

# Reliability and cost optimisation of complex electric power networks using ant colony algorithm

Lukasz Piątek<sup>1,\*</sup>

<sup>1</sup>Częstochowa University of Technology, Institute of Information Technology, 42-200 Częstochowa, Poland

**Abstract.** The article presents a new approach towards reducing an overall cost of electric power network with maintaining its reliability. Goals are achieved by implementing an ant colony algorithm with a cut-set method as a method for reliability evaluation. The algorithm solves the problem of multi-objective optimisation, where both the network cost and network reliability index, known as unavailability, should be minimalised. The network cost is considered as a linear function of overall length of network's connections. For reliability evaluation in the cut-set method, real empiric data of hazard rate for overhead power lines are used. Parallel-series network structure, equivalent by means of reliability to analysed network, is generated through the cut-set method to compute unavailability of trial solutions. Sections of the structure are generated on the basis of minimum cut set, found by the algorithm for finding one- and two- minimum cuts. As used algorithm for finding minimum cuts has linear complexity, the evaluation of trial solutions is computationally effective. An example, presented in this article, provides figure of optimal network configurations found by the algorithm.

## 1 Introduction

Overhead power lines are exposed to various environmental conditions that threatens outages in continuity of electric power delivery to the end consumers. Various consumers have different endurance to power outages. As a typical household can bear up with several hours power outage, for industrial, business and utility customers, lack of power continuity could create damage and money loss. As some customers are willing to pay more for reliability of their power supply, it is up to electric grid operator to provide the best possible service.

To achieve better reliability, the grid operator has two choices. Either an improvement in reliability of key components of the network or an introduction of some redundancy into the network. Both solutions come with drawbacks because usually they require additional spending on building and maintaining the network structure. Improving reliability of such network components as overhead lines cannot be easily done, as they are vulnerable to weather conditions. Thus, fast response repair teams are used, that can repair a fault as soon as possible. The second solution, adding redundancy, provides alternative paths the can be used while some components of the network are damaged and under repair.

The designing of such power network that has the highest reliability with the lowest cost states a multi-objective optimisation problem. This paper addresses this problem by providing an approach based on adapted ant colony optimisation algorithm [1]. The

modification introduced into the algorithm involves pheromone value updates based on the value of fitness function, heuristic search of solution space, as well as effective reliability estimation of trial solutions.

Similar problems of optimisation were considered in different studies. The approach in which there is a need to maintain connections between all network nodes was analysed in [2,3]. Networks where only  $k$  nodes must be connected were considered in [4,5,6,7]. Simulated annealing was considered in [8]. Heuristic algorithms were proposed in [4,5,6]. Genetic algorithms for network optimisation were proposed in [9,10,11]. In [11] a Monte-Carlo method it was used to evaluate the reliability of networks coded by the individuals in population. In [10,11] the reliability estimation it was done by computing the sum of maximum distance between nodes. The author of this paper also designed a genetic algorithm for this problem in [12].

## 2 Ant colony optimisation

Observation on biological processes and behaviours led to design of new algorithms used in problem solving. The ant colony optimisation algorithm is based on how insects search for food. First proposed in [1], it consists in allowing a population of ants searching for trial solutions by exploring the solution space through random walking. The crucial point is that each ant deposits an information about recent movements in the form of chemical component called in biology a pheromone. The levels of pheromones are then read by other ants and used by them as a guidance in their walk.

\* Corresponding author: [l.piatek@el.pcz.czest.pl](mailto:l.piatek@el.pcz.czest.pl)

The more ants choose a path, the more likely it is for an ant to decide to follow that path. The probability of an ant choosing to move from state  $x$  to state  $y$  is given by:

$$p_{xy}^k = \frac{\tau_{xy}^\alpha \cdot \eta_{xy}^\beta}{\sum_{z \in \text{allowed}_y} \tau_{xy}^\alpha \cdot \eta_{xy}^\beta} \quad (1)$$

where  $\tau_{xy}$  is the pheromone level assigned to the connection  $(x, y)$  and  $\eta_{xy}$  is some heuristic value designed to help ants achieve their goal. Parameters  $\alpha$  and  $\beta$  allow to control balance between pheromone levels and the heuristic value  $\eta_{xy}$ . Just as in nature, the pheromone level is not constant but changes over algorithms iteration. First, as said before, every ant deposits some amount of pheromone while passing from state  $x$  to state  $y$ , and second the pheromone fades over time. The evanescence of pheromone is regulated by the pheromone evaporation coefficient  $\rho$  of the algorithm. For each connection the pheromone level  $\tau_{xy}$  is updated as follows:

$$\tau_{xy} \leftarrow (1 - \rho)\tau_{xy} + \sum_k (\Delta\tau_{xy}^k) \quad (2)$$

where  $\Delta\tau_{xy}^k$  is the pheromone value deposited by  $k$ -th ant.

The classic application of ant colony optimisation algorithm is for the traveling salesman problem. Other applications are e.g. Job-shop scheduling problem, Quadratic assignment problem, Capacitated vehicle routing problem. This paper presents a new approach where ant colony algorithm has been adapted to deal with a multi-objective problem of optimisation of network structure. The adaptation involves a heuristic based on connections reliability and a new way of updating pheromone levels that takes into consideration the value of fitness function.

### 3 Network reliability estimation with cut-set algorithm

The evaluation of network reliability is necessary to correctly identify the best solution and to guide the algorithm presented in this paper. For this purpose, the cut-set method is used [13]. This method allows to compute reliability of connection between two distinct nodes in a graph. The nodes are called the source  $s$  and the sink  $t$ . The cut-set method consists in replacing a complex network by an equivalent, by means of reliability parallel-series structure. Each section of the equivalent structure is generated based on one minimal cut. The minimal cut is a set of connection that when damaged, cause disconnection of source and sink, but the repair of any single connection of this set can restore the link between source and sink. Finding all minimal cuts in a graph is a separate task. Any algorithm for finding these cuts can be used.

Once minimal cuts are found the computation depends on whether network components are renewable or not. As in this paper, the object of interest is a power grid, it is necessary to assume that connections can be repaired. Because power lines are put into an operation after fault in time that is usually short, compared to their

operational time, we can assume that simultaneous outage of more than two connection is very unlikely. This allows to consider only 1 and 2 elements minimal cut in the cut set method. Of course, any catastrophic failure, that involves all area of the network, like flood or weather storms, can destroy many connections. But for such events network redundancy is not enough to prevent failure anyway. In this paper, we deal only with connections failures that are independent.

The reliability of network connections is usually measured empirically. Failures of the network elements occur with the mean time which is called Mean Time To Failure – MTTF. The repair rate is referred as Mean Time To Repair – MTTR. These are the two network component reliability indicators used as data for the optimisation algorithm. Additionally, an index named failure rate  $\lambda$  is frequently used for describing components reliability. Since a typical component of electric power network is characterised by a constant failure rate, the number of failures occurring over a period of time referred to original, total population has the exponential distribution. Then, the relationship between component's MTTF and failure rate can be defined as  $MTTF = \lambda^{-1}$ .

The study in [14] provides values of failure rate and MTTR for various components of electric power networks. For typical overhead distribution lines, the value of hazard rate per circuit kilometre  $\lambda_p$  is 0.0625 per year and the value of MTTR is 4.0 hours, regardless of line length. Since the failure rate value is given per circuit kilometre, we need to treat a line as a series of one-kilometre length lines. Then, the following formula is used to compute reliability indices of a connection between two network's nodes:

$$\lambda = d * \lambda_p \quad (3)$$

where  $d$  is the connection length in kilometres.

For renewable components the unavailability  $U$  parameter can also be used. The unavailability is the probability of the component being in the down state. Similarly, the unavailability of power network is the probability of the network to be in the down state, in which the network is unable to fulfil its designed task. As power networks are most of the time on stand-by, it is easier to operate with the unavailability parameter.

Once we have identified all minimal cuts and formed on that basis the equivalent parallel-series network, formulae provided in [13] can be used to compute the unavailability of the network. For one component section the unavailability is:

$$U_{1comp} = \lambda \cdot MTTR \quad (4)$$

where  $\lambda$  is the hazard rate of the component in this section and MTTR is the component's mean repair time. For a parallel connection of two component, with hazard rate respectively  $\lambda_1$  and  $\lambda_2$  and the repair time  $MTTR_1$  and  $MTTR_2$ , the unavailability is:

$$U_{2comp} = \lambda_1 \cdot \lambda_2 \cdot MTTR_1 \cdot MTTR_2 \quad (5)$$

Formulae (4,5) are only valid if  $MTTF \gg MTTR$  for each network's component. This is true for most real electric power systems. Finally, the unavailability of the parallel-series network  $U_n$  is obtained by adding up the unavailability of each of the  $m - one-$  element sections and each of the  $n - two-$ elements parallel sections:

$$U_n = \sum_{i=0}^m U_{1comp}^i + \sum_{i=0}^n U_{2comp}^i \quad (6)$$

The special case is when a network has no minimal cut through less than three components. In such a case, because the probability of simultaneous failure of more than two components is very low in practical systems, the network is treated as perfectly reliable with the unavailability value of zero.

#### 4 Algorithm for optimisation of reliability of complex network

The presented in this chapter algorithm solves the multi-objective optimisation problem of designing a network connecting two nodes with the maximum reliability and minimum cost. It is assumed, that the reliability and cost of network connections are provided as data. The designed network connects source node  $s$  with sink node  $t$  through network nodes defined as data for the algorithm. Algorithm can use arbitrary subset of nodes for constructing the network.

As mentioned in chapter 3, it is more convenient to operate with unavailability rather than reliability parameters. With unavailability, the fitness function for optimisation is defined as follows:

$$F(x) = w_c f_c(x) + w_u f_u(x) \quad (7)$$

and

$$w_c + w_u = 1 \quad (8)$$

where  $w_c$  is the network cost weight and  $w_u$  is the network unavailability weight.  $f_u(x)$  is the network unavailability. The normalised network cost in  $f_c(x)$  is defined as the sum of costs  $C(i, j)$  of every connection from connections set  $E$ , compared to the cost  $C_{max}$  of full connected network:

$$f_c(x) = \frac{\sum_{(i,j) \in E} C(i,j)}{C_{max}} \quad (9)$$

The argument of functions (7) and (9) is a vector of bits values representing the network's connections set. The vector length is  $n(n-1)/2$ , where  $n$  is the number of nodes. If there is a connection between two nodes, the appropriate bit is set to one.

The goal of the algorithm is to minimise the fitness function. The algorithm consists of seven steps. The first two are preparations steps, while the rest are in a loop that is an iteration of algorithm.

*Step 1)* start with positioning all  $N_{ant}$  ants on the source node.

*Step 2)* apply a start value  $\tau_0$  of pheromone for every connection.

*Step 3)* through successive iteration move ants from node  $x$  to node  $y$  with probability given by formula:

$$p_{xy}^k = \begin{cases} \frac{\tau_{xy}^\alpha \cdot \eta_{xy}^\beta}{\sum_{z \in allowed_y} \tau_{xy}^\alpha \cdot \eta_{xy}^\beta} & \text{if } d(y, t) < d(x, t) \\ 0 & \text{if } d(y, t) > d(x, t) \end{cases} \quad (10)$$

where  $\eta_{xy}$  is the value of MTTF of a connection between nodes  $x, y$ . It is forbidden for any ant heading to the sink, to move to a node which is located further from sink node than the node where the ant is. If an ant reaches out the sink node, it starts moving back to the source node. For returning ant the probability of move is:

$$p_{xy}^k = \begin{cases} \frac{\tau_{xy}^\alpha \cdot \eta_{xy}^\beta}{\sum_{z \in allowed_y} \tau_{xy}^\alpha \cdot \eta_{xy}^\beta} & \text{if } d(y, s) < d(x, s) \\ 0 & \text{if } d(y, s) > d(x, s) \end{cases} \quad (11)$$

Thus, we have two kinds of ants. Moving to the sink or moving to the source. In implementation of the algorithm two data structure are used for each kind of ants. For every iteration, ants that arrived at the sink node are moved to the data structure responsible for returning ants. Similarly, ants returning to the sink are moved to the data structure responsible for ant going to the sink node.

Eventually this creates the situation that is similar to one found in nature. Ants travel back and forth, transporting food from sink node to source node.

In every move each ant deposits pheromone  $\Delta\tau_{xy}^k$ . The value of additional pheromone is stored in the data structure used later when applying formulae (2) in Step 6.

*Step 4)* compute the value of fitness function (7). The graph for the evaluation is created on the basis of connection pheromone level. Every connection where the level of pheromone passes a threshold value  $\tau_T$  is considered as a valid graph edge.

*Step 5)* compare computed fitness function to fitness function from previous iteration. If the fitness function is worse than that from previous iteration, apply special procedure. First, compute the average pheromone value for all connection from the graph from previous iteration. Then, set this average as a new pheromone value for every connection from previous iteration. Thus, connection with low pheromone level, that are important for improving fitness function are revived by increasing the level of pheromone.

*Step 6)* finally, for every iteration pheromone evaporation is applied (3) and new pheromone from ants' move is deposited.

*Step 7)* if there is no improvement of the value of fitness function during  $10 \cdot n$  iterations, end the algorithm. Otherwise jump to *Step 3*.

Conducted tests of the performance of the algorithm showed that it can have reduced computational time compared to genetic algorithm. The genetic algorithm maintains a set of trial solutions and evaluates all these solutions for every iteration. The presented algorithm evaluates only one solution every iteration. Although the presented algorithm requires more iteration to find

solution, the completion of every iteration is faster than in genetic algorithm. However, unlike the genetic algorithm, the algorithm finds only one solution. Sometimes it is desirable to see a set of solutions with the best value of fitness function. The network operator can then choose the desirable network configuration. The algorithm does not always find the best solution but a solution with fitness function very close to the best solution. Even then, that solutions are sufficient in practical application. It should be noted that the cut set method together with statistical relations (4-6) do not provide the exact value of unavailability. It only is an estimation.

### 5 Example

To illustrate the algorithm, we include an example of optimising seven nodes notional network. The problem is to find a set of connections that form a network which provide minimum unavailability at reasonable costs. The failure rate and MTTR for overhead electrical lines are given in [14]. For a typical overhead line, the failure rate has the value of 0,0625 fault per year per circuit kilometre. Mean time to repair has the value of 4.0 hours, regardless of line length. The coordinates of the nodes are given in table 1. It is assumed that the cost of a line is proportional to the line length and that the normalised cost is as (9).

**Table 1.** Node coordinates list.

| Node       | x [km] | y [km] |
|------------|--------|--------|
| 1 (source) | 0      | 24     |
| 2          | 11     | 11     |
| 3          | 28     | 30     |
| 4          | 42     | 11     |
| 5          | 66     | 22     |
| 6          | 73     | 5      |
| 7 (sink)   | 106    | 15     |

In order to balance the values in function (7), the weights are used. The normalised network cost takes on values between 0 and 1. In the sample network, its unavailability takes on values around  $10^{-6}$ . Thus, the adopted values of weights are  $w_c=0,00007$  and  $w_u=0,99993$ . Algorithm users can freely set up these weights to fit their needs. The value of ant algorithm parameters is:  $N_{ant} = 400$ ,  $\tau_0=0.1$ ,  $\tau_T=0.1$ ,  $\alpha=1$ ,  $\beta=0.5$ ,  $\rho=0.1$ .

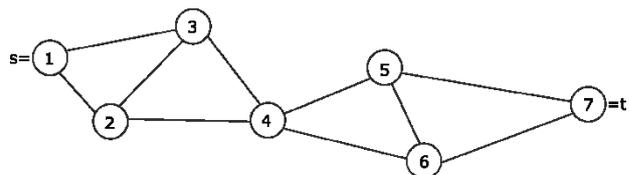
The optimal network found by the algorithm is shown in Figure 1. The value of unavailability for this network is  $2.8 \cdot 10^{-6}$  and the normalised network cost is 27% of the full connected network. By multiplying the computed network unavailability by the number of hours in a year, we can also express the network unavailability as 0.025 hours/year.

### 6 Conclusions

The presented algorithm provides an effective way to find reliable, low-cost networks. The adaptation of ant colony algorithm involves defining new heuristic

suitable for the problem, new procedure to guide the algorithm by the value of fitness function, as well as using the cut-set method to evaluate reliability. For performance reasons, the algorithm uses the time-efficient, linear algorithm for finding one- and two-elements minimal cuts in the cut-set method.

The algorithm has better computational efficiency than the genetic algorithm applied to similar problems. Nevertheless, only one solution is found. It is not possible to find a set of solutions with best value of fitness function. The algorithm deals with renewable network components. The algorithm can be used in designing reliable electrical networks.



**Fig. 1.** The optimal network found by the algorithm.

### References

1. A. Colorni, M. Dorigo et V. Maniezzo, actes de la première conférence européenne sur la vie artificielle, Paris, France, Elsevier Publishing, 134-142, (1991).
2. B. Dengiz, F. Altıparmak, A.E. Smith, IEEE Transactions on Reliability, **46**, Issue 1, pp. 18-26, (January 1997).
3. M. Gen, R. Cheng, S.S. Oren, Advances in Engineering Software, **32**, Issue 9, pp. 731-744, (September 2001).
4. K.K. Aggarwal, Y.C. Chopra, J.S. Bajwa, Microelectronics and Reliability, **22**, Issue 3, pp. 341-345, (May 1982).
5. K.K. Aggarwal, Y.C. Chopra, J.S. Bajwa, Microelectronics and Reliability, **22**, Issue 3, pp. 347-351, (May 1982).
6. Y.C. Chopra, B.S. Sohi, R.K. Tiwari, K.K. Aggarwal, Microelectronics and Reliability, **24**, pp. 911-913, (March 1984).
7. B. Liu, K. Iwamura, Computers and Mathematics with Applications, **39**, pp. 59-69, (January 2000).
8. V. Ravi, B.S.N. Murty, P.J.Reddy, IEEE Transactions on Reliability, Vol. **46**, pp. 233-239, (March 1997).
9. A. Kumar, R.M. Pathak, Y.P. Gupta, IEEE Transactions on Reliability, **44**, pp. 63-72, (January 1995).
10. A. Kumar, R.M. Pathak, Y.P. Gupta, H.R. Parsaei, Computers and Industrial Engineering, **28**, pp. 659-670, (May 1995).
11. B. Liu, K. Iwamura, Computers and Mathematics with Applications, **39**, pp. 59-69, (January 2000).

12. Ł. Piątek, M. Rudnicki, International Journal of Applied Electromagnetics and Mechanics, **43**, pp. 93-102, (2013)
13. R. Billinton, R. Allan: *Reliability Evaluation of Engineering Systems*, 2nd ed. (New York: Plenum Press 1992).
14. R. E. Brown: *Electric Power Distribution Reliability*, Boca Raton (CRC Press 2009)