I J C T A, 9(40) 2016, pp. 257-266

ACO Approaches to Routing

Shalini Aggarwal*, Pardeep Kumar and Shuchita Upadhyaya*****

ABSTRACT

ACO is a technique for solving a variety of optimization problems such as machine learning, routing, bioinformatics, assignment, subsets, scheduling, etc. ACO is based on a very simple creature in nature called ant. Although a single ant may not perform any specific task, yet colony of ants can prove to be quite useful in finding shortest path from their nest(source) to the food source(destination). Generally ants communicate by sensing pheromone trails deposited by the other ants on the path they travel. This survey primarily focuses on network routing applications using ACO. The survey contributes in a three-fold manner viz. providing an idea about the various approaches designed using ACO for solving routing problems, comparing some of the major researches involving applications of ACO in routing and discussing similarities and differences between AntNet and ABC approaches.

Keywords: ACO, AntNet, ABC, MACO.

I. INTRODUCTION

With the advancement of Internet and telecommunication, demand of fast, secure and accurate network has also grown. Due to this more and more complex network systems are being made. The complex networks are faced with a number of problems such as high cost, load balancing, congestion control, routing, etc. Therefore the challenges are to develop more sophisticated networks (may be intelligent) to solve these problems. One of such technique inspired by social insects i.e. ant used to solve problems in telecommunication and networks is ACO. Although an ant [1] basically is a very simple creature, yet a colony of ants can perform a number of important tasks such as foraging, building nests [1], [2], etc. Interestingly, most of the times when ants travel in search of food, they find shortest path from their nest (source) to the food source (destination). An important question arises here is that how most of the ants travel through the same path? The answer lies in the fact that they use stigmergy [2], [3] for communicating with each other. This stigmergic communication is achieved via pheromone [3] layering by ants while travelling through a path. During last two decades computer scientist have successfully tried and transformed the models of ant colonies into optimization and other similar applications [1], [4], [5]. This emerging and interesting field is Ant Colony Optimization (ACO) [2], [4 – 7]. Since its inception, ACO has been applied in a number of applications such as telecommunication networks [8], [9], [10] industrial applications [11], routing [2], machine learning, subset, bioinformatics [12], etc. This paper surveys the use of ACO in network routing applications. The application of ACO in network routing is the simulation of ants travelling, pheromone layering and maintaining data structures while travelling from source to destination.

The more detailed explanation of ACO is given in section II, while the section III discusses challenges in ACO routing based on problems of traditional routing and IV focuses on the various approaches of network routing using ACO. Section V provides a review of extensions of the two major ACO approaches

* Tecaher Fellow, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra (Haryana), India, *E-mail: shaliniagg07@gmail.com*

** Associate Professor, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra (Haryana), India, *E-mail: mittalkuk@gmail.com*

*** Professor, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra (Haryana), India, *E-mail: shuchita_bhasin@yahoo.com*

viz. AntNet and ABC. Section VI gives the similarities and differences between AntNet and ABC approach. Last section VI gives conclusion and future perspective.

II. ANT COLONY OPTIMIZATION(ACO)

ACO can be easily understood by the fig. 1(1) shows the various possible paths from nest to food source. Fig. 1(2) shows a situation when multiple ants are travelling from nest to food source. The ants are moving from all the possible paths. After some time most or all the ants travel through the shortest path as shown in fig. 1(3). This happens due to the more pheromone layering on the shortest path. The ants that travel through the shortest path will reach the food source(destination) first and come back to the nest(source) earlier as compared to the ants which travels through longer paths. As the ants which were travelling through the shortest path will come back first to the nest, they will deposit more (double) pheromone on the shortest path. On the basis of this pheromone, the new ants will prefer to travel through this shortest path. After some time all ants will be following the path as shown in figure 1(3).

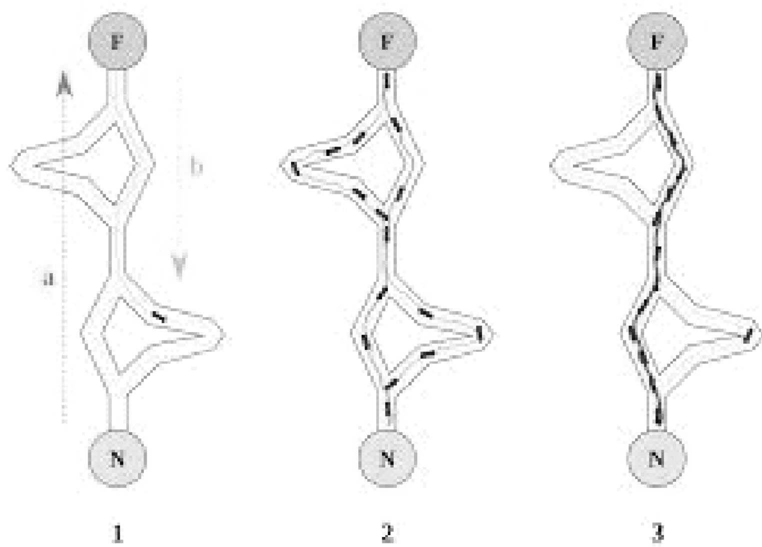

Figure 1: Ants travelling from Nest to Destination [1]

A basic flowchart representing the ACO working is shown in fig. 2.

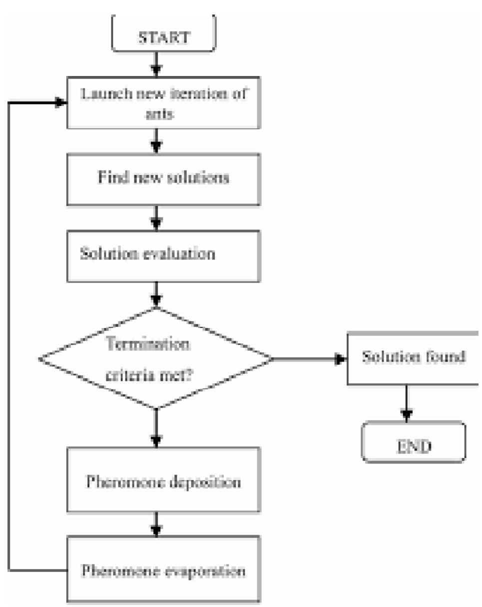

Figure 2: Flowchart showing the working of ACO

The pheromone layering has been done in different manners by different researchers. Deneubourg et al. [3] performed an experiment for studying behavior of ant. In their experiment they considered the double bridge of equal length with nest on one side and food source on the other side of the bridge. The ants start travelling at both the bridges to find food and in the process deposit pheromone on that path. After sometime one of the bridge finds more pheromone deposit and most of the remaining ants follow that path. It was concluded from their experiment that even if both the bridges were of equal lengths, the bridge on which more ants were travelling finally becomes the only path for finding the food by almost all of the ants.

Goss et al. [13] discussed a variant of double bridge experiment with one bridge being shorter that other. It was found that after sometime most of the ants were travelling through the shorter path because the shorter path was having more pheromone. Schoonderwoerd et al. [10] and Beckers et al. [14] also described the different ways to deposit pheromone on the path.

Although most of the ACO approaches use the basic problem solving techniques of biological ants, but some variations are done according to the requirement of the problem and solution. When using artificial ants some of the biological traits of the ants may not be used and some other traits may be added to have a better solution using heuristic approaches.

III. CHALLENGES IN ROUTING

While working with ACO in network routing, the major challenges to be considered are

(1) How the routing information will be handled?

(2) What will be the routing overhead?

(3) How the issues of adaptation and stagnation will be handled?

In traditional routing algorithms such as RIP or OSPF, nodes in the network depend upon the routing information provided by all the immediate neighbors of the node. This process continues for all the nodes and in this manner the complete routing table is formed. Routing table in RIP is dependent on the distances between the nodes, while in OSPF, the routing table is depends on the link-state information stored at all links in all of the identified paths. In ACO, the paths are explored in parallel and independently. As an ant arrives at a node, it updates the pheromone value corresponding to the source node in the path. Therefore all the entries of pheromone table at a node can be modified independently and in this manner the complete pheromone table is formed independently.

In traditional routing the transmission of routing table is done by every node to its every neighbor as in case of RIP, and a link-state-packet is transmitted to every other node via flooding in OSPF. The overhead can be very large in RIP as complete routing table needs to be transmitted to every neighbor. In case of OSPF, though the size of LSP is much smaller, yet due to flooding duplicate copies of same LSP may be received by various nodes via different paths, which result in increase of unnecessary overhead. Routing in ACO is achieved by transmitting ants, which generally have a very small size as compared to LSP or a complete routing table. Therefore the overhead in ACO is generally much lesser than traditional routing.

In case of dynamic networks, the changes to be made in LSP or large routing tables may incur large overheads for routing and may lead to slower speed of the network. In ACO, the changes to be made for dynamic network may be handled comparatively easily by piggybacking ants in data packets with more frequent transmission as the ant's size is very small. However, ACO approach also suffers from the problem of stagnation. Stagnation is the stage when all the ants choose only one path. As ants follow a single path they will deposit more and more pheromone on that path and the path will become optimum very soon. Then obviously that path will become congested and therefore the network will become standstill. This is the condition of stagnation. Another problem is the problem of local optima. If most of the ants start to follow a non optimum path, then that path will be considered as optimum path. The actual optimum paths may not be exploited by the ants.

IV. ACO IN ROUTING

This section provides an overview of the three main approaches in applying ACO in network routing, i.e. AntNet, ABC and MACO.

AntNet: AntNet[15], [16] algorithm was designed for Asymmetric Packet Switched networks for optimizing the performance of overall network instead of focusing on shortest or minimal path. AntNet uses two types of ants (exploration agents), which are forward (FRANTs) and backward ants (BKANTs). FRANTs are launched at a regular time interval from every node to a random destination. FRANTs use normal queues to experience the true network conditions. FRANTs collect information about trip time to each node, nodes traversed, and traffic existing at each node. The FRANTs travel in an asynchronous and concurrent manner with the data traffic.

As a FRANT reach to its destination, it copies its structure into BKANT and dies. BKANT then travels in backward direction towards the source node via exactly same path as travelled by FRANT but in reverse order. The reason for using two types of ant agents in AntNet is that the FRANTs are basically used for collecting information such as noting down the node numbers in the path and trip times. They are not used for any updates in the routing table at the nodes. The BKANTs receive this information from the FRANTs and accordingly updates the routing table at the nodes. BKANTS use priority queues to quickly circulate the information to the nodes.

The nodes in the network maintains various data structures such as M_{vk}, a statistical model containing information about end to end delays, T_{vk}, a pheromone table, and the routing tables to be updated by BKANTs. The updation in the pheromone and routing tables are done depending on the quality of the path identified by the FRANTS.

The data structure M_{vk}, which is a statistical model, is actually a vector for storing the mean and variances calculated on the basis of the delays measured by the ant agents and moving observation window to have a good estimation of the best trip time. The routing table stored at each node includes entries for each neighbor as well as each destination of the node. For example, if a network consists of n nodes and a specific node is having m neighbors, then the routing table at that node will have n-1 rows and m columns. The values stored in the table corresponding to each neighbor(n) and each destination(d) are the probability(P_{dn}) of going through this link for the destination d. As the rows of the table represent the probabilities of reaching to the destination via various neighbors, the one row sum should be 1.

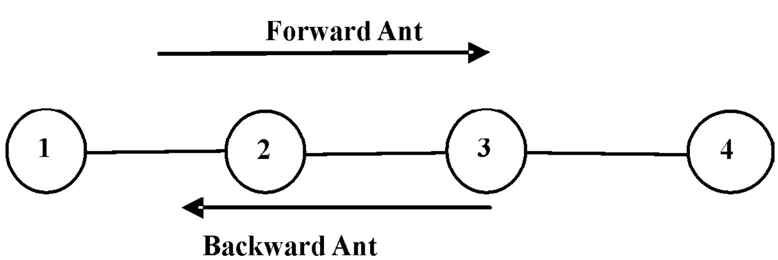

Figure 3: Example Network showing forward and backward Ants

For example in a simple network, as shown in fig. 3, when a FRANT travels from first node to the fourth node via second and third nodes, it gathers trip time along with information regarding local data traffic at each node. When FRANT reaches at destination node which is assumed to be 4 in this case, BKANT is activated. As explained above the BKANT uses the information gathered by FRANT to travel backward and updates the routing tables accordingly. The routing tables are updated according to the goodness of trip times. Goodness is calculated using current trip time and the best trip time calculated statistically.

Table 1
Probability table updated by backward ant at node 3

Destination\next node	*2*	*4*
1		
2		
4		P44

Table 2
Probability table updated by backward ant at node 2

Destination\next node	*1*	*3*
1		
3		P33
4		P43

Table 3
Probability table updated by backward ant at node 1

Destination\next node	*2*
2	P22
3	P32
4	P42

The tables 1, 2 and 3 show the updation by the backward ants while travelling back from destination to source node.

The new FRANTs are influenced by the updation done by BKANT in the routing table, which in turn is dependent on the path goodness as calculated by earlier FRANTs. When BKANT updates the routing table, it increases the probability P_{df} and decreases the probabilities P_{dn} corresponding to all other neighbours. This increase or decrease in the probabilities is done on the basis of a reinforcement value *r*, which is calculated for the new trip time on the basis of current routing table and local traffic statistics. The value of r lies between 0 and 1 and tells about the goodness of the newly experienced delay. The pheromone table is updated by increasing the probability P_{df} and decreasing the probabilities P_{dn}. The values are updated in a manner so that sum of all the probabilities still sum to 1.

$$P_{df} = P_{df} + r\,(1 - P_{df}) \tag{1}$$

$$P_{dn} = P_{dn} - rP_{dn} \tag{2}$$

However, by using both FRANT and BKANT, the routing overhead in AntNet Algorithm is doubled.

ABC: Ant Based Control (ABC) [17] algorithm was the first algorithm inspired by the behavior of ant colonies in network routing. The approach was designed for routing in telephone networks for load balancing. The principle on which this technique works comprises of mobile routing agents called ants. These agents traverse through the network and explore it randomly. The routing tables are updated on the basis of current state of the network. Each node in the network comprises of capacity C which shows the number of calls that can be accommodated, probability of being a destination and probabilistic routing table.

ABC uses only one type of ant named as forward ant. These ants are generated from every node to a randomly chosen destination at regular intervals. When travelling from one node to another, they select the next node according to the values in the pheromone table. Upon arriving at a node, the forward ant updates the routing table entries immediately against their source node. In other words it the pheromone value correponding to its previous node is incremented. For example, as in fig. 3, an ant is launched from node 1 to node 4. When the ant reaches node 2; p_{11} value at the node 2 and when it reaches at node 3, p_{12} value at node 3 are updated. Hence, the updation is done only in the entry corresponding to node 1 which is the source node. Here, the updated information in node 4 only effects the routing ants that have node 1 as their destination node.

Table 4
Pheromone update table at Node 2

Destination\next node	*1*	*3*
1	P11	
3		
4		

Table 5
Pheromone update table at node 3

Destination\next node	*2*	*4*
1	P12	
2		
4		

Table 6
Pheromone update table at node 4

Destination\next node	*3*
1	P13
2	
3	

Tables 4, 5 and 6 shows the entries done by a forward ant when travelling from source node to destination node at each intermediate node.

It is important to note that ants moving from source node can actually influence only those ants travelling in reverse direction or for which the source node is the destination. In simple words, it can be said that the ants travelling from source S to destination D may influence other ants that are travelling from any node in the network towards S and ants travelling from source S to destination D may get influenced by other ants that are travelling from D to any other node in the network[17].

The entries corresponding to the node from which the ant has arrived are updated according to the following formula as given in equation (3).

$$P = (P\_\text{old} + \Delta P) / (1 + \Delta P) \quad \ldots\ldots\ldots\ldots\ldots\ldots \text{(3)}$$

where P is the updated probability according to the probability increment “P. Other entries corresponding to the node are decremented according to equation (4).

$$P = (P\_\text{old}) / (1 + \Delta \mathrm{P}) \tag{4}$$

For identifying shorter paths, visiting heavily congested nodes, and avoiding stagnation, three methods have been introduced by Schoonderwoerd et al.[10], which include the aging, delaying of ants and noise to find shorter and optimum paths.

Shorter paths can be identified by reducing the value of ΔP gradually depending on the age of the ant. Age of the ant depends on the length of the path it travels. Longer the path more will be the age of the ant and therefore the ants which are travelling via shorter paths will have more impact on the routing table updates as compared to the ants which travels through longer paths. The equation(5) below indicates a proposal for calculation of ΔP.

$$\Delta P = ((d/age) + c) \tag{5}$$

where c and d are constants.

For avoiding the ants to travel through congested nodes, a delay may be added to the ants so that lesser number of ants will be travelling to their neighbours via this path. This delay will prevent other ants to travel via this path and the new ants will be able to find alternate paths avoiding congestion. The delay will also increase the age of ants again resulting in a lesser pheromone deposit and therefore lesser probability of selecting the congested path.

The equation(6) shown below indicate a delay calculation formula as described by Schoonderwoerd *et al.* [17]. The delay is expressed in discrete time steps.

$$Delay = a.e^{-bs} \tag{6}$$

Where ‘a’ and ‘b’ are constants and ‘s’ denotes the spare capacity of the current node.

Stagnation can be avoided by adding noise. The basic purpose of adding noise is to have diversity in finding the paths. As in case of ants it is not guaranteed that always the shortest path will be found. It may be the case that if the initial ants select a longer path, the other ants may also follow that path. As more and more pheromone is deposited on that path, that path may yield the final path, which obviously is incorrect. Adding noise will enable the ants to select a path in a random fashion and without considering the influence of the pheromone table. By adding noise, more diverse paths can be found, therefore even if a bad path has been selected by earlier ants, the better paths may still be identified by the newer ants and in this manner stagnation can be avoided.

MACO: Multiple ant colony approach was proposed firstly by Verela and Sinclair [18]. The proposed approach was based on problems found in virtual wave length path routing and wavelength allocations. The proposed strategy of wavelength path routing is to allocate the least wavelengths for each link by distributing the wavelength requirements evenly over all the links. The concept of pheromone attraction and repulsion is used in this type of routing. The pheromone attraction works in a similar manner as in case of ACO, but the pheromone repulsion is used to repel the different wavelength so that they can be distributed over different links. A probability function similar to the ACO applications is generated; however this function is created using the different degrees of pheromone attraction and repulsion. In this technique the ants of different colonies have different colours. The ants travel in the same manner as in case of AntNet. The ants check the pheromone deposit by the ants of same colony. If the pheromone deposit of the same colony is high, then the ants are attracted to that path. However, the pheromone deposit of the ants of different colony is high, the ants are repelled. Obviously, after sometime ants of different colours will choose their own path. In this manner multiple optimum paths can be identified.

Sim and Sun [19] proposed an approach based on MACO for load balancing tasks in connection-oriented networks. MACO was used to find optimal paths. ABC [10] and AntNet [15] are storing only one routing table at each node. So if more than one optimal path is found, then data can be routed through only one of the optimal paths. MACO is used to address such a problem. In this approach, ants of different colonies deposit pheromone of different colours. Ants are attracted or repelled according to the pheromone colour i.e. if the colour of pheromone is from the same colony; ants will be attracted otherwise they will be repelled. In this manner each colony will choose its own path and multiple paths for different types of ants can be found. The issue of how many colonies should be generated was not considered in this paper.

II. EXTENSION OF ABC AND ANTNET

This section discusses some of the extensions of the ABC and AntNet approaches.

One of the extensions of ABC was found by Guerin [20] and Bonabeau et al.[21]. They proposed the smart ants to solve the problem of routing and load-balancing in circuit-switched networks. The smart ants updates the entries for each node they pass, instead of updating the entries only for the source node. The smart ants behave exactly in the same way as that of forward ants in ABC. But, one of the limitations of smart ants is that it is more complex due to the reason that it performs more pheromone updates at every intermediate node than ants in ABC. But the results achieved using smart ants were better than that of ABC ants.

Subramanian et al.[22] proposed two types of ants i.e. regular ants and uniform ants. This research was primarily focused on packet-switched networks. In this paper, the regular ants are the ants similar to the ABC ants. The only difference being that regular ants use the total cost of a path to deposit pheromone instead of ant's age. A regular ant with a lesser cost path will deposit more pheromone than the ant travelling through a high cost path. On the other hand, the uniform ant chooses the next nodes randomly. Another difference between regular and uniform ants lies in the fact that the accumulated cost is used in forward direction by the regular ant while in uniform ants it is used in backward(reverse) direction. It is assumed by the uniform ants that a node knows the distances to its neighbor nodes. Overall the results of this approach were found to be satisfactory.

Heusse et al.[23] proposed a cooperative asymmetric forward(CAF) technique for routing in packet-switching networks. CAF works similar to ABC except that a CAF ant updates routing table using the cost in reverse direction as stored by the data packets. CAF ants can be used for both symmetric and asymmetric networks. This approach may not be very useful if data packets do not travel on both sides on a path. Another problem with this approach was that it needs to maintain a reverse routing table.

A variant of AntNet was proposed in [24]. In this technique priority queues were used and the forward ants travel on the basis of these priority queues. The backward ants estimates the trip times and perform other tasks such as updating the local traffic statistics and deposit the pheromone (probability) accordingly. The routing information in this technique is more accurate as compared to AntNet. The results found using this approach are better as compared to the AntNet.

Another extension of AntNet was found by Baran and Sosa[25]. They proposed five major features differences from AntNet, which are intelligent initialization, intelligent pheromone updates when a link or node fails, adding noise to avoid stagnation, using deterministic criteria for selecting a node and using a limited number of ants in the system. The purpose of these five improvements in the AntNet were aimed at less overhead, lower cost, and using intelligent approaches to choose next node. But sometimes due to restricted number of ants its adaptiveness may decrease. Overall it was found to be a very good extension of AntNet.

III. DIFFERENCES AND SIMILARITIES IN ABC AND ANTNET APPROACHES

Differences

1. Antnet is used to improve the overall performance of the network instead of focusing on shortest path whereas ABC is used for load balancing in telecommunication network.

2. The most important difference lies in the fact that in ABC ants update pheromone trails on node by node basis, while in AntNet, the updation is carried out by backward ants when going back to the source node. Therefore in ABC approach, ants do not require to travel in reverse direction and go back to their source node.
3. In AntNet, selected set of ants(namely backward ants) have the privilege to deposit more or less pheromone depending on the problem, which can prove to be very useful for different types of applications. But the use of backward ants increases the network traffic in return trip. As ABC does not use backward ants, network traffic is less in this approach.
4. Another important differences can be found in the fact that in AntNet, local traffic models are used to score the ant traveling time, while ABC does not use such types of models and therefore is faster than AntNet.
5. AntNet uses the concept of learning from local queue information and ant's private memory to improve the decision taken by the ant and balance the pheromone updates accordingly. ABC does not use any such concept.
6. In ABC approach cycles can be formed as it does not use the information contained in its sub-paths, while AntNet use s such type of information and therefore cycles are not formed.

Similarities

1. Both approaches do not put any limit on the amount of pheromone they deposit on each link/path.
2. There is no limit on the number of ants to be used.
3. Both techniques use uniform distribution of probability in the initial stage which does not reflect the state of network.

IV. CONCLUSION

Traditional routing can have a number of problems such as network failure, slow response time, large overhead, congestion problem and high cost. ACO has become a very good technique for routing, removing most of the problems faced by traditional routing. ACO is a technique based on real ants replicating their work in a network. Under ACO, a number of approaches have been proposed, but the most important researches were AntNet and ABC approach. Survey and comparison of ABC and AntNet and their extensions in applying ACO in routing has been given in this work. The techniques for multipath routing viz. MACO was also discussed in detail. It may be concluded that a hybrid approach using the good qualities of both ABC and AntNet can yield better results in the direction of routing optimization.

REFERENCES

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Inspiration for optimization from social insect behavior," *Nature*, vol. 406, pp. 39–42, July 2000.

[2] M. Dorigo, G. D. Caro, and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artif. Life*, vol. 5, no. 2, pp. 137–172, 1999.

[3] J. L. Deneubourg, S. Aron, S. Goss, and J. M. Pasteels, "The self-organizing exploratory pattern of the argentine ant," *J. Insect Behav.*, vol. 3, pp. 159–168, 1990.

[4] M. Dorigo, E. Bonabeau, and G. Theraulaz, "Ant algorithms and stigmergy," *Future Gener. Comput. Syst.*, vol. 16, no. 8, pp. 851–871, 2000.

[5] M. Dorigo and G. D. Caro, "The ant colony optimization metaheuristic," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. New York: McGraw-Hill, 1999.

[6] V. Maniezzo and A. Carbonaro, "Ant colony optimization: An overview," in Metaheuristic Int. Conf., Angra dos Reis, Brazil.

[7] M. Dorigo and T. Stutzle, "The ant colony optimization metaheuristic: Algorithms, applications, and advances," in *Handbook of Metaheuristics* , F. Glover and G. Kochenberger, Eds. Norwell, MA: Kluwer.

[8] K.M. Sim and W.H. Sun, Ant colony optimization for routing and load-balancing: Survey and new directions, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2003.

[9] G. Di Caro, Ant colony optimization and its application to adaptive routing in telecommunication networks, Ph.D. dissertation, Universit´e Libre de Bruxelles, Brussels, Belgium,2004.

[10] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, "Ants for Load Balancing in Telecommunication Networks," Hewlett Packard Lab., Bristol, U.K., Tech. Rep. HPL-96-35, 1996.

[11] J. Bautista and J. Pereira, Ant algorithms for assembly line balancing, in Proc. ANTS 2002, ser. LNCS, M. Dorigo et al., Eds., Springer Verlag, vol. 2463, pp. 65–75, 2002.

[12] P. Sivakumar and K.Elakia, "A Survey of Ant Colony Optimization", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 6, Issue 3, pp. 574-578, March, 2016.

[13] S. Goss, S.Aron, J.L. Deneubourg and J.M. Pasteels," Self –organized shortcuts in the Argentine ant", Naturwissenschaften, 1989.

[14] R. Beckers, J. L. Deneubourg, and S. Goss, "Trails and U-turns in the selection of a path by the ant lasius niger," *J. Theoret. Biol.*, vol. 159, pp. 397–415, 1992.

[15] G. D. Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *J. Artif. Intell. Res.*, vol. 9, pp. 317–365, 1998.

[16] G. D. Caro and M. Dorigo, "AntNet: A Mobile Agents Approach to Adaptive Routing," Univ. Libre de Bruxelles, Brussels, Belgium, Tech. Rep. IRIDIA/97-12, 1997.

[17] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, "Antbased load balancing in telecommunications networks," *Adapt. Behav.*, vol. 5, no. 2, 1996.

[18] N. Varela and M. C. Sinclair, "Ant colony optimization for virtual-wavelength- path routing and wavelength allocation," in *Proc. Congress Evolutionary Computation*, Washington, DC, July 1999, pp. 1809–1816.

[19] K. M. Sim and W. H. Sun, "Multiple ant-colony optimization for network routing," in *Proc. 1st Int. Symp. Cyberworld*, Tokyo, Japan, November 2002, pp. 277–281.

[20] S.Guerin, "Optimization Multi-Agents en Environment Dynamique: Application au Routage Dans Les Reseaux de Telecommunications", DEA, Univ. Rennes I, Ecole Nat. Super. Telecommun. Bretagne, France, 1997.

[21] E. Bonabeau, F. Henaux, S. Guerin, D. Snyer, P. Kuntz and G. Theraulaz, "Routing in telecommunications networks with ant-like agents," in Proc. of Intelligent Agents Telecommunications Applications, Berlin, Germany, 1998.

[22] D. Subramanian, P.Drushel, and J.Chen, "Ants and reinforcement learning: A case study in routing in dynamic networks", in Prc. of Int. Joint Conf. Artificial Intelligence, Palo Alto, CA, 1997, IJCAI-97, pp. 832-838.

[23] M. Heusse, D. Snyers, S. Guérin, and P. Kuntz, "Adaptive Agent-Driven Routing and Load Balancing in Communication Networks," ENST Bretagne, Brest, France, Tech. Rep. RR-98001-IASC, 1998.

[24] "Two ant colony algorithms for best-effort routing in datagram networks," in *Proc. 10th IASTED Int. Conf. Parallel Distributed Computing Systems*, 1998, pp. 541–546.

[25] B. Baran and R. Sosa, "A new approach for AntNet routing," presented at the *Proc. 9th Int. Conf. Computer Communications Networks*, Las Vegas, NV, 2000.