Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 4, pp. 2603–2613, DOI 10.12694/scpe.v25i4.2904

DYNAMIC SECURITY RULE OPTIMIZATION BASED ON DEEP LEARNING AND ADAPTIVE ALGORITHMS

FEILU HANG*, LINJIANG XIE†, ZHENHONG ZHANG‡, YUTING LIU§, AND JIAN HU¶

Abstract. Compared with traditional computer network security identification techniques, deep learning algorithms are based on their own distributed network structure for storage, processing, classification, comparison, and automatic identification functions. The distribution and dynamism of cloud databases increase the difficulty of route prediction and recognition in the cloud, affecting the efficiency of cloud computing. In response to the above issues, the author proposes a dynamic path optimization process for cloud databases based on adaptive immune grouping polymorphic ant colony algorithm. By setting up two states of ant colony, reconnaissance ant and search ant, and introducing an adaptive polymorphic ant colony competition strategy, the defect of general ant colony algorithms being prone to falling into local optima is improved; On this basis, an artificial immune algorithm with fast global search capability is further integrated to improve the search ant path optimization process, improving search speed and accuracy. Simulation experiments show that the IPANT algorithm outperforms the other three algorithms, maintaining a throughput of 1000 kbps and relatively stable; The data of OSPF, SPF, and FR are not significantly different, significantly lower than IPANT. The immune polymorphic ant colony algorithm (IPANT) has the lowest time delay for packet routing and performs better than the other three algorithms, with FR and SPF having higher latency. It has been proven that this algorithm can better solve convergence speed and global optimization problems, and can quickly and reasonably find the database to be accessed in the cloud.

Key words: Adaptive polymorphic ant colony competition strategy, Immune polymorphic ant colony algorithm, Cloud database, Dynamic path optimization

1. Introduction. With the rapid development of artificial intelligence, internet technology, and automated recognition technology, more deep learning methods such as decision tree classification algorithms, gradient classification algorithms, neural network algorithms, convolutions, etc. are gradually being rapidly applied in computer network security recognition. Compared to traditional computer network security identification technology methods, deep learning algorithms make up for the shortcomings of computer network security intelligent management information ability, intuitive, and nonlinear data information adaptive ability based on their own distributed structure storage, processing, classification, comparison, and automatic recognition functions, thereby improving the network security data calculation ability and data information efficiency value, expanded the application scope of computer network security under deep learning mode[1]. On the basis of deep learning, in order to achieve efficient, accurate, and high-quality computer network security identification and management technology, the algorithm feature advantages of deep learning algorithms such as feature vector extraction, recognition, information combination optimization, and classification are applied to computer network security identification management. Through systematic analysis of its principles, architectural functional features, and platform application implementation, a set of security, economy, and intelligence is designed and constructed, provide a scientific reference for the design, implementation, and application of computer network

*Information security operation and maintenance center of Information Center of Yunnan Power Grid Co., LTD., Kunming, Yunnan, China, 650106 (Corresponding author, hangfeilu2021@163.com)
†Information security operation and maintenance center of Information Center of Yunnan Power Grid Co., LTD., Kunming, Yunnan, China, 650106
‡Network Security Management Center of Information Center of Yunnan Power Grid Co., LTD., Kunming, Yunnan, China, 650106
§Information security operation and maintenance center of Information Center of Yunnan Power Grid Co., LTD., Kunming, Yunnan, China, 650106
¶Network Security Management Center of Information Center of Yunnan Power Grid Co., LTD., Kunming, Yunnan, China, 650106

security identification management system diagrams for deep learning algorithms.

In order to scientifically and comprehensively grasp the characteristics and functions of computer network security identification management technology based on deep learning algorithms, the design and application of the technology platform in security identification management are based on the principles of scientificity, intuitiveness, security management, and scalability. Furthermore, based on the basic principles of design and application, enhance the functionality of network security identification technology in the context of deep learning, and enhance the application of deep learning methods and identification management core technologies. In the basic principles of design, scientific prediction and evaluation of computer network security forms are carried out through neural networks and convolutional learning methods. Compared with traditional security management methods, deep learning methods improve the accuracy of security evaluation prediction results and overall management efficiency; In terms of information scalability, machine learning presets the scalability of security protection tools in the security design process based on the current situation of computer systems, enabling the expansion and upgrading of new functions in security protection situations, and achieving full coverage of computer network security management[2].

As shown in Figure 1.1, the design process of computer network security identification management technology in the context of deep learning is presented. In the computer network security management technology platform, interactive mechanism structure is mainly used to achieve the sharing and co construction of interpretation mechanism, deep learning method inference, and data knowledge acquisition. At the same time, the functions of each module and important components are as follows: In terms of interpretation mechanism, coding data quantification evaluation is implemented for computer network security, and relevant predictions are made based on the evaluation results for the collected data information and situation values; In terms of deep learning methods and inference mechanisms, conduct situation assessment based on selected data, generate the required format data, and then predict and evaluate the current computer network security situation through neural network algorithms or convolutional algorithms to ensure computer network security[3]. The advantages of computer network identification security management mechanisms are mutually collaborative. When realizing the interoperability and sharing of mechanism functions, deep learning algorithms are used to demonstrate the current computer security situation and strengthen the technical capabilities of network security management. From the above scholars' research, it can be analyzed that polymorphic ant colony algorithms can basically solve the defects of ant colony algorithms in the path optimization process, and are suitable for cloud environments. However, during the search cycle, the number of the two ant colonies is not adjusted at any time, which can easily cause the search process to be too slow or stagnant, and the search ants still need to improve in terms of global search. Therefore, the author utilizes a grouping polymorphic ant colony algorithm with social morphology, and introduces an adaptive polymorphic ant colony competition strategy based on this. The strategy function Pg is used to reasonably adjust the number of reconnaissance ants and search ants, better improving the shortcomings of premature and local optima in general ant colony algorithms; In response to the shortcomings of the global search ability of search ants in polymorphic ant colony algorithms, an artificial immune algorithm with fast global search ability is organically integrated to further improve its optimization process: each search ant is regarded as an antibody based on the reconnaissance element and the path that may be the optimal solution, and the shortest path in reality is regarded as an antigen to solve the matching degree between the two, forming a new adaptive immune polymorphic ant colony algorithm, simultaneously solving the problem of global optimal solution and algorithm convergence speed, thereby optimizing the path of cloud databases. Finally, simulation experiments were conducted from three perspectives: Adaptive polymorphic ant colony competition strategy, immune polymorphic ant colony algorithm, and cloud database throughput and grouping delay to demonstrate the effectiveness of the algorithm.

2. Deep learning polymorphic ant colony algorithm and adaptive competition strategy.

2.1. Basic Polymorphic Ant Colony Algorithm Model. There are three main types of ant colonies in polymorphic ant colony algorithms: Reconnaissance ants, search ants, and worker ants. Among them, the task of the worker ant colony is only responsible for feeding back from the confirmed optimal path, and the author did not consider it when designing the cloud database path optimization algorithm[4]; Reconnaissance ants are mainly responsible for local reconnaissance, searching around each node in the cloud database and leaving reconnaissance results (reconnaissance elements) to provide assistance for searching ants; Search ants

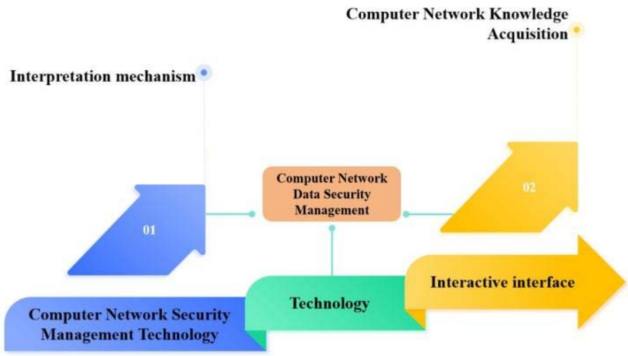

Fig. 1.1: Schematic diagram of computer network security identification management using deep learning algorithms

are mainly responsible for global search. At each node, they select the next node based on the reconnaissance elements left by the reconnaissance ants and the original information elements of each path, until the best path is found and marked.

1) Reconnaissance ants place m reconnaissance ants on n nodes, with each reconnaissance ant scouting n-1 other nodes centered around its node. The reconnaissance results are combined with existing MAXPC (prior knowledge) to form reconnaissance elements, denoted as s [i] [j], marked on the path from node i to j, the search ant colony can calculate the state transition probability $P_{ij}^{k}$and adjust the amount of information on each path based on the marked detection elements and existing pheromones[5]. The calculation formula for s [i] [j] is shown in Equation 2.1.

$$s[i][j] = \begin{cases} \overline{d_{ij}}/d_{ij} & \text{If node j is within MAXPC of i} \\ & 0 \quad otherwise \end{cases} \tag{2.1}$$

Among them, $\overline{d_{ij}}$ represents the minimum distance from node i as the center to other (n-1) nodes.

The amount of information on each path at the initial time is

$$\tau_{ij}(0) = \begin{cases} C \times s[i][j] & \text{if s[i][j]} \neq 0 \\ C \times \overline{d_{ij}}/\overline{\overline{d_{ij}}} & otherwise \end{cases} \tag{2.2}$$

Among them, $\overline{\overline{d_{ij}}}$ represents the maximum distance from node i as the center to other (n-1) nodes, and C represents the initial concentration of pheromones on each path.

2) The probability of state transition from node i to node j during the movement of ant k (k=1,2,...,m) is $P_{ij}^{k'}(t)$, and its value is

$$P_{ij}^{k'}(t) = \begin{cases} \frac{\tau_{ij}^{\alpha}(t) \times \eta_{ij}^{\beta}(t)}{\sum\limits_{s \neq tabu_k} \tau_{is}^{\alpha}(t) \times \eta_{is}^{\beta}(t)} & \text{if j} \neq tabu_k, \text{ands[i][j]} \neq 0 \\ & 0 \quad otherwise \end{cases} \tag{2.3}$$

After an iteration, the pheromone concentrations on each path need to be readjusted:

$$t_{ij}(t+1) = \begin{cases} (1-\rho) \times \tau_{ij}(t) + \rho \times \Delta\tau_{ij} & \text{ifs[i,j]} \neq 0 \\ (1-\rho) \times \tau_{ij}(t) & otherwise \end{cases} \tag{2.4}$$

Among them, $\Delta\tau_{ij}$ is the sum of the information released by all ants on the path (i, j) in this cycle, $\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$. The amount of information left by k ants on paths (i, j) in this cycle is $\Delta\tau_{ij}^{k}$, and the calculation formula is shown in Equation 2.5.

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q\times(\overline{d_{ij}}/d_{ij})}{L_k} & \text{If ant k passes through (i,j) and afters[i,j]} \neq 0 \\ 0 \quad otherwise \end{cases} \tag{2.5}$$

According to Equation 2.5: each search ant only leaves an appropriate amount of pheromones (local information $(\overline{d_{ij}}/d_{ij})$ combined with global information L_k) on the path that may be a component of the optimal solution based on the reconnaissance element. Is it a component of the optimal solution determined by s [i, j] Decision[6].

2.2. Adaptive Polymorphic Ant Colony Competition Strategy. In order to fully leverage the roles of search ants and reconnaissance ants, promote better integration of polymorphic ant colony algorithm and immune algorithm, and ensure the optimality of the improved algorithm, an adaptive polymorphic ant colony competition strategy is introduced, which reasonably allocates the number of two ant colonies and adjusts them at any time based on the cycle results. Firstly, set the competition strategy function, as shown in Equation 2.6.

$$P_g = F(X) = \int_{-\infty}^{X} f(x)\,dx \tag{2.6}$$

where f (x) is the normal distribution probability density function, that

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{2.7}$$

Among them: σ is the standard deviation, μ is a mathematical expectation. Due to compliance with standard normal distribution $\mu = 0$ $\sigma = 1$.At the beginning of the search, if the number of reconnaissance ants $m_0 = m \times P_g$ is the same as the number of search ants $m_1 = m \times (1 - P_g)$, then the variable $X = 0, P_g = 0.5$. For each iteration completed, X in Equation 2.6 is updated based on the search results[7].

Then set the adjustment method. Among them, APL (average path length) is used to represent the average path length of ant colony search, and a constant of $k_0 (1 < k_0 < 1.005)$ is set.

a) When $APL_{\text{Reconnaissance ant}} < k_0 \times APL_{\text{Search for ants}}$ is reached, it can be seen that the average path length searched by reconnaissance ants is shorter, and their ability to search for the best path is better than that of search ants. According to Equation 2.8: update X:

$$X = X + \frac{\sigma}{(\sigma + b_1)} \tag{2.8}$$

Among them, b_1 is a constant, and b_1 is taken as 100. At this point, as X increases and P_g also increases, the number of reconnaissance ants will increase in the next round.

b) When $APL_{\text{Reconnaissance ant}} < k_0 \times APL_{\text{Search for ants}}$ is reached, it can be seen for the best path, and update X according to Equation 2.9.

$$X = X \tag{2.9}$$

Just keep the number of two ant colonies consistent with the previous generation.

In order to avoid the complete disappearance of one of the ant colonies, AA regulation is used.

$$P_g = \begin{cases} P_{gmin} \quad if \quad P_g \leqslant P_{gmin} \\ P_g \quad if \quad P_{gmin \leqslant P_g \leqslant P_{gmax}} \\ P_{gmax} \quad if \quad P_g \geqslant P_{gmin} \end{cases} \tag{2.10}$$

Experimental simulation shows that the introduction of polymorphic ant colony algorithm and adaptive competition strategy can significantly improve the search performance of ant colony and improve the defect of general ant colony algorithms being prone to falling into local optima; And apply it to the improved new algorithm in the following text to further improve the dynamic path optimization effect of cloud databases.

3. Cloud Database Dynamic Path Optimization Process Based on Immune Polymorphic Ant Colony Algorithm.

3.1. Algorithm design concept. The basic idea of immune polymorphic ant colony optimization algorithm is to divide ants used for dynamic path planning in cloud databases into two categories according to the idea of group polymorphic ant colony algorithm. Among them, reconnaissance ants use each node placed in the cloud database for local reconnaissance and leave reconnaissance elements. The speed of global search based on polymorphic ant colony algorithm is relatively slow, and the artificial immune algorithm with fast and random global search ability is integrated to improve the functional part of ant search in the ant colony algorithm. Search ants use artificial immune algorithms to conduct global searches based on the reconnaissance elements left by the reconnaissance ants[8]. Consider the path that the search ant considers to be the possible optimal solution as an antibody, calculate the affinity (matching degree) between the antibody and the antigen, select the one with good affinity as the new antibody, obtain the optimal feasible solution, and generate the initial distribution of pheromones. Search for ants and optimize them based on reconnaissance elements and pheromones to improve solution efficiency. The improved model is conducive to finding optimal feasible solutions and improving solution efficiency. It can effectively improve the convergence speed of algorithms and the globality of optimal solutions that need to be solved in dynamic path queries of cloud databases, and achieve load balancing of computing resources.

3.2. Steps for optimizing dynamic paths in cloud databases. The steps of immune polymorphic ant colony optimization algorithm are as follows:

a) Before starting the search, place m reconnaissance ants on m nodes of the cloud database. Each reconnaissance ant evaluates the resource utilization rate of its neighboring cloud database nodes within the local range of the database node, and combines prior knowledge to form the reconnaissance element s [i] [j], among them, s [i] [j] represents the probability that node S chooses to use node j's resources through node i. The calculation formula for s [i] [j] is shown in equation (12): where i, j=0, 1, 2,..., m-1, $i \neq j$, η for neighbor Table.

$$s[i][j] = \frac{\sigma_{ij}}{\delta_{ij}} \tag{3.1}$$

b) Whenever user U searches on the platform, submit a search task Q to the source node N_s. The source node N is detected in the neighbor table n. If no neighbor nodes are found, task Q is temporarily placed in N_s.

c) Set the amount of information on each path at the initial time.

$$t_{ij}(0) = \begin{cases} C \times s[i][j] & if s[i][j] \neq 0 \\ C \times \overline{\sigma_{ij}}/\overline{\overline{\sigma_{ij}}} & otherwise \end{cases} \tag{3.2}$$

d) Before the search starts, the $P_g = F(X) = \int_{-\infty}^{X} f(x)\,dx$ will adjust the number of two ant colonies at any time by introducing the designed competition strategy.

e) The ant scheduling module AS of N_s generates search ant ANTs. The search ants select the most likely optimal path based on the reconnaissance elements left by the reconnaissance ants on m cloud database nodes, and roughly calculate the calculation amount of information search task Q, which is recorded as $\sigma(Q)$. Assuming the search period TTL is 0, calculate the grid size to obtain the longest period TTL_{max}. Calculate the available search ability of the source node N_s born as search ants and the estimated time T_l for N_s to start the next information search task, and then place them in the tabu(ANT_s) of ANTs.

f) Design the search part of search ants from the perspective of artificial immunity. If the path traveled by the ANTs search ants (m ants) in the taboo Table (i.e. the path that the search ants believe may be the optimal solution) is considered as antibody a, then the antibody population Ab can be expressed as $Ab = [a_1, a_2, ..., a_m]$, and m is the number of antibodies, i.e. the number of search ants; The antigen is the antibody with the shortest path length, which is the actual fastest path to reach the cloud database during the ant search process. The parameter that represents the degree of matching between antibodies and antigens is affinity. The affinity function is defined as

$$affinity(a_i) = \frac{[F - dist(a_i)]}{[\sum_{j=1}^{m}(F - dist(a_j)) + \varepsilon]} \tag{3.3}$$

Among them, dist (ai) is the path length of antibody a; F is the longest path among all antibodies, $F = max(dist(a_1), dist(a_2), ..., dist(a_m))$. In order to prevent the shortest path from becoming the same and intimacy saturation in later search tasks, coefficients are added to the denominator $\varepsilon(0 < \varepsilon < 1)$. The larger the affinity (ai): the greater the degree of matching between the antibody and antigen, and the more likely it is to be the shortest path.

g) Optimize the antibodies in the taboo list to generate new antibodies. The process of selecting new antibodies is as follows: Based on the calculated affinity, the antibodies with a high degree of matching are placed in the memory antibody library N_{Ab}, and the optimized antibodies are compared with the original antibodies. If the path length is shorter, the original antibodies are replaced.

h) Antibody selection. The newly generated antibodies are arranged in the taboo table according to their affinity matching degree from high to low. The first four antibodies are selected to form a new antibody group, and then proceed to step f) for cyclic optimization.

i) Simultaneously initializing parameters τ_C, τ_G, m_1, m_2. Follow step e) to obtain a shorter search path and update the pheromone. Among them, τ_C is the pheromone constant τ_G is the conversion value of the pheromone of the immune polymorphic ant colony algorithm result, m_1 is the current number of detected ants, and m_2 is the current number of searched ants[9]. Update pheromones while forming a new antibody population:

$$\tau_{ij}(0) = \begin{cases} \tau_C + \tau_G & m_1 \neq m_2 \\ t_c & otherwise \end{cases} \tag{3.4}$$

j) After n hours, m only searches for ants and ends searching for various nodes in the cloud database. Calculate the final function value Lk (h=1,2,..., n) of the antigen to output the optimal solution.

4. Simulation experiments. In order to better validate the effectiveness of the improved immune polymorphic ant colony algorithm in the dynamic path search process of cloud databases, the author selected an example from the internationally recognized TSPLIB test library for simulation experiments. Simulate and analyze the introduction of adaptive polymorphic ant colony competition strategy in polymorphic ant colony algorithm, the improvement of immune polymorphic ant colony algorithm using artificial immune algorithm, and the comparison of immune polymorphic ant colony algorithm with other algorithms in average throughput and group delay in cloud database path search. The experimental simulation software of the first two systems is MATLAB 7.0; the latter chose NS-2 from the ISI Institute of the University as the experimental simulation platform.

4.1. Simulation analysis of adaptive polymorphic ant colony competition strategy. The author selected datasets from four typical examples of the TSP problem, Att48, St70, Lin105, and Ch150, and conducted 40 tests based on different Pg values to verify whether polymorphic ant colonies and adaptive competition strategies can effectively improve search capabilities, among them, $\alpha = 1, B = 5, Q = 50$. $P_g = 0$ represents a polymorphic ant colony algorithm with only search ant colonies, $P_g = 1$ represents a polymorphic ant colony algorithm with only reconnaissance ant colonies, and adaptive PR represents a polymorphic ant colony algorithm with adaptive polymorphic ant colony competition strategy. The initial value of P_g is set to 0.5, and the maximum number of cycles is set to 4 times the number of cities[10]. The specific experimental results are shown in Table 4.1. In the table, N_{best} is the number of times the optimal solution was obtained within the experimental iteration number;.

From the comparative data shown in Figure 4.1 and Table 4.1, it can be seen that the polymorphic ant colony algorithm using adaptive P_g has improved convergence speed and accuracy, and is superior to the polymorphic ant colony algorithm with $P_g = 0$ and $P_g = 1$, because it combines the advantages of reconnaissance ants and search ants, and applies adaptive competition strategies to reasonably adjust the number of the two ant

Table 4.1: Test examples based on different Pg values (40 tests)

Problem S_0	Method	N_best	MTL	MITL	METL	σ/%
	P_g=0	4	23794	32515	33641	0.39
Att4833524	P_g=1	2	23794	32515	33602	0.25
	Adaptive P_g	7 23792	32515	33565	0.16	
	P_g=0	5	676	666	670	0.32
St70678.5975	P_g=1	6	677	666	669	0.21
	Adaptive P_g	7	676	666	668	0.07
	P_g=0	2	14563	14372	14447	0.51
Lin10514382.996	P_g=1	2	14571	14372	14436	0.44
	Adaptive P_g	8	14471	14372	14387	0.10
	P_g=0	0	6273	6114	6180	1.32
Ch1506110.9	P_g=1	0	6288	6167	6278	1.75
	Adaptive P_g	2	6243	6100	6150	0.8

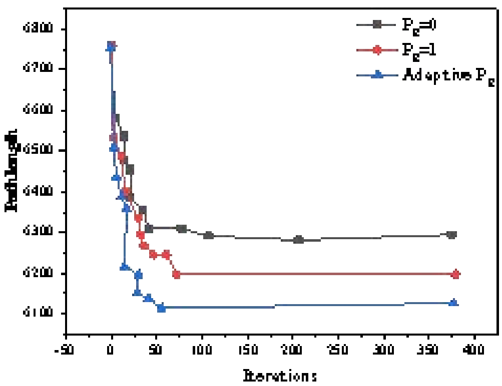

Fig. 4.1: Comparison of Convergence Process

colonies, ensuring global and local balance[11]. MTL (maximum value of tour length) is the longest path value; MITL (minimum value of tour length) is the shortest path value; METL (mean value of tour length) is the mean path length; σ is the average error (%): $\sigma\frac{\sum_{i=1}^{30}(S_{Ti}-S_0)}{30S_0}\times 100\%$ (where S_{Ti} is the shortest path for the i-th time and S_0 is the known shortest path). Figure 4.1 shows the comparison of convergence processes based on different P_g in the Ch150 instance.

4.2. Simulation analysis of immune polymorphic ant colony algorithm path optimization. Regarding whether the improved immune polymorphic ant colony algorithm can achieve faster search for global optimal solutions, the author selects Oliver 10 in the TSP problem as a simulation example, which is often used to verify the effectiveness of a certain algorithm. The coordinate data of Oliver 10 is as follows:

Abscissa x=5 10 8 20 6 1 21 9 30 25;

Ordinate y =20 2 12 16 3 20 30 35 20 8.

Its known shortest path length is 106.740.

Simulate the improved immune polymorphic ant colony algorithm and basic ant colony algorithm using MATLAB 7.0. In the experiment, take $\alpha = 1, \beta = 3, \rho = 0.3, Q = 50$. The experimental results are shown in Table 4.2, and the comparison of path evolution between the two algorithms is shown in Figures 4.2 and 4.3. The result of the number of iterations is a conclusion drawn after 10 experiments[12].

Table 4.2: Comparison of the experimental results of the basic polymorphic ant colony algorithm

The maximum short circuit strength length	Number of iterations(immune polymorphism)	Number of iterations(basic polymorphism)	Time ratio (Immune polymorphism /basic polymorphism)
108.324	9	18	1:1
107.746	14	29	1:3
107.193	28	47	1:3
106.740	32	58	1:4

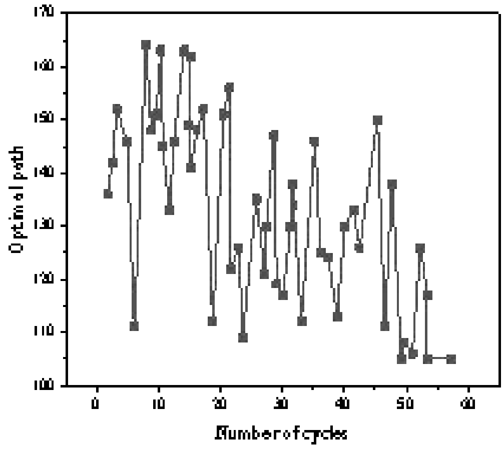

Fig. 4.2: Pathway evolution curve of basic polymorphic ant colonies

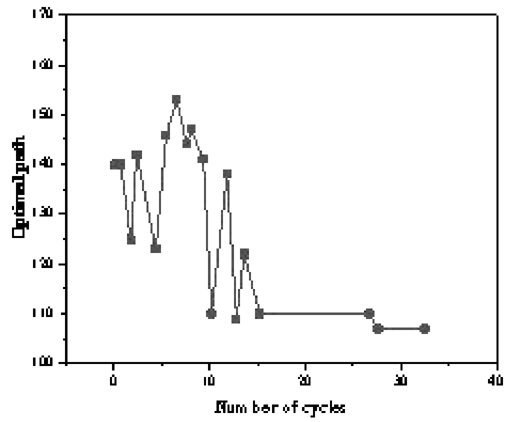

Fig. 4.3: Pathway evolution curves of immunopolymorphic ant colonies

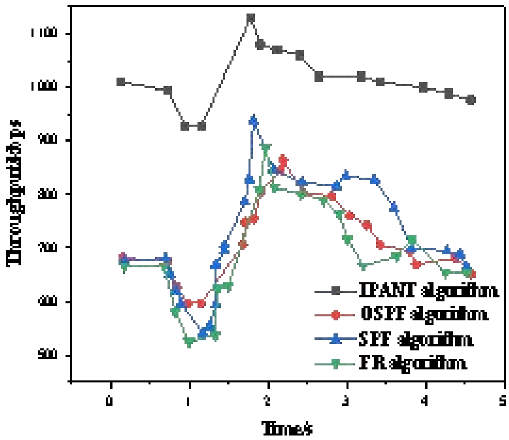

Fig. 4.4: Average throughput

From Table 4.2, it can be seen that using the immune polymorphic ant colony algorithm, an average of 32 generations can obtain the optimal solution of 106.740; Using the basic polymorphic ant colony algorithm, an average of 58 generations can be obtained, and the time is much longer than the immune polymorphic ant colony algorithm. From Figures 4.2 and 4.3, it can also be seen that the basic polymorphic ant colony algorithm has unstable iterations and multiple iterations; The improved immune polymorphic ant colony algorithm significantly reduces the number of iterations, is stable, easy to converge, and can quickly find satisfactory solutions. So the improved immune polymorphic ant colony algorithm can effectively solve the convergence speed of the algorithm and the global optimization problem of the optimal solution.

4.3. Cloud Database Throughput and Packet Delay Simulation Analysis. Due to the fact that the author's research object is cloud databases rather than general databases, simulation needs to further consider the characteristics of cloud databases. However, the role of a data network depends on many components that interact in nonlinear and unpredictable ways, so choosing a meaningful testing environment is very difficult[13]. The method followed is to define a finite set of adjustable components composed of various classifications. The experimental model selected NS-2 from the South California University ISI Research Institute as the experimental platform. Throughput and data packet delay were selected as the evaluation criteria for the effectiveness of the new algorithm, and three representative excellent routing algorithms in communication networks were selected, namely the open shortest path first algorithm, the shortest path first algorithm, and the flow based routing algorithm, to compare with the immune polymorphic ant colony algorithm. Take the average of 30 experiments as experimental data, and each simulation time is 1000 virtual seconds.

The average throughput is shown in Figure 4.4. From the experimental results in Figure 4.4, it can be seen that the IPANT algorithm outperforms the other three algorithms, maintaining a throughput of 1000 kbps and relatively stable; The data of OSPF, SPF, and FR are not significantly different, significantly lower than IPANT[14-15].

The empirical distribution of packet delay is shown in Figure 4.5. From the experimental results in Figure 4.5, it can be seen that the immune polymorphic ant colony algorithm (IPANT) has the lowest time delay for packet routing and performs better than the other three algorithms, with FR and SPF having higher latency.

5. Conclusion. In order to solve the problem of difficult route prediction and recognition in the cloud due to the dynamic instability of cloud databases, the author proposes an adaptive immune polymorphic ant colony algorithm for path optimization in cloud databases. By combining the group polymorphism ant colony intelligent search algorithm with cloud database search with strong global search ability, and integrating artificial

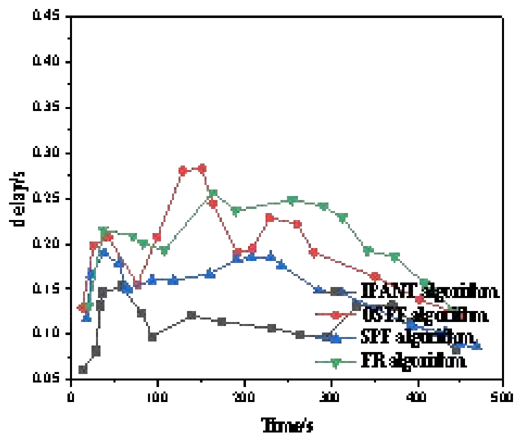

Fig. 4.5: Empirical distribution of grouping delays

immune algorithm in the process of polymorphism ant colony optimization, it effectively solves the problems of abundant and dynamic resources in cloud databases, slow convergence speed of traditional optimization algorithms, and falling into local optima, achieving load balancing. Compared with other routing algorithms, good experimental results and effects were obtained. However, the new algorithm has some shortcomings in improving the grouping delay of cloud databases. Although it has significant advantages compared to SPF and FR algorithms, it has little difference in latency compared to OSPF algorithm, and the experimental effect needs to be improved. However, due to the complexity of the testing environment and the limitations of experimental conditions, the author's current research is only conducted in simulated cloud environments. In the future, it will be applied to real cloud databases for experiments, in order to better improve algorithms and improve the efficiency of path optimization. In addition, the next step of work is mainly focused on designing and developing cloud platforms that can truly apply the designed algorithms to practice.

REFERENCES

[1] Z.-J. W , H.-B. G , X.-H. W ,et al.Adaptive learning rate optimization algorithms with dynamic bound based on Barzilai-Borwein method.Information Sciences: An International Journal, 2023;236(1): 19-33.
[2] Chen C , Liu H .Dynamic ensemble wind speed prediction model based on hybrid deep reinforcement learning.Advanced Engineering Informatics, 2021; 48(8):101290.
[3] Hu A , Gong X , Guo L .Joint Optimization-Encryption Model Based on Auto Encoder Neural Network, Variable-Length Dynamic S-Box and New Stream Encryption.International journal of bifurcation and chaos in applied sciences and engineering, 2022;8(10): e18287.
[4] Pérez-Aracil, Jorge,Camacho-Gómez, Carlos, Lorente-Ramos E ,et al.New Probabilistic-Dynamic Multi-Method Ensembles for Optimization based on the CRO-SL.arXiv e-prints, 2022; 661(1): 012021.
[5] Zakhrouf M , Bouchelkia H , Kim S ,et al.Novel insights for streamflow forecasting based on deep learning models combined the evolutionary optimization algorithm.Physical Geography, 2021;43(1): 49-57.
[6] Marcelin K , Mcheick H , Kouame K M .Designing Adaptive Mechanism for COVID-19 and Exacerbation in Cases of CODP Patients Using Machine Learning Approaches.Science Publishing Group, 2021;34(3): 3065-3072.
[7] Sharif A , Li J P , Saleem M A ,et al.A Dynamic Clustering technique based on Deep Reinforcement Learning for Internet of Vehicles.Journal of Intelligent Manufacturing, 2021; 32(5):96-102
[8] Pulvirenti L , Tresca L , Rolando L ,et al.Eco-Driving Optimization Based on Variable Grid Dynamic Programming and Vehicle Connectivity in a Real-World Scenario.Energies, 2023;34(5): 9354-9361.
[9] Wu B , Zhang S .Energy Management Strategy for Dual-Motor Two-Speed Transmission Electric Vehicles Based on Dynamic Programming Algorithm Optimization.SAE International Journal of Electrified Vehicles, 2021;74(1):10.
[10] Li J , Lan F , Chen J .Intelligent vehicle visual pose estimation algorithm based on deep learning and parallel computing for dynamic scenes.Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 2022;39(8):

3563-3570.

[11] Zhang Y , Yang X , Liu S .Data-driven predictive control for floating offshore wind turbines based on deep learning and multi-objective optimization.Ocean engineering, 2022;58(7):45-49.

[12] Yu Q , Lin Q , Zhu Z ,et al.A dynamic multi-objective evolutionary algorithm based on polynomial regression and adaptive clustering.Swarm and Evolutionary Computation, 2022;874(71):71.

[13] Hu Z , Ma X , Sun H ,et al.Many-objective optimization algorithm based on adaptive reference vector.Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 2021;96(1):40.

[14] Li W , Zeng Y , Wu Y ,et al.Dynamic manifold Boltzmann optimization based on self-supervised learning for human motion estimation.IET image processing, 2022;75(4):16.

[15] Tian H , Tian C , Li K ,et al.Dynamic operation optimization based on improved dynamic multi-objective dragonfly algorithm in continuous annealing process.Journal of Industrial and Management Optimization, 2023, 19(8):6159-6181.

Edited by: Zhigao Zheng
Special issue on: Graph Powered Big Aerospace Data Processing
Received: Nov 23, 2023
Accepted: Dec 15, 2023