Received: 22 July 2022 | Revised: 13 August 2022 | Accepted: 20 August 2022

IET Networks

DOI: 10.1049/ntw2.12056

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

An ant colony genetic fusion routing algorithm based on soft define network

Kaixin Zhao | Yong Wei | Yang Zhang

Department of Computer Science and Technology, Henan Institute of Technology, Xinxiang, China

Correspondence

Kaixin Zhao, Department of Computer Science and Technology, Henan Institute of Technology, Xinxiang 453003, China.
Email: zhaokx_2008@126.com

Handling Editor: Christoph Sommer

Funding information

Henan Science and Technology Development Plan Project, Grant/Award Number: 222102240046

Abstract

Aiming at the problem that there are many paths in data forwarding in soft define network (SDN) network, and the optimal path is difficult to find, combined with the advantages of ant colony algorithm and Genetic algorithm (GA), a routing control strategy based on the ant colony genetic fusion algorithm is proposed. This algorithm absorbs the advantages of high speed in the early stage of GA search and high efficiency in the late stage of ant colony algorithm search; the improved ant colony genetic fusion algorithm is applied to the SDN routing process for simulation tests. The simulation experiment shows that the throughput and link utilization of the proposed algorithm are about 30% higher than that of the single optimal particle swarm optimization algorithm.

KEYWORDS

ant colony algorithm, genetic algorithm, SDN, throughput delay

1 | INTRODUCTION

At present, intelligent algorithms are widely used in path planning for network models because of their high efficiency. Among them, the ant colony algorithm is a distributed optimization method with a positive feedback mechanism, which can use the information generated in the early stage to further accelerate the solution speed [1]. A Genetic algorithm (GA) is a heuristic search algorithm with high operation efficiency and strong search ability [2]. The particle swarm optimization algorithm has the advantages of few parameter settings, fast convergence speed, simple implementation, and so on. At present, these three algorithms are widely used in the field of path planning [3]. In Ref. [4], the author applies the GA optimised by chromosome and the fitness function to robot path planning, which improves the speed at the initial stage of path search, but the algorithm will spend a lot of time on redundant iteration in the later stage of path search. In Ref. [5], the author uses the random method and the ascending order method to improve the initial population generation process of the GA to avoid the algorithm falling into local optimization in routing, but serious network congestion will occur when the load is large. In Ref. [6], the author solves the problem of ant colony algorithm stagnation when solving TSP(Travelling Salesman Problem). Based on the edge pheromone update mechanism and pheromone initialisation strategy, but in the early stage of path search, due to the lack of pheromone, the solution efficiency is low.

The ant colony algorithm [7, 8] is well used in routing, but its global search ability is weak, the algorithm is easy to fall into local optimal solution, and the convergence speed is slow. The genetic algorithm [9, 10] is fast in the initial stage of the search, but it will spend a lot of time on redundant iteration because it cannot make full use of the feedback information in the system in the later stage of the search. Based on the above shortcomings, this paper proposes a fusion algorithm of the ant colony and GA to solve the optimal path problem in the soft define network (SDN) network.

1.1 | Contributions

Following are the main contributions of this paper:

- We propose an improved ant colony algorithm to improve the search efficiency of the optimal path.

- We propose a routing control strategy based on the ant colony genetic fusion algorithm, which combines the advantages of ant colony algorithm and GA to avoid the generation of local optimal path in the early stage of ant colony algorithm and the redundant iteration in the later stage of GA.
- We apply the ant colony genetic fusion algorithm to SDN routing and verify the advantages of the algorithm proposed in this paper through simulation tests

The rest of the paper is organised as follows. A brief overview of related work is provided in Section 2. Section 3 provides an improved scheme of ant colony algorithm and GA. Section 4 describes our proposed solution approach to the fusion of ant colony and GA. Section 5 presents the simulation results for the performance evaluation of the proposed work compared to existing schemes. Section 6 provides concluding remarks and future research directions.

2 | RELATED WORK

For the routing problem in SDN network, the ACA (Ant Colony Algorithm) can improve the accuracy of the results, but it also has some defects, such as local optimization and long search time. The optimization of ACA can consider the following two important characteristics – probabilistic state transition rules and pheromone update rules. In addition, high robustness combined with the advantages of other algorithms is also one of the improvement directions of ACA [11–13].

For the probabilistic state transition rule, In Ref. [14], the author uses the roulette selection method to construct a feasible solution, because it does not need to be improved to the ACA α And β. Parameters reduce the computational complexity to O (1) to avoid falling into local optimization. However, reducing the importance of pheromones may lead to slow convergence of the algorithm. In Ref. [15], the author introduces a fault elimination factor into the probabilistic state transition rule to avoid selecting the fault node in the network as much as possible and increases the diversity of feasible solution search to improve the accuracy of the algorithm. In Ref. [16], the author believes that the existing probabilistic state transition rules are easy to lead to locally optimal solutions and need to dynamically adjust the balance parameters: initially set a larger balance parameter, ants tend to nodes with larger pheromones, reduce the value of the balance parameter after iterations, ants choose new nodes to avoid local optimization, and finally improve the balance parameters again to speed up the convergence process of the algorithm.

Considering the pheromone update rule, in Ref. [17], the author believes that some abnormal nodes may affect the quality of the global optimal solution and proposes a pheromone update rule to exclude abnormal nodes, screen the nodes with abnormal pheromones in the iterative optimal solution, and the subsequent iterations give priority to the normal nodes, so as to guide the ants to the new search area. In Ref. [18], the author found that when the pheromone volatilisation coefficient is constant, the search ability of the algorithm for nodes with low pheromone is not strong, so a pheromone dynamic volatilisation mechanism is proposed, and the volatilisation coefficient changes dynamically according to the value of pheromone after each iteration. In Ref. [19], the author improved the max min ant system to avoid the result falling into the local optimal solution: the vertex pheromone is initially the maximum value, the pheromone is strictly limited to the maximum range in the ant colony search process, and only the pheromone of the vertex in the iterative optimal solution or the global optimal solution is updated after each iteration. However, these pheromone update rules cannot take into account the accuracy and convergence of the algorithm, and the application effect in the SDN measurement node selection problem is not good.

Considering the pheromone update rule, in Ref. [20], the author believes that some abnormal nodes may affect the quality of the global optimal solution and proposes a pheromone update rule to exclude abnormal nodes, screen the nodes with abnormal pheromones in the iterative optimal solution, and the subsequent iterations give priority to the normal nodes, so as to guide the ants to the new search area. In Ref. [21], the author found that when the pheromone volatilisation coefficient is constant, the search ability of the algorithm for nodes with a low pheromone is not strong, so a pheromone dynamic volatilisation mechanism is proposed, and the volatilisation coefficient changes dynamically according to the value of pheromone after each iteration. In Ref. [22], the author improved the max min ant system to avoid the result falling into the local optimal solution: the vertex pheromone is initially the maximum value, the pheromone is strictly limited to the maximum range in the ant colony search process, and only the pheromone of the vertex in the iterative optimal solution or the global optimal solution is updated after each iteration. However, these pheromone update rules cannot take into account the accuracy and convergence of the algorithm, and the application effect in the SDN measurement node selection problem is not good.

2.1 | Soft define network

In 2006, the clean slate project led by Stanford University and jointly launched by NSF and several industrial manufacturers, the project research team proposed a new network innovation architecture SDN, which is an implementation method of network virtualisation. Its core is to realize the separation of network data plane and control plane through the OpenFlow technology and realize network programmability, which brings convenience to the intelligent management, scalability, and elastic change of the network. Then, foreign Cisco, juniper, HPE, NEC, arista, IBM, and other manufacturers have successively released SDN switches supporting OpenFlow. Domestic Huawei, ZTE, Shengke, and other equipment manufacturers have also rapidly developed SDN-related products. Mobile, Unicom, telecom, and other operators have also put forward to SDN solutions on the Internet. The specific network architecture is shown in Figure 1.

It can be seen from Figure 1 that the SDN network adopts a hierarchical structure, which is divided into three layers: application layer, control layer, and infrastructure layer, mainly including four planes and two interfaces, namely, application plane, control plane, management plane, data plane, south interface and north interface. The data plane and control plane communicate through the south interface, and the control plane and application plane communicate through the north interface. The control plane is used to generate routes and the data plane is used to forward data.

As the main development direction of the next generation Internet, SDN realizes the rapid routing and data forwarding through the separation of the data plane and control plane. With the continuous application of streaming media technology, the network will face huge traffic challenges, and the advantages and disadvantages of routing algorithm determine the forwarding efficiency of data flow. Therefore, how to make SDN controller obtain the network state in real time and calculate the optimal path through an efficient routing algorithm is the hotspot of SDN network research [23].

2.2 | SDN controller

The SDN control plane is composed of one or more SDN controllers. It is the control centre of the network, which centrally manages the underlying network switching equipment, monitors the status, makes forwarding decisions, processes and schedules the flow of the data plane, and opens multiple levels of programmable capabilities to the upper applications. The control plane mainly interacts with the data plane through the south interface protocol and uses the uplink channel to uniformly monitor and count the information reported by the underlying switching equipment to realize link discovery and topology management. The downstream channel is used to implement unified control over network equipment and to realize strategy formulation and flow table item distribution. It also provides flexible network resource abstraction for upper business applications and resource management systems through the north interface. The controller sends packet_out message with LLDP (Link Layer Discovery Protocol) packets, according to the received packet_in message sent by the switcher, obtains switch status information, realizes network address, VLAN, route learning, monitors switch working status, and completes the network topology view update. Then, the flow table is generated according to the global network resource view and a certain customisation strategy, and the flow table is distributed to the network devices in the data plane. The controller sends packet_Out message to discover the device and the port corresponding to the device. The specific interaction process is shown in Figure 2.

3 | PROBLEM FORMULATION

3.1 | Ant colony algorithm

The nt colony algorithm (ACA) is a bionic method, which simulates the foraging behaviour of ant colonies [24, 25]. During the foraging process, ants will leave pheromones on the path they pass. When many ants pass the same path, the pheromone concentration on the path will increase, the

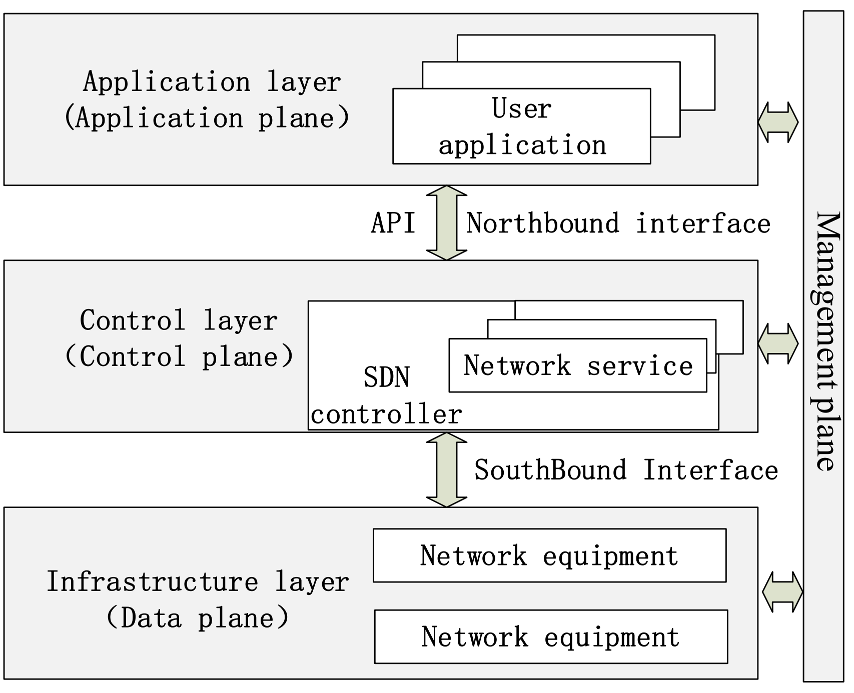

FIGURE 1 Soft define network (SDN) network architecture diagram

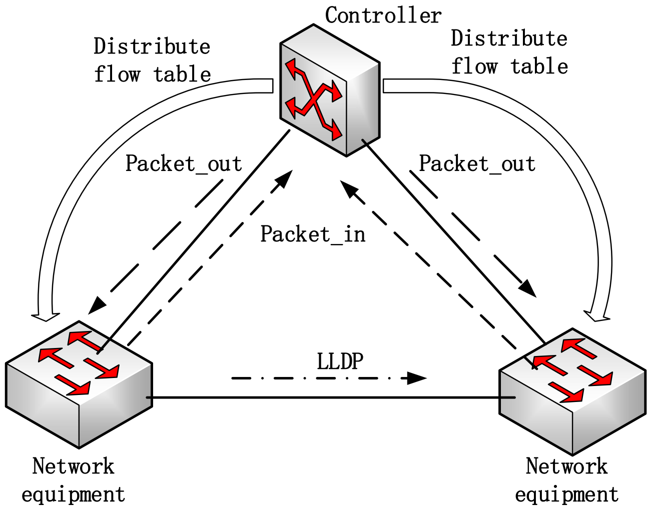

FIGURE 2 Interaction process between controller and network equipment

probability of subsequent ants choosing the path will increase, and ants can dynamically adjust the path according to changes in the environment; finally, the whole ant colony tends to the optimal path.

In a graph with n nodes, set the starting point and ending point, place m ants at the starting point in the graph, and the probability calculation formula of the k ($k = 1, 2\ldots m$) ant for each node may be passed in the next step according to Formula (1) [10, 26].

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{s\in allowed_k}\tau_{is}^{\alpha}(t)\eta_{is}^{\beta}(t)}, j \in \text{allowed}_k \\ 0, \qquad j \notin \text{allowed}_k \end{cases} \quad (1)$$

α is the pheromone heuristic factor in Formula (1), which reflects the influence of pheromone on the selection probability when selecting the moving direction. The larger the parameter is, the closer the connection between ants in the system is. β is the information expectation factor, which reflects the influence of the length of the selected path on the selection probability. If this parameter is too large, the selection will tend to the greedy algorithm of the current shortest path. $p_{ij}^{k}(t)$ represents the probability that ant k moves from node i to node j at time t, and $n_{ij}(t)$ is a heuristic function that reflects the expected degree of ant transfer from element i to element j [27–29].

The ant colony algorithm is a heuristic method to solve the combined optimal path problem. The algorithm has distributed computing, a positive feedback mechanism, good parallelism, robustness, and scalability. Considering that the network routing in SDN is also related to the problem of optimal path selection, the ant colony algorithm can be applied to SDN network for routing calculation. However, the ant colony algorithm is blind in the early stage of the search, and it is time-consuming and prone to stagnation in the late stage of the search. If the ant colony algorithm is directly applied to SDN, the efficiency is low, so it needs to be optimised before use.

3.2 | Improvement of ant colony algorithm

The better the path searched by ant colony, the more likely to find a better shortest path near this path. Therefore, focussing the search scope of the path on the vicinity of the best path can greatly improve the efficiency of the algorithm. The improved algorithm updates the pheromone globally as shown in Formula (2).

$$\tau_{ij}^{\text{new}} = (1-\rho) * \tau_{ij}^{\text{old}} - \varepsilon * \frac{L_{\text{worst}}}{L_{\text{best}}} \quad (2)$$

ε is a parameter introduced into the algorithm in Formula (2). Assuming that all ants complete the construction of the path solution from the starting point to the end point as an iteration, L_{worst} is the worst solution for each iteration and L_{best} is the best solution for each iteration, τ_{ij}^{new} represents the new pheromone value and τ_{ij}^{old} represents the old pheromone value.

After each ant completes the search of the path, it needs to update the path it has travelled. The local update method refers to the basic ant colony algorithm model. The local update method is as follows:

$$\tau_{ij}^{\text{new}} = (1-\rho) * \tau_{ij}^{old} + \tau_{ij} \; \rho \in (0,1) \quad (3)$$

$$\Delta\tau_{ij} = \begin{cases} \dfrac{Q}{L_k}, & \text{if the ant passes the path}(i,j) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where ρ is information volatilisation coefficient in Formula (3), L_k represents the path distance obtained by the ant k, and Q represents the strength of pheromone, and τ_{ij}^{old} is the change of pheromone.

3.3 | Genetic algorithm

The genetic algorithm is a search optimization method based on the evolutionary genetic mechanism [30, 31]. It is a computational model of biological evolution process that simulates the natural selection and genetic mechanism of Darwin's biological evolution theory. It is a method to search for the optimal solution by simulating the natural evolution process. The algorithm expresses the solution of the problem as a chromosome, generates a set according to a certain coding method in the algorithm, and then selects, crosses, and mutates to simulate the survival of the fittest, evolutionary reproduction, and gene mutation in the biological world, so as to achieve a high-quality optimal solution [7, 32–34].

The genetic algorithm has good parallelism and strong versatility, and the algorithm has good global optimization and stability, and simple operation, but it is also easy to fall into the local optimal solution when it solves to a certain range [30, 35–37].

4 | OPTIMIZATION AND FUSION OF INTELLIGENT ALGORITHM AND GENETIC ALGORITHM

Ant colony algorithm and GA are two intelligent bionic algorithms that are widely used at present and have been applied to related fields, such as scientific research and engineering [38–41]. The positive feedback mechanism of ant colony algorithm makes it have better global optimization ability and distributed parallel computing ability, but in the initial stage of the search, due to the lack of pheromone, the solution efficiency is low. The genetic algorithm is fast in the initial stage of search and suitable for large-scale search, but it will spend a lot of time on redundant iteration because it cannot make full use of the feedback information in the system in the later stage of the search. In order to overcome the shortcomings of the two intelligent algorithms, ant colony and GA can be effectively fused.

4.1 | Integration strategy

Through a lot of analyses, research studies, and experimental demonstrations, it is found that the overall situation of ant colony and GA in solving speed and time is shown in Figure 3.

It can be seen from Figure 3 that the GA has a fast solution speed in the initial $t_0 \sim t_c$ time period of search. After the t_c time, the efficiency of the GA begins to decline rapidly. At the initial stage of ant colony algorithm search, due to the lack of pheromone information, the efficiency is low at $t_0 \sim t_c$ time. After t_c time, the efficiency of the ant colony algorithm begins to rise rapidly and finally reaches a higher stable state. The idea of the fusion algorithm is to initialise the pheromone of the ant colony algorithm with the optimal solution generated by the GA in the early stage of the fusion algorithm. In the later stage of the fusion algorithm, the fast convergence speed of the ant colony algorithm is used to find the optimal path. In this way, it combines the advantages of high speed in the early stage of GA and high efficiency in the later stage of ant colony algorithm. In the GA stage, the evolution rate of the sub-generation population in the iteration process is counted. If the evolution rate of consecutive n generations is less than the minimum evolution rate of the preset iteration times, the GA terminates and the ant colony algorithm executes.

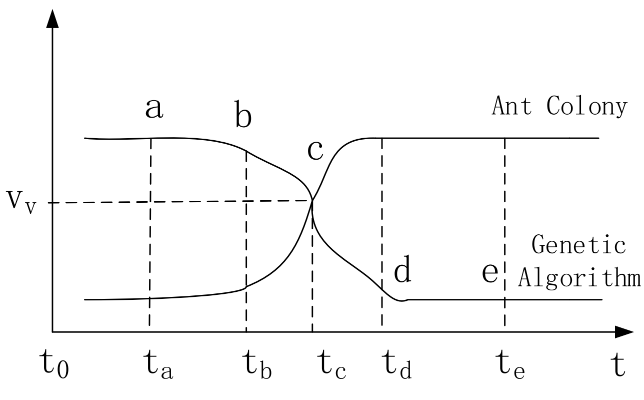

FIGURE 3 Speed time curve of ant colony algorithm and Genetic algorithm (GA)

4.2 | Fusion algorithm process

In the fusion algorithm, set the minimum number of iterations of the GA as $G_{\min}$, the maximum number of iterations as $G_{\max}$, and the minimum evolution rate as G_{ratio}. When the evolution rate of continuous G_{end} generations within the given number of iterations is lower than G_{ratio}, the GA search will be terminated, and the information obtained by the GA will be used to initialise the initial value of pheromones in the ant colony algorithm, which will be transferred to the ant colony algorithm for the solution. The flow chart of fusion algorithm is shown in Figure 4; the steps of the algorithm are as follows:

Step 1 initialise the crossover probability pc, mutation probability pm, and the maximum evolution algebra $G_{\max}$, the minimum evolution algebra $G_{\min}$, the minimum evolution rate G_{ratio}, and the evolution end algebra G_{end}.

Step 2 set the population size as S to get the initial population G, make $G_{\min} < G < G_{\max}$, code according to the actual problem, determine the fitness function and calculate the fitness value of individuals in the population.

Step 3 decode the population individuals and perform selection, crossover and mutation operations. Compare the new individual with the individual in the original parent population, replace the advantages and disadvantages of the individual according to the results, and select the excellent individual as the new next generation of offspring.

Step 4 if $G_{\min} < G < G_{\max}$ and the evolution rate of $G_{\text{end}} > G_{\text{ratio}}$, turn to step 3, otherwise turn to step 5.

Step 5 initialise the initial value of pheromone of the ant colony algorithm with the better solution generated by the genetic algorithm. Set the maximum cycle times of ant colony algorithm as $N_{\max}$, the number of ants as m, and the cycle times k as 0.

Step 6 each ant selects the next node according to the state movement rule Formula (1).

Step 7 when ant k reaches the end point, update the information concentration on the road section it passes by according to Formulas (2–4).

Step 8 repeat step 6 and step 7 until all ants reach the destination.

Step 9 update the worst path length and its included road segment information of this iteration, and the globally optimal path length and its included road segment information. Reset the position of m ants to the starting point and empty the taboo table.

Step 10 if the number of cycles $k > N_{max}$, the programme ends and the optimal path is output, otherwise go to step 6.

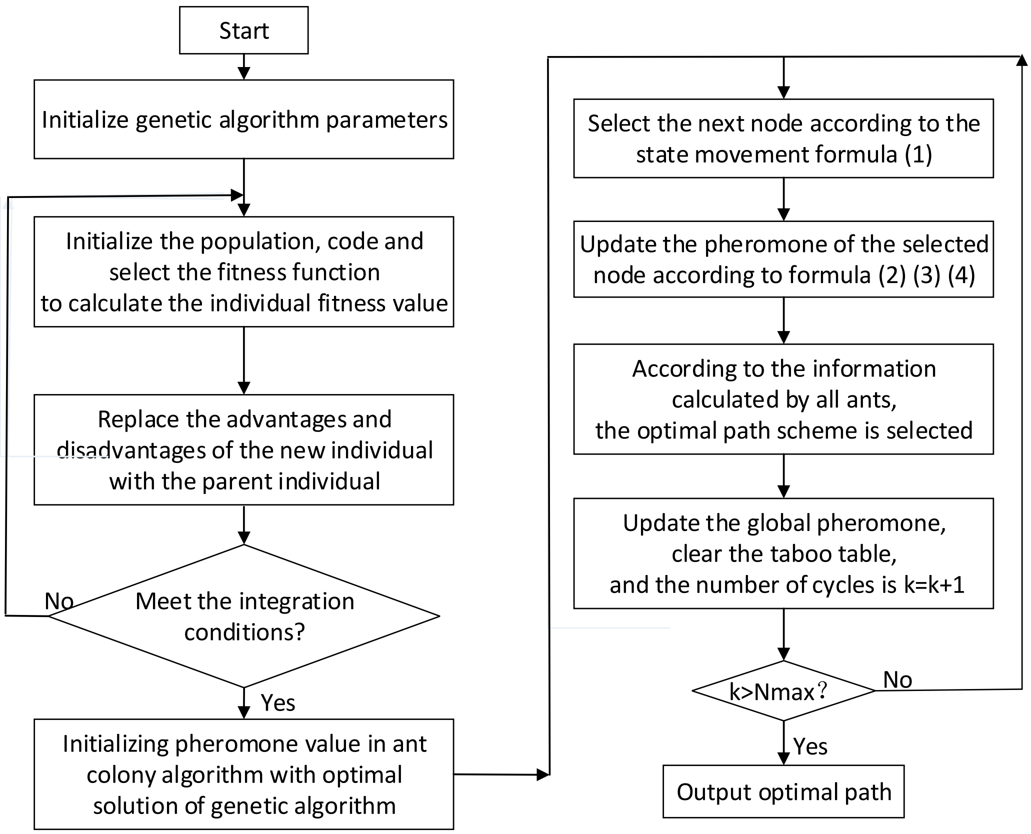

FIGURE 4 Flow chart of fusion algorithm

5 | PERFORMANCE EVALUATION

5.1 | Experiment design

In this paper, the mininet simulator is used to construct the network topology to simulate the real network environment. The network topology adopts the fat tree network topology of four pods, and the Ryu controller is used as the SDN controller. The switch adopts OpenFlow switch. The main reason for using fat tree topology is that there are multiple available links between any two points as shown in Figure 5.

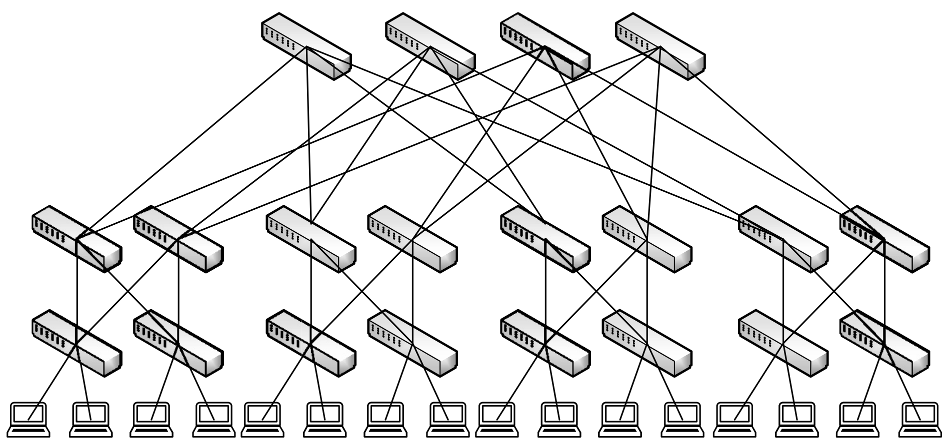

FIGURE 5 Experiment topology

The experiment is compared from four aspects: packet loss rate, average delay, network throughput, and resource utilization. In this paper, the network topology uses the iperf tool to generate analog traffic, uses the ping command to test, and has four data sending ends and four data receiving ends. Conduct 10 experiments for each case and then take the average value of the experimental results. The experiment simulates that the host sends large flows to the network randomly and defines the proportion of the number of hosts sending large flows to the network as the network load. The algorithm in this paper is compared with the equivalent multipath algorithm (ECMP), ant colony algorithm (ACA), GA, and particle swarm optimization algorithm (PSO). The effectiveness of the algorithm proposed in this paper (ACA-GA) is verified by analysing the experimental results. The parameter settings of each algorithm are shown in Table 1.

5.2 | Link utilization comparison

Network link utilization refers to the ratio of the number of links used in data forwarding to the total number of links. Under the same load, the higher the link utilization, the more uniform the network traffic distribution, and the higher the efficiency of the routing algorithm. Figure 6 shows the comparison results of link utilization in the network. As the network load increases, the link utilization gradually increases. ECMP distributes the data packets of the same destination node evenly to multiple equivalent paths according to a specific distribution method. Therefore, after the load reaches a certain degree, the algorithm lacks the judgement of the difference in network links, so its link utilization is the lowest. When the network load reaches a certain degree, the link utilization rate of the ant colony algorithm, GA, and particle swarm algorithm for the calculation of network link load is higher than ECMP. The ant colony algorithm and genetic fusion algorithm proposed in this paper can give full play to the fast search speed of the GA in the early stage and the positive feedback mechanism of the ant colony algorithm in the later stage. Therefore, the link utilization rate has been greatly improved compared with the particle swarm algorithm, ant colony algorithm, and GA. Compared with the ECMP algorithm, when the load increases to 1, the link utilization increases by about 30%.

TABLE 1 Parameter values of each algorithm

Parameter name	Parameter value	Algorithm
Population size	20	Heredity and ant colony
Crossover probability	0.9	Heredity
Mutation probability	0.01	Heredity
Weighting factors ω1 and ω2	1	Heredity
α, β, ρ	2, 6, 0.3	Ant colony
ω, c1, c2, τ1, τ2	1, 0.5, 0.5, 0.5	Particle swarm

5.3 | Comparison of average network throughput

The average throughput of the network refers to the average throughput of the network system in unit time, which directly reflects the transmission performance of the network. Figure 7 shows the comparison of average throughput. It can be seen that when the number of data streams is less than 60, the throughput of each algorithm is the same and shows a linear increasing trend. However, with the increase of data traffic, the gap between the network throughputs of the five algorithms widens. When it reaches 120, the throughput of the ECMP algorithm reaches saturation. This is because when the traffic increases, the ECMP algorithm is easy to cause network congestion, resulting in its average throughput reaching a

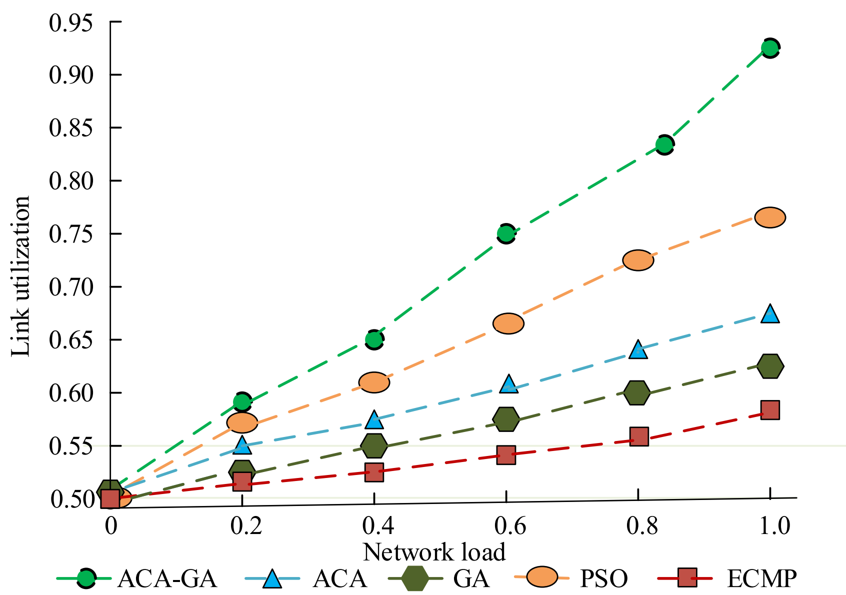

FIGURE 6 Comparison of link utilization

saturation value. As for other intelligent algorithms, the network throughput increases linearly, and with the increase of traffic, the ant colony algorithm and genetic fusion algorithm proposed in this paper increase the average network throughput because the GA searches the path quickly in the early stage and the ant colony algorithm produces the optimal path quickly in the late stage. In addition, among the five algorithms, the throughput is nearly 6 times higher than the ECMP algorithm with the lowest efficiency. Compared with the particle swarm optimization algorithm, the throughput is also improved by nearly 30%.

5.4 | Delay comparison

Compare the average packet delay of the five algorithms by gradually increasing the number of bytes at the sending end. The simulation results are shown in Figure 8. The abscissa represents the number of bytes sent by the data sending end, and the ordinate represents the average packet delay. As can be seen from Figure 8, as the number of transmissions gradually increases, the average packet delay of the five algorithms gradually increases. When the number of bytes sent by the sender is 4000, the average delay of the five algorithms has little difference. This is because when the number of bytes sent by the sender is small, the link load is relatively light, so the average delay difference is small. With the gradual increase of the number of bytes sent by the sender, the ECMP algorithm increases rapidly. The reason is that the ECMP algorithm only allocates network traffic according to the equivalent path, which is prone to congestion. By reasonably setting the weight coefficient and learning factor, the particle swarm optimization algorithm can make the calculation path better than the ant colony algorithm and GA. By using the

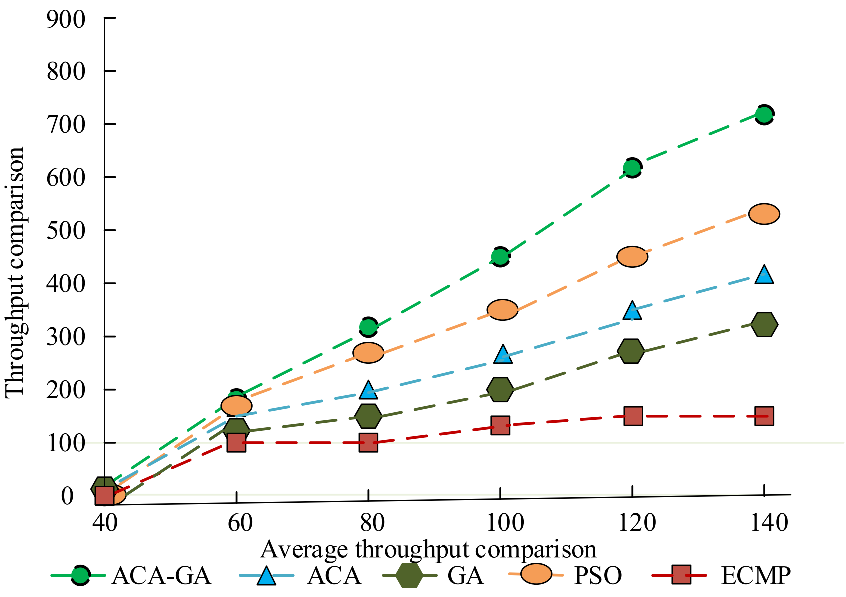

FIGURE 7 Average throughput comparison

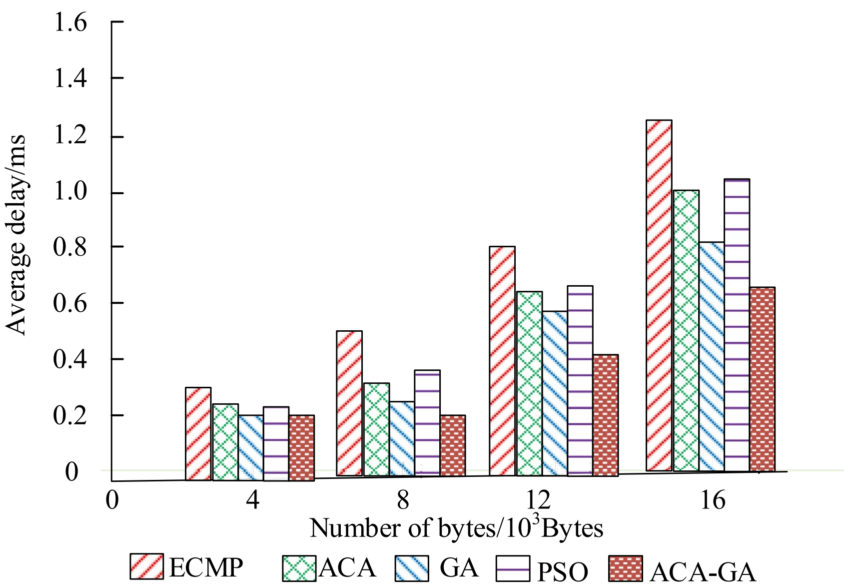

FIGURE 8 Average delay

ant colony genetic fusion algorithm proposed in this paper, because it combines the advantages of the ant colony algorithm and GA, the generated route is fast and the path is better, so it improves the data forwarding rate and reduces the data delay.

6 | CONCLUSIONS

Aiming at the phenomenon of the unreasonable path and easy network congestion in the process of data forwarding in SDN network, this paper combines the advantages of the ant colony and GA, integrates the ant colony and GA, and applies the fused algorithm to SDN network. Through simulation tests, it can be seen that the algorithm ACA-GA proposed in this paper improves the utilization of network links compared with the traditional ECMP algorithm and the single intelligent algorithms ACA, GA, and PSO. The average throughput of the network increases, and the average delay of the network decreases. By applying the routing strategy proposed in this paper, the SDN routing table converges quickly and finds a better route. Therefore, it has a certain application value in the future SDN network routing.

AUTHOR CONTRIBUTIONS

Kaixin Zhao designs paper and experiment. Yong Wei conducts paper simulation test. Yang Zhang arranges and draws icons.

ACKNOWLEDGEMENT

The project is supported by Henan Science and Technology Key Project (222102240046).

CONFLICT OF INTEREST

The authors declare that there are no conflicts of interest.

DATA AVAILABILITY STATEMENT

All the experimental data used to support the findings of this study are included within the article.

ORCID

Kaixin Zhao https://orcid.org/0000-0002-8375-7687

REFERENCES

1. Rafal, S.: Improving Ant Colony Optimization efficiency for solving large TSP instances. Appl. Soft Comput., 108653 (2022)
2. KimKwan, J., et al.: Early prediction of sepsis onset using neural architecture search based on genetic algorithms. Int. J. Environ. Res. Publ. Health. 19(4), 2349 (2022). https://doi.org/10.3390/ijerph-tnqh_9;19042349
3. Amir, A., Saeedreza, A., Tayebeh, A.: A novel optimized routing algorithm for QoS traffic engineering in SDN-based mobile networks. ICT Express. 8(1), 130–134 (2022)
4. Fang'ai, C., Xu, Y.: Building performance optimization for university dormitory through integration of digital gene map into multi-objective genetic algorithm. Appl. Energy. 307 (2022)
5. Meihua, Wu, et al.: Path planning of mobile robot based on improved genetic algorithm. In: Proc of Chinese Automation Congress, pp. 6696–6700. IEEE Press (2017)
6. Jiaxyu, N., et al.: A best-path-updating information-guided ant colony optimization algorithm. Inf. Sci.(433), 142–162 (2018)
7. Ting, Li, Xie, Q., Zhang, H.: Design of college scheduling algorithm based on improved genetic ant colony hybrid optimization. Secur. Commun. Network. 2022 (2022)
8. Tang, Y.Lu., Zamira, M.: Research on the flow space planning model of a classical garden based on an ant colony optimization algorithm. J. Math. 2022 (2022)
9. Deng, X., Zhang, D.: Trajectory optimization of interceptor missile based on hybrid genetic algorithm. In: Proceedings of 2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT 2021), vol. 2021, pp. 1292–1296. https://doi.org/10.26914/c.cnkihy.2021.065600
10. Panizo-LLedot, A., Bello-Orgaz, G., Camacho, D.: A Multi-Objective Genetic Algorithm for detecting dynamic communities using a local search driven immigrant's scheme. Future Generat. Comput. Syst. 110(prepublish), 960–975 (2020). https://doi.org/10.1016/j.future.2019.10.041
11. Zheng, Ma, et al.: Research on the on-demand scheduling algorithm of intelligent routing load based on SDN. Int. J. Internet Protoc. Technol. 14(1) (2021)
12. Abbas El-Hefnawy, N., Abdel Raouf, O., Askr, H.: Dynamic routing optimization algorithm for software defined networking. Comput. Mater. Continua (CMC) 70(1), 1349–1362 (2022). https://doi.org/10.32604/cmc.2022.017787
13. Zhang, He, et al.: Picking path planning method of dual rollers type safflower picking robot based on improved ant colony algorithm. Processes. 10(6), 1213 (2022). https://doi.org/10.3390/pr10061213
14. Mehrdad Ahmadi, K., et al.: Optimal coordination of PSS and SSSC controllers in power system using ant colony optimization algorithm. J. Circ. Syst. Comput. 31(04) (2022)
15. Wang, J., Wu, X.: Personalized original ecotourism route recommendation based on ant colony algorithm. Wireless Commun. Mobile Comput., 2022–9 (2022). https://doi.org/10.1155/2022/6783567
16. Li, C., et al.: Path planning for mobile robots based on an improved ant colony algorithm with Gaussian distribution. J. Phys. Conf. 2188(1), 012005 (2022). https://doi.org/10.1088/1742-6596/2188/1/012005
17. DiDebora, C., et al.: A novel ant colony algorithm for solving shortest path problems with fuzzy arc weights. Alex. Eng. J. 61(5), 3403–3415 (2022)
18. Deng Jian, J., Zhang, L., Wu, Da.: Structural optimization design based on improved ant colony algorithm. J. Phys. Conf. 2173(1) (2022)
19. Zhao, H., Zhao, J.: Improved ant colony algorithm for path planning of fixed wing unmanned aerial vehicle. In: MATEC Web of Conferences, 355 (2022)
20. Li, Y., et al.: Mobile robot path planning based on angle-guided ant colony algorithm. Int. J. Swarm Intell. Res. (IJSIR). 13(1), 1–19 (2022). https://doi.org/10.4018/ijsir.302603
21. Ge, D.: Optimal path selection of multimodal transport based on ant colony algorithm. J. Phys. Conf. 2083(3), 032011 (2021). https://doi.org/10.1088/1742-6596/2083/3/032011
22. Lin, M., et al.: Ant colony algorithm for container-based microservice scheduling in hybrid cloud. J. Phys. Conf. 1994(1), 012028 (2021). https://doi.org/10.1088/1742-6596/1994/1/012028
23. Flauzac, O., et al.: An SDN approach to route massive data flows of sensor networks. Int. J. Commun. Syst. 33(7), e4309 (2020). https://doi.org/10.1002/dac.4309
24. Gilani, S.S.A., et al.: SDN mesh: an SDN based routing architecture for wireless mesh networks. IEEE Access. (99)1 (2020)
25. Allah Bukhsh, Z., Stipanovic, I., Doree, A.G.: Multi-year maintenance planning framework using multi-attribute utility theory and genetic algorithms. Eur. Transport. Res. Rev. 12(1), 3 (2020). https://doi.org/10.1186/s12544-019-0388-y
26. Jia, L., et al.: Solving the dynamic energy aware job shop scheduling problem with the heterogeneous parallel genetic algorithm. Future Generat. Comput. Syst. 108(prepublish), 119–134 (2020)
27. Yan, C., et al.: Improved adaptive genetic algorithm for the vehicle insurance fraud identification model based on a BP neural network. Theor.

Comput. Sci. 817(prepublish), 12–23 (2020). https://doi.org/10.1016/j.tcs.2019.06.025
28. Zarei, B., Meybodi, M.R.: Detecting community structure in complex networks using genetic algorithm based on object migrating automata. Comput. Intell. 36(2), 824–860 (2020). https://doi.org/10.1111/coin.12273
29. Hanna, G., et al.: A simulation-optimisation genetic algorithm approach to product allocation in vending machine systems. Expert Syst. Appl. 145(C), 113110 (2020)
30. Rashid, S.Ur., Walia, M.: Optimization of data acquisition in wireless sensor networks based on ant colony algorithm and genetic algorithm. J Res Sci Eng. 3(5) (2021)
31. KueiHsiang, C., Muhammad Nursyam, R.: A hybrid MPPT controller based on the genetic algorithm and ant colony optimization for photovoltaic systems under partially shaded conditions. Energies. 14(10), 2902 (2021)
32. Kristjanpoller, F., et al.: Fleet optimization considering overcapacity and load sharing restrictions using genetic algorithms and ant colony optimization. Artif. Intell. Eng. Des. Anal. Manuf. 34(1), 104–113 (2020). https://doi.org/10.1017/s0890060419000428
33. Blagoveshchenskaya, E.A., Mikulik, I.I., Strüngmann, L.H.: Ant colony optimization with parameter update using a genetic algorithm for travelling salesman problem. In: CEUR Workshop Proceedings, vol. 2803, pp. 20–25 (2020)
34. Jain, A., et al.: An ensemble of bacterial foraging, genetic, ant colony and ParticleSwarm approach EB-GAP: a load balancing approach in CloudComputing. Recent Adv. Comput. Sci. Commun. 15(5) (2022)
35. Ni, W., et al.: Neural network optimal routing algorithm based on genetic ant colony in IPv6 environment. Comput. Intell. Neurosci. 2021, 3115704–13 (2021). https://doi.org/10.1155/2021/3115704
36. Xiang, Ma., Chekole, W.T.: Research on communication optimization of power carrier sensor control network based on ant colony algorithm. J. Sens., 2022 (2022)
37. Charis, N., Lyridis Dimitrios, V.: A comparative study on Ant Colony Optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles. Ocean. Eng., 255 (2022)
38. Zhao, H., Zhang, C.: An ant colony optimization algorithm with evolutionary experience-guided pheromone updating strategies for multi-objective optimization. Expert Syst. Appl. 201 (2022)
39. Kai, F., et al.: Path optimization of agricultural robot based on immune ant colony: B-spline interpolation algorithm. Math. Probl Eng., 2022 (2022)
40. Wang, Q., et al.: Continuous space ant colony algorithm for automatic selection of orthophoto mosaic seamline network. ISPRS J. Photogrammetry Remote Sens. 186, 201–217 (2022). https://doi.org/10.1016/j.isprsjprs.2022.02.011
41. Li, C., Jiang, K., Luo, Y.: Dynamic placement of multiple controllers based on SDN and allocation of computational resources based on heuristic ant colony algorithm. Knowl. Base Syst. 241 (2022)

How to cite this article: Zhao, K., Wei, Y., Zhang, Y.: An ant colony genetic fusion routing algorithm based on soft define network. IET Netw. 1–10 (2022). https://doi.org/10.1049/ntw2.12056