Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 2648 – 2660
© TÜBİTAK
doi:10.3906/elk-1801-261

Optimization of large electric power distribution using a parallel genetic algorithm with dandelion strategy

Jorge SABATTIN[1], **Guillermo FUERTES**[2], **Miguel ALFARO**[1], **Luis QUEZADA**[1], **Manuel VARGAS**[3,*]

[1]Department of Industrial Engineering, Faculty of Engineering, Aut noma de Chile University, Providencia, Chile
[2]Faculty of Engineering, Sciences, and Technology, Bernardo OÕHiggins University, Santiago, Chile
[3]Department of Industrial Engineering, Faculty of Engineering, University of Santiago, Santiago, Chile
[4]Faculty of Engineering and Technology, San Sebasti n University, Santiago, Chile

Received: 28.01.2018 • **Accepted/Published Online:** 03.05.2018 • **Final Version:** 28.09.2018

Abstract: The study of electrical distribution of primary networks design is oriented to reduce the construction costs and the energy losses by transmission. The topology for the implementation of distribution networks may vary according to the geographical characteristics of the final users and requires specialized optimization solutions with metaheuristics to improve the energy performance of the electrical power systems. A parallel genetic algorithm (PGA) is proposed to optimize a tree-based topology for large-scale electric power distribution networks. The proposed PGA uses the dandelion code, which allows obtaining tree-feasible solutions within each iteration of the PGA. This cannot be achieved with other metaheuristic approaches directly. Eight cores are used simultaneously. We achieve a 22.05% improvement when compared to the tree-feasible solutions obtained with its sequential version. Moreover, the computational time required by the PGA is on average 23 times lower than the sequential version. Finally, we find feasible solutions for instances of the problem with up to 50,000 nodes.

Key words: Genetics algorithms, parallel algorithms, power distribution, power system planning

1. Introduction

Planning of the primary distribution of electric energy compromises the use of limited natural resources. For this reason there is increasing scientific and economic interest in the optimal distribution of energy. According to [1, 2], in countries like Chile, Spain, Sweden, and England the costs of power distribution companies are established in accordance with reference network models, which consider average distribution losses of power and energy; this comparison encourages the companies to deliver efficient service. The primary electric energy distribution network starts from the substation, which receives the energy from the transmission line and distributes it through the feeders and up to the transformers or consumption endpoints.

In this research, the problem is finding the layout of a network of electrical distribution in a radial form, with the minimum expense in its construction and distribution of the electric energy. With large networks finding an optimal solution is not possible; for this reason, metaheuristic techniques are used that find solutions close to the optimum. In [3–5] optimization approaches were implemented that focus on different design topologies. Additionally, the optimization of the network has to take into account other metrics in order to make the whole distribution process of the network more efficient [6, 7]. These metrics are known as node-based and

*Correspondence: manuel.vargas@uss.cl

branching methods. Node-based methods include the Z-bus method [8], Newton–Raphson-based algorithms [9], and fast decoupled power flow-based algorithms [10] while branching-based methods are the backward/forward sweep-based methods [11] and loop impedance [12].

Commonly, the construction is installed in a geographical area where no previous network exists. This process is known as Greenfield. Under this approach, the geographical location of the substations and the consumption points, as well as the substation capacities and the energy requirements of the consumption points, are already known [13]. Thus, the problem consists of finding a tree network configuration of minimum construction costs and low total energy consumption.

The electric energy distribution from the substation to the consumption points can be represented using a tree-graph representation. For this purpose, we assume that the substation is equivalent to the root node, while the remaining nodes represent the consumers. Finally, the arcs of the network represent the feeders. There must be a relation between the real model and the theoretical model. An n-node graph can be represented by means of $n^{n\text{-}2}$ different trees. Furthermore, the network uses connectors for the different sections, which introduce a higher combinatorial degree in the problem [14, 15]. The main contribution of this paper is to propose a new method, the parallel genetic algorithm (PGA), using dandelion code to solve the optimization problem. The coding used allows us to explore the whole space of feasible solutions for optimal trees; this research improves on previous works [16].

These types of problems have been solved using diverse optimization techniques. These are classical methods such as mixed integer linear programming [17, 18], nonlinear dynamic programming [19, 20], and metaheuristic approaches such as tabu search [21], particle swarm optimization [22], bacterial foraging [23], simulated annealing [24], bee colony [25], and genetic algorithms (GA) [26, 27]. Others authors [28] studied the effects of the indicators of complexity of industrial structures. In [26] the authors solved a similar problem, but their study was mainly focused on the comparison between two separation techniques rather than solving the problem directly. Also, in [26], a heuristic algorithm was proposed for solving a variant of this problem. The algorithm selects the location of the transformers and builds a meshed network. This optimizes the individual location of the transformers in a Greenfield area. However, the proposed algorithm does not take into account the risk of falling into local optimal solutions. Finally, in [27] a probabilistic flow method was proposed to optimize the location of the substations and radial connectors for another variant of the electric power distribution problem. In a previous work [29] we dealt with the same problem of this paper by using a sequential GA. However, in this paper, we further propose a PGA and compare it with the sequential GA [30]. For this purpose, we use 8 processors in our proposed PGA. In what follows we assume that the locations of the substations and the location of the consumption points are known. Thus, the main goal in our problem is to find a tree network topology that minimizes the construction and energy costs of the whole network. Considering that difficulty grows as the size of the problem grows, obtaining the optimal solution in large networks becomes difficult in computational terms.

In [31] a distribution problem in a large geographical area was developed with an improved GA. The algorithm finds the location and optimal size of the high and medium voltage substations, as well as the location of feeders between the medium and high voltage substations. The strategy used in the algorithm is to make a forecast of the long-term power consumption, dividing the geographic area into smaller blocks; this allows GA to simplify the model, leaving the network with 90 nodes and also reducing the amount of power lines.

In general, with the use of GAs to resolve the distribution system Greenfield planning (DSGP) that consists of planning a new network in a place where one did not exist before, the systems fail because of the new solutions obtained with the operators of the GA. They lead to nonfeasible solutions, which must be penalized so that the objective function does not consider them in the solution of the problem.

Therefore, the feasible solution space is not easily explored. Most of the models that solve variants of this type of problem focus on the optimization of the GA operators. This technique only presents solutions for models of small magnitude. Reality-tuned network models for midsize cities involve approximately 50,000 consumption points, implying that the coding mechanism can affect GA performance. An appropriate encoding method can represent the problem more clearly, help to explore the space of feasible solutions, and make it more efficient [32].

In order to encode the tree, in this paper the dandelion code is used. The dandelion code was initially proposed by Picciotto [33] in his doctoral thesis. The code establishes a bijective function between an array of integers and a tree digraph. Furthermore, it complies with an effective representation of coverage trees, such as locality (similar coding corresponds to similar trees), feasibility (the operators, who manage the coding, have to produce just coverage trees), and efficiency (the operations have to be computationally efficient) [34]. This code has also been used for multiobjective optimization design of data network topologies [35].

In order to propose a PGA with the dandelion strategy we model the DSGP using a graph $G = (U,\ A)$ where U represents the substation set and the consumption nodes, while the set A represents a set of directed arcs that connect all consumption nodes among them and with the corresponding substations. There is a subset G called Γ, where Γ represents all tree graphs that can connect all network nodes in A. In this paper, it is considered that each node can connect to any other node within the system. In order to start running the proposed PGA, an initial population is necessary. We generate this initial population by using the Prims algorithm [36]. Each of the solutions of the initial population is represented with dandelion code.

This paper is organized as follows. In Section 2, we give a brief description of the solution of the problem, the mathematical formulation of the problem is presented, and the PGA algorithm is explained. In section 3, we conduct computational numerical experiments in order to compare our PGA with its sequential version, and finally the main conclusions of the paper are given in section 4.

2. Materials and methods

In order to represent an electrical distribution tree of N nodes, with M substations and N - M consumption nodes, we will construct the array C_n with n natural numbers. All nodes are represented with natural numbers between 1 and n and this corresponds to the dandelion code C_n.

Moreover, an artificial station labeled with the number 0 is considered to form the tree from the array. Furthermore, to node 0, or the root node, is assigned the first M numbers of the array. Now we form the array A_c with the elements 1, 2, ..., n-1. Then the dandelion code is applied.

To clarify the tree representation, an example is shown. For a given electrical distribution tree with 3 distribution substations and 13 consumption nodes, it can be represented with an array of 16 elements, which are equivalent to natural numbers between 1 and 16. The artificial node is a fictional substation, which is equivalent to the root node. The substations of this electrical distribution tree are derived from the artificial substation. After the tree has been built, substation 0 is eliminated; therefore, the array is only formed by natural numbers. The array is given as C_{16}=1 3 4 12 11 1 10 9 7 4 5 7 9 10 5 2, in which 0 is imposed from its position to position 3, and then this array is placed just below an array of A_c with 15 positions. The functional relationship is established between the elements of the first row to which corresponds one element of the second row in Table 1. Each time an item in A_c is associated, it is removed from the first array and it is characterized by the function $\phi()$. With this function we have that $\phi(1)\rightarrow$ 0. Element 1 is removed from the first array, $\phi(2)\rightarrow$ 0, 2 is removed from the first array, $\phi(3)\rightarrow$ 0, and 3 is removed from the first array, ensuring that M substations are connected to the root node, in this case node 0. Then it continues with the consumption nodes as follows: $\phi(4)\rightarrow$ 12, 4 is removed from the first array, $\phi(12)\rightarrow$ 7, 12 is removed from the first array, and the algorithm is finished when the first array is empty.

Table 1. Array for function.

$$A_c = \begin{vmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 0 & 0 & 0 & 12 & 11 & 1 & 10 & 9 & 7 & 4 & 5 & 7 & 9 & 10 & 15 \end{vmatrix}$$

In this relation, the cycle $\phi(4)\rightarrow$ 12, $\phi(12)\rightarrow$ 7, $\phi(7)\rightarrow$ 10, $\phi(10)\rightarrow$ 4 and the cycle $\phi(5)\rightarrow$ 11, $\phi(11)\rightarrow$ 5 are produced. These cycles are left in a set of cycles ξ, in this case $\xi = \{(4, 12, 7, 10), (5, 11)\}$. In order to build the tree associated with this array we proceed as follows: Step 1. Connect the cycles of the set ξ to each substation alternately; this is a subtree. Then connect the element n to the last element of the last cycle, in the example 16 to 11. Step 2. Connect all elements according to function $\phi()$. For this specific example, Figure 1a depicts the geographical distribution of the fictional substation, substations, and consumption nodes. Figure 1b depicts the electrical distribution tree that corresponds to array C_{16}. In order to generate a balanced network, the first cycle is connected to the first substation, the second cycle to the second substation, and so on until all of the cycles are connected and the last node is connected to the last cycle.

2.1. Mathematical formulation

Furthermore, the model must comply with the restrictions imposed on the problem. These restrictions are given in [18], and the following notation is applied:

Parameters

$G = (U, A);\ U = \{r\} \cup M \cup N$,

r : Artificial node,

M : Set of substations,

N : Set of consumption nodes,

W : Set of conductors,

A_i^- : Set of arcs incoming from node i,

A_i^+ : Set of arcs outgoing from node i,

$B \subset A;\ B = \{a;\ a = (r, m_j),\ j = \{1, 2, ..., M\}\}$,

R_w : Electrical resistance of type w conductor,

d_j : Power required by consumer node j,

α, β : Adjustment parameters.

Variables

x_a : Power flow in arc a,

$y_a = 1$: If arc a is part of a tree, and $y_a = 0$ otherwise,

$z_{aw} = 1$: If a conductor type w is assigned to arc a,

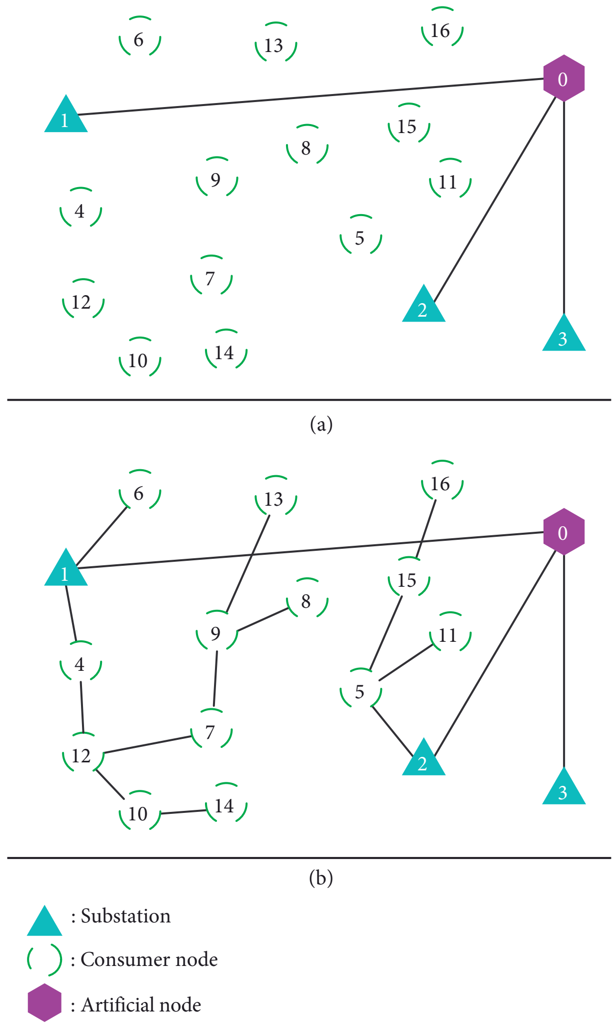

Figure 1. (a) Geographical distribution of the fictional substation, substations, and consumption nodes; (b) electric distribution tree corresponding to array C_{16}.

$z_{aw} = 0$: If a conductor type w is not assigned to arc a,

${V_a}^s$: Voltage at the beginning of arc a.

The mathematical formulation is as follows:

$$\min f(x) = \alpha \sum_{\alpha \in A} R_w z_{aw} (\frac{x_a}{V_a^s})^2 y_a + \beta \sum_{\alpha \in A} c_w z_{aw} y_a, \tag{1}$$

where the first addend is the electric energy cost and the second addend is the network construction cost function. The function is subject to the following restrictions:

$$\sum_{\alpha \in A_i^-} x_a y_a - \sum_{\alpha \in A_i^+} x_a y_a = d_j \qquad \forall j \in U \tag{2}$$

$$\sum_{\alpha \in A_i} R_w z_{aw} (\frac{x_a}{V_a^s})^2 y_a = \sum_{\alpha \in A_i} x_a - \sum_{j=1}^{|M|+|N|} d_j \tag{3}$$

$$\sum_{\alpha \in A} y_a = |M| + |N| \tag{4}$$

$$\sum_{w \in W} z_{aw} = 1 \qquad \forall a \in A \tag{5}$$

$$x_a \leq k_w z_{aw} y_a \qquad \forall a \in A \tag{6}$$

$$x_a \geq 0 \qquad \forall a \in A \tag{7}$$

$$V_a^S = 0 \geq 0 \qquad \forall a \in A \tag{8}$$

$$y_a = 0 \vee 1 \qquad \forall a \in A \tag{9}$$

$$z_{aw} = 0 \vee 1 \qquad \forall a \in A, w \in W \tag{10}$$

Eq. (2) ensures the energy of each node and Eq. (3) generates the balance of network. The condition of Eq. (4) represents the tree topology of network. The condition of Eq. (5) guarantees that one conductor is assigned to each arc and Eq. (6) guarantees the satisfaction of the capacity of each arc. Eqs. (7), (8), (9), and (10) guarantee a valid range for each variable.

The construction cost between nodes is proportional to the Euclidean distance measured from the arc that is between the nodes, multiplied by the cost of the feeder. Moreover, the cost of the losses is proportional to the square of the current flowing through each stretch, and it is determined using a power flow algorithm [37]. The method used in the investigation is described in the flow chart provided in Figure 2.

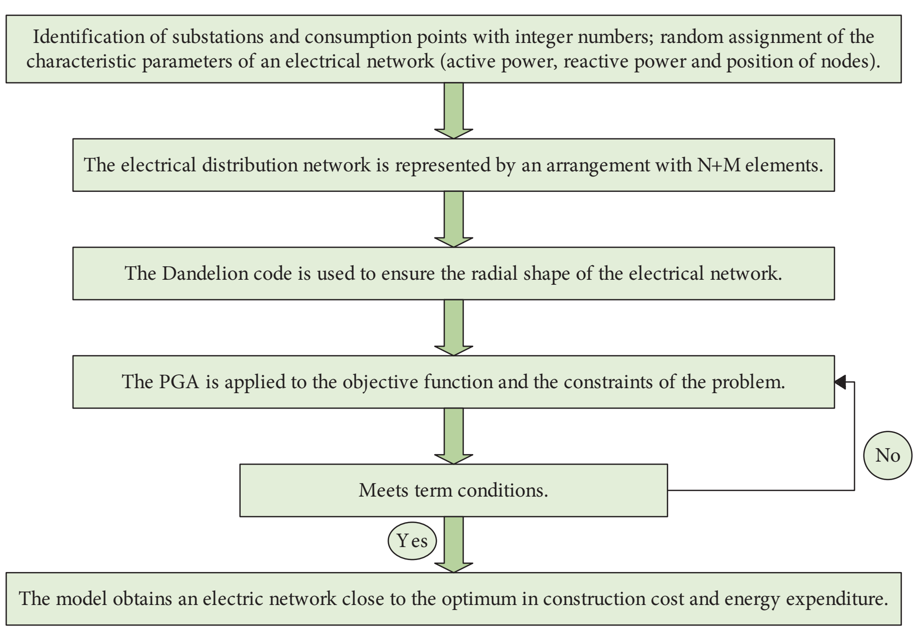

Figure 2. Flowchart for PGA.

The PGA operates in a genotype and phenotype mode. For the genotype, a string of integers is used, which corresponds to the dandelion code, while for the phenotype a tree-type network topology is used, and the progress of the populations is made using classical selection, crossover, and mutation operators, as depicted in the algorithm. The operators used are detailed below:

1. The selection operator used is a tournament of five individuals [38].
2. The crossover operator uses a crossing of two points [38]. The mutation operator is generational. However, in each generation the best current parents (elitism) replace 10% of the worst individuals generated.
3. The evaluation function is represented in Eq. (1).
4. These GAs can be very demanding in terms of computational load and memory. In order to improve the computation time of complex problems, parallel processing is used. There are several techniques for parallel processing. In this work, the master-slave model is used, and a scheme of it is shown in Figure 3. In this model the computational processing is performed on multiple processors, each of which evaluates the objective function. One processor is responsible for organizing and distributing new populations until a stopping criterion stops Algorithm 1.
5. In most of the problems solved using the PGA, the parallel processors evaluate the objective function because that is where the highest computational complexity and time consumption occurs [39].

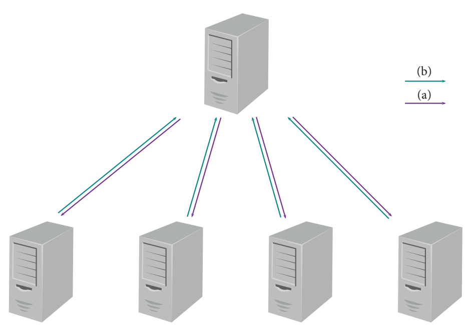

Figure 3. Master-slave for PGA.

3. Results and discussion

The used hardware was a computer cluster with 2.10 GHz AMD Opteron 6272 Processors with 64 cores (8 per experiment were used) and 128 GB RAM. The cluster used GNU/Linux as the operating system with CentOS 6.2 distribution. The software used to obtain the simulation was C++, and the open MP library was used [39] to achieve parallel processing.

The set of instances to solve and those that will be used for calibration have to be defined in the numerical experiment. The determining factor in the generation of instances is the representatives compared to reality. For example, in a city like Santiago, Chile, there are about 30 substations and over 50,000 consumption points; therefore, these numbers have provided a general idea regarding instance size, which must be generated to get representative results. In this paper, four large instances have been generated: 35,000 nodes, 40,000 nodes, 45,000 nodes, and 50,000 nodes, labeled as follows: C35, C40, C45, and C50, respectively.

Algorithm 1: Parallel genetic algorithm

```
Result: Multiple cores solutions
Initial generation();
Generation ← 1;
while Generation ≤ Maximum number generations do
    do interchange;
    evaluation of new individual;
    enddo interchange;
    for t ← 0 to Numberindividuals do
        Selection ();
        Crossover ();
        Mutation ();
    end
    Generating new population ();
    Generation ← Generation + 1;
end
```

The simulations have the following input data: active power P and reactive power Q and *(x, y)*, which have the position of the nodes in the geographical space. The proof instances consider 500 points corresponding to the consumption nodes and 20 points equivalent to the substation. For each node, P is randomly generated, with values between 0 and 1. However, Q is generated to fulfill electric power relationships, satisfying the condition $Q = P\tan(\theta)$, where θ is generated to satisfy the condition $\tan(\theta) \leq 0.8$. The active power and the reactive power of the substations are generated in the same way. Besides, the total power of the substations must be greater than the sum of the power of the consumption nodes, including the losses.

In order to simulate the geographical location of the nodes, positions in the *(x, y)* axes are randomly generated normalized values between 0 and 1. The type of conductor used in the simulations appears in Table 2.

Table 2. Type of conductor.

Conductor	Impedance ohm * km * 10^{-5}	Resistance ohm * km * 10^{-3}	Current (A)	Cost, dollars * 10^3
1	1.0	1.6	0.08429	8
2	1.0	0.8	0.12644	9
3	0.9	0.5	0.14232	10
4	0.8	0.4	0.180721	12
5	0.8	0.8	0.21073	13
6	0.7	0.4	0.24624	17

In this paper, the calibration of the parameters is done based on the type of problem to be solved, not on the size of the instance used to solve it. Therefore, the calibration of the parameters is made considering only one instance of the problem. The calibration parameters that will be used throughout the rest of the paper are given in Table 3.

The following parameters are initially studied: 30 to 50 individuals are used to characterize the population size, while a crossover probability between 0.90 and 0.95 and a mutation probability between 0.01 and 0.22 are used. The study results of the mutation are shown in Table 4. The tests indicate that the best mutation

Table 3. Calibration parameters.

Parameters	Value
Base energy (KVA)	1000
Available energy (KVA)	21,000
Energy required (KVA)	17,000
Operation time (years)	10
Base voltage (KV)	12
Number of consumers	500
Numbers of substations	20

performance is 0.16. In the final criterion, the number of generations is used, which is 500 in this case. For each instance of the input data, the following was considered: 20 substations, base equal to 1000 KV, and operation time of about 10 years. Figure 4 shows that the PGA starts with a good initial solution; then it moves away from the initial solution and finally it converges to a good quality solution. The good initial solution is explained because the initial population is generated with the Prim algorithm, which only considers construction costs.

Table 4. Results of the mutation.

Averages for mutation (dollars)	0.01	0.02	0.04	0.06
Installation	167,895	169,553	187,395	170,235
Losses	39,757,425	18,641,845	6,069,982	189,316
Total	139,925,320	18,811,398	6,257,377	359,551
Averages for mutation (dollars)	0.08	0.10	0.12	0.14
Installation	171,649	161,885	160,996	166,902
Losses	519,619	1,462,243	1,570,343	157,148
Total	691,268	1,624,128	1,731,339	324,050
Averages for mutation (dollars)	**0.16**	0.18	0.2	0.22
Installation	**157,098**	166,722	157,471	156,557
Losses	**121,282**	2,839,201	215,340	1,684,931
Total	**278,381**	3,005,924	372,810	1,841,488

In order to validate the results obtained with the PGA, these results are compared with the lower bound. The lower bound is obtained by applying the Prim algorithm to the construction costs of the network. Table 5 and Table 6 show the PGA parameters used in all instances, and Table 7 shows the comparison of the PGA with the GA.

Table 5. Parameters of PGA.

Instance	Values for (C35, C40, C45, C50)
Substation	20
% Crossover	0.9
% Mutation	0.16
Population	50
Generations	500
Cores	8

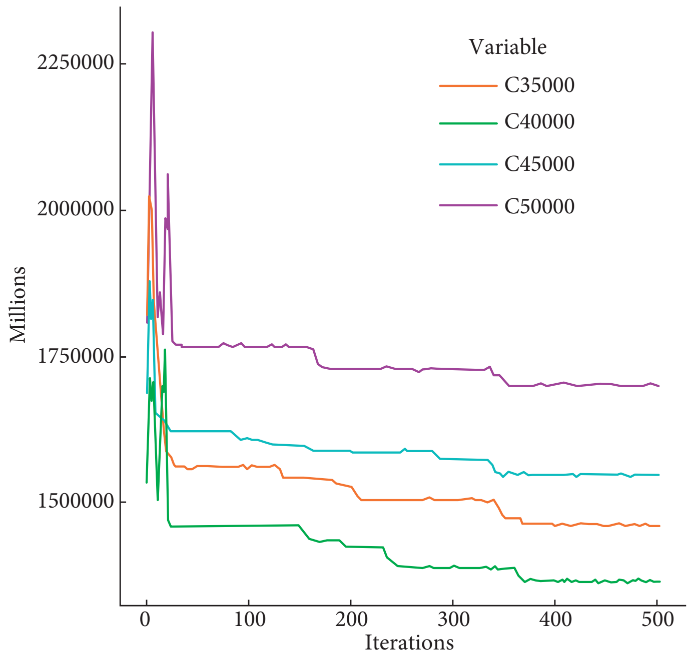

Figure 4. PGA trend in all instances.

Table 6. Results of PGA.

GKA - instance	C35	C40	C45	C50
Consumers	35,000	40,000	45,000	50,000
Lower bound	970,306	1,035,929	1,098,224	1,160,860
Crosses	22,536	22,448	22,400	22,530
Mutations	15,106	15,074	15,224	15,166
Time (s)	13,558	16,798	23,955	28,630
Average solution cost $	1,456,958	1,500,802	1,544,645	16,978,444

The PGA scheme gets better solutions than the GA. In all instances, these improvements go from the range 10.53% to 22.05%. However, the time used by the PGA is 23.75 times less than the GA.

Table 7. Comparison of PGA with GA.

Instance	C35	C40	C45	C50
PGA result (dollars)	1,456,958	1,500,802	1,544,645	1,697,844
GA result (dollars)	1,610,336	1,824,886	1,885,239	1,961,640
PGA vs. GA %	10.53	21.59	22.05	15.57
PGA time (h)	3.7	4.67	6.65	7.95
GA time (h)	87.28	110.90	127.00	142.23
PGA vs. GA	23.15	23.75	19.10	17.89

Parada et al. [17] used the Simulated Annealing Procedure tool in a network of 30,000 nodes, obtaining an average solution cost of 1,648,478, while the same authors using the Tabu Search Procedure for the same number of nodes obtained an average cost of 1,451,307. In this work with the PGA tool for the instance of

35,000 nodes, an average solution cost of 1,456,958 is obtained, a value that is within the expected solution range.

4. Conclusions

In this work, a PGA scheme is proposed in order to find the topology with the lowest transmission power and the lowest construction cost of a large electric distribution network. The networks uses thousands nodes and arcs, which generate a higher combinatorial problem with millions of feasible solutions. These combinatorial problems have been solved using diverse metaheuristic techniques like bacterial foraging, simulated annealing, bee colony, and genetic algorithms. Further, a new coding scheme is used for the PGA, which finds a solution at 22.05% lower cost than the same optimization used with the sequential GA. However, a considerable decrease is noticeable in computational time. The proposed model shows that the code is efficient using the PGA; besides, this model allows finding good solutions to large problems in relatively short processing times.

Acknowledgment

This research was supported by the DICYT (Scientific and Technological Research Bureau) of the University of Santiago of Chile and the Department of Industrial Engineering.

References

[1] Agrell PJ, Grifell-Tatje E. Wearing out the regulator: industry response to non-credible high-powered regulatory regimes. In: IEEE 10th International Conference on the European Energy Market; 27–31 May 2013; Stockholm, Sweden. New York, NY, USA: IEEE. pp. 1-8.

[2] Gonzalez-Sotres L, Mateo C, Sanchez-Miralles A, Alvear M. Large-scale MV/LV transformer substation planning considering network costs and flexible area decomposition. IEEE T Power Deliver 2013; 28: 2245-2253.

[3] Hashemi F, Alizade AR, Zebardast SJ, Ghadimi N. Determining the optimum cross-sectional area of the medium voltage feeder for loss reduction and voltage profile improvement based on particle swarm algorithm. In: IEEE 10th International Conference on Environment and Electrical Engineering; 8-11 May 2011; Rome, Italy. New York, NY, USA: IEEE. pp. 1-5.

[4] Vujicic MD, Vasovic S, Radojicic M, Vasovic JV, Nesic Z. Multi-criteria approach to optimization of optimal feeder conductor size. Metal Int 2012; 17: 47-52.

[5] Jannaty H, Moshtagh J. Siting, sizing, timing and determining the service area of MV/LV transformers: a novel approach. Arab J Sci Eng 2012; 37: 683-695.

[6] Abulâ Wafa AR. A network-topology-based load flow for radial distribution networks with composite and exponential load. Electr Pow Syst Res 2012; 91: 37-43.

[7] Mateo C, Gomez T, Sanchez-Miralles A, Peco JP, Candela A. A reference network model for large-scale distribution planning with automatic street map generation. IEEE T Power Syst 2011; 26: 190-197.

[8] Chen TH,Chen MS, Hwang KJ, Kotas P, Chebli EA. Distribution system power flow analysis â a rigid approach. IEEE T Power Deliver 1991; 6: 1146-1152.

[9] Teng JH. A novel and fast three-phase load flow for unbalanced radial distribution systems. IEEE T Power Syst 2002; 17: 1238-1244.

[10] Aravindhababu P, Ashokkumar R. A fast decoupled power flow for distribution systems. Electr Pow Compo and Sys 2008; 36: 932-940.

[11] Wu WC, Zhang BM. A three-phase power flow algorithm for distribution system power flow based on loop-analysis method. Electr Pow Syst Res 2008; 30: 8-15.

[12] Luo GX, Semlyen A. Efficient load flow for large weakly meshed networks. IEEE T Power Syst 1990; 5: 1309-1316.

[13] Gan CK, Mancarella P, Pudjianto D, Strbac G. Statistical appraisal of economic design strategies of LV distribution networks. Electr Pow Syst Res 2011; 81: 1363-1372.

[14] Moreira JC, Miguez E, Vilacha C, Otero AF. Large-scale network layout optimization for radial distribution networks by parallel computing. IEEE T on Power Deliver 2011; 26: 1946-1951.

[15] Moreira JC, Miguez E, Vilacha C, Otero AF. Large-scale network layout optimization for radial distribution networks by parallel computing: implementation and numerical results. IEEE T Power Deliver 2012; 26: 1468-1476.

[16] Parada V, Ferland JA, Arias M, Schwarzenberg P, Vargas L. Heuristic determination of distribution trees. IEEE T Power Deliver 2010; 25: 861-869.

[17] Lotero RC, Contreras J. Distribution system planning with reliability. IEEE T Power Deliver 2011; 26: 2552-2562.

[18] Shu J, Wu L, Li Z, Shahidehpour M, Zhang L, Han B. A new method for spatial power network planning in complicated environments. IEEE T Power Syst 2012; 27: 381-389.

[19] Wong S, Bhattacharya K, Fuller JD. Electric power distribution system design and planning in a deregulated environment. IEEE P-Gener Transm D 2019; 3: 1061-1078.

[20] Popovic Z, Popovic D. Graph theory based formulation of multi-period distribution expansion problems. Electr Pow Syst Res 2010; 80: 1256-1266.

[21] Rodrigues B, Cossi AM, Contreras J, Sanches JR. Multiobjective multistage distribution system planning using tabu search. IET Gener Transm Dis 2014; 8: 35-45.

[22] Ziari I, Ledwich G, Ghosh A. Optimal integrated planning of MVLV distribution systems using DPSO. Electr Pow Syst Res 2011; 81: 1905-1914.

[23] Singh S, Ghose T, Goswami SK. Optimal feeder routing based on the bacterial foraging technique. IEEE T Power Deliver 2012; 27: 70-78.

[24] Parada V, Ferland J, Arias M, Daniels K. Optimization of electrical distribution feeders using simulated annealing. IEEE T Power Deliver 2004; 19: 1135-1141.

[25] El-Zonkoly AM. Multistage expansion planning for distribution networks including unit commitment. IET Gener Transm Dis 2013; 7: 766-778.

[26] Jimenez-Estevez GA, Vargas LS, Marianov V. Determination of feeder areas for the design of large distribution networks. IEEE T Power Deliver 2010; 25: 1912-1922.

[27] Kocaman AS, Huh WT, Modi V. Initial layout of power distribution systems for rural electrification: a heuristic algorithm for multilevel network design. Appl Energ 2012; 96: 302-315.

[28] Alfaro M, Vargas M, Fuertes G, SepÃºlveda-Rojas J. Proposal of two measures of complexity based on Lempel-Ziv for dynamic systems: an application for manufacturing systems. Math Probl Eng 2018; 2018: 8692146.

[29] Briceno Vicente W, Caire R, Hadjsaid N. Probabilistic load flow for voltage assessment in radial systems with wind power. Int J Elec Power 2012; 41: 27-33.

[30] Sabattin J, Bolton CC, Arias M, Parada V. Evolutionary optimization of electric power distribution using the dandelion code. J Electr Comp Eng 2012; 2012: 738409.

[31] Najaï¬ S, Hosseinian S, Abedi M, Vahidnia A, Abachezadeh S. A framework for optimal planning in large distribution networks. IEEE T Power Syst 2009; 24: 1019-1028.

[32] Swarnkar A, Gupta N, Niazi K. A novel codification for meta-heuristic techniques used in distribution network reconfiguration. Electr Pow Syst Res 2011; 81: 1619-1626.

[33] Picciotto S. How to encode a tree. PhD, University of California, San Diego, CA, USA, 2009.

[34] Delbem ACB, de Lima TW, Telles GP. Efficient forest data structure for evolutionary algorithms applied to network design. IEEE T Evolut Comput 2012; 16: 829-846.

[35] Zhang L, Lampe M, Wang Z. Topology design of industrial ethernet networks using a multi-objective genetic algorithm. In: 6th International ICST Conference on Communications and Networking in China; 17–19 August 2011; Harbin, China. New York, NY, USA; IEEE. pp. 735-741.

[36] Prim RC. Shortest connection networks and some generalizations. J Bell Syst 1957; 36: 1389-1401.

[37] Eiben AE, Smith JE. Introduction to Evolutionary Computing. 2nd ed. Berlin, Germany: Springer, 2015.

[38] Alba E. Parallel Metaheuristics: A New Class of Algorithms. New York, NY, USA: Wiley, 2005.

[39] Dagum L, Menon R. OpenMP: an industry standard API for shared-memory programming. IEEE Comput Sci Eng 1998; 5: 46-55.