Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 781041, 12 pages
doi:10.1155/2012/781041

Research Article

An Enhanced Genetic Algorithm to Solve the Static and Multistage Transmission Network Expansion Planning

Luis A. Gallego, Marcos J. Rider, Marina Lavorato, and Antonio Paldilha-Feltrin

Departamento de Engenharia Elétrica, Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista "Júlio de Mesquita Filho" (UNESP) Avenida Brasil 56, 15385-000 Ilha Solteira, SP, Brazil

Correspondence should be addressed to Luis A. Gallego, gallegopareja@gmail.com

Received 10 October 2011; Accepted 18 November 2011

Academic Editor: Edgar Carreno

Copyright © 2012 Luis A. Gallego et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An enhanced genetic algorithm (EGA) is applied to solve the long-term transmission expansion planning (LTTEP) problem. The following characteristics of the proposed EGA to solve the static and multistage LTTEP problem are presented, (1) generation of an initial population using fast, efficient heuristic algorithms, (2) better implementation of the local improvement phase and (3) efficient solution of linear programming problems (LPs). Critical comparative analysis is made between the proposed genetic algorithm and traditional genetic algorithms. Results using some known systems show that the proposed EGA presented higher efficiency in solving the static and multistage LTTEP problem, solving a smaller number of linear programming problems to find the optimal solutions and thus finding a better solution to the multistage LTTEP problem.

1. Introduction

1.1. Metaheuristics. Metaheuristics present a common basic strategy in the search process for an optimal solution to a complex problem. The metaheuristic search process takes place through the use of transitions in the search space. This search process can be carried out from one single point or from a group of points from the search space [1]. The transition process throughout the search space is carried out according to the particular strategy of each metaheuristic. In this context, the neighborhood concept is fundamental. The transition in the search space is carried out from the present point (solution) to the neighbor that is considered more interesting by the given metaheuristic strategy. However, in the outcome of a metaheuristic, the definition of the size and shape of the present solution is fundamental.

To solve a problem using metaheuristics, a group of necessary decisions must be taken in order to adequately characterize a problem. The main choices are the following: (1) a codification proposal, (2) a form of evaluating the quality of the solution proposed (objective function, fitness, etc.), (3) the decision of whether to carry out transitions through only feasible solutions or also through the unfeasible solutions, and (4) the choice of the neighborhood. In this context, each metaheuristic adds only the search strategy that should be adopted and the stop criterion.

The codification is a way to unambiguously specify a solution proposal or, in other words, to identify an element of the search space of the problem. The elements of the codification vector are not always related to the decision variables of the problem, and it must also be observed that many problems do not have mathematical modeling. Therefore, there are many ways of coding a problem: the chosen codification can be fundamental in the outcome of the metaheuristic. Each coding solution proposal must allow the objective function of the problem or an equivalent one to be found in a simple and clear manner.

The transitions in the search space are usually carried out only through feasible solutions (neighborhoods with many feasible elements). This kind of case occurs in most of the optimization problems in the operation of electric power systems. On the other hand, there are problems for which feasible solutions can be difficult to find (all of the chosen neighborhoods involve only unfeasible solutions) and, in this case, the best proposal consists of carrying out transitions through feasible and unfeasible solutions. This type of case occurs in some electric power system operation problems

and in practically all power system expansion planning problems.

1.2. The Transmission Network Expansion Planning Problem. The objective of the transmission network expansion planning (TNEP) is to find the transmission network that must be built in order to meet the demand growth considered in a long-term scenario, minimizing investment costs and providing electric energy to all system consumers. Then, a solution to a planning problem specifies *where*, *how many*, and *when* new equipment must be installed in an electric system, so that it operates adequately within a specified planning horizon. The TNEP problem can be approached from either static or multistage point of view. In the first case, only the quantity and location of the new elements of the network are determined. In the second case, besides the quantity and location, the timing for building the new reinforcements in the network is taken into account.

The mathematical model of the TNEP corresponds to a mixed integer nonlinear programming problem. This is due to the fact that such model includes nonlinear functions, as well as integer and real variables. Furthermore, the TNEP problem presents a high combinatorial explosion, generated by the number of alternatives that can be explored. Transmission systems are usually of great size; consequently, we are dealing with a problem of difficult solution and great mathematical complexity. There are several methodologies proposed in the specialized literature to solve the TNEP problem. Initially, in [2] Garver proposes a linear power flow estimation method to efficiently determine a preliminary network that can be used to determine the optimal network. In [3–5] constructive heuristic algorithms have been used to solve the TNEP problem. Mathematical models based on classical optimization techniques, such the Benders' decomposition [6–8] and branch and bound methods [9, 10], have also been used to solve the TNEP problem. Intelligent metaheuristic algorithms such as (1) simulated annealing, (2) tabu search, (3) genetic algorithms, and (4) harmony search algorithm have been proposed in [11–14], respectively, to solve the TNEP problem. The metaheuristics are methodology that solves combinatorial optimization problems with excellent solutions and low computational cost, especially for medium and large problems. Also, the metaheuristics are more robust in terms of processing time and find better solutions than the other classical optimization methods, such as the Benders decomposition and branch and bound methods. Thus, for example, while analyzing a huge and complex problem for the relaxed model called transportation, in [9], it was verified that the Benders decomposition algorithm presents great difficulty in convergence and the branch and bound algorithm finds quality solutions but does not converge due to the prohibitive processing time. However, the metaheuristics present the best solution for the dc model with acceptable processing time [13].

Several methodologies have been developed to include concepts of security and reliability in the TNEP problem. In [15] a mathematical model and a methodology are presented to solve the TNEP problem considering the $(n-1)$ security criterion. In [5, 14] a probabilistic reliability criterion is used to solve the same problem. The TNEP problem has been incorporated in an electricity market environment in [16–18]. The AC load flow model has been used in [19, 20] to consider the active and reactive power planning at the same time. The TNEP using AC model can be solved using a metaheuristic as shown in [19]. In this case, to determine the unfeasibility of the investment proposal, a nonlinear programming problem (instead of a linear programming problem one) is solved. In this way, the nonlinear programming problem (NLP) problem consumes most of the processing time of the metaheuristic. To solve the NLP problem, an efficient and robust algorithm necessary.

A robust metaheuristic is presented in this paper to solve the TEP problem. The metaheuristic proposed is a modification of the proposal presented by Chu and Beasley [21] for the generalized assignment problem, a multiconstrained problem that presents very similar characteristics to the TEP problem. The Chu-Beasley algorithm can be considered a modified genetic algorithm but is significantly different from the traditional genetic algorithm and other modified versions that can be found in the specialized literature. Therefore, our proposal is compared to the original Chu-Beasley proposal and to a traditional genetic algorithm.

1.3. Contributions. The main contributions of this paper are threefold.

(1) An enhanced genetic algorithm to efficiently solve the static and multistage TNEP problem. The proposed algorithm found, in some test systems, better solutions than the ones reported in the specialized literature.

(2) A critical comparative analysis between the proposed genetic algorithm and traditional genetic algorithms.

(3) An efficient solution of the operative problems in the TEP problem (linear programming problems).

This paper is a natural extension of the work presented in [22]. In this case, the authors have implemented the TNEP problem using faster and more efficient software. The proposed genetic algorithm considers an initial population, in which all individuals are feasible. Also, the search for the best solution is performed only over feasible regions of the TEP problem, and, finally, efficient tools for solving the operative problems in the TEP problem are used. The proposed methodology finds better solutions, with lower computational burden, as compared with the results reported in [22].

2. Traditional Genetic Algorithm and the Chu-Beasley Proposal

The main characteristics of the traditional and the Chu-Beasley genetic algorithms will be analyzed with a focus on the differences in operators and strategies.

2.1. Traditional Genetic Algorithm. The fundamental theory on genetic algorithms can be found in [23–27]. The

traditional genetic algorithm can be summarized in the following steps.

(1) Specify the control parameters (population size, recombination rate, mutation rate, etc.).

(2) Specify genetic algorithm characteristics such as codification type, form of setting up an initial population, unfeasibility manipulation, selection type, need and form of standardization, and so forth.

(3) Find an initial population which becomes the current population.

(4) Find the fitness of the current population and update the best solution found in the process, if possible.

(5) If the stop criteria are satisfied, stop. Otherwise, continue the process.

(6) Implement selection.

(7) Implement recombination.

(8) Implement mutation, replacement of the current population and return to step 4.

The basic theory of genetic algorithms suggests randomly setting up an initial population because the quality of the final solution must be independent of how the initial population was set up. However, some studies show that a randomly generated initial population generally takes a very long time to find quality solutions and, in addition, these solutions are of poorer quality than those of an initial population set up using efficient strategies that consider specific characteristics of the problem. Therefore, a good-quality initial population generated using a fast and robust heuristic algorithm is considered more promising.

The manipulation of unfeasibilities is another relevant topic. Most genetic algorithms applied in the optimization of power systems incorporate the unfeasibilities into the fitness through penalization or eliminate the unfeasible proposal altogether [28–31]. Usually unfeasible solutions can be eliminated in the optimization of power systems operation problems because they rarely appear. However, in electric power systems planning, the penalization strategy is used because the unfeasible proposals are numerous and usually dominant and because proposals with "small unfeasibilities" can produce excellent feasible solutions after applying the genetic operators. The great difficulty is to correctly adjust the penalty parameter. In [32] there is a detailed analysis of this form of manipulating unfeasibilities as well as of other alternative proposals.

The selection type and the necessity for standardization are analyzed separately from the other genetic operators. The basic theory of genetic algorithms considers that a problem is standardized when there is a maximization problem and the fitness values are all nonnegatives. The first genetic algorithms also suggest using the so-called proportional selection, in which a standardized problem creates a number of offspring that is proportional to the fitness value. Due to this, all fitness values must be nonnegative. As the number of offspring generally is not integer, the selection process is finished using a roulette wheel. Proportional selection thus requires standardization of the problem and has two clearly differentiated parts: (1) determination of the number of offspring using the proportional relation creating a number of non-integer offspring, (2) found integer values of the number of offspring using the roulette wheel.

Proportional selection, however, presents various problems: (1) when it is a minimization problem, it must be transformed into an equivalent maximization problem; (2) when negative fitness values appear, a strategy must be used to make all the fitness values of the current population nonnegative; (3) when the so-called superconfigurations appear, that is, excellent-quality solution proposals with the high probability of generating many offspring, there is a loss of diversity and a premature convergence; (4) when all elements of the population present very close fitness values, there is also a loss of diversity, transforming the selection into a random process. The two last problems are solved by proportional selection using modifications in the selection process such as limiting the number of offspring and/or using linear scaling mechanism selection, which modifies the fitness values by approximating them when they are far apart (in the case of superconfiguration) or separating them when they are too close. In this manner, scaling mechanism selection is the same proportional selection with an additional step: the modification of the fitness values in the current population.

Selection by tournament does not present any of the problems of proportional selection and is very simple to implement. In this proposal, the number of offspring is generated in n_{pop} games for an n_{pop} size population. In each step, a game is carried out that consists of choosing random k elements of the population, with the element of the population with the best-quality fitness considered a winner and given the right to generate one offspring. After n_{pop} games, the selection process ends. This case involves only the calibration and choosing of the k, the number of elements that participate in each game, which is easily calibrated and usually presents small values such as $k = 2, 3, 4$. Logically, selection by tournament does not require a roulette wheel because it generates an integer value for the number of offspring. In this way, selection by tournament does not require standardization; thus if this decision is made, standardization issues like maximization or minimization become irrelevant.

Recombination and mutation are operators that are highly dependent on the type of codification used. When the codification is binary, both operators are implemented in a trivial way. However, if another type of codification is used, the operators must be redefined.

The traditional genetic algorithm has a generational substitution; that is, the elements of the current population are substituted by offspring generated through genetic operators. This strategy can eliminate the best solutions still found from the current population. The use of the elitism strategy is a form of dealing with this problem. In this proposal, the best solutions (elite solutions) are passed on to the next generation, preserving the best solutions. Therefore, the current population inherits the best solutions from the previous population.

Finally, the most critical problem for the traditional genetic algorithm is the loss of diversity. Generally, a genetic algorithm does not verify if some elements of the population are repeated. If a control of diversity is not implemented, the best solutions become dominant in the new populations, leading to a loss of diversity. This problem is partially solved using two strategies: (1) using more and more elevated mutation rates and (2) verifying and eliminating the repeated solutions and carrying out a recomposition of the population, which normally involves randomly generating new solutions. This last proposal is rarely used because it is computationally expensive and the substitution process is not efficient. Therefore, the loss of genetic diversity is one of the biggest problems of traditional genetic algorithms. An interesting analysis on this topic can be found in [32].

2.2. Chu-Beasley Genetic Algorithm. Chu and Beasley [21] presented a genetic algorithm to solve the generalized assignment problem. It can be summarized in the following steps.

(1) Specify the control parameters (population size, recombination rate, mutation rate, etc.).

(2) Specify genetic characteristics of the algorithm: codification type, initial population assembly, manipulation of unfeasibilities, selection type, and so forth.

(3) Randomly find an initial population to become the current population. Find the fitness and unfitness of the current population.

(4) Implement a selection to choose only two generating solutions.

(5) Implement the recombination and preserve an offspring.

(6) Implement the mutation of the preserved offspring.

(7) Implement a phase of local improvement.

(8) Decide if the improved offspring can enter the population, substituting an element from the population.

(9) If the stop criterion is not satisfied, return to step 4. Otherwise, end the process.

Chu and Beasley presented a modified genetic algorithm with very special particularities. Here we present a summary of the most relevant aspects of the Chu-Beasley genetic algorithm with special attention to those proposals that are significantly different from a traditional genetic algorithm.

The Chu-Beasley genetic algorithm (CBGA) suggests the random generation of a population just as in the basic genetic algorithms. However, it can be observed that this proposal produces an initial population with all the unfeasible elements that is very distant from the feasibility for the case of complex problems.

The CBGA presents an innovative proposal for the manipulation of unfeasibilities. This approach presents a proposal for storing the objective function (fitness vector) and the unfeasibilities (unfitness vector) in separate form and using each for different purposes. The proposal eliminates the necessity of choosing the penalization parameter when both pieces of information are gathered into one fitness alone. The fitness is used in the selection process, and the unfitness is used along with the fitness in the substitution process to decide whether the generated offspring should be incorporated into the population by substituting it for an element of the population. The CBGA uses tournament selection, considered one of the most efficient and simple-to-implement methods for selection.

The Chu-Beasley proposal is significantly different from the traditional genetic algorithms in its process of substitution of the population elements. The traditional genetic algorithm carries out a generational substitution, substituting all (or almost all) the elements of the population, and generally diversity verification is not performed. The CBGA suggests substituting only one element from the current population in each step. This facilitates two strategies that are crucial for the development of the algorithm: (1) allowing the production of improved offspring using a local optimization; (2) allowing absolute control of the diversity of the elements in the current population. These two proposals cannot be implemented efficiently in a traditional genetic algorithm with generational substitution.

The CBGA suggests the implementation of a local improvement phase of the generated offspring. This local improvement phase can be a very simple local search or a sophisticated strategy that takes the specific characteristics of the problem into consideration. However, the local improvement phase has two phases for the multiconstrained problems: (1) an unfeasibility improvement phase (2) a quality improvement phase. This way, if the generated offspring is unfeasible, there must be an attempt to make this offspring feasible, which is not always possible. Next, the quality of the offspring must be improved by searching its neighborhood.

The CBGA suggests substituting an element of the current population for a generated offspring, preserving the complete diversity; that is, all population elements must be different. Therefore, if the generated offspring is equal to an element from the population, then this offspring is discarded. Otherwise, the process follows this strategy: (1) if the generated offspring is unfeasible, then it is verified whether the unfeasibility is smaller than the unfeasibility of the element from the population with the largest unfeasibility; if this is the case, then the substitution is carried out, and if this is not the case, the generated offspring is discarded. (2) If the generated offspring is feasible, it must substitute for the element with the largest unfeasibility. Logically, if all elements of the population are feasible, for the exchange to be possible, it must be verified that the generated offspring presents better quality than the lowest-quality element of the population.

In summary, the generated offspring can enter the population if it is different from the rest of the elements of the population and if it is more promising than any of the elements, verifying first the unfeasibility and next the quality of the objective function of the feasible solution proposals. In this context, the unfitness is used to arrange the elements of the population that are unfeasible and as an "unfeasibility measurement," and the fitness represents only the original objective function of the problem and is used to arrange feasible solution proposals, as in the selection process. This logic

of initially prioritizing the search for unfeasible solutions, neglecting the quality and later preserving the feasibility of population, has been widely analyzed by researchers in recent studies [32, 33].

The proposal for the substitution in the CBGA has many relevant aspects: (1) all elements of the population are different; (2) the substitution logic increases the number of feasible elements; (3) due to the previous observation, the process finds a current population only with feasible solutions, and this stage can be reached in a period that depends on the type of problem and the local improvement strategy; (4) the best solutions are always preserved because in each substitution process a solution of inferior quality is eliminated. The last observation means that the strategy works better than the elitism of the traditional genetic algorithms. Nevertheless, the great advantage of the CBGA is the absolute control of diversity. Thus, in complex problems with great difficulty finding feasible solutions, it might be interesting to increase the size of the population to allow the storage of feasible solutions of diverse genetic compositions.

3. Enhanced Genetic Algorithm (EGA)

A metaheuristic that represents a modified version of the Chu-Beasley proposal is presented in this section. The proposal is directed towards its application in the optimization of power systems complex problems, where there is a great application and diversity of proposals related to heuristic algorithms to solve these complex problems, especially constructive heuristic algorithms. The proposal suggests modifying the Chu-Beasley algorithm in three areas: (1) the generation of the initial population, (2) the local improvement phase of the generated offspring and (3) the control of diversity can be extended.

This proposal consists of how an initial population of good quality can be generated and diversified using heuristic algorithms and very simple additional strategies. Thus, in most applications, the initial population can be made up of only feasible solution proposals, making the function of the unfitness vector from the Chu-Beasley proposal not very active or relevant. In the local improvement phase, fast and efficient heuristic algorithms can also be used, which in most cases can totally eliminate the unfeasibility of the generated offspring, which can in turn improve the quality of the objective function. The Chu-Beasley proposal for local improvement in the generalized assignment problem, in most cases, does not eliminate the unfeasibility, and the improvement in quality is also primitive. Thus in this step, efficient heuristic algorithms from the specialized literature for each type of electric power systems problem can also be used.

The control of diversity can be easily extended. In the Chu-Beasley proposal, control is limited to verifying that all the elements of the population are different. Nevertheless, practice indicates that this diversity proposal is not sufficient in multimodal and complex problems. Frequently, the individuals of a population can be different, but this can be constrained to small differences; as a consequence, the current population may represent a reduced number of regions from the search space. A simple way to deal with this problem consists of extending the diversity. In this way, an offspring can enter the current population if it meets the following parameters: (1) it presents better quality than the lowest-quality stored solution, (2) it is different from each of the elements of the population in a minimum number of elements of the coded vector.

The proposed metaheuristic can be summarized in the following steps.

(1) Specify the control parameters (population size, recombination rate, mutation rate, etc.).

(2) Specify EGA characteristics: codification type, setting up the initial population, manipulation of the infeasibilities, choice of selection by tournament, and so forth.

(3) Find an initial population with efficient, fast, and robust heuristic algorithms. The proposal is to prioritize the use of algorithms that generate only feasible solutions. Set up the fitness of the initial population.

(4) Implement a selection by tournament to choose only two generating solutions.

(5) Implement a recombination and preserve only one offspring.

(6) Implement mutation in the preserved offspring.

(7) Implement a local improvement in the preserved offspring using efficient heuristic algorithms.

(8) Decide whether the improved offspring can enter the population, substituting an element after verifying the substitution test.

(9) If the stop criterion is not satisfied, return to step 4. Otherwise, end the process.

4. Application in the Transmission Network Expansion Planning Problem

The presented metaheuristic is especially useful in complex problems with multimodal and multiconstrained characteristics. Most of the complex problems in electric power systems fall into this category. Thus, an application for the static and multistage planning problem in the electric power transmission network expansion is presented.

The static transmission network expansion planning problem is also considered a relevant and complex problem in power systems. In this case, static means that the planning is carried out in one stage. Thus, given the network configuration for a certain year and the peak generation/demand for the next stage, along with other data such as network operation limits, costs, and investment constraints, determining an expansion plan with minimum cost is desirable, to optimize where and what type of new equipment should be installed. This is a special case of a more general problem called multistage expansion planning in which it is also desirable to know when to install new pieces of equipment. In this type of problem, the initial stages of the expansion planning studies are focused on when the basic topology

of the future network is determined. Network topologies synthesized by the proposed approach will then be further analyzed and improved by testing their performance with other analytical tools such as power flow, short circuit, transient and dynamic stability analysis [34, 35]. The static and multistage transmission network expansion planning problems are mixed-integer nonlinear programming problems with many optimal solutions. In this case, optimization techniques also only find optimal local solutions for real large-scale and complex problems. The mathematical model for the transmission network expansion planning problem can take on its most complex forms when more detailed operation models are considered, such as security and operation planning in competitive markets. Metaheuristics applied to the transmission network expansion planning problem can be found in [29, 30, 36, 37].

4.1. Static Planning Modeling. The mathematical model for the static transmission network expansion planning problem, using the DC model, presents the following format [34]:

$$\text{Min} \quad v = \sum_{(i,j)\in\Omega} c_{ij} n_{ij} \tag{1}$$

$$\text{s.t.} \quad \mathbf{S} f + g = d, \tag{2}$$

$$f_{ij} - \gamma_{ij}\left(n_{ij}^0 + n_{ij}\right)(\theta_i - \theta_i) = 0, \tag{3}$$

$$\left|f_{ij}\right| - \left(n_{ij}^0 + n_{ij}\right)\overline{f_{ij}} \le 0, \tag{4}$$

$$0 \le g \le \overline{g}, \tag{5}$$

$$0 \le n_{ij} \le \overline{n_{ij}}, \tag{6}$$

$$n_{ij} \text{ integer}, \ f_{ij} \text{ and } \theta_j \text{ unbounded}, \tag{7}$$

$$\forall (i, j) \in \Omega, \tag{8}$$

where c_{ij}, γ_{ij}, n_{ij}, n_{ij}^o, f_{ij}, and $\overline{f_{ij}}$ represent, respectively, the cost of a circuit that can be added to the i-j right-of-way, the susceptance of that circuit, the number of circuits added to the i-j right-of-way, the number of circuits in the base case, the total power flow, and the corresponding maximum power flow to the circuit in the i-j right-of-way. v is the investment, $\mathbf{S}$ is the branch-node incidence matrix of the power system, f is a vector with f_{ij} elements, θ_j is the phase angle in j bus, g is a vector with g_k elements (generation in k bus) whose maximum value is $\overline{g}$, d is the demand vector, $\overline{n_{ij}}$ is the maximum number of circuits that can be added to the i-j right-of-way, and Ω is the set of all right-of-ways.

Constraint (2) represents the conservation of power in each node. This constraint models Kirchhoff's Current Law (KCL) in the equivalent DC network. Constraint (3) is an expression of Ohm's Law for the equivalent DC network, so Kirchhoff's Voltage Law (KVL) is implicitly taken into account and these constraints are nonlinear.

4.2. Multistage Planning Modeling. In multistage planning, the planning horizon is divided into several stages, for example in five-year-long stages, and in that context the equipment that should be installed in every planning stage needs to be determined. Considering an annual discount rate I, the present values of the investment costs, for the reference year t_0, with an initial year t_1, with a horizon of $t_T - t_1$ years, and with T stages, are the following [22, 29, 38]:

$$c(x) = (1-I)^{t_1-t_0} c_1(x) + (1-I)^{t_2-t_0} c_2(x) + \cdots + (1-I)^{t_T-t_0} c_T(x), \tag{9}$$

$$c(x) = \delta_{\text{inv}}^1 c_1(x) + \delta_{\text{inv}}^2 c_2(x) + \ldots + \delta_{\text{inv}}^T c_T(x)$$

$$\delta_{\text{inv}}^T = (1-I)^{t_t-t_0}, \quad t = 1, \ldots, T, \tag{10}$$

where, x represents the investment variables (lines to be constructed) and $c_t(x)$ represents the investment in the t stage. The DC model for the multistage planning problem assumes the following form [22, 29, 38]:

$$\text{Min} \quad v = \sum_{t=1}^{T}\left[\delta_{\text{inv}}^t \sum_{(i,j)\in\Omega} c_{ij} n_{ij}^t\right] \tag{11}$$

$$\text{s.t.} \quad \mathbf{S}^t f^t + g^t = d^t, \tag{12}$$

$$f_{ij}^t - \gamma_{ij}\left(n_{ij}^0 + \sum_{m=1}^{t} n_{ij}^m\right)\left(\theta_i^t - \theta_j^t\right) = 0, \tag{13}$$

$$\left|f_{ij}^t\right| \le \left(n_{ij}^0 + \sum_{m=1}^{t} n_{ij}^m\right)\overline{f_{ij}}, \tag{14}$$

$$\underline{g}_j^t \le g_j^t \le \overline{g}_j^t, \tag{15}$$

$$\underline{n}_{ij}^t \le n_{ij}^t \le \overline{n}_{ij}^t, \tag{16}$$

$$\sum_{m=1}^{T} n_{ij}^t \le \overline{n}_{ij}, \tag{17}$$

$$n_{ij}^t \text{ integer}, \ \theta_j^t \text{ unbounded}, \tag{18}$$

$$t = 1, 2, \ldots, T. \tag{19}$$

The variables are the same from the static planning except t that represents the stages.

4.3. Efficient Solution of Linear Programming Problems (LPs). In this work the methodology presented in [39] is applied to efficiently solve the LP problems (operative problems of the expansion planning problem) during the execution of the EGA. The methodology is divided in two steps: (1) reduction of the number of variables and constraints of the problem and (2) solution of the LP problem by means of a bounded dual simplex algorithm with a constraint relaxation strategy. The resulting linear programming problem only features a single equality constraint and a set of inequality constraints equal to the number of circuits of the electrical system. An important characteristic of this problem is that only a reduced number of inequality constraints are active in the

optimal solution. Due to this particular feature, an efficient strategy can then be formulated to solve the LP problems. This strategy consists of relaxing and solving the LP problem considering only the equality constraint plus the limits of the generation variables in the buses. After that, a ranking of the inequality constraints is performed in order to identify the one with the highest degree of infeasibility. Subsequently, this constraint is incorporated into the problem and it is solved again. This procedure is repeated until all the constraints of the problem are satisfied. A more detailed description of this procedure can be consulted in [39].

The results presented in [39], for various transmission systems, show that the computational time to solve the planning problem is lower than the time using a MINOS routine of Fortran. In this work, the methodology proposed in [39] was implemented in C++ and tested in a computer with a Core Duo processor of 3 GHz and 4 GB of RAM memory. Different tests were performed for various transmission systems to verify the efficiency of the proposed algorithm. For example, the Colombian system with 93 buses and 155 lines was implemented executing 1000 LPs (every LP obtains the optimal solution of the problem) with an execution time of approximately 3 seconds.

4.4. Codification in the Static Planning. The individual is a solution proposal for the planning problem or, better, is the topology made up of all the lines added to the system corresponding to an investment proposal. In this paper, a decimal codification to the number of circuits added (a solution proposal) is used.

In the transmission network expansion problem (TNEP), the individual of the EGA is represented by a vector size *nl*, that is, the number of right-of-ways. Each member of this vector corresponds to a right-of-way of the system that is being analyzed, and where new lines can be constructed. Each member can vary its value from 0 to the maximum number of lines that can be added to the respective branch. Thus, in the codification shown in Figure 1, branch 1-2 has 2 new lines, branch 3-4 has 1 new line, and so forth.

The method proposed in this paper does not demand that the characteristics of the lines between two buses be equal, they can work with various types of circuits between two buses. In this case the only change occurs in the form of the LP problem to be solved. The number of individuals in the EGA population, for the transmission network expansion problem, depends on the dimension of the system.

4.5. Codification in the Multistage Planning. In multistage planning, a solution proposal, that is, an individual of the EGA, is represented by a matrix with dimension $(n_{\text{est}} \times n_l)$, n_{est} represents the number of stages and n_l the number of candidate branches of the system. In each stage, the lines that will be constructed in that stage are presented. For example, in the codification shown in Figure 2, the branch 3–5 has 2 lines added in the first stage and 1 line added in the *n* stage. In reality, the coding separates the operation problems for each stage and therefore smaller n_{est} LP problems are solved. In summary, to identify the quality of an investment proposal,

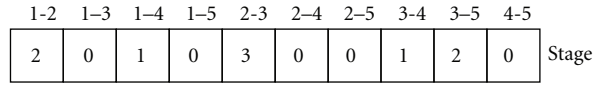


Figure 1: Static planning codification.

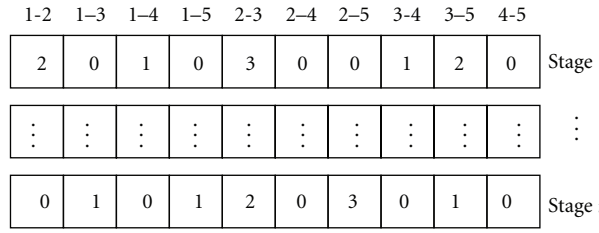


Figure 2: Multistage planning codification.

n_{est} LP problems must be solved, representing the longest part of the process.

4.6. Knowledge Incorporation in the Initialization. Considering a population randomly generated to solve the transmission network expansion planning using GA, the algorithm may need a higher computational effort, especially in medium and large systems. A way to simplify the GA is to create the initial population efficiently [13].

The initial population is set up using the Villasana-Garver-Salon (VGS) [4] constructive heuristic algorithm (CHA), which presents excellent-quality feasible solutions using additional strategies like those presented in Section 2. In the VGS algorithm, two basic concepts are observed: (1) when the integrality constraints of investment variables in the CHA are relaxed, the VGS model is transformed into an LP problem used to identify the most adequate circuit to be added to the system during the CHA interactive process; (2) each added circuit obeys the two Kirchhoff laws, so the final solution is feasible for the DC model.

4.7. Objective Function. For each individual in a population, the objective function (*fitness*) can be easily calculated. The *fitness* represents the total costs of planned lines to be constructed. The objective function is used to implement the selection and the substitution of an individual in the population.

4.8. Selection. The proposed selection of parents uses a competition based on tournaments: in each game there are *k* individuals from the current population that compete to be one of the parents. The procedure is as follows: *k* individuals are randomly chosen from the current population; the individual that has better fitness will be parent number 1. After this, the procedure is repeated and parent number 2 is then determined. Parents 1 and 2 must be different from each other. Then the parents go on to the recombination phase.

To solve large problems, it is usually more efficient to use a larger population. In this case, this means great topological diversity. However, this enlargement of population should be followed by an increase in the number of individuals in the tournament selection.

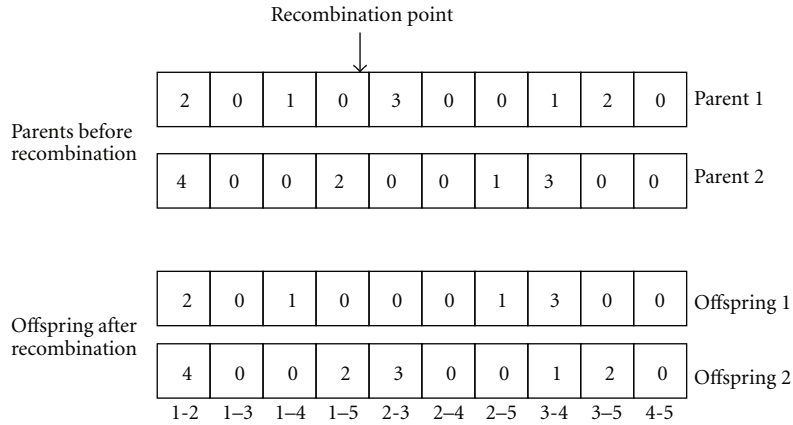


Figure 3: Single-point recombination (static codification).

4.9. Recombination. A single-point recombination is used in which a recombination point is randomly chosen and two offsprings are created. Each offspring has a piece of each parent topology, separated by the recombination point.

In the traditional GA, the two offsprings can be part of the population in the next generation. In the EGA proposal, only one offspring can do so. Thus, with the same probability, only one offspring is chosen, while the other one is eliminated. In Figure 3, we have an example of recombination where the second offspring is eliminated for static planning problems.

In Figure 4 is shown a recombination between multistage codifications when the second offspring is eliminated. The recombination must be done between stages with the same order, but the recombination point can be chosen random for different stages.

4.10. Mutation. The mutation is made in accordance with [29]. In this kind of mutation, just one point (branch) is chosen aleatorily for each stage, and one circuit of this point (branch) is added or retired aleatorily. Figures 5 and 6 show examples of mutation to the static and multistage codifications, respectively.

4.11. Local Improvement of an Offspring. The local improvement of an offspring is one of the main EGA contributions. It is made up of two types of improvement: unfeasibility and objective function improvements.

4.11.1. Improving the Unfeasibility. If the created offspring is unfeasible, that is, if it has load shedding, then the unfeasibility should be eliminated using the CHA proposed by [4].

Considering that the main objective of this phase is to completely eliminate offspring unfeasibility, the CHA proposed by VGS will add lines to the individuals to eliminate the unfeasibility, thus solving only the LP problems in each CHA step.

The Figure 7 presents the procedures for selection, recombination, mutation and elimination of the infeasibility of the offspring. Note that the proposed EGA generates only feasible solutions.

4.11.2. Improving the Objective Function. After the execution of the CHA, some lines are added to the individual in the local improvement phase. Some of the individual lines are unnecessary and should be discarded so that the individual (solution proposal) is not too expensive. Thus, all the solution proposal lines are listed in decreasing order of costs and all the lines are removed one by one. The lines that do not present load shedding when the removal of the individual is simulated are unnecessary and therefore discarded.

The lines that remain are those whose resulting solution proposal presents load shedding when their removal is simulated.

In this case, the strategy is different from that of the original Chu-Beasley proposal. In the proposal presented here, the generated offspring is made feasible, eliminating the work of the unfitness function

4.12. Convergence. The simulation process of the proposed genetic algorithm (EGA) is stopped when the incumbent (best found investment proposal) does not change after a certain predefined number of iterations.

5. Tests and Results: Static Planning

The proposed EGA to solve the transmission expansion planning problem is tested using two electrical power systems familiar to the specialized literature. The first system is the South Brazilian system, which has 46 buses and 79 branches; the second is the Colombian system of 93 buses and 155 branches.

5.1. South Brazilian System. This system has 46 buses, 79 branches, and a total demand equal to 6,880 MW. The electrical data can be found in [34]. The necessary investment to solve the planning problem for the south Brazilian system is v = 72,870,000 US$, and the following lines are added: n_{02-05} = 1, n_{05-06} = 2, n_{13-20} = 1, $n_{20\text{-}21}$ = 2, n_{20-23} = 1, $n_{42\text{-}43}$ = 1, and n_{46-6} = 1. The EGA found the solution after solving 600 LPs with a run time of 1.6 seconds on average. The initial population had 50 individuals, and the parents were selected by tournament with k = 3.

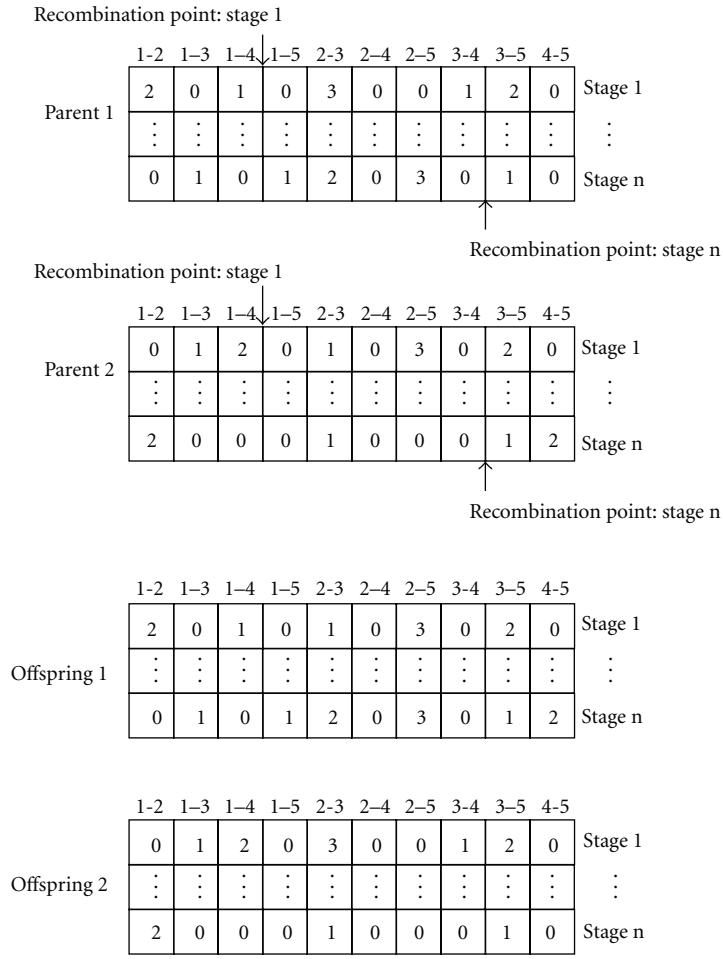


Figure 4: Single-point recombination (multistage codification).

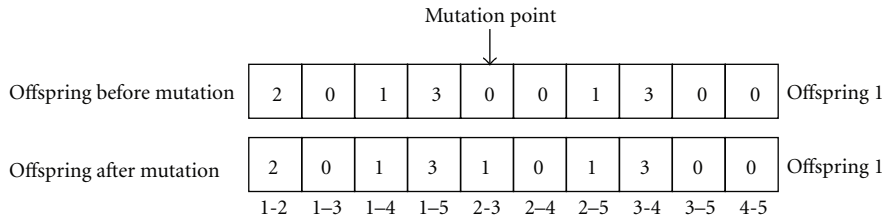


Figure 5: Mutation (static codification).

5.2. Colombian System. This system has 93 buses, 155 branches, a total demand equal to 14,559 MW, and the capacity for 5 lines to be added to each branch. This system represents a good test of the proposed methodology because it is a real system.

The proposed EGA was executed twenty times in order to determine effectiveness of this algorithm. In all tests carried out the EGA found its best solution with an investment equal to $v = 560{,}002{,}000$ US$, with a shedding load of 0.38 MW; the following lines are added: $n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-65} = 1$, $n_{30-72} = 1$, $n_{55-57} = 1$, $n_{55-84} = 1$, $n_{56-57} = 1$, $n_{55-62} = 1$, $n_{27-29} = 1$, $n_{29-64} = 1$, $n_{50-54} = 1$, $n_{62-73} = 1$, $n_{54-56} = 1$, $n_{72-73} = 1$, $n_{19-82} = 2$, $n_{82-85} = 1$, and $n_{68-86} = 1$. The EGA found the solution after solving 4,302 LPs with a run time of 16 seconds, on average. The initial population had 100 individuals, and the parents were selected by tournament with $k = 3$.

5.3. Results Analysis: Multistage Planning. The EGA was tested to solve the multistage transmission network expansion planning as well. The algorithm was tested in the Colombian system [29]. The available data allows a three-stage planning, namely P_1, P_2, and P_3. The P_1 stage is between the years 2002 and 2005, the P_2 stage is between the years 2005 and 2009, and the P_3 stage is between the years 2009 and 2012. The annual discount rate I is equal to 10%. Obviously, the circuits added to P_1 appear in the objective function

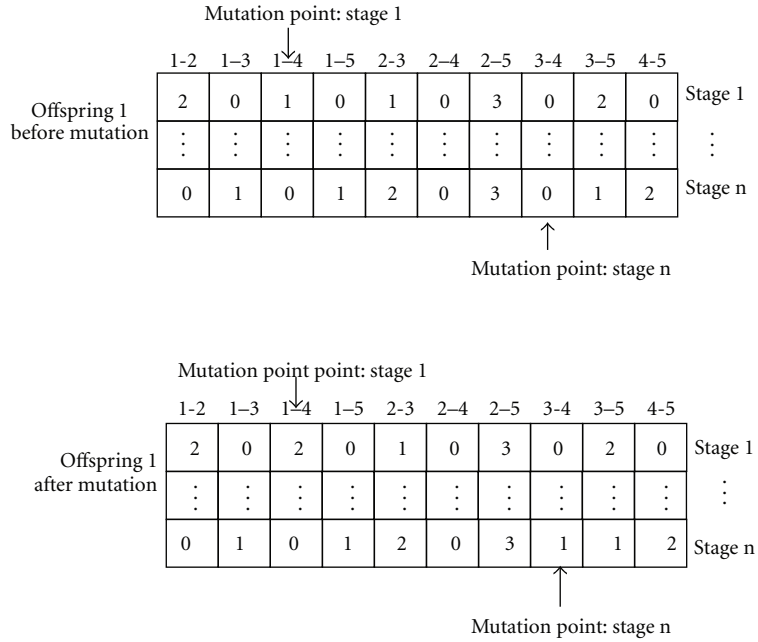


Figure 6: Mutation (multistage codification).

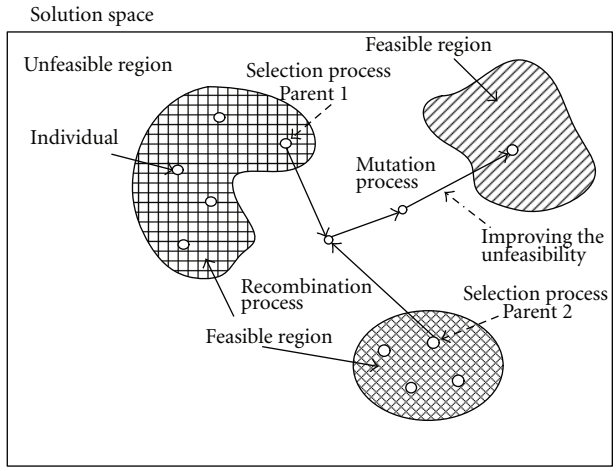


Figure 7: Local improvement process of the offspring.

with their nominal costs and those added to P_2 and P_3 are multiplied by 0.729 and 0.478, respectively. The proposed EGA was implemented with an initial population of 130 individuals, generated randomly, with feasibility generated through the CHA. A value of $k = 3$ was chosen for the tournament selection process.

As in TNEP static, the EGA was executed twenty times and the best proposal found by the proposed EGA has a capital cost equal to $v = 491{,}012{,}200$ US$ and shedding load of 0.4123 MW. Table 1 presents the best and worst investment proposals found by the EGA (stored in the matrix of codification) for an execution, as well as the proposals reported in the literature [22, 29, 38]. Note that all investment proposals stored in the EGA, at the end of the iterative process, have lower investment costs than those previously reported in the literature. The best solution found by the EGA was found after solving 47,520 LPs with a run time of 160 seconds on average. Of the twenty executions, in fourteen executions the EGA found the above proposal, while in the other six executions the EGA found different proposal with a higher investment cost, the proposal with worst investment being $v = 500{,}423{,}000$ US$.

6. Comparative Analysis between the Chu-Beasley and the Proposed EGA

In order to test the robustness and efficiency of the proposed algorithm and compare it with the performance of the Chu-Beasley genetic algorithm, a real-size power system (Colombian system) is used. Tests were carried out with both static and multistage TNEP problems. In the first case, the Chu-Beasley genetic algorithm found the same solution as the proposed EGA; however, the number of PLs executed was 68,441, with a computational time of 4.12 minutes. Comparing these results with the ones found using the proposed EGA (see Section 5.2) it can be noted that, for this particular system, the EGA is nearly 16 times faster than the traditional Chu-Beasley genetic algorithm.

On the other hand, the best solution found by the Chu-Beasley genetic algorithm for the multistage TNEP problem presents an investment cost of $v = 500{,}423{,}000$ US$. This solution was found in 856 iterations after solving 101,166 PLs in 4.71 minutes. In an attempt to find better solutions, the same algorithm was set to run up to 20,000 iterations; however, after solving 564,393 PLs no better solutions were found. For this same problem the proposed EGA found better solutions in less computational time as presented in Section 5.3.

Table 1: Investment proposals for the multistage expansion planning.

	Proposal of [29]	Proposal of [22]	Proposal of [38]	Better proposal by the EGA	Worst proposal by the EGA
Total investment cost	514,356,716 US$	503,717,434 US$	505,802,470 US$	491,012,200 US$	498,511,200 US$
Total load of shedding	0.0000 MW	0.0000 MW	0.0000 MW	0.4123 MW	0.000 MW
Investment cost stage 1	316,440,000 US$	338,744,000 US$	338,744,000 US$	338,744,000 US$	352,014,000 US$
Load of shedding stage 1	0.000 MW	0.000 MW	0.000 MW	0.000 MW	0.000 MW
Circuits proposal stage 1	$n_{56-81} = 1$, $n_{55-57} = 1$ $n_{56\text{-}57} = 1$, $n_{55-62} = 1$ $n_{45-81} = 1$, $n_{82-85} = 1$	$n_{57-81} = 2$, $n_{55-57} = 1$, $n_{55-62} = 1$, $n_{45-81} = 1$, $n_{82-85} = 1$	$n_{57-81} = 2$, $n_{55-57} = 1$, $n_{55-62} = 1$, $n_{45-81} = 1$, $n_{82-85} = 1$	$n_{57-81} = 2$, $n_{55-57} = 1$, $n_{55-62} = 1$, $n_{45-81} = 1$, $n_{82-85} = 1$	$n_{57-81} = 2$, $n_{55-57} = 1$, $n_{55-62} = 1$, $n_{19-82} = 1$, $n_{82-85} = 1$
Investment cost stage 2	122,011,272 US$	76,362,750 US$	76,362,750 US$	76,362,750 US$	66,688,920 US$
Load of shedding stage 2	0.0000 MW	0.0000 MW	0.0000 MW	0.0000 MW	0.0000 MW
Circuits proposal stage 2	$n_{56\text{-}57} = 1$, $n_{27-29} = 1$, $n_{62-73} = 1$, $n_{72\text{-}73} = 1$, $n_{19-82} = 1$	$n_{27-29} = 1$, $n_{62-73} = 1$, $n_{72\text{-}73} = 1$, $n_{19-82} = 1$	$n_{27-29} = 1$, $n_{62-73} = 1$ $n_{72\text{-}73} = 1$, $n_{19-82} = 1$	$n_{27-29} = 1$, $n_{62-73} = 1$ $n_{72\text{-}73} = 1$, $n_{19-82} = 1$	$n_{27-29} = 1$, $n_{62-73} = 1$, $n_{72\text{-}73} = 1$
Investment cost stage 3	75,905,444 US$	88,610,684 US$	90,695,720 US$	75,905,444 US$	79,808,140 US$
Load of shedding stage 3	0.0000 MW	0.0000 MW	0.0000 MW	0.4123 MW	0.0000 MW
Circuit proposal stage 3	$n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-65} = 1$, $n_{30-72} = 1$, $n_{55-84} = 1$, $n_{29-64} = 1$, $n_{19-82} = 1$, $n_{68-86} = 1$	$n_{52-88} = 1$, $n_{15-18} = 1$, $n_{55-84} = 1$, $n_{55-62} = 1$ $n_{29-31} = 1$, $n_{29-64} = 1$ $n_{68-86} = 1$	$n_{52-88} = 1$, $n_{15-18} = 1$, $n_{55-84} = 1$, $n_{55-62} = 1$, $n_{29-31} = 1$, $n_{29-64} = 2$ $n_{68-86} = 1$	$n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-65} = 1$, $n_{30-72} = 1$, $n_{55-84} = 1$, $n_{29-64} = 1$, $n_{19-82} = 1$, $n_{68-86} = 1$	$n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-65} = 2$, $n_{55-84} = 1$, $n_{29-64} = 1$, $n_{19-82} = 1$, $n_{68-86} = 1$

7. Conclusions

An enhanced genetic algorithm was applied to static and multistage long-term transmission network expansion planning. The achieved results for medium and large systems show excellent performance. The parameters for each simulation were determined after exhaustive tests.

The Chu and Beasley proposal can be considered a special type of genetic algorithm, but it represents a different form of recomposition of the population that allows a local improvement phase and control of diversity. The proposal in this work is to additionally generate an initial population using fast, efficient heuristic algorithms; promote better implementation of the local improvement phase; have more efficient control of genetic diversity.

The enhanced genetic algorithm showed itself to be more efficient than the other metaheuristics in solving the static planning problem in that it required solving fewer LP problems to find the optimal solutions. It was also more efficient in solving the multistage planning problem, finding a better solution after solving fewer LP problems than those reported in the literature.

An efficient metaheuristic is important when the mathematical model of expansion planning considers other requirements because these requirements make it necessary to solve more LP problems. Some examples of more complex transmission expansion planning models are planning with security constraints, planning in a fully open market, and planning with uncertainty.

The proposed metaheuristic can be used in other power system problems, especially when there is difficulty in finding feasible proposals, multimodal problems, or problems with very complex constraints and varied decision variables.

References

[1] F. Glover and G. A. Kochenberger, *Handbook of Metaheuristics*, Springer, 2003.

[2] L. L. Garver, "Transmission network estimation using linear programming," *IEEE Transactions on Power Systems*, vol. 89, no. 7, pp. 1688–1697, 1970.

[3] A. Monticelli, A. Santos, M. V. F. Pereira, S. H. Cunha, B. J. Parker, and J. C. G. Praca, "Interactive transmission network planning using a least-effort criterion," *IEEE Transactions on Power Apparatus and Systems*, vol. 101, no. 10, pp. 3919–3925, 1982.

[4] R. Villasana, L. L. Garver, and S. J. Salon, "Transmission network planning ussing linear programming," *IEEE Transactions on Power Apparatus and Systems*, vol. 104, no. 2, pp. 349–356, 1985.

[5] M. V. F. Pereira and L. M. V. G. Pinto, "Application of sensitivity analysis of load supplying capability to interactive transmission expansion planning," *IEEE Transactions on Power Apparatus and Systems*, vol. 104, no. 2, pp. 381–389, 1985.

[6] R. Romero and A. Monticelli, "Hierarchical decomposition approach for transmission network expansion planning," *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 373–380, 1994.

[7] R. Romero and A. Monticelli, "Zero-one implicit enumeration method for optimizing investments in transmission expansion planning," *IEEE Transactions on Power Systems*, vol. 9, no. 3, pp. 1385–1391, 1994.

[8] S. Binato, M. V. F. Pereira, and S. Granville, "A new Benders decomposition approach to solve power transmission network

design problems," *IEEE Transactions on Power Systems*, vol. 16, no. 2, pp. 235–240, 2001.

[9] S. Haffner, A. Monticelli, A. Garcia, and R. Romero, "Specialised branch-and-bound algorithm for transmission network expansion planning," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 148, no. 5, pp. 482–488, 2001.

[10] M. J. Rider, A. V. Garcia, and R. Romero, "Transmission system expansion planning by a branch-and-bound algorithm," *IET Generation, Transmission and Distribution*, vol. 2, no. 1, pp. 90–99, 2008.

[11] R. Romero, R. A. Gallego, and A. Monticelli, "Transmission system expansion planning by simulated annealing," *IEEE Transactions on Power Systems*, vol. 11, no. 1, pp. 364–369, 1996.

[12] R. A. Gallego, R. Romero, and A. J. Monticelli, "Tabu search algorithm for network synthesis," *IEEE Transactions on Power Systems*, vol. 15, no. 2, pp. 490–495, 2000.

[13] R. A. Gallego, A. Monticelli, and R. Romero, "Transmission systems expansion planning by an extended genetic algorithms," *IEE Proceedings Generation, Transmission and Distribution*, vol. 145, no. 3, pp. 329–335, 1998.

[14] A. Verma, B. K. Panigrahi, and P. R. Bijwe, "Harmony search algorithm for transmission network expansion planning," *IET Generation, Transmission and Distribution*, vol. 4, no. 6, pp. 663–673, 2010.

[15] I. D. J. Silva, M. J. Rider, R. Romero, A. V. Garcia, and C. A. Murari, "Transmission network expansion planning with security constraints," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 152, no. 6, pp. 828–836, 2005.

[16] S. de la Torre, A. J. Conejo, and J. Contreras, "Transmission expansion planning in electricity markets," *IEEE Transactions on Power Systems*, vol. 23, no. 1, pp. 238–248, 2008.

[17] L. P. Garcés, A. J. Conejo, R. García-Bertrand, and R. Romero, "A bilevel approach to transmission expansion planning within a market environment," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1513–1522, 2009.

[18] P. S. Georgilakis, "Market-based transmission expansion planning by improved differential evolution," *International Journal of Electrical Power and Energy Systems*, vol. 32, no. 5, pp. 450–456, 2010.

[19] L. A. Gallego, M. J. Rider, R. Romero, and A. V. Garcia, "A specialized genetic algorithm to solve the short term transmission network expansion planning," in *Proceedings of the IEEE Bucharest PowerTech*, pp. 1–7, Bucharest, Romania, July 2009.

[20] M. J. Rider, L. A. Gallego, R. Romero, and A. V. García, "Heuristic algorithm to solve the short term transmission network expansion planning," in *Proceedings of the IEEE Power Engineering Society General Meeting (PES '07)*, pp. 1–7, June 2007.

[21] P. C. Chu and J. E. Beasley, "A genetic algorithm for the generalised assignment problem," *Computers and Operations Research*, vol. 24, no. 1, pp. 17–23, 1997.

[22] I. J. De Silva, M. J. Rider, R. Romero, and C. A. Murari, "Genetic algorithm of chu and beasley for static and multistage transmission expansion planning," in *Proceedings of the IEEE Power Engineering Society General Meeting (PES '06)*, Montreal, Canada, June 2006.

[23] D. E. Goldberg, *Genetics Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Reading, Mass, USA, 1989.

[24] C. R. Reeves, *Modern Heuristics Techniques for Combinatorial Problems*, McGraw Hill, 1995.

[25] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, 1996.

[26] S. M. Sait and H. Youssef, *Iterative Computer Algorithms with Applications in Engineering*, Wiley-IEEE Computer Society Press, 1996.

[27] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Artificial Intelligence, Springer, Berlin, Germany, 1996.

[28] K. N. Mïu, H. D. Chiang, and G. Darling, "Capacitor placement, replacement and control in large-scale distribution systems by a ga-based two-stage algorithm," *IEEE Transactions on Power Systems*, vol. 12, no. 3, pp. 1160–1166, 1997.

[29] A. H. Escobar, R. A. Gallego, and R. Romero, "Multistage and coordinated planning of the expansion of transmission systems," *IEEE Transactions on Power Systems*, vol. 19, no. 2, pp. 735–744, 2004.

[30] E. L. D. Silva, H. A. Gil, and J. M. Areiza, "Transmission network expansion planning under an improved genetic algorithm," *IEEE Transactions on Power Systems*, vol. 15, no. 3, pp. 1168–1175, 2000.

[31] W. M. Lin, F. S. Cheng, and M. T. Tsay, "Distribution feeder reconfiguration with refined genetic algorithm," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 147, no. 6, pp. 349–354, 2000.

[32] S. Venkatraman and G. G. Yen, "A generic framework for constrained optimization using genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 4, pp. 424–435, 2005.

[33] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2–4, pp. 311–338, 2000.

[34] R. Romero, A. Monticelli, A. Garcia, and S. Haffner, "Test systems and mathematical models for transmission network expansion planning," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 149, no. 1, pp. 482–488, 2002.

[35] G. Latorre, R. Darío Cruz, J. M. Areiza, and A. Villegas, "Classification of publications and models on transmission expansion planning," *IEEE Transactions on Power Systems*, vol. 18, no. 2, pp. 938–946, 2003.

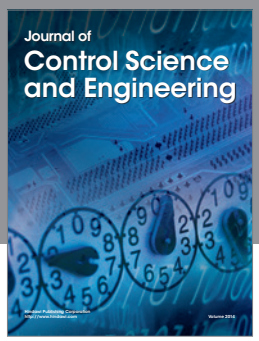
[36] R. A. Gallego, A. Monticelli, and R. Romero, "Comparative studies on non-convex optimization methods for transmission network expansion planning," *IEEE Transactions on Power Systems*, vol. 13, no. 3, pp. 822–828, 1998.

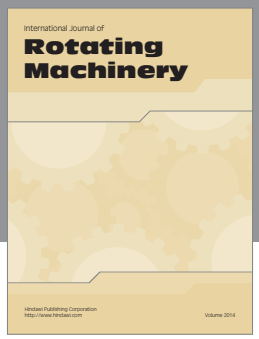
[37] E. L. Da Silva, J. M. Areiza Ortiz, G. C. De Oliveira, and S. Binato, "Transmission network expansion planning under a Tabu Search approach," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 62–68, 2001.

[38] T. Sum-Im, G. A. Taylor, M. R. Irving, and Y. H. Song, "Differential evolution algorithm for static and multistage transmission expansion planning," *IET Generation, Transmission and Distribution*, vol. 3, no. 4, pp. 365–384, 2009.

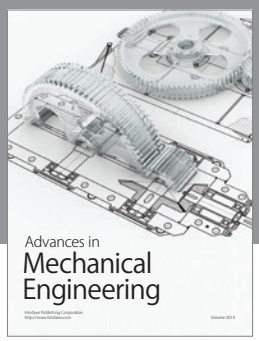
[39] S. H. M. Hashimoto, R. Romero, and J. R. S. Mantovani, "Efficient linear programming algorithm for the transmission network expansion planning problem," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 150, no. 5, pp. 536–542, 2003.



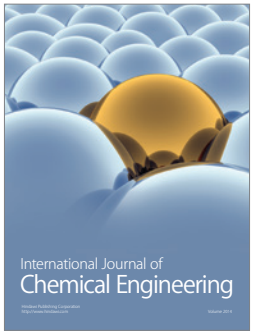




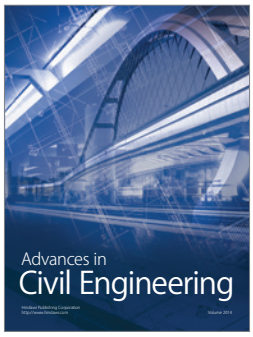


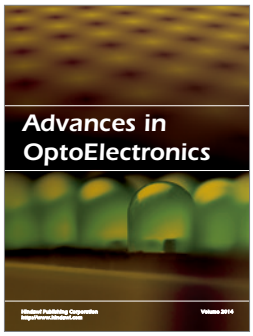




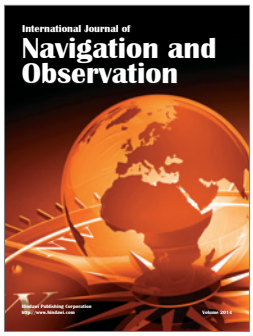


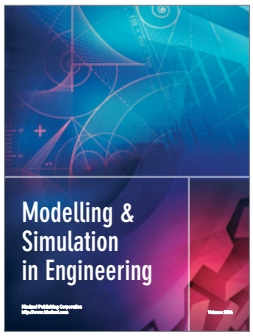






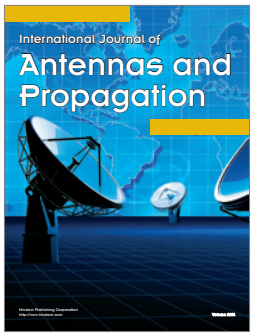




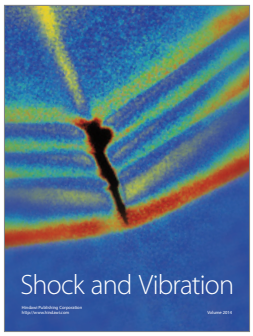














Hindawi

Submit your manuscripts at
http://www.hindawi.com