

# Optimization of Cable Layout Design in a Wind Farm: A Hybrid Approach

M. Tifroute and H. Bouzahir

Laboratory of Engineering Systems and Information Technologies, ENSA Ibn Zohr University, Agadir, Morocco

## Abstract

In this paper, a hybrid algorithm based on modified Ants Colony Optimization (ACO) and Artificial Immune Algorithm (AIA) for solving the Steiner Minimal Tree Problem (SMTP) is introduced. Since the Steiner Tree Problem is NP-hard, we design an algorithm to construct high quality Steiner trees in a short time which is suitable for real time multicast routing in networks. After the breadth - first traversal of the minimal graph obtained by ACO, the terminal points are divided into different convex hull sets, and the full Steiner tree is structured from the convex hull sets partition. The Steiner points can be vaccinated by AIA to get an optimal graph. The average optimization effect of AIA is shorter than the minimal graph obtained using ACO, and the performance of the algorithm is shown. We give an example of application in wind farm network design.

**Keywords:** Steiner Problem, Ant Colony Algorithm, Artificial Immune Algorithm.

## 1. Introduction

Let  $n$  be a natural number and  $D = \{a_i\}_{i=1, \dots, n}$  be a given set of points in the Euclidean plane. A natural task which arises in many optimization problems is finding a connected graph with minimal length whose set of vertices is  $D$ . If we are not allowed to add other vertices to the graph, the solution is called a minimal spanning tree. Finding such a solution is a purely combinatorial problem. One can tackle the problem by considering the complete tree (i.e., the tree containing all possible edges with set of vertices  $D$  and weights on the arcs given by the distance between the corresponding vertices). Then, one is asked to find a connected subgraph of the given weighted graph, which contains all the vertices and which has a minimal length. The problem becomes much more difficult if one has the possibility to add new vertices to the given set of points. This variant of the problem is called the Steiner problem. It means finding a connected graph with minimal length among all graphs whose set of vertices contains the given set of points. The additional vertices are called the Steiner points.

In this paper, a hybrid algorithm based on Ants Colony Algorithm (ACO) and Artificial Immune Algorithm (AIA) is proposed for solving the Steiner tree problem in a planar graph.

## 2. Minimal Steiner Tree Problem

The objective of the Euclidean Steiner Tree Problem (ESTP) is to determine the minimal length tree (with respect to the Euclidean metric) spanning a set of terminal points, while permitting the introduction of extra Steiner points  $S$  into the tree to reduce its overall length. A topology is a configuration of terminal points and Steiner points where the connections are specified, but the locations and number of the Steiner points are not. A topology is said to be a Steiner topology if every Steiner point has degree 3 and every terminal node has degree at most 3. A Steiner topology where all terminal points have degree 3 is a Full Steiner Topology (FST) (see Fig.1. for two simple examples). Any non-FST can be identified with a FST where some edges have zero length. Such a FST is called degenerated (see Fig. 2 for illustrative examples). The cause of degeneration is due to the distribution of terminal points. A non-FST can be decomposed into some FST's. A solution to the ESTP is called a Minimal Steiner Tree (MST). It can be shown that a MST has a Steiner topology. Other know results regarding MST's include the following conditions (see [4]):

- Angle condition: angles between edges connecting a Steiner point and its 3 neighbors are all 120 degrees.
- SMT is a concatenation of FST's on subsets of terminal points.
- SMT for a problem on  $n$  terminal points has at most  $n - 2$  Steiner points.

Corresponding authors

E-mails: [mohamed.tifroute@gmail.com](mailto:mohamed.tifroute@gmail.com); [hbouzahir@yahoo.fr](mailto:hbouzahir@yahoo.fr)

© 2016 International Association for Sharing Knowledge and Sustainability

DOI: 10.5383/ijtee.11.02.005

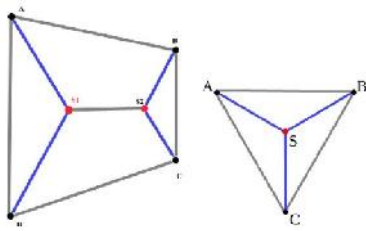


Fig. 1. FST for 3 points and 4 points

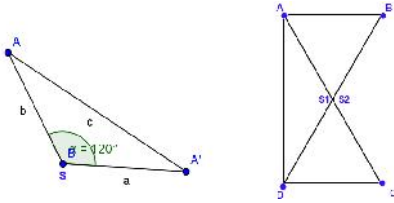


Fig. 2. Degenerated FST

### 3. Minimal Network Route Construction using the Ant Colony Optimization Algorithm

Ant colony optimization (ACO) paradigm is included in relatively recent intelligent agents, based on ants' biological inspiration [2]. By tracking the ants natural behavior, it is discovered that they can find the shortest path from the ant hill to a food source in absence of visual information or a direct communication between them. Also, ants can be adapted to environmental changes. ACO tries to use such real ants' skills to solve optimization problems. Hereafter, we present an ACO particular approach, adapted from graph theory for wind farm optimal cable route determination. The solution uses a graph with  $n$  vertex and all edges between them. Each edge  $(i, k)$  of a complete graph is associated with a pheromone concentration  $(\tau_{ik})$ , used for the route chosen by an ant from the supposed colony. Initially, the  $\tau_{ik}$ 's are set to small positive values (i.e., 0.01). In the ACO algorithm (see [3]), the minimal route length was initialized with a high value  $L_{min} = \infty$ . The number of ants  $na$  will be distributed as evenly as possible between the graph vertices. It is admitted that the number of vertices  $n$  and the number of ants  $na$  is chosen such that  $na = m \times n$  ( $m$  has an integer value). That is, in each node, there will be  $m = na/n$  ants. Also, we note that  $Nod_j$  represents the place where each ant  $j$  is located at each time. According to the proposed problem, the optimization process contains the restriction that an ant must pass through each node without forming no cycle. Each ant route selection is done in tabu list, which contains the elements that describe the sequence of visited nodes. After the ants' distribution on the graph nodes, the tabu list assigned to each ant will be initialized by the node where that ant was distributed. Further, the ants should move in different graph nodes until the tabu lists are complete. Each ant will then make a complete graph tour. Next, for every ant  $j$ , the starting node  $i = Nod_j$  and the destination node  $k^*$  are considered. The  $k^*$  should not be included in tabu list. The probability for an ant  $j$  in a node  $i$  to choose the node  $k$  as its destination is defined as follows:

$$P_{ik}^j = \begin{cases} \frac{(\tau_{ik})^\alpha \cdot (\eta_{ik})^\beta}{\sum_{p \in \text{Tabu}_j} (\tau_{ip})^\alpha \cdot (\eta_{ip})^\beta} & \text{if } k \notin \text{Tabu}_j(i) \\ 0 & \text{if } k \in \text{Tabu}_j(i) \end{cases}$$

where  $\text{Tabu}_j(i)$  is the set of nodes that are neighbors to node  $i$  that can be visited by the ant  $j$ .  $\tau_{ik}$  and  $\eta_{ik}$  represent respectively the amount of pheromone in the edge  $ik$  and visibility (distance between nodes) probability. If  $\beta = 0$ , the  $P_{ik}^j$  probabilities depend only on pheromone concentration. Also, if  $\alpha = 0$ , the  $P_{ik}^j$  depend only on the nodes visibility. When all the ants pass through all the graph nodes, each ant route will be closed without returning to the origin node. Practically, this aspect is the ACO algorithm adaptation to the studied problem. Further, according to the ACO, the route lengths for all ants should be calculated and will be stored as the minimal length, which coincides with the final iteration. Before switching to another step, the pheromone concentration must be updated on each graph edge as follows

$$\tau_{ik} = \rho \cdot \tau_{ik} + \Delta\tau_{ik}$$

where  $\rho$  is a parameter set to 0.1 in most of the experiments.  $1 - \rho$  is called the pheromone evaporation rate.  $\Delta\tau_{ik}$  represents the pheromone concentration correction on the edge  $(i, k)$  determined by total ant number who move from  $i$  to  $k$ , using the equation:

$$\Delta\tau_{ik} = \sum_{j=1}^{na} \Delta\tau_{ik}^j$$

where  $\Delta\tau_{ik}^j$  represents the deposited pheromone quantity on edge  $(i, k)$  by ant  $j$ , determined as follows:

$$\tau_{ik}^j(t) = \begin{cases} \frac{Q}{L_j(t)} & \text{if } k \in \text{Tabu}_j, i = \text{Tabu}_j(p) \text{ and } k = \text{Tabu}_j(p+1) \\ 0 & \text{otherwise,} \end{cases}$$

Finally, the stopping criterion coincides with the maximum number of iterations  $T_{max}$ : while  $t < T_{max}$ , reset tabu lists of the ants and the procedure is restarted by resetting first element of every tabu list with current node number where each ant is located. We should mention that ACO algorithm (see [3]) is cooperative, versatile and robust.

### 4. Artificial Immune Algorithm (AIA)

AIA is a recent branch of stochastic search algorithms. It is classified as a population-based metaheuristic method. Many authors claim that AIA is a more efficient problem-solving strategy in optimization. The original intention is inspired by the simulation of the physiological immune systems of natural living organisms defending the body from foreign pathogens (bacteria or virus). The mechanisms work by first recognizing foreign substances known as antigens. Immune systems then generate a set of antibodies to eliminate the antigens. The mechanisms are able to recognize which antibodies are better at eliminating the antigens and producing more variations in the next generation of antibodies. Each antibody is assigned a value called affinity showing the capability of that antibody to eliminate antigens. The antigen and affinity in the AIA are

respectively equivalent to the problem to be solved or the objective function and the feasible solution for a conventional optimization method. The terminology ( see Fig. 3) of AIA is:

- Antigen: Problem to be solved, i.e., the objective function.
- Antibody: Solutions to the problem to be solved.
- Vaccine: Feature information extraction from the prior knowledge of the problem to be solved.
- Fitness function: Judgment of individuals in the population quality index.

#### 4.1. Immune Algorithm for Obtaining the SMT

##### 4.1.1 Dividing the minimal graph obtained using ACO into different convex hull sets

A non-FST can be decomposed into some full Steiner trees. The ideal partition is the critical issue. The Steiner points are inoculated as vaccine to the minimal graph obtained by ACO. For the terminal points set  $V = \{a_1, a_2, \dots, a_n\}$  in the plane, we follow the breadth - first traversal sequence on the minimal graph obtained using ACO in order to divide the nodes into different subsets. We notice that the nodes on the segmentation may appear in two different subsets as in Fig. 4.

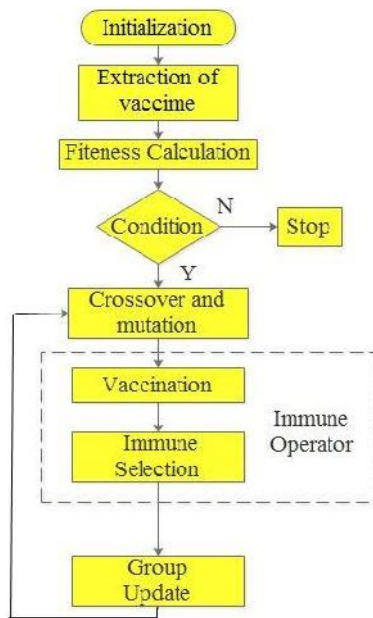


Fig. 3. Flowchart of Artificial Immune Algorithm

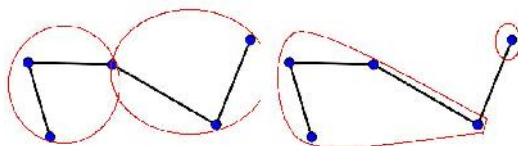


Fig. 4. Two different convex hulls partitions for 5 points

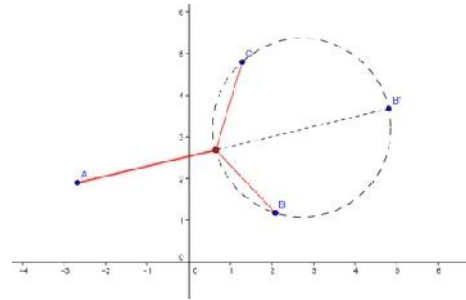


Fig. 5. Vaccines of 3 points in convex hull

In the division of a convex hull, one should respect the following principles:

- Ensure that the minimal graph obtained by ACO is carved up by adding the maximum number of Steiner points.
- The topology structure of a convex hull should not be a FST.
- The number of Steiner points is greater than the maximum number of convex hull sets.

##### 4.1.2. Vaccination method

For each convex hull set, we find Steiner points using Melzak method (see [6]), and the added Steiner point is the extracted vaccine. For three points, the obtained full Steiner topology is sole. However, for four points, the obtained full Steiner topology is two or one, as in Fig. 6.

##### 1.1.3. Inoculation in minimal graph obtained by ACO

The extracted vaccines will be inoculated to the minimal graph obtained by ACO as in Fig. 6, for instance.

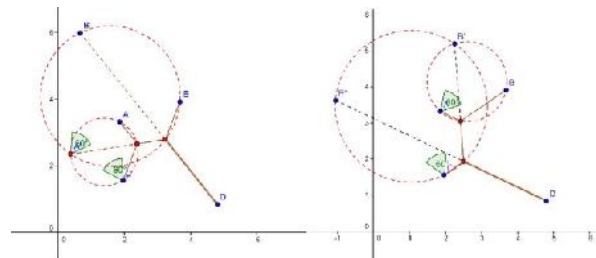


Fig. 6. Two different vaccines of 4 points

For the set of terminal points  $V$ , let  $E(e_i)$  and  $G(g_i)$  be respectively the edges of the minimal graph obtained by ACO and after vaccination. We have:

1.  $L(ACO) = \sum_{i=1}^{n-1} e_i$
2.  $L(SMT) = \sum_{i=1}^m (g_i)$
3.  $f(v) = \frac{L(SMT)}{L(ACO)}$

Algorithm:

- Step 1: Set  $V$  as the initial Steiner tree solution obtained by ACO and compute its length.
- Step 2: Make all possible partitions to convex hulls.
- Step 3: Extract all kinds of vaccines depending on the obtained convex hulls: define the set of vaccines.

- Step 4: Select vaccines, vaccinate in initial solution and calculate the Steiner tree length, then delete the used vaccines from the set of vaccines.
- Step 5: Calculate the fitness function, if its value is shorter than before, save the present inoculation, else cancel.
- Step 6: If  $f(v) < 0,87$  or set of vaccines is null, end, else go to Step 4.

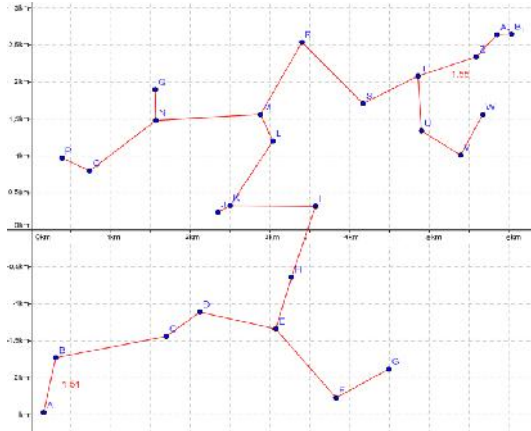
**5. Application in Designing a Wind Farm Cable Route**

Radial or branched wind farm cable design is divided in two phases: one uses ACO algorithm building the minimal cable route length and the second improves the network by adding supplementary branch nodes using AIA. The wind farm structure or cable route improvement can be achieved by shifting the source nodes to the ends and vice versa, with the particularity that the latter would be preferable because the ends power flows are known. By use of the method above, farm arborescence cable route optimization application can be designed. The allocation uses a combination of ACO and Artificial Immune Algorithm and the objective function is obtaining a minimal cable route length (between wind turbines and the step-up station) with radial structure restrictions.

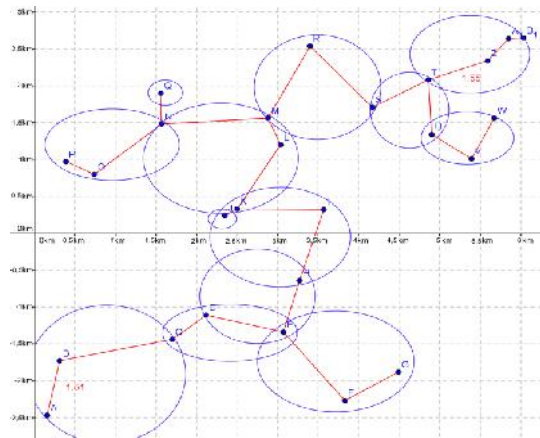
Our objective is to follow a hybrid approach that gives the optimal cable route for a wind farm with 26 wind turbine. The input data of test are presented in the following Fig. 8.

In a first step, we use the methodology aforementioned that is based on the ACO successive search technique. The obtained minimal length for the wind farm cable route is 16700m. This configuration is shown in Fig. 9.

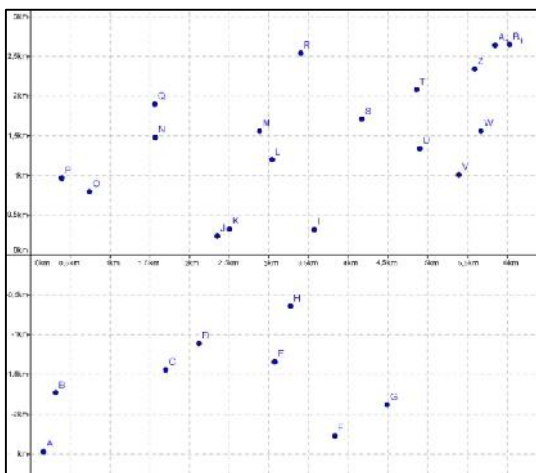
In a second step, to allow new route construction, 14 Steiner points are added by AIA and the minimal wind farm cable route length is reduced to 15040m.



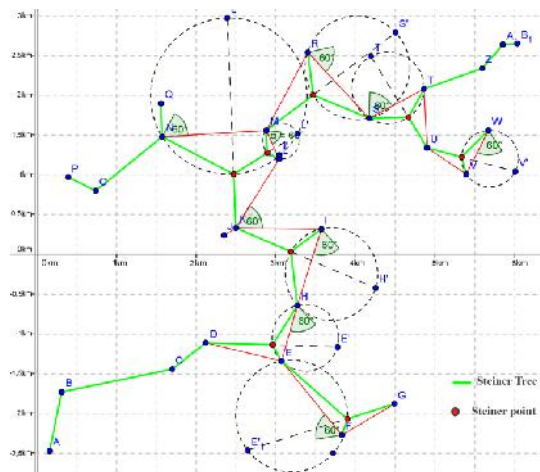
**Fig. 9. Wind Farm Cable Route Optimization using ACO Algorithm**



**Fig. 10. Optimal Convex Hulls Partition**



**Fig. 8. Initial Positioning of Turbines in the Analyzed Wind Farm**



**Fig. 11. Steiner Minimal Tree Obtained using the Artificial Immune Algorithm**

**Table 1. Results**

Number of Turbines	Number of convex hulls	Number of Vaccines	Length L(ACO)	Length L(SMT)	Ratio
26	13	14	16700m	15040m	0,9

### 6. Conclusion

The proposed approach is capable of finding the optimal wind farm cable route in a short time and less computational effort comparing to exact methods, such as Kruskal and Prim Algorithms.

### References

- [1] A. O. Ivanov and A. A. Tuzhilin, Minimal networks: the Steiner problem and its generalizations. CRC Press, 1994.
- [2] S. Das, S. V. Gosavi, W. H. Hsu and S. A. Vaze, An ants colony approach for the Steiner tree problem, in Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'02), Morgan Kaufmann Publishers Inc., Orlando, 2002, p. 135
- [3] B. Neagu, Gh. Georgescu and O. Ivanov, A new approach for electric energy distribution network routes optimization, Bul. Inst. Polit. Iasi, f 4, 2013, pp.133-142.
- [4] E. N. Gilbert and H. O. Pollak, Steiner minimal trees, SIAM J. Appl. Math., Vol. 16, 1968, pp. 1-29.
- [5] L. N. de Castro and J. Timmis, Artificial Immune Systems: A Novel Paradigm to Pattern Recognition, In J. M. Corchado, L. Alonso, and C. Fyfe (eds.), Artificial Neural Networks in Pattern Recognition, University of Paisley, 2002, pp. 67-84.
- [6] Z. A. Melzak, On the problem of Steiner, Canad. Math. Bull., Vol. 4, 1961, pp. 143-148.