

## An ant colony algorithm and improvement for the state transition diagram search

Xiao Da<sup>1, a</sup>, Shengli Liu<sup>1, b</sup>, Yuefei Zhu<sup>1, c</sup>, Xiadong Wang<sup>2, d</sup>, Li Rui<sup>1</sup>

<sup>1</sup>Computer network department, ZhengZhou Information Science University, 450001, ZhengZhou, HeNan, China

State Key Laboratory of Mathematical Engineering and Advanced Computing, 450001, ZhengZhou, HeNan, China

<sup>2</sup>National Key Laboratory of Science and Technology on Information System Security, 100101, Beijing, China

<sup>a</sup>email :shuttle12@163.com, <sup>b</sup>email: liu\_shengli9163@yahoo.com.cn, <sup>c</sup>email: Zyf0136@sina.com, <sup>d</sup>email:wangdx3958@163.com

**Keywords:** The state transition diagram; Ant colony algorithm; Sequential logic chip; Shortest path; Fast state test;

**Abstract.** Aiming at the state transition quick extraction problem for sequential logic chip, has designed a state transition graph extraction algorithm based on ant colony algorithm model, besides that, the ant colony algorithm model has been introduced in details, and according to the characteristics of the graph state transfer, the ant colony algorithm has been improved, has designed the state transition graph realization method in details oriented to the ant colony algorithm. Experimental tests show that, the ant colony algorithm is designed in this paper can realize fast state transition graph extraction.

### 0 Introduction

The generating principle of the state transition diagram based on ATPG is by means of exerting excitation signals on the tested chip, and record the response signals of the chip. By statistic data of the excitation signal and response signal that chip received, we analysed the relationship between the input and output to restore the state transition diagram of the chip. When state transition diagram generated, the excitation signal should be ensured that it can cover all input space of tested chip, in order to tested chip can traverse to all the working conditions. But for the temporal logic chips, chip state will change after exerting excitation. In order to conduct follow-up tests, the chip need to be driven from the current state to a new under test state. How to search out the shortest path from state transition diagram affects the efficiency of the test. Therefore, after researching, this paper designed a shortest path search algorithm based on an ant colony algorithm.

### 1 A model of ant colony algorithm

$n$  represents state number of the current state transition diagram;

$m$  represents the number of ants selected from the process of ant colony algorithm;

$i$  represents node order number, is the subscript of current state ( $0 \leq i \leq n-1$ );

$j$  represents node order number, is the subscript of target state points expected to move to ( $0 \leq j \leq n-1$ );

$k$  represents the order number of ants ( $0 \leq k \leq m-1$ );

$W_{ij}$  represents the path length between state points  $S_i$  and  $S_j$  ;

$\eta_{ij}$  represents the visibility of path  $(i, j)$ , it is also an important factor that the path is chosen by ants;

$\lambda_{ij}$  represents the pheromone strength of path  $(i, j)$ ;

$\Delta\lambda_{ij}^k$  represents amount of pheromone on unit length that ant k left on the path ;

$P_{ij}^k$  represents the probability that ants choose path  $(i, j)$  to move from the current state  $S_i$  to  $S_j$ ;

$t$  represents the current moment, when building a model of ant colony algorithm, the value of  $t$  is time point that time interval is equal , in order to calculate conveniently;

$\alpha$  represents Information stimulating factor;

$\beta$  represents path expectation stimulating factor;

$N(si)$  represents transfer gather that the current state can transfer next;

At the time  $t$ , the probability calculation formula that ants from node  $i$  to  $j$  is:

$$P_{ij}^k(t) = \begin{cases} \frac{[\lambda_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{r \in N(si)} [\lambda_{ir}(t)]^\alpha [\eta_{ir}(t)]^\beta} & j \in N(si) \\ 0 & j \notin N(si) \end{cases}$$

$$\lambda_{ij}(t+1) = (1-\rho)\lambda_{ij}(t) + \Delta\lambda_{ij}(t), \quad \Delta\lambda_{ij}(t) = \sum_{k=0}^m \Delta\lambda_{ij}^k(t), \quad \eta_{ij}(t) = \frac{1}{d_{ij}} * C$$

## 2 The improvement of ant colony algorithm

### 2.1 The improvement of transfer path expectation $\eta_{ij}(t)$

Design the ant colony algorithm in the state transition diagram which logic chip correspond to, in order to achieve the shortest path search. It is very different with traditional TPS (Travelling Salesman Problem). All the distance between the adjacent nodes is equal in state transition diagram. The distance between the adjacent nodes is noted  $d_{ij} = 1$ , so the calculation of transfer path expectation  $\eta_{ij}(t)$  will be more simple.

$$\eta_{ij}(t) = \frac{1}{d_{ij}} * C = C$$

### 2.2 the calculation of pheromone $\Delta\lambda_{ij}^k(t)$ that ant k left

There are different kinds of calculation method for  $\Delta\lambda_{ij}^k(t)$ . The Italian scholar M.Dorigo has gave three kinds of calculation model, called Ant Quantity System, Ant Density System and Ant Cycle System. The calculation method of the three model corresponding  $\Delta\lambda_{ij}^k(t)$  as follows:

$$\text{Ant Quantity System: } \Delta\lambda_{ij}^k(t) = \begin{cases} \frac{Q}{d_{ij}} & (i, j) \in L \\ 0 & (i, j) \notin L \end{cases}$$

$(i, j) \in L$  represents the path  $(i, j)$  that ants have passed,  $(i, j) \notin L$  represents the path  $(i, j)$  that ants have not passed.

$$\text{Ant Density System: } \Delta\lambda_{ij}^k(t) = \begin{cases} Q & (i, j) \in L \\ 0 & (i, j) \notin L \end{cases}$$

$$\text{Ant Cycle System: } \Delta\lambda_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & (i, j) \in L \\ 0 & (i, j) \notin L \end{cases}$$

$Q$  represents the pheromone that an ant carries,  $L_k$  represents the path that ants walked to complete a cycle path. In Ant Quantity System, when calculating the pheromone on the path  $(i, j)$ , we use the average of pheromone on the path. The longer the path between path  $(i, j)$ , the less pheromone increasing. Ant Density System used pheromone values directly that ants carried, equivalent to the

ant release pheromones at the same rate every moment, and the value has nothing to do with the length of path  $(i, j)$ . Pheromone that Ant Cycle System used is the average of the circular path. The simulation results show that the convergence rate of the Ant Cycle System faster. But the Ant Cycle System before calculation, need to get a complete circulation path. So for large-scale state transition diagram, it is not too easy to implement.

According to the characteristics of state transition diagram, the distance between adjacent nodes is constant 1. Therefore in the process of calculation of the adjacent nodes  $(i, j)$ , Ant Quantity System and Ant Density System get the same calculation formula. And in order to ensure that the calculating results ratio is no difference, quantitative coefficient  $\mu$  is introduced, the  $\Delta\lambda_{ij}^k(t)$  calculating formula is as follows:

$$\Delta\lambda_{ij}^k(t) = \begin{cases} Q^* \mu & (i, j) \in L \\ 0 & (i, j) \notin L \end{cases}$$

### 2.3 Improvements of convergence performance

The core of ant colony algorithm is calculating the value  $P_{ij}^k(t)$ , and all the ants transfer path on the basis of the value. The calculation of the value depends on the  $\lambda_{ij}(t)$  and  $\eta_{ij}(t)$ , and the variation regularity of these two numerical directly affects the convergence performance of the ant colony algorithm. In the search problem of state transition diagram  $\eta_{ij}(t)$  has simplified as a constant, so the convergence of the ant colony algorithm will be decided by  $\lambda_{ij}(t)$ . In the update model of calculating  $\lambda_{ij}(t)$ , selection is the Ant Quantity System model, and convergence rate of the model is slower than Ant Cycle System. In order to assure the convergence speed of algorithm, we need to improve the algorithm's convergence performance.

Once the ants  $g_1$  reached the destination node, indicate that the way which ants have just gone is better than other paths and mark ant  $g_1$  as the backbone ants. Although along the path of the far may also can reach the destination node, it seems unreasonable that we regard the path that the ants arrived as the optimal path. But, in fact, the moment is better relatively speaking to those who can not reach the destination node path. And if there is other ant  $g_2$  along the shorter path to reach the destination node,  $g_2$  will earlier to find the optimal path than  $g_1$ . Therefore, marking backbone of ant  $g_1$  that it can find the optimal path does not affect  $g_2$  become backbone of ants earlier.

If ant  $g_1$  is marked as the backbone of ants, show that the path should be given the key consideration. The pheromone that ant left in the circulation path should be worthy of reference for other ants particularly. But because ant colony algorithm is not saved the path that they passed in the process of search and if we save walking paths that an ant passed, it will consume too much computing resources. So after the ant  $g_1$  being marked as the backbone ants, fix all the pheromone on the path that the ant walked is not easy to implement. And analysis from the angle of probability, can find that it is higher possibility that the ant return path is also the original path after it arrived at the destination node. Unless there are other ants to correct some nodes on the path making the ants optional path be better. So strengthening the pheromone of the ant backbone can adjust to strengthen the pheromone that the ant carry later.

Therefore, in order to make ant colony algorithm converge faster, should set a variable for pheromone that the ant carry to describe. The modified calculation formula is as follows:

$$\begin{cases} Q_{ci+1} = (1 + \gamma)Q_{ci} \\ Q_{c0} = Q \end{cases} \quad i = 0, 1, 2, \dots$$

$\gamma$  as pheromone strengthening coefficient of backbone ants.  $c_i$  represents that the current path loops  $i$ .

By means of strengthening pheromone of the backbone ant pheromone constantly, will make pheromone on the path of backbone ant being stronger than any other paths obviously. And the

strengthening the path will attract more ants to going along the path, so the algorithm convergence speed will be improved obviously. The range of the value of  $\gamma$  value will directly affect the convergence performance, the big value may lead to algorithm into the local optimal solution prematurely.

### 3 The experiment test

In the implementation of ant colony algorithm, six parameters need to set, they are  $m$ 、 $\alpha$ 、 $\beta$ 、 $\rho$ 、 $Q$ 、 $\mu$  and  $\gamma$ . Different setting of these parameters will directly affect the search results and search time of ant colony algorithm. At present there is not a good theoretical support for parameter, and in the common ant colony algorithm implementation, the values of these parameters are also based on experience and simulation experiments. First give an experiential range of each parameter, then test in the simulation environment and choose appropriate value for each parameter.

At present the common range of each parameter are  $m \approx n/2$ 、 $0.05 \leq \alpha \leq 0.5$ 、 $0.05 \leq \beta \leq 5$ 、 $0.1 \leq \rho \leq 0.8$ 、 $10 \leq Q \leq 10000$ 、 $0.001 \leq \mu \leq 0.1$  and  $0.05 \leq \gamma \leq 0.4$ . In this experiment, the selected parameters after simulation experiment are  $m = 0.3n$ 、 $\alpha = 0.25$ 、 $\beta = 1$ 、 $\rho = 0.4$ 、 $Q = 2000$ 、 $\mu = 0.02$  and  $\gamma = 0.2$ .

This article selects three typical logic circuits to get the state transition diagram. Using ant colony algorithm search the shortest path between two nodes that selected randomly in the diagram and each state transition diagram test 10 times. After the simulation test, get the test results as table 1. The test result shows that the ant colony algorithm designed in this paper can find approximate shortest path between any two nodes effectively in the state transition diagram.

Table 1 Test results of ant colony algorithm

state transition diagram	average of path length	of optimal solution	worst solution	shortest path length	optimal approximate degree	average times of iterations
1	82.4	76	85	76	1.084	27
2	519.7	501	538	492	1.056	95
3	3823.2	3782	3867	3746	1.021	187

The computation formula of optimal approximation in the table is as follow:

The optimal approximation = the average of path length/the shortest path length

Using ant colony algorithm that this paper designed to test and the results show that the results of the shortest path between two nodes that searched in the state transition diagram are very close to the real shortest path. In three state transition diagrams, their highest optimum approximate degree of the average path length is 1.084 and the minimum is 1.021. If the value is close to 1, it shows that the search path is closer to optimal path. And when search the state transition diagram, if the diagram scale is bigger, the optimal solution we get is closer to the real shortest path. And it can find from the searching results that the optimal solution we found on one occasion is the real shortest path in the searching process of the first state transition diagram. This shows that the ant colony algorithm designed in this paper may find the path just the same as the real shortest path. And even though the optimal path it found is not the real shortest path, we can find from the optimal approximate degree that the difference between result and the real shortest path is not too big.

### 4 Conclusion

Ant colony algorithm is a probability algorithm and the characteristics of probability algorithm is that the result we finally get is the approximate optimal results. Although sometimes the search results we get is not the optimal path, ant colony algorithm is a good compromise in calculation complexity and calculation accuracy for the shortest path search in the state transition diagram.

## Acknowledgment

This work is supported by the National Science-Technology Support Plan Project of China (No. 2012BAH47B01), by the Natural Science Foundation of China (No. 61271252), and by the Municipal Science and Technology Innovation Team Project of Zhengzhou (No. 10CXTD150).

## References

- [1] Dorigo, M.Maniezzo, V.Colormi,A. Ant system:optimization by a colony of cooperating agents. IEEE Transactions on SMC,1996,26(1):8~41.
- [2] Ozgur Baskan, Soner Haldenbilen,Huseyin Ceylan, Halim Ceylan.A new solution algorithm for improving performance of ant colony optimization.Applied Mathematics and Computation, Volume 211,Issue 1,1May 2009.75~84
- [3] Guan-Zheng TAN,Huan HE,Aaron SLOMAN.Ant Colony System Algorithm for Real-Time Globally Optimal Path Planning of Mobile Robots.Acta Automatica Sinica, Volume 33,Issue 3,March 2007.279~285
- [4] Meijuan Gao, Jin Xu, Jingwen Tian.Mobile Robot Global Path Planning Based on Improved Augment Ant Colony Algorithm.Genetic and Evolutionary Computing,2008.WGEC 08.Second International Conference on 25-26 Sept.2008.273~276
- [5] Nattapat Attiratanasunthron, Jittat Fakcharoenphol.A running time analysis of ant Ant Colony Optimization algorithm for shortest paths in directed acyclic graphs. Information Processing Letters, Volume 105, Issue 3,31 January 2008.88~92
- [6] Bin Yu, Zhong-Zhen Yang, Baozhen Yao. An improved ant colony optimization for vehicle routing problem. European Journal of Operational Research, Volume 196,Issue 1,1 July.2009.171~176