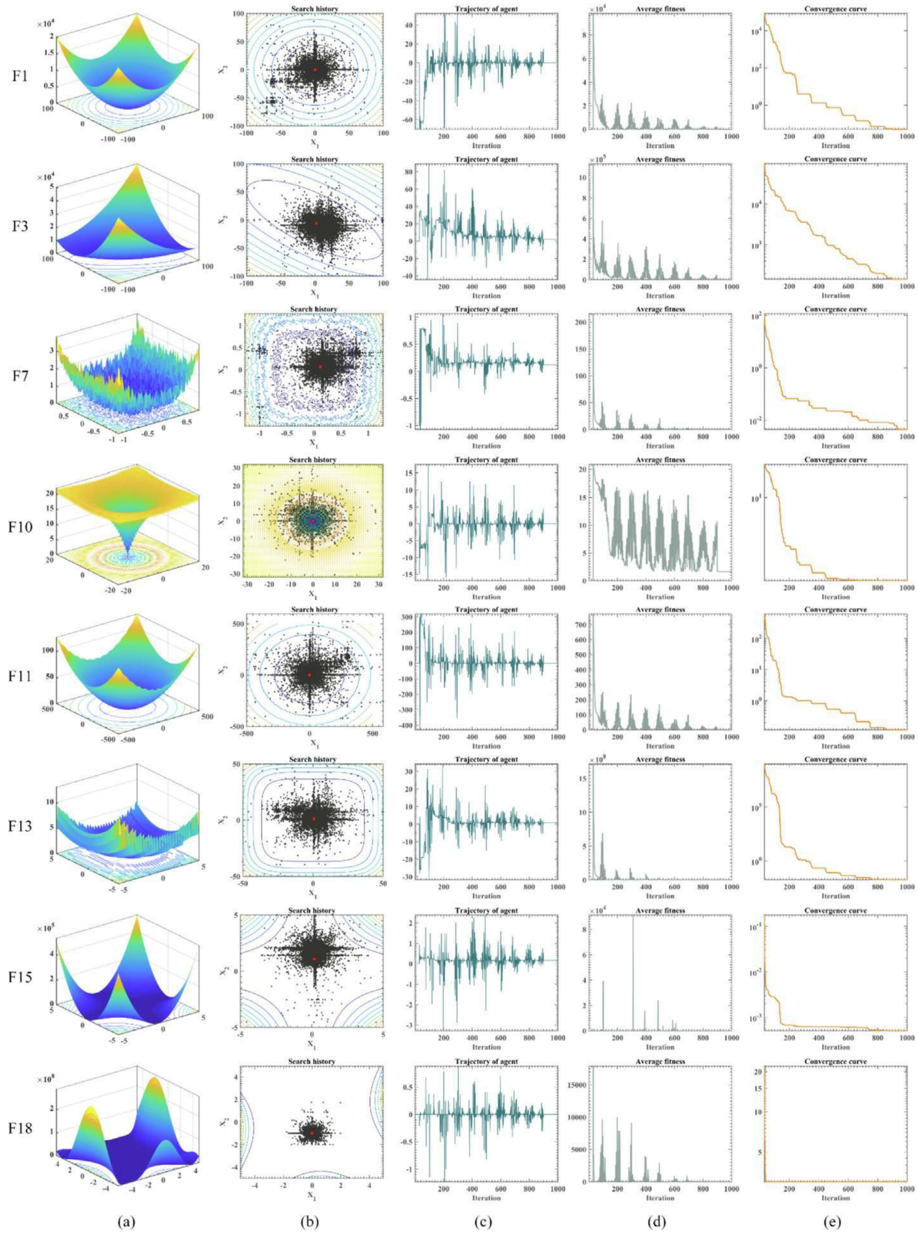

Fig. 7. Qualitative analysis experiment of RIME.

Fig. 7(c) records the trend of the first dimension of the RIME's agent throughout the iterations. It can be seen that the RIME algorithm has very long search steps in F1, F3, F10, F11, F15, F18 due to the role of the soft-rime search strategy in the early search phase, which is very beneficial for the algorithm to go out of local optimality and find the global optimal solution. The hard-rime puncture mechanism plays a significant role in the later iterations of the algorithm, resulting in shorter search steps, which helps to improve the accuracy of the optimal solution. On the other hand, the agent dimensions also oscillate less with the number of iterations. This facilitates the algorithm to converge quickly during the exploitation phase.

Fig. 7(d) records the optimal fitness values of the RIME after each iteration. It can be seen that, again, due to the search agents of the algorithm actively adjusting their search positions as they are updated, the fitness values for each iteration follow the agents with regular fluctuations. This indicates that the RIME algorithm is effective in searching for the optimal solution with agent updates, which is in line with the design of the algorithm.

Fig. 7(e) shows the overall iterative convergence curve of RIME. For most of the tested functions, the algorithm improves the quality of the solution as the number of iterations increases and does not fall into a local optimum. Further, Fig. 7(e) also shows that whether on simple single-peaked test functions such as F1 and F3 or complex multi-peaked and composite functions such as F7 and F13, RIME algorithms can search and develop incrementally during iterations through the combined action of the soft-rime search strategy and hard-rime puncture mechanism, avoiding as much as possible the algorithm falling into the local optimum trap.

In summary, the characteristics of RIME include: 1) It can quickly find the global approximate optimal solution, centralize exploitation and improve solution accuracy. 2) While ensuring convergence speed, the search position is actively changed during updates to improve the algorithm's global exploration capability and ability to jump out of local optima. 3) In the process of finding the optimal solution, the RIME algorithm has a unique stepped exploration and exploitation approach, which allows the algorithm to continuously RIME algorithm has a unique stepped exploration