Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Active Pair Ant Colony Optimization using Natural Evolution to Solve Symmetric Travelling Salesman Problem (STSP)

Manimozhi*
Applied Electronics Engineering &
Anna University, Tamil Nadu, India

Rajkumar Sampath
Electronics and Communication Engineering &
Anna University, Tamil Nadu, India

Abstract— ***An Active Pair ant system is proposed for the integrative optimization problems to improve Efficiency rate using Natural Evolution. The proposed method is stimulated by the knowledge that there are numerous colonies of ants in the normal world and constructed with the pair colonies of ants. Each colony initiator ant performs search solution using Natural Evolution like Genetic Algorithm to find the minimum time travel path by collaborates with follower ants until no better solution is found. Then the communication between the paired colonies is achieved to build new pheromone distributions for each colony using Ant system (AS) Pheromone update, and the ants start their search procedure again in each separate colony, depends on the new pheromone distribution and Genetic Algorithm. The Two colony Ant system produces typical efficiency rate, so we establish Genetic Population to forebode succeeding Optimal Nodes to minimize the time travel between the nodes to achieve a high Efficiency rate. The proposed algorithm is examined by simulating the Symmetric Traveling Salesman Problem (STSP) with elevated efficiency rate. Research study results show that the proposed method performs better than the Existing scheme.***

Keywords— ***Ant Search, Ant System (AS), Symmetric Travelling Salesman Problem (STSP), Pheromone, Pair ACO, Genetic Algorithm (GA)***

I. INTRODUCTION

In recent years, many research works have been carried to ant colony optimization (ACO) techniques in different domains. It is a quite innovative technique and has been successfully used in many applications especially in computational optimization. An ACO algorithm based on the behaviour of real ants and ant colonies in determining the shortest path between food sources and nests. Ants can communicate with one another through chemicals called pheromones (Phe) [3]. The ants release pheromone on the ground while walking from their nest to food and then go back to the nest. The ants move according to the amount of pheromones, the better-off the pheromone trail on a path is, the more likely it would be followed by other ants. So the shorter path has a higher density of pheromone in probability, ants will influence to choose a shorter path. Through this method, ants will eventually find the shortest path. Artificial ants act like the behaviour of real ants, but can solve much more difficult problem than real ants can.

ACO has been generally applied to solving various combinatorial optimization problems such as Traveling Salesman Problem (TSP), Job-shop Scheduling Problem (JSP), Vehicle Routing Problem (VRP) [6], Quadratic Assignment Problem (QAP), etc. The Traveling Salesman Problem (TSP) is one of the most intensively studied problems in computational mathematics, this method ant to find the best possible way of visiting all the cities exactly once and returning to the starting point. But it's difficult to find minimum tour because of NP-hard [11] problem and unfavourable efficiency rate of ants in optimization. We are developing a solution using ACO for Symmetric TSP to minimize the cost of Travelling between each city visit and obtain maximal efficiency rate. Ant System has a powerful capacity to find out solutions to combinational optimization problems, it has the problems of stagnation and premature convergence and the convergence speed of ACO is very slow. This Problem increases when the travelling size will increase. Therefore, several extensions and improvements are introduced to solve this complexity, So Xiao-Fan Zhou [2] introduced a two colony ant systems to minimize the complexity. This method consists of two traditional Ant systems [8], where each system gives updated pheromone trail in each iteration (ite). These two Ant systems communicate their own pheromones with each other. These updated pheromones create new pheromones for next iterations. This method reduces the premature convergence problem of ant system, but it's not giving stable efficiency rate for all TSP Databases.

In 2010, Marco Dorigo al., [3] has presented single colony cooperative ant system to obtain maximum effective tour based on distance and pheromone distribution of ants. But this algorithm fails to achieve precise solution due to ant traffic and insufficient capable of ants memory. Ants provide a tour result before pheromone table got stable .so the best and worst value of path has big margin and this value affects error and efficiency of whole system. To overcome this problem modern age researcher proposed multi colony or multi objective ant system.

In 2012, Xiao-Fan Zhou al.,[2] has introduced multi colony system to get improved result when compared to single objective ant systems. Each individual colony had their own pheromone table. This table updated at the end of each iteration this time all ants share their pheromone table with each other. So all the ants got knowledge about other ants as well as the colonies. The search solution procedure is doubled here so it will reduce ant traffic and shared information boosts the memory of the ant system. Two Traditional ant systems are used to determine stagnation and premature convergence and the convergence speed, but it's failing to give stable efficiency rate.

So we have combined another well-known optimization approach called *Genetic Algorithm* [4, 5] when communication between two pheromones takes place. This Optimization avoids unwanted tours in total populated cities and reduces average path length simultaneously. Due to this factor we will get maximized Efficiency. In the remainder of the paper, we first describe the travelling salesman problem in section II, and then we describe the traditional ACO and the two colonies ACO in detail in section III. Section IV describes the proposed ACO in details. Then Section V demonstrates the effectiveness by some simulation results and discussion. Finally, a conclusion is given in section VI. All steps are mentioned in Block diagram shown in Fig.1

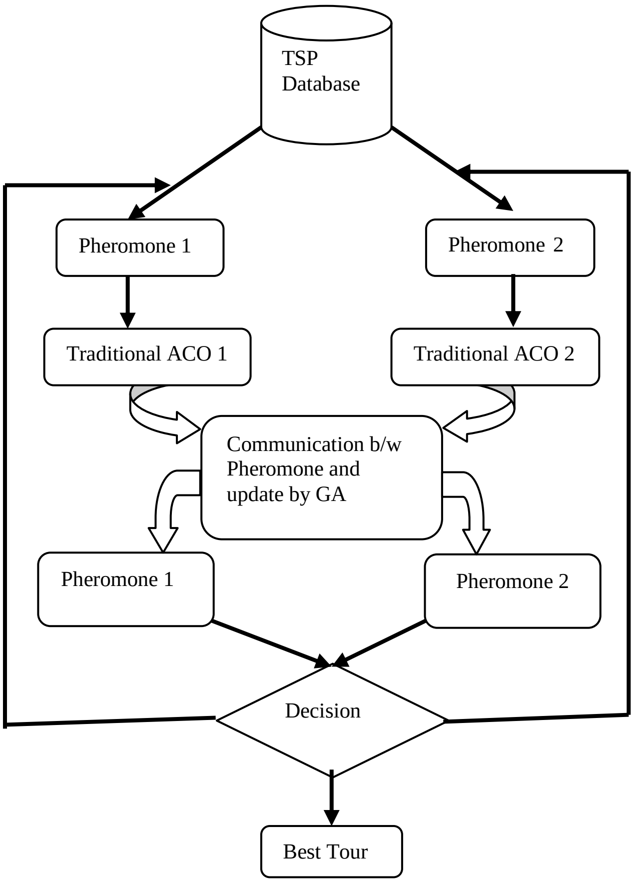


Fig 1: Block Diagram of a TSP problem

II. TRAVELLING SALESMAN PROBLEM

Travelling Salesman Problem (TSP) is one of the most familiar difficult problems of time. A salesperson must visit '**N'** cities, passing through each city only once, beginning from one of the city that is considered as a nest or starting city and returns to it. The cost of the tour depends on total travel length. The problem is to find the order of the minimum cost route, that is, the order of visiting the cities in such a way that the cost is the minimum. This is one of the well-known and extensively studied problems in Discrete or combinational optimization and asks for the shortest round-trip of minimal total cost visiting each given city (node) exactly once. Travelling salesman problem has a NP-hard problem and it is so easy to describe and so difficult to solve. Graph theory defines the problem as finding the *Hamiltonian* cycle [13, 14] with the least weight for a given complete weighted graph. Hamiltonian cycle (tour) provides a closed path tour, which visits every vertex of the graph. TSP trails *Branch and Bound* theory with the help of distance matrices. TSP has two types (i).Symmetric TSP (ii).Asymmetric TSP [9]. For symmetric TSP, the distances between the cities are independent of the direction of traversing the arcs, that is, $D_{ij} = D_{ji}$ for every pair of nodes. In the Asymmetric TSP (ATSP) at least for one pair of nodes i, j we have $D_{ij} \neq D_{ji}$. In this paper, we choose Symmetric TSP. Fig.2 Shows how Travelling salesman problem finds a best solution in symmetric path.

A complete weighted graph G = (N, Ed) can be used to represent a TSP, where N is the set of '**N'** cities and **Ed** is the set of edges (paths) fully connecting all cities. Each edge $(i,j) \in Ed$ is allocated a cost D_{ij} , which is the distance between cities i and j. If i and j same in the network means it's the distance must be zero everywhere. So the distance matrix **D** diagonal must be Zero or infinite.

$D_{ij} = 0 \in i == j$. D_{ij} can be defined in the Euclidean space and is given as follows:

$$D_{ij} = D_{ik}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (1)$$

Distance Matrix must satisfy following 3 conditions: (i) Must be a Square Matrix (ii) Symmetric

(iii) Triangular Equality. TSP follows these 3 properties we should conclude salesperson visits every city in one time in a single tour. Triangularly Equality can represent as follows

$$D_{ij} + D_{jk} >= D_{ik} \qquad (2)$$

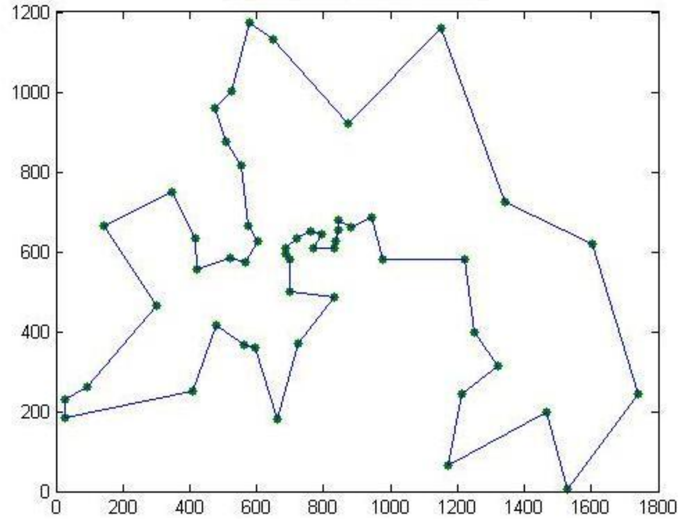


Fig 2: An example of optimal solution of a TSP problem

These are the basic requirements when the symmetric travelling salesman problem is run. The Database satisfies these properties database is ready to proceed with ACO.

III. EXISTING TWO ANT COLONY OPTIMIZATION IN TSP

The complex common activities of ants have been much studied by science, and we are now finding that these natural activities of ant patterns can provide models for solving complex combinatorial optimization problems. The challenge to develop algorithms inspired by one feature of ant activities, the capability to find what we would call shortest paths, has become the field of ant colony optimization (ACO) [7,15], the most successful and widely accepted algorithmic method based on ant activities. Ants system introduced by *Dorigo* [3] in the year of 2012.

Let us consider a symmetric TSP with **N** cities. Let **M** be the total number of ants, assumed constant over time. For an ant located in a city *i*, the transition from city *i* to city *j* depend on:

(1) Whether or not city j has already been visited. Each ant has a tabu list that contains all the cities that the ant has already visited. Let $J_k(i)$ be the set of cities that remain to be visited by ant *k* when ant *k* is currently on city *i.*

(2) The distance D_{ij} between *i* and *j*. $D_{ij} = D_{ji}$ for a symmetric TSP.

(3) The amount of "artificial pheromone" on the edge connecting *i* to *j*, denoted by (*i*,*j*).

Let $Phe_{ij}(t)$ be the total amount of pheromone on edge (*i*, *j*) at time **t**. Time is incremented by 1 when all ants have completed a tour. The initial quantity of pheromone on edges is assumed to be a small-scaled positive constant **c**: for all (*i*,*j*),$Phe_{ij}(t = 0)$ = **c**. At the beginning of each iteration, ants are placed randomly in the cities. When on a city **i**, ant **k** select which city **j** to move to. To do so ant **k** checks the salesman list associated with city **i**, which is a list of preferred cities to be visited from city i: instead of examining all possibilities from any given city, unvisited cities in the salesman list are examined first, and only when all cities in the salesman list have been visited are other cities examined. The salesman list of a city contains the N_l closest cities. Cities in the salesman list are ordered by increasing distance, and the list is scanned sequentially. Ant **k** first chooses the next city to hop to from the list, and, if all cities in the candidate list have already been visited, selects city j according to:

$$j = \left\{ \begin{array}{ll} Arg \max\limits_{u \in J_k(i)} \left\{ [T_{iu}]^{\alpha} . [d_{iu}]^{-\beta} \right\} & if\ q \le q_0 \\ J & Otherwise \end{array} \right\} \qquad (3)$$

Where **q** is a real random variable uniformly distributed in the interval [0; 1], **q0** is a tunable parameter ($0 \le q0 \le 1$), and $J \in J_k(i)$ is a node that is randomly selected according to probability

$$p_{i,J}^{k}(t) = \frac{[T_{iu}]^{\alpha} . [d_{iu}]^{-\beta}}{\sum\limits_{u \in J_k(i)} [T_{iu}]^{\alpha} . [d_{iu}]^{-\beta}} \qquad (4)$$

Where α and β [7] are two tunable parameters that control the relative influences of trail intensity ($Phe_{iJ}(t)$) and distance (D_{iJ}). If $\alpha = 0$, the closest cities are more likely to be selected: this corresponds to a *classical stochastic greedy* algorithm with multiple starting points because ants are initially randomly circulated on the nodes. If $\beta = 0$, only pheromone magnification is at work: this method will lead to the instant selection of tours which may be far from optimal. $q \le q0$ corresponds to an exploitation of the network, that makes use of distances between cities and of existing pheromone trails by choosing the best local compromise between distance and pheromone concentration, whereas $q > q0$

favors more exploration. Cutting exploration by tuning q0 allows concentrating the activity of the system on the best solutions, instead of letting it explore constantly.

Pheromone trails are updated locally and globally:

***Local update**:* When, while performing a tour, ant **k** is on city i and selects city j ($\in J_k(i)$) as the next city to hop to, the pheromone concentration of (i,j) is immediately strengthened by a fixed amount Phe_0. The trail decays simultaneously, so that:

$$\tau_{i,j} \leftarrow (1-\rho_l).\tau_{i,j} + \rho_l.\tau_0 \qquad (5)$$

Where ρ_l ($0 \le \rho_l \le 1$) is parameter leading the local trail decay.

Global update: The ant that performed the best tour since the beginning of the trial is allowed to globally update the concentrations of pheromone on the corresponding edges. To improve the solutions found by ants, a local search procedure is performed. Each ant's tour is modified by applying 2 - opt a certain number of times, denoted by σ. If, after application of the sequence of σ2-opt swaps, no better solution has been found the original tour is kept, otherwise the better tour is kept. $Phe_{ij}(t)$ is then modified by an amount $\Delta Phe_{ij}(t)$ as follows:

$$\Delta\tau_{i,j}(t) = \begin{cases} \dfrac{Q}{(L_+)^\gamma} & if\,(i,j) \in T_+, \\ 0 & Otherwise \end{cases} \qquad (6)$$

Where Q is a tunable parameter, T+ has been the best tour since the beginning of the trial and L+ is its length γ= 1 is a parameter introduced in this study. Only the best tour is reinforced through the global update. Here again, trail decay is implemented:

$$\tau_{i,j}(t+1) \leftarrow (1-\rho_g).\tau_{i,j} + \rho_g.\Delta\tau_{i,j}(t) \qquad (7)$$

Where ρ_g ($0 \le \rho_g \le 1$) is parameter prevailing the two global trail decay. The information's are updated in the pheromone table it will continue up to the total number of node **N**, and this method is defined as ***Traditional Ant System*** .For two colonies we doing same procedure one again in parallel. Pseudo code [15] for *Traditional Ant System* is described below

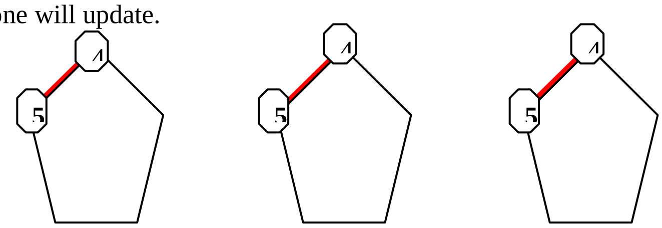
Procedure *Traditional Ant System for TSP Starts here*
    Initialize parameters pheromone trails
    While (termination condition not satisfy) **do**
        Construct solutions based on below condition:
        1) Randomly select the initial nodes
        2) Fix on the next component according to probability, which is based on the
           Pheromone and heuristic information update trails
    end-while
end-procedure

A. Two colony Ant System

Independently implemented solution search procedure for each colony is used to form a new pheromone distribution. In each colony, the pheromone distribution extends to a relative stable state after extending the termination conditions. Through pheromone communication [1, 2], the relative steady state turns to be relatively unstable, and this state contributes to the next solution search procedure through developing diversification. As to the means of the communication, we first present the pheromone distribution in each colony by $Phe_{ij}^C(t)$, which represents the pheromone concentration of *arc* (*i, j*) in colony *C* (*C*= 1 or 2), after *t* time of communication. Note that $Phe_{ij}^C(0)$ is the initialized pheromone distribution. Before each communication, pheromone distribution is obtained as $Phe_{ij}^C(t)$, and two colonies as follows:

$$Phe_{ij}^1(t),\ Phe_{ij}^2(t) \qquad (8)$$

In order to acquire new pheromone distributions $Phe_{ij}^C(t)$, (t+1), an important procedure in the existing method is applied. That is averaging their concentrations as follows after the search process in both colonies terminated. The Fig.3 shows how the pheromone will update.



Phe^1_{45} Phe^2_{45} $Phe^1_{45}(t+1) = [Phe^1_{45}(t) + Phe^2_{45}(t)]/2$

Fig 3: pheromone update

$$Phe_{ij}^1(t+1) = Phe_{ij}^2(t+1) = \frac{Phe_{ij}^1(t) + Phe_{ij}^1(t+1)}{2} \qquad (9)$$

Through this means of communication, new pheromone distributions $Phe_{ij}^{C}(t+1)$ are prepared for the next search procedure these processes followed in traditional ant algorithm after getting updated pheromone. We introduce our proposed Genetic Algorithm after getting best search solution from two colony ant system. Proposed method and Simulations are discussed in chapter IV.

IV. PROPOSED GENETIC ALGORITHM IN TWO COLONY STSP

Each iteration the pheromone memory is update successfully. This updated table unstable means it goes to another tour otherwise it finish their tours. After pheromones communication [1,2] we introduced Genetic Algorithm for optimization to reduce total cost value of that tour that reduction helps to improve efficiency rate. This process started at the time of Evaluation of ants. Evolution predicts total tour length (L) as well as Evaluation value (E). The solutions obtained by ants are represented as the chromosomes. The evaluation E_k of k-th ant is decided as

$$E_{jk} = N/L_k \tag{10}$$

Where E_k shows the quality of obtained tour. The GA-ACO algorithm gives the ants with high fitness to the future generation, to obtain the best solution. We perform scaling and leading of the fitness f_k from the evaluation. The scaling is decided as

$$f_k = \frac{[\chi(E_k - E_{ave}) + (E_{\max} - E_k)]E_{ave}}{E_{\max} - E_{ave}} \tag{11}$$

Where χ is the scaling parameter, E_{ave} is the average of the ant evaluation and E_{max} is the best evaluation in the populated chromosomes. The individuals, whose genetic information is given to the future generation, are chosen by according to the probability; this rule is called ***roulette selection*** [16]**.**

The choice probability of k-th ant is decided by

$$P_{GA,k} = \frac{f_k}{\sum_{k=1}^{M} f_k} \tag{12}$$

It should be marked that it is possible for the slow ant to be chosen as the ant whose genetic information is given to the future generation, although the slow ant cannot use the genetic information to choose the next visit city.

Crossover: Parents are chosen from the total possible population, and these parents create their children. This operation is repeated until the number of children reaches maximum limit. Here selected parents are identically same. However, the number of parents participating in the crossover is decided by a crossover rate *Pc*. This kind of crossover is called as *swapping* [4]. There are various ways of the crossover, in this paper, we use Partially-mapped crossover operator (PMX). PMX [16] is two-point crossover [10] and is shown in Fig. 4.

PMX selects two cut points along the strings, which represent the parent tours, at random. The substrings between the cut points are exchanged for genes of the other parents.

Mutation: After a crossover is performed, the mutation has taken place. The probability of the mutation is decided by the mutation rate *Pm* [16]. In this research study, we use the inverse mutation, and it is shown in Fig. 5. This mutation selects two cut points along the strings, which represent the tours, at random. The substrings between the cut points are inversed. This method is called as *flipping* [4].

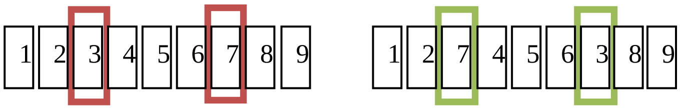


Fig 4: Cross over between two healthy parents

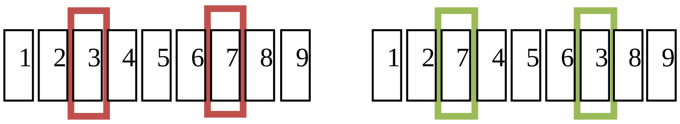


Fig 5: Inverse Mutations in Child

Update the genetic information to pheromone: After the genetic operators, the obtained tour length *Gk* (*t*) is calculated. The genetic information Δ*gki j* (*t*) transferred to the next generation by *k*-th ant is decided as

$$\Delta g_{k_{ij}}(t) = \begin{cases} 10/G_k & if\ (i,j) \in T_k(t) \\ 0, & Otherwise \end{cases} \tag{13}$$

Where *Tk*(*t*) is the tour obtained by *k*-th ant, and *Gk*(*t*) is its length. *gi j*(*t*) of each path (*i*; *j*) is updated depending on its Δ*gki j* (*t*).where the genetic information's are initialized to *g*0 at every iteration *t*.

$$g_{jk}(t+1) = g_0 + \sum_{k=1}^{M} \Delta g_{k_{ij}}(t) \tag{14}$$

Let $t = t + 1$. The Process will repeat until $t = t_{max}$.

Pseudo code for *Proposed Active pair Ant System* is described below

Algorithm *Ant System with Two Colonies* set parameters

```
for i=1 to 2 do
    Initialize pheromone distribution Phe_i
end-for
while(termination condition not satisfy) do
    for i=1 to 2 do
        while(termination condition not satisfy) do
        Construct ant solutions update pheromones by traditional ACO
        end-while
    end-for
    for i=1 to 2 do
        Phe1ij (t + 1) = Phei2j (t + 1) ← [Phe1ij (t) + Phei2j (t)] /2
    end-for
    Evaluation of Ants (Chromosome) for all Phe
    Select healthy parents
    Crossover; Mutation
    Update Genetic information in pheromone table (Phe1ij (t + 1) and Phei2j (t + 1) )
end-while
end Algorithm
```

V. SIMULATION

This Simulation was developed in Matlab platform. The basic system requirements are Intel 2.66GHz i3 processor, 4 GB RAM and Matlab version 8.0.0.783 software is require to run this simulation. All TSP databases taken from the TSPLIB [12] for simulation. The maximum iteration is set to 1e5, pheromone trail to (α) 1; heuristic value to (β) 5; evaporation coefficient (ρ) to 0.5 [7] and colony size is set to total number of nodes in each TSP database. Comparing the optimal solutions, the worst solutions, the average solutions, and the efficiency rate of the three algorithms in Table I.

TABLE I. SIMULATION RESULUT

TSP	Optimal	Algorithm	Best	Average	Error (%)	Worst	Time(m)	Efficiency Rate (%)
eil51	426	ACS	426	427.26	0.460	434	120	54
		TACS	426	426.8	0.188	432	135	81.2
		Proposed	426	426.5	0.117	428	152	88.3
berlin52	7542	ACS	7542	7542	0.000	7542	128	100
		TACS	7542	7542	0.000	7542	143	100
		Proposed	7542	7542	0.000	7542	165	100
eil76	538	ACS	538	540.56	0.476	549	181	52.4
		TACS	538	538.32	0.059	546	192	94
		Proposed	538	538.32	0.059	546	205	94
rat99	1211	ACS	1212	1232.40	1.767	1254	249	0
		TACS	1211	1216.72	0.472	1247	260	52.8
		Proposed	1211	1213.2	0.181	1235	279	81.9
kroA100	21282	ACS	21282	21400.04	0.555	21912	252	44.5
		TACS	21282	21294.72	0.060	21577	262	94
		Proposed	21282	21290.08	0.030	21485	275	97
d198	15780	ACS	16130	16716.56	5.935	17513	248	0
		TACS	15850	15994.24	1.358	16341	261	0
		Proposed	15805	15870.5	0.573	16227	278	42.7

The comparison is done based on the similar calculation time. In this paper, as the proposed method is based on ant system and two colony ACS, the performance of it was evaluated with both methods. For each TSP problem on an algorithm, 1000 runs were executed. Performance of each algorithm was evaluated using *Best*, *Average*, *Error*, *Worst*, *Timeavg*,*Rate*,[2,10] which mean the best solution, average solution, average excess rate from optimum, worst solution, average executing time, efficiency rate during 1000 runs.

VI. SIMULATION RESULT AND DISCUSSION

The performance of the proposed model is compared with the Traditional Ant system and the two colony Ant System and tabulated in Table 1. The error calculations and Efficiency rate under various TSP Database are given in Figure 6 and Figure 7.

A. Efficiency Rate comparison

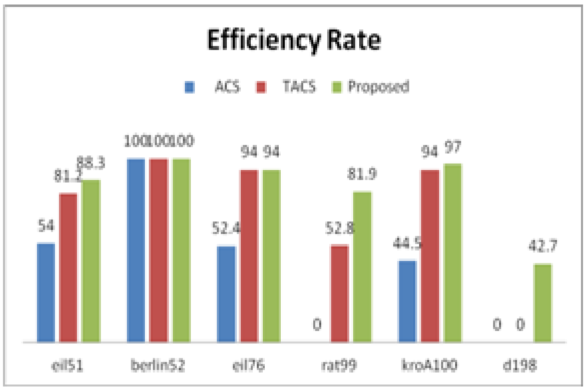


Fig 6: Efficiency rate comparisons

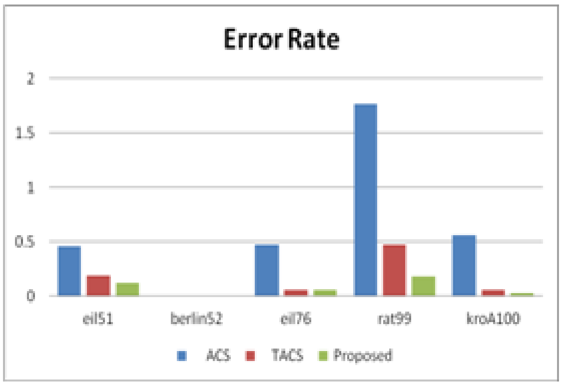
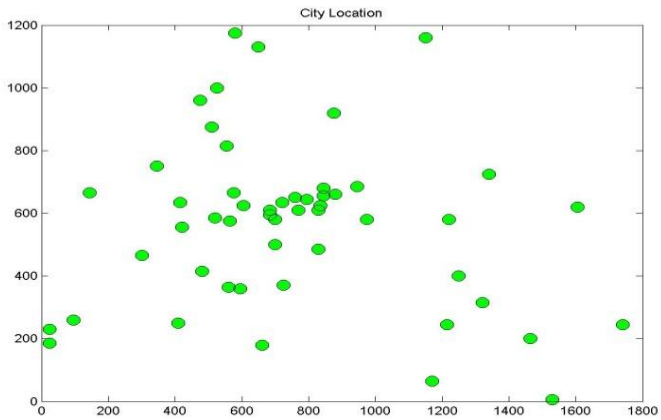


Fig 7: Error rate comparisons

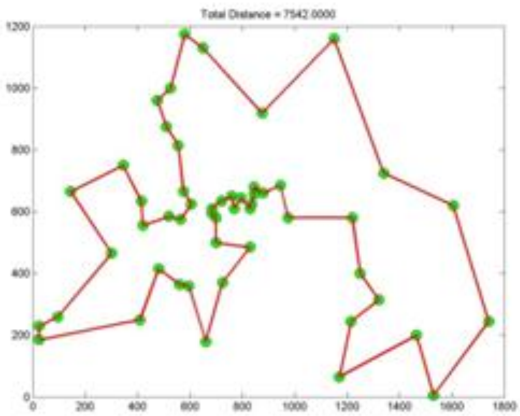
Discussion: Table 1 shows the all TSP database output, Best, Worst, Average, Error and Efficiency Rate. From the Table, it can be seen that in most cases, the proposed system efficiency and error values are better than two colony system. This can be further proved the error performance, which is adorned in Fig.6 & Fig.7, in which mean system error and efficiency is calculated between six different test TSP cases and the performance was evaluated. The error is relatively less in the proposed system when compared to the traditional and existing systems. The Optimal route selection and distance in iteration are shown in Fig.8. The Efficiency rate calculated between ideal error and system output error. The efficiency rate is as follows,

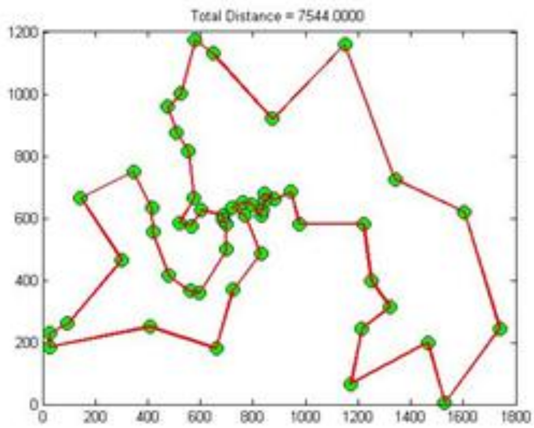
$$\text{Efficiency rate} = (\text{Ideal Error} - \text{Database error}) \qquad (15)$$

B. Results Comparison between proposed and existing methods

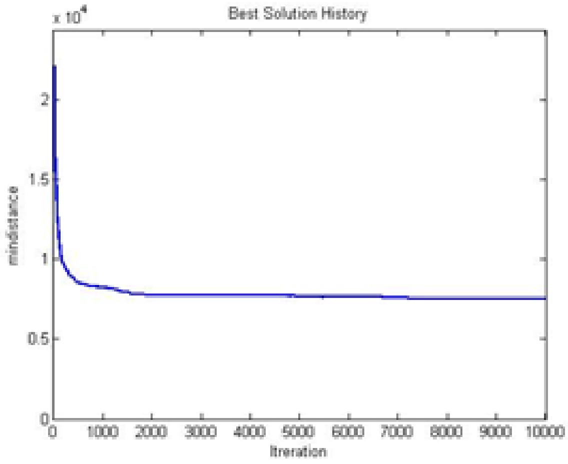


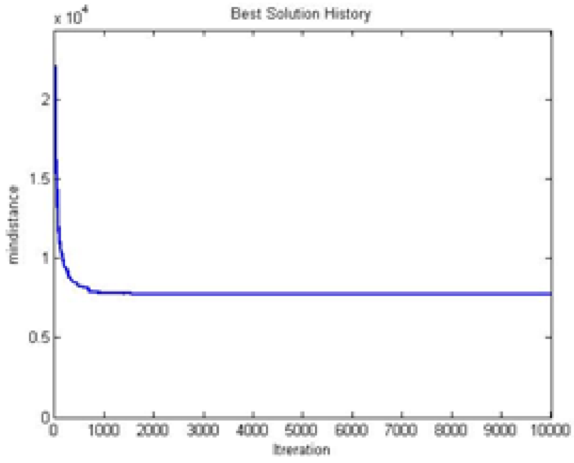
1





2a 2b





3a 3b

Fig 8: Comparative results for Existing and proposed methods for berlin52 (1) Initial node location, (2a) Existing Optimal route, (2b) Proposed Optimal route, (3a) Existing best distance history (3b) Proposed best distance history

VII. CONCLUSION

An ant system with two colonies using genetic algorithm, as an improved ACO algorithm, has been proposed in this paper. In the proposed method, we introduced a two-colony mechanism with natural evolution methods like genetic Algorithm. Ants in each colony first search solutions separately by the rule of the traditional ACO, and after the search action in each colony terminated, communication between colonies is boosted by genetic gens to prepare new pheromone information for the next search pattern, as a method of enhancing the diversification and efficiency rate of the algorithm. The proposed algorithm was applied to the TSP, and to verify the performance of it, several TSP benchmark problems were simulated. From the results, we find that the improved ACO algorithm has very high performance in searching solution comparing with the traditional ACO algorithm and excising two colony systems.

REFERENCES

[1] Manuel Lopez-Ibanez and Thomas Stutzle, "The Automatic Design of Multi objective Ant Colony Optimization Algorithms", IEEE Transactions on Evolutionary Computation, Vol.16, No.6, pp.861 -875, 2012.
[2] Xiao-Fan Zhou , Li-Qing Zhao ; Ze-Wei Xia ; Zhi-Qiang Chen, "An ant system with two colonies and its application to Traveling Salesman Problem", International Conference on Natural Computation, Chongqing, pp. 744 - 748, 2012.
[3] Dorigo, M., & Birattari, M. (2010). Ant colony optimization. In Encyclopedia of Machine Learning (pp. 36-39). Springer US.
[4] Chunxiang Wang, Xiaoni Guo,"A hybrid algorithm based on genetic algorithm and ant colony optimization for Traveling Salesman Problems", 2nd International Conference on Information Science and Engineering (ICISE), Hangzhou, China, pp 4257-4260,2010.
[5] D.Kaur,M. M.Murugappan, "Performance enhancement in solving Traveling Salesman Problem using hybrid genetic algorithm", Annual Meeting of the North American, Fuzzy Information Processing Society (NAFIPS), New York City, NY,pp 1-6,2008.
[6] John E. Bell, Patrick R. McMullen. Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics, vol.18, no.1, pp.41-48, 2004.
[7] Zar Chi Su Su Hlaing, May Aye Khine, "An Ant Colony Optimization Algorithm for Solving Traveling Salesman Problem", International Conference on Information Communication and Management (IPCSIT), vol.16 ,pp – 54-59, 2011 .
[8] M. Dorigo, L. M. Gambardella, Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation, vol.1, No.1, pp. 53-66, 1997.
[9] Luca Maria Gambardella, Marco Dorigo, "Solving Symmetric and Asymmetric TSPs by Ant Colonies", IEEE Transactions on Evolutionary Computation, pp- 622-627, 1996.
[10] Saloni Gupta, Poonam Panwar, "Solving Travelling Salesman Problem Using Genetic Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), Vol.3, No.6, 2013.
[11] Feng-Tse Lin, Cheng-Yan Kao and Ching-Chi Hsu," Applying the genetic approach to simulated annealing in solving some NP-hard problems", IEEE Transactions on Man and Cybernetics,Vol.23 , No.6 ,2002.
[12] TSPLIB:http://www.iwr.uniheidelberg.de/groups/comopt/software/TSPLIB95/tsp/
[13] Marco Dorigo, Gianni Di Caro, Luca M. Gambardella,"Ant Algorithms for Discrete Optimization", Artificial Life, Vol.5, pp - 137—172, 1999.
[14] L. Chen and K. Aihara, Chaotic Simulated Annealing by a Neural Network Model with Transient Chaos, Neural Networks, vol.8, no.6, pp. 915-930, 1995.

[15] ACO Algorithms for the Traveling Salesman Problem, John Wiley & Sons, 1999.
[16] Sho Shimomura, Haruna Matsushitaz and Yoshifumi Nishio", Ant Colony Optimization using Genetic Information for TSP", International Symposium on Nonlinear Theory and its Applications NOLTA2011, Kobe, Japan, pp-48-51, 2011.

ABOUT AUTHOR

Manimozhi N. Received the B.E. degree in Electronics and Communication Engineering and M.E. degrees in Applied Electronics from Anna University Chennai in 2010 and 2014, respectively. Currently working as a Assistant Professor in Vandayar College of engineering, Thanjavur, Electronics and communication department.

Rajkumar Sampath. Received the B.E. degree in Electronics and Communication Engineering from Anna University Chennai in 2010. Currently working as a Software Engineer in private firm. He has more than 4 years of Industrial experience in Digital Image Processing, Optimization Methodology, and Communication Theory.