IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 25, 2022, accepted April 6, 2022, date of publication April 11, 2022, date of current version April 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3166153

A Branch and Bound Algorithm for Transmission Network Expansion Planning Using Nonconvex Mixed-Integer Nonlinear Programming Models

REINALDO T. ZOPPEI[1], MARCOS A. J. DELGADO[1], LEONARDO H. MACEDO[2], (Member, IEEE), MARCOS J. RIDER[3], (Senior Member, IEEE), AND RUBÉN ROMERO[2], (Senior Member, IEEE)

[1]Institute of Exact and Natural Sciences, Federal University of Rondonópolis, Rondonópolis, Mato Grosso 78736-900, Brazil
[2]Department of Electrical Engineering, São Paulo State University, Ilha Solteira, São Paulo 15385-000, Brazil
[3]Department of Systems and Energy, University of Campinas, Campinas, São Paulo 13083-852, Brazil

Corresponding author: Rubén Romero (ruben.romero@unesp.br)

This work was supported in part by the Coordination for the Improvement of Higher Education Personnel (CAPES)—Finance Code 001, in part by the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant 305852/2017-5, and in part by the São Paulo Research Foundation (FAPESP) under Grant 2015/21972-6 and Grant 2018/20355-1.

ABSTRACT The branch and bound (BB) algorithm is widely used to obtain the global solution of mixed-integer linear programming (MILP) problems. On the other hand, when the traditional BB structure is directly used to solve nonconvex mixed-integer nonlinear programming (MINLP) problems, it becomes ineffective, mainly due to the nonlinearity and nonconvexity of the feasible region of the problem. This article presents the difficulties and ineffectiveness of the direct use of the traditional BB algorithm for solving nonconvex MINLP problems and proposes the formulation of an efficient BB algorithm for solving this category of problems. The algorithm is formulated taking into account particular aspects of nonconvex MINLP problems, including (i) how to deal with the nonlinear programming (NLP) subproblems, (ii) how to detect the infeasibility of an NLP subproblem, (iii) how to treat the nonconvexity of the problem, and (iv) how to define the fathoming rules. The proposed BB algorithm is used to solve the transmission network expansion planning (TNEP) problem, a classical problem in power systems optimization, and its performance is compared with the performances of off-the-shelf optimization solvers for MINLP problems. The results obtained for four test systems, with different degrees of complexity, indicate that the proposed BB algorithm is effective for solving the TNEP problem with and without considering losses, showing equal or better performance than off-the-shelf optimization solvers.

INDEX TERMS Branch and bound algorithm, mixed-integer nonlinear programming, optimization, transmission network expansion planning.

NOMENCLATURE

A. FUNCTIONS

υ Objective function of the problem

B. INDICES

i, j Indices for buses
ij, ji Indices for corridors
ref Index for the reference bus

The associate editor coordinating the review of this manuscript and approving it for publication was Zhiyi Li.

C. SETS

Ω_b Set of buses
Ω_c Set of corridors

D. PARAMETERS

C_{ij} Construction cost of a line on corridor ij
D_i Active power demand at bus i
$\overline{F}_{ij}$ Maximum power flow of a line on corridor ij
$\mathcal{G}_{ij}$ Conductance of a line on corridor ij
$\underline{G}_i$ Minimum active power generation at bus i
$\overline{G}_i$ Maximum active power generation at bus i
$\overline{N}_{ij}$ Number of lines that can be constructed on corridor ij

N_{ij}^o Number of existing lines on corridor ij
R_{ij} Resistance of a line on corridor ij
X_{ij} Reactance of a line on corridor ij
Λ Load shedding cost

E. CONTINUOUS VARIABLES

f_{ij} Total power flow on corridor ij
g_i Active power generation at bus i
ℓ_{ij} Losses on corridor ij
q_i Artificial generation at bus i
θ_i Voltage phase angle at bus i

F. INTEGER VARIABLE

n_{ij} Number of lines constructed on corridor ij

I. INTRODUCTION

The branch and bound (BB) algorithm, developed by Land and Doig [1] in the 1960s, is a classic method for solving mixed-integer linear programming (MILP) problems. Its famous *divide-and-conquer* logic when solving an MILP problem first relaxes the integrality requirement of the integer variables of the original problem so that a linear programming (LP) problem can be solved. Then, a search procedure is carried out until the solution to the MILP problem is found.

The expression *branch* is related to the technique used by the algorithm to choose an integer variable with a current noninteger value. From this choice, two new LP subproblems are created that immediately divide the search region. This process is repeated until the algorithm finds the integer solution to the original problem. This strategy divides the search space into subregions, in addition to allowing regions to be rejected where the problem has no integer solution. The expression *bound* defines rules for fathoming subproblems not yet analyzed. Fathoming occurs when the algorithm detects that those subproblems, if solved, would not present an improvement in relation to the best integer solution found thus far in the process (incumbent solution). The algorithm ends the process when there is no remaining subproblem to be analyzed.

The great advantage of the BB algorithm is the guaranteed convergence toward the global solution of the problem if it exists. However, it should be noted that this guarantee is due to the type of problem that is being solved and not due to an attribute of the algorithm. Convergence to the global optimal solution occurs as a consequence of the convexity of the problem.

Because the BB algorithm is effective for MILP problems, some works have already presented extensions to apply it to solve nonconvex mixed-integer nonlinear programming (MINLP) problems [2]. This type of approach requires a support solver to solve the nonlinear programming (NLP) subproblems. These works have shown that the traditional BB algorithm for MILP problems is ineffective for solving nonconvex MINLP problems.

This work deals with the difficulties that inhibit the effectiveness of the direct use, in nonconvex MINLP problems, of a traditional BB algorithm for MILP problems. The transmission network expansion planning (TNEP) problem is used to test the proposed BB algorithm. This is a well-known optimization problem in the field of power systems, widely analyzed in the specialized literature and commonly studied by engineers and researchers in electrical energy companies due to its importance from an economic and computational point of view [3]–[5].

The mathematical model most frequently used to formulate the TNEP problem is a nonconvex MINLP problem, which considers the dc representation of the operation of the transmission system. This problem presents a very high number of locally optimal solutions due to its combinatorial nature. Several works have presented relaxed models to enable the development of solution methods compatible with the available computational tools [3].

The solution to the TNEP problem defines an optimal expansion plan of a power transmission system, adequate to maintain the continuous supply of power and to ensure that an expected increase in demand is met. In other words, it determines *where*, *how many*, and *when* new transmission lines should be built in a stipulated planning horizon in order to meet the demand growth at a minimum cost, while ensuring that there is no load shedding.

The dc model is considered by many researchers the ideal model to represent the operation of transmission systems for planning purposes since it presents a balance between complexity and precision. However, it disregards transmission losses that may not be negligible, leading to expansion plans with insufficient capacities [6]. Therefore, when solving the TNEP problem, it is very important that electrical losses of the transmission system be accounted for in the formulation.

One way of solving the TNEP problem when the dc model is used is to convert it into an MILP problem using a disjunctive formulation [7], [6]. Reference [7] proposes a disjunctive model for the TNEP problem without considering transmission losses, while [6] presents a model for the TNEP problem with a piecewise linearization for the losses. The advantage of such an approach is that the original nonconvex MINLP problem becomes an MILP problem that has the same solution as the original problem (when transmission losses are not considered). The disadvantage of such an approach is that the resulting model has a significantly larger number of constraints and continuous and discrete variables, besides a larger search space, which makes it difficult to solve large systems.

Mathematical decomposition has also been used to solve the TNEP problem. Reference [8] presents a Benders decomposition approach to solve the disjunctive model for the TNEP problem. Reference [9] proposes a decomposition framework for the TNEP problem considering renewable generation and contingency scenarios. Finally, [10] presents a Benders decomposition strategy to solve the TNEP problem with simultaneous investments in generation expansion considering contingency and load uncertainty.

Another approach, considered in this article, is to use a BB algorithm to solve the TNEP problem directly. Reference [11] proposed a BB algorithm to solve the TNEP problem using the transportation model, which is an MILP problem. Reference [12] also proposed a BB algorithm, but with an interior-point method to solve the TNEP problem using the dc model, which is an MINLP problem. In contrast to this article, [12] does not allow the flexibility of choosing different optimization solvers for solving the NLP subproblems of the BB procedure. This article presents other improvements in relation to [12], especially regarding how to treat infeasibility and convergence issues. Reference [13] presents a BB algorithm to solve the TNEP problem using the dc model without considering losses.

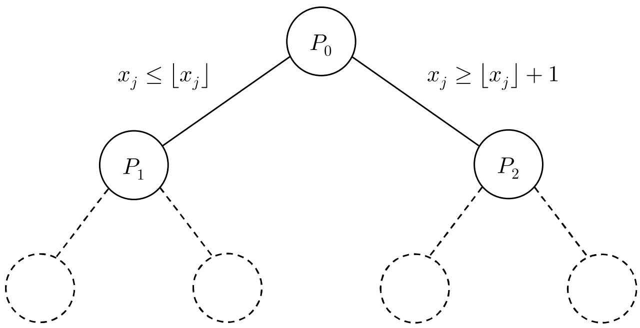


FIGURE 1. Branch and bound tree.

Metaheuristics [14] have also been widely used to solve the TNEP problem. Among the several metaheuristics available, it is possible to find works that use the particle swarm optimization algorithm [15], [16], the harmony search algorithm [17], [18], the artificial bee colony algorithm [19], the multistart algorithm with path relinking [20], and evolutionary optimization [21]. These approaches are robust and adaptable, and they provide good-quality results, but they have a number of drawbacks, including a high computing demand, parameter tuning, and the establishment of a stop condition. Furthermore, they are unable to ensure convergence to optimality [14].

More recently, [22] has proposed the use of the ac operation model in the TNEP problem. This type of problem is mainly solved using heuristics and metaheuristics [23] and convex or linearized formulations [24]. It should be noted that it is still difficult to deal with the ac TNEP problem because metaheuristics cannot ensure the optimality of a solution, while convex or linearized formulations can also lead to infeasible solutions to the original problem.

The main contributions of this work are (i) to present strategies to make the BB algorithm effective when dealing with nonconvex MINLP problems, (ii) to apply the proposed algorithm to solve the TNEP problem using the dc model with and without losses, and (iii) to carry out an evaluation of the performance of several local and global optimizations for solving nonconvex formulations for the TNEP problem.

The remainder of this article is organized as follows. Section II presents the traditional BB algorithm for MILP problems, and Section III discusses the issues of the traditional BB algorithm when solving nonconvex MINLP problems. Section IV presents the proposed strategies to adapt the BB algorithm for solving nonconvex MINLP problems; Section V presents the formulation for the TNEP problem solved in this article. Section VI presents the tests and results, and lastly, Section VII presents the conclusions of the work.

II. BB ALGORITHM FOR MIXED-INTEGER LINEAR PROGRAMMING PROBLEMS

Without loss of generality, a minimization MILP problem will be considered, as shown in (1) and (2).

$$(P) \qquad \text{minimize } \upsilon(\mathbf{x}) \tag{1}$$

$$\text{subject to } \mathbf{x} \in C, \tag{2}$$

where C is a convex set, $\mathbf{x} = (x_j)_{j \in I}$ is a vector where some components $j \in I$ are required to be integer values, I is a set of indices of the variables, and υ is a linear function. The maximization version is analogous to (1). In the BB algorithm, first, the integrality condition of the MILP problem (P) is disregarded, and a linear programming problem (P_0) is obtained. The *branch* procedure is then applied, i.e., problem (P_0) is divided into two linear programming subproblems, designated (P_1) and (P_2). This division occurs by selecting an integer variable x_j, which has a noninteger current value (division variable) that generates the subproblems (P_1) and (P_2). When considering $x_j = \lfloor x_j \rfloor + \mathfrak{f}_j$, where $\lfloor x_j \rfloor$ represents the largest integer less than x_j, i.e., the integer part of x_j obtained via truncation, and $\mathfrak{f}_j$ represents the fractional part of x_j, i.e., the truncated part, the subproblems (P_1) and (P_2) are represented as shown in (3)–(4) and (5)–(6).

$$(P_1) \qquad (P_0) \tag{3}$$

$$x_j \leq \lfloor x_j \rfloor \tag{4}$$

and

$$(P_2) \qquad (P_0) \tag{5}$$

$$x_j \geq \lfloor x_j \rfloor + 1. \tag{6}$$

If the optimal solution to the original problem, which is an MILP, is not found, the process continues with a new subdivision of both (P_1) and (P_2), each one generating two more new subproblems. In this case, the BB tree is shown in Fig. 1.

The steps of the BB algorithm that searches for a global solution to MILP problems are as follows.

1. *Initialization:* Let the value of the incumbent solution $\upsilon^* \leftarrow \infty$; initialize subproblem counter $k \leftarrow 0$; initialize the list L of candidate subproblems (P_k) with the solution of the corresponding LP problem P_0. If the solution of P_0 is integer, then let $\upsilon^* \leftarrow \upsilon_0$ and remove P_0 from L. Go to Step 2.
2. *Convergence test:* Check the list L; if $L = \emptyset$, then the process is over, and the current incumbent solution υ^*

is the optimal solution for the original MILP problem. Otherwise, go to Step 3.

3. Select and remove a candidate subproblem (P_k) from the list L; solve it and store the optimal solution υ_k. It follows from the linearity of the problem that this solution υ_k is a lower bound for all solutions of its descending subproblems.
4. *Bound step:* A candidate subproblem (P_k) will be fathomed in the following cases: (i) if the solution υ_k of the problem (P_k) is infeasible; (ii) if the solution υ_k is greater than the current incumbent solution υ^*, i.e., if $\upsilon_k > \upsilon^*$; or (iii) if the solution υ_k of the problem (P_k) is also a solution to the original MILP problem, i.e., if the integrality conditions are met. In Case (iii), if $\upsilon_k < \upsilon^*$, the incumbent solution is updated, $\upsilon^* \leftarrow \upsilon_k$; the list L must be revised so that all nodes that present $\upsilon_k \geq \upsilon^*$ are fathomed, and k must be updated. After this step, if the candidate subproblem (P_k) has been fathomed, then return to Step 2; otherwise, go to Step 5.
5. *Branch step:* From the solution of the subproblem (P_k), select an integer variable x_j with a noninteger current value for separation and add the two new descendant subproblems generated to the candidate list L. The two new subproblems are of the form (7)–(8) and (9)–(10):

$$(P_k) \tag{7}$$
$$x_j \leq \lfloor x_j \rfloor \tag{8}$$

and

$$(P_k) \tag{9}$$
$$x_j \geq \lfloor x_j \rfloor + 1, \tag{10}$$

i.e., the new subproblems are generated from the *parent* subproblem (P_k), one with an additional constraint $x_j \leq \lfloor x_j \rfloor$ and the other with an additional constraint $x_j \geq \lfloor x_j \rfloor + 1$, where $\lfloor x_j \rfloor$ is the largest integer less than x_j. After adding these two new subproblems to the list L, let $k \leftarrow k+2$ and return to Step 3.

III. THE INEFFECTIVENESS OF THE BB ALGORITHM FOR MINLP PROBLEMS WITH NONCONVEX SEARCH REGION

Suppose one wants to solve a nonconvex MINLP problem with the BB algorithm defined in Section II. For MILP problems, the traditional method is effective for determining the global solution; however, the next example points out flaws in the algorithm when it is used to solve nonconvex MINLP problems. After the example, a discussion about where the traditional BB algorithm fails will be carried out.

A. CASE STUDY: ANALYSIS OF A NONCONVEX PURE INTEGER NONLINEAR PROGRAMMING PROBLEM

Consider the nonconvex pure (with only discrete variables) integer nonlinear programming (PINLP) problem (11)–(16):

$$\text{minimize } \upsilon = x_1 + 6x_2 \tag{11}$$

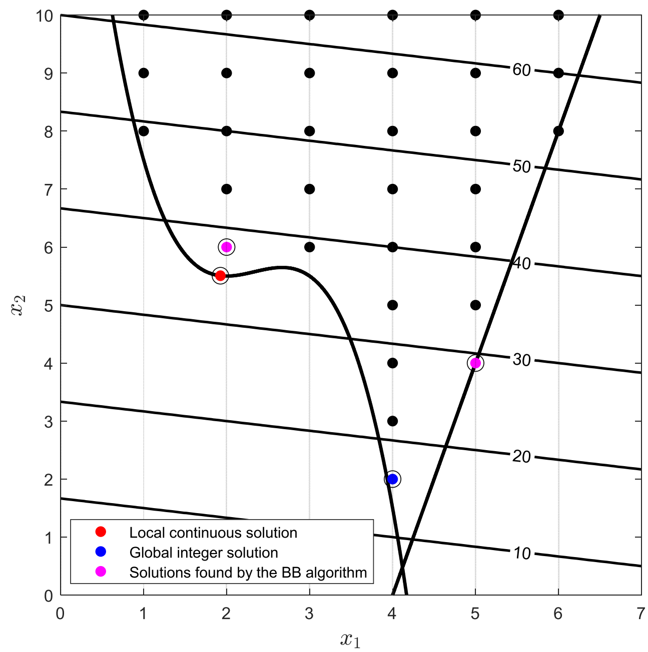


FIGURE 2. Nonconvex region of the NLP problem (11)–(16).

$$\text{subject to } (x_1-2)(x_1-2)(x_1-3) - 5.5 + x_2 \geq 0 \tag{12}$$
$$x_2 - 4x_1 + 16 \geq 0 \tag{13}$$
$$x_1 \geq 0 \tag{14}$$
$$x_2 \geq 0 \tag{15}$$
$$x_1, \quad x_2 \in \mathbb{Z}. \tag{16}$$

Let us use the solvers KNITRO v12.4.0 [25] and IPOPT v3.12.13 [26] to solve the NLP subproblems of the BB algorithm. To start the BB algorithm, note that, by relaxing the integrality of the problem, we get a local optimal solution of the NLP (P_0). In this case, both solvers found the same solution $(x_1, x_2) = (1.92509, 5.50603)$ with an objective function value $\upsilon = 34.9613$ (see Fig. 2).

From the initial solution, it is possible to apply the traditional BB algorithm to try to solve the problem (11)–(16). Using the solver KNITRO or IPOPT to solve NLP subproblems, it is possible to obtain either a solution with an objective function value $\upsilon^* = 29$ at point $(x_1, x_2) = (5, 4)$ or a solution with an objective function value $\upsilon^* = 38$ at point $(x_1, x_2) = (2, 6)$, as shown in Fig. 2, depending on the choices of the separation variable at each node and the next subproblem to be analyzed. However, the global optimal solution with an objective function value $\upsilon^* = 16$ at the point $(x_1, x_2) = (4, 2)$ cannot be reached. Besides that, the problem still has other local optimal solutions that are better than the ones found using the traditional BB algorithm.

B. WHERE THE TRADITIONAL BB ALGORITHM FAILS

Note that, even for a problem with only two variables, the counterexample (11)–(16) above has shown the deficiency of the traditional BB algorithm when dealing with problems with a nonconvex search space. The algorithm works, but it is not successful, as the nonlinearity and nonconvexity are complicating factors that interfere in the functioning of a search algorithm.

For nonconvex NLP problems, there is no guarantee of convergence or of finding a globally optimal solution. Moreover, there is a dependence on the starting point, since the problem may present many locally optimal solutions. Often, the algorithms reach local optimal solutions that are far away from the known global optimal solution. This can induce the elimination of an important node from the search tree and exclude a possible path that would lead to a better-quality optimal solution. This wrong elimination can occur whenever the algorithm finds a low-quality local solution. The nonconvexity of the feasible region, in turn, hinders the convergence of NLP problems and contributes to a deficient performance of the traditional BB algorithm defined to solve MILP problems.

Now, we must define the modifications that should be made to the traditional BB algorithm defined for solving MILP problems so that the algorithm becomes efficient in solving nonconvex MINLP problems. Therefore, we will define an improvement strategy for each point where flaws were observed.

Motivated by the counterexample, the necessity to build a BB algorithm that is effective for solving nonconvex MINLP problems is evident. Thus, it is necessary to establish approaches for dealing with the following:

1. Handling NLP subproblems;
2. Handling the nonconvexity of the problem;
3. Defining the fathoming rules;
4. Detecting the infeasibility of an NLP subproblem.

In addition to these challenges based on the nature of MINLP problems, it is critical to examine ways to improve the efficiency and effectiveness of the BB algorithm by:

5. Defining the rule for choosing the variable to separate the subproblems;
6. Defining the rule for choosing the next subproblem to be solved.

In the next section, we present strategies to answer each one of these topics.

IV. HOW TO TREAT NONCONVEX MINLP USING A BB ALGORITHM

When solving MINLP problems using the BB algorithm, it is necessary to establish a tool for solving the NLP subproblems that arise in each node of the BB tree.

A. HOW TO DEAL WITH THE NLP SUBPROBLEMS

In this work, the solvers KNITRO, IPOPT, LOQO [27], and MINOS [28] were used to solve the NLP subproblems. These four solvers were chosen after testing some popular solvers for NLP problems; these four were the most effective. The possibility of using more than one tool to solve the NLP subproblems gives more credibility to the results, besides allowing for a comparison between the tools. In this context, it is worthwhile to highlight the characteristics of the solvers that will be used to solve NLP subproblems in the proposed BB algorithm for MINLP problems.

The KNITRO solver is an optimization software that seeks local optimal solutions to large-scale NLP problems. Additionally, it presents two versions of the BB algorithm, a usual version for solving MILP problems and a specialized version for convex MINLP problems. In this work, the KNITRO solver will be used to solve the NLP subproblems of the proposed BB algorithm and will also be used to try to solve the nonconvex MINLP problem directly, allowing the performance of the proposed approach to be evaluated. The KNITRO solver offers several configuration options to approach a problem in different ways, according to the difficulties presented by each specific problem; if the user does not want to contend with this aspect, there is a setting whereby the solver itself chooses the kind of tool to use when searching for a solution.

The IPOPT solver is an open-source software for solving NLP problems, capable of solving general large-scale nonlinear programming problems. It uses a primal-dual interior-point algorithm with a linear search filter method to improve its convergence. Moreover, it is not capable of dealing with MINLP problems directly.

The LOQO solver is adequate for solving smooth constrained optimization problems. It is based on the interior-point method applied to a sequence of quadratic approximations. LOQO can handle a variety of problems, including linear or nonlinear, convex and nonconvex, and constrained and unconstrained, as long as the defining functions are smooth (at the points evaluated by the algorithm). LOQO provides a globally optimal solution for convex problems, while for nonconvex problems, it iterates from the specified starting point to find a locally optimal solution.

The MINOS solver can solve both linear and nonlinear optimization problems. It uses a linearly constrained Lagrangian method for large-scale sparse nonlinear problems. It is especially efficient at solving nonlinear objectives with linear and near-linear constraints.

B. HOW TO TREAT THE NONCONVEXITY OF THE MODEL

The nonconvexity of the feasible region is one of the complicating factors in MINLP problems, due to the existence of a large number of locally optimal solutions. It is possible that the BB algorithm will find a local optimal solution of poor quality and, through the fathoming rules, eliminate some promising nodes from the BB tree. This affects the operation of the traditional BB algorithm for solving nonconvex MINLP problems.

For the linear case, with a convex feasible region, the function maintains a linear increase/decrease, which allows for control over each solution υ_k found when solving each subproblem of the BB tree; thus, we will always have $\upsilon_k > \upsilon_{k-1}$ for the minimization case, which guarantees the functionality of the fathoming rules and the convergence to the global optimal solution. On the other hand, for NLP problems with nonconvex feasible regions, it is impossible to have control of υ_k in relation to υ_{k-1}: it may occur that $\upsilon_k \geq \upsilon_{k-1}$ or $\upsilon_k \leq \upsilon_{k-1}$, and the lack of a predefined control strategy leads the algorithm to function without the certainty of convergence. Even if the algorithm converges, the quality of the result cannot be guaranteed. The main problem, in this case, is that the solver may not converge to the global optimal solution for the relaxed problem of a node. This situation was verified for the problem illustrated in Fig. 2.

To circumvent this type of problem, it is necessary to add a safety factor $\varepsilon > 0$ to the algorithm. The security factor, $\varepsilon > 0$, increases the number of nodes to be analyzed in the BB tree, but also provides a security margin so that, after a node is analyzed and present $\upsilon_k \geq \upsilon^*$, the BB algorithm continues to preserve the descendant nodes that have a solution υ_k greater than υ^*, up to a limit $\upsilon^* + \varepsilon$, as this can prevent the loss of a promising node in the search for a better solution. In this case, it is necessary to redefine this fathoming rule, rewriting it as follows: the subproblem k that has a solution $\upsilon_k > \upsilon^* + \varepsilon$ will be discarded.

Note that when $\varepsilon = 0$, the fathoming rule number 2 (see Section IV-C) becomes the same one used in the traditional BB algorithm. If ε is a large number, then the fathoming rule number 2 will be ignored, and the BB algorithm will have a higher chance of finding the global optimal solution of the problem, at the expense of a larger BB tree and a higher computational effort. Therefore, there must be a balance when choosing the value of ε, so that the BB tree does not increase too much without a significant effect on the obtained result.

C. FATHOMING RULES

The fathoming rules that eliminate the necessity of analyzing the descendants' nodes of a parent node are:

1. When the algorithm finds a feasible solution to the original nonconvex MINLP problem, the subproblem in question is fathomed;
2. For minimization problems, the subproblem is fathomed if it has a solution $\upsilon_k > \upsilon^* + \varepsilon$, because there is no guarantee that υ_k is the best possible solution at this node for nonconvex MINLP problems, i.e., due to the nonconvexity, the guarantee that $\upsilon_k > \upsilon_{k-1}$ always occurs is lost;

1) When a subproblem is infeasible, it should be fathomed.

D. HOW TO TREAT INFEASIBILITY AND CONVERGENCE ISSUES

When solving NLP problems, it is not always an easy task to detect infeasible problems, as well as it is not possible to guarantee convergence. This contributes to the deficient functioning of the traditional BB algorithm for MILP problems when applied directly to nonconvex MINLP problems.

It should be noted that many nodes of the BB tree will present infeasible subproblems. Optimization solvers usually demand high computational effort to verify that NLP problems are infeasible. The idea is, therefore, to obtain more efficiency, robustness, and effectiveness in the procedure by solving NLP problems with higher dimensions and that are always feasible, which allows for smooth transitions in the search space of the problem.

To identify the cases of the infeasibility of NLP subproblems in a simple manner, a set of nonnegative artificial variables combined with a positive penalization parameter will be included in the constraints and in the objective function (for minimization problems) to identify infeasibility and make the algorithm reject the infeasible problems by minimizing the artificial variables.

By incorporating a penalization term into the objective function, to force the algorithm to try to zero out the artificial variables associated with it, it is likely to reach a feasible solution. However, for this to occur, it will be necessary to establish a penalization parameter with a high value. There is no criterion for determining the ideal value of this parameter: it was observed in the tests that a fixed parameter can be good for a given system and bad for another. Furthermore, a small parameter may not produce the goal because the algorithm may choose to assign positive values to the artificial variables; on the other hand, a large parameter may not allow convergence because the algorithm will try to zero out the artificial variables associated with the penalization parameter at all costs, affecting the convergence to a good-quality solution.

As we want a BB algorithm for solving nonconvex MINLP problems, it is more adequate to define a penalization parameter for each problem. Thus, for each problem, a penalization parameter Λ will be defined as a function of the objective function coefficients. Specifically, when the objective function is linear, e.g., $\upsilon(\mathbf{x}) = \sum_{j \in I} c_j x_j$, we can define $\Lambda := 2\|c\|$, where $\|c\| := \sqrt{\sum_{j \in I} c_j^2}$ is the norm of the objective function coefficients for a set of indices I. Thus, the penalization parameter Λ will have the value associated with the problem in question, related to the objective function, which allows greater flexibility in solving different systems by guaranteeing generality without the rigidity of a fixed value for all problems.

As for the convergence of each NLP subproblem, the BB algorithm will be programmed to try to solve the NLP subproblem that has not converged again by starting exactly where it left off. Up to three attempts are considered at each node. After these three attempts, the NLP subproblem will be considered to have not converged, and the respective node will be discarded. This strategy avoids dropping a node when the solver did not converge on the first try.

E. CHOOSING THE SEPARATION VARIABLE AND THE NEXT SUBPROBLEM TO ANALYZE

To improve the performance of the BB algorithm, a variety of approaches can be used, such as pseudocosts [29], [30].

When a variable is required to take on an integer value, the pseudocosts quantify the estimated degradation of the objective function. This information is utilized to reduce the size of the BB tree, allowing for the solution of a lower number of NLP subproblems.

The values of the pseudocosts, P_j^- and P_j^+ are defined in (17) and (18) for each integer variable of the TNEP problem, x_j:

$$\upsilon_k^- - \upsilon_k = P_j^- \mathfrak{f}_j^k \tag{17}$$

$$\upsilon_k^+ - \upsilon_k = P_j^+ \left(1 - \mathfrak{f}_j^k\right), \tag{18}$$

where υ_k is the value of the optimal solution of the objective function of subproblem k, υ_k^- is the value of the optimal solution of the objective function of the descendant subproblem of subproblem k with the additional constraint $x_j \leq \lfloor x_j \rfloor$, υ_k^+ is the value of the optimal solution of the objective function of the descendant subproblem of subproblem k with the additional constraint $x_j \geq \lfloor x_j \rfloor + 1$, and $\mathfrak{f}_j^k$ is the fractional part of the variable x_j in the solution of subproblem k.

To choose the separation variable, (19) will be used to estimate the variable j that provides the greatest deterioration in the value of the objective function:

$$\max_j \left\{ \min \left\{ P_j^- \mathfrak{f}_j^k, P_j^+ \left(1 - \mathfrak{f}_j^k\right) \right\} \right\} \tag{19}$$

Note that negative pseudocosts might occur as a result of the nonlinearity and nonconvexity of the problem, suggesting that the solution obtained for the NLP subproblem k is only a local optimal solution. Moreover, to obtain the pseudocosts, it is necessary to solve two NLP subproblems, which increases the computational effort of the BB algorithm.

To avoid these drawbacks related to the calculation of the pseudocosts, in the proposed BB algorithm, these values are defined at the beginning of the algorithm with the corresponding values of the coefficients of the objective function, and they do not change.

The proposed algorithm performs a breadth-first search to determine the next subproblem to be analyzed.

V. THE MINLP MODEL FOR THE TNEP PROBLEM CONSIDERING LOSSES

The nonconvex MINLP model for the TNEP problem considering the dc operation of the transmission network with transmission losses is presented in (20)–(29):

$$\text{minimize } \upsilon = \sum_{ij \in \Omega_c} C_{ij} n_{ij} + \Lambda \sum_{i \in \Omega_b} q_i \tag{20}$$

$$\text{subject to } \sum_{ji \in \Omega_c} \left(f_{ji} - \frac{\ell_{ji}}{2} \right) - \sum_{ij \in \Omega_c} \left(f_{ij} + \frac{\ell_{ij}}{2} \right) + g_i = D_i - q_i \quad \forall i \in \Omega_b \tag{21}$$

$$f_{ij} = \left(N_{ij}^o + n_{ij}\right) \left(\theta_i - \theta_j\right) / X_{ij} \quad \forall ij \in \Omega_c \tag{22}$$

$$\ell_{ij} = \mathcal{G}_{ij} \left(N_{ij}^o + n_{ij}\right) \left(\theta_i - \theta_j\right)^2 \quad \forall ij \in \Omega_c \tag{23}$$

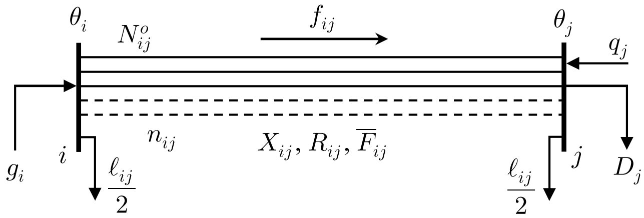


FIGURE 3. Illustration of the power balance constraint (21).

$$\left|f_{ij}\right| + \frac{\ell_{ij}}{2} \leq \left(N_{ij}^o + n_{ij}\right) \overline{F}_{ij} \quad \forall ij \in \Omega_c \tag{24}$$

$$\underline{G}_i \leq g_i \leq \overline{G}_i \quad \forall i \in \Omega_b \tag{25}$$

$$0 \leq q_i \leq D_i \quad \forall i \in \Omega_b \tag{26}$$

$$0 \leq n_{ij} \leq \overline{N}_{ij} \quad \forall ij \in \Omega_c \tag{27}$$

$$\theta_{ref} = 0 \tag{28}$$

$$n_{ij} \in \mathbb{Z} \quad \forall ij \in \Omega_c. \tag{29}$$

The objective function υ, presented in (20) includes the total investment cost of expanding the transmission system and the load shedding cost, which avoids the occurrence of infeasible NLP subproblems. Constraint (21) is the power balance equation, which corresponds to the application of Kirchhoff's current law to the system. Note that this constraint includes the artificial generation q_i, which allows the constraint to be fulfilled in any situation. Constraint (22) corresponds to the systematic application of Kirchhoff's voltage law to the system. Constraint (23) provides an estimate of the losses on corridor ij as a function of the conductance of a line, $\mathcal{G}_{ij} = R_{ij} \big/ \left(R_{ij}^2 + X_{ij}^2\right)$, the number of lines installed on corridor ij, and the voltage angle phase difference across corridor ij. Constraint (24) is the power flow transmission capacity of a branch. Constraint (25) is the generation capacity of the generation buses, and constraint (26) is the artificial generation capacity of the buses. Constraint (27) limits the maximum number of new lines that can be constructed on corridor ij. Constraint (28) defines an angular reference for the system. Finally, constraint (29) represents the integer nature of the investment variable n_{ij}. Fig. 3 illustrates the power balance constraint at buses i and j. Note that the total losses for a corridor, ℓ_{ij}, are concentrated at the terminal buses of the corridor.

In the presented model, the objective function (20) is linear, as well as constraints (21) and (24)–(29). However, constraints (22) and (23) are nonlinear and nonconvex, since they present the product of variables and squared variables. Moreover, note that (22) is a quadratic equality, and when losses are not considered in the problem, the model becomes a nonconvex mixed-integer quadratically-constrained programming (MIQCP) problem.

Fig. 4 shows f_{ij} as a function of n_{ij} and $\theta_i - \theta_j$ for $X_{ij} = 1$, while Fig. 5 shows ℓ_{ij} as a function of the same variables for $\mathcal{G}_{ij} = 1$. The possible values of f_{ij} and ℓ_{ij} when n_{ij} is an integer are represented by the red curves in Fig. 4 and Fig. 5, respectively, while the entire surfaces of both figures

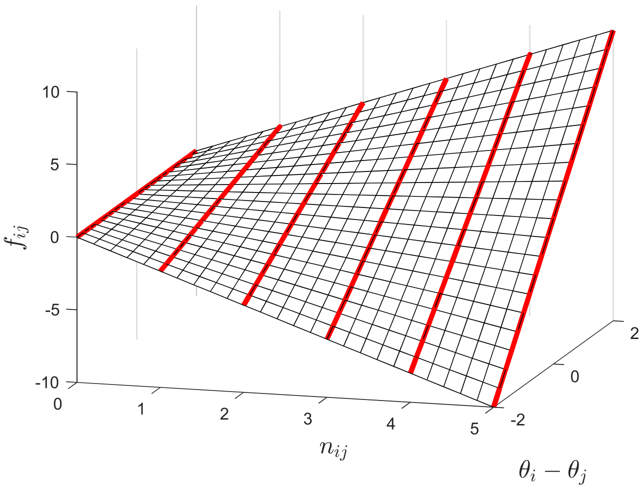


FIGURE 4. Illustration of constraint (22) for $X_{ij} = 1$.

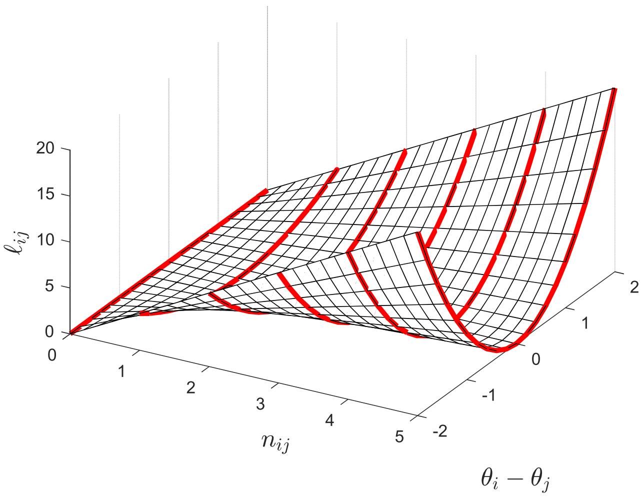


FIGURE 5. Illustration of constraint (23) for $\mathcal{G}_{ij} = 1$.

represent the values for f_{ij} and ℓ_{ij} when the integrality of n_{ij} is relaxed. It can be seen that both constraints are nonlinear and nonconvex, even when the integrality of n_{ij} is relaxed.

VI. TESTS AND RESULTS

The proposed BB algorithm to solve nonconvex MINLP problems was implemented in the modeling language AMPL [31]. The open-source solver IPOPT v3.12.13 and the commercial solvers KNITRO v12.4.0, LOQO v7.03, and MINOS v5.51 were used, with default settings, to solve the NLP subproblems of the BB tree. For the fathoming rules, it was considered that $\varepsilon = 5\%$, obtained after several trials. It should be noted that this value of ε was defined based on experimental tests and demonstrated to be conservative without increasing too much the size of the BB tree and the computational times to solve the problem. For other problems, it may be adequate to perform an analysis to define more adequate values of ε.

Tests were performed using the 6-bus Garver's system, the IEEE 24-bus system, the 46-bus Southern Brazilian system, and the 93-bus Colombian system with and without generation rescheduling. Note that losses can only be considered in the cases with generation rescheduling, since, for the case without generation rescheduling, the values of the generation at the generation buses are fixed. Complete data for all the systems are available in [32]. All tests were carried out on a computer with a 3.4 GHz AMD Ryzen™9 5950X processor with 32 GB of RAM.

It should be highlighted that the 93-bus Colombian system represents a large instance of the TNEP problem. Note that the complexity of a system for TNEP is not directly related to the number of buses. In fact, the complexity of the problem is much more related to the stress level of the network and the number of transmission lines that must be constructed for an adequate operation of the network.

For comparison purposes, the MINLP model (20)–(29) was also solved directly using off-the-shelf optimization solvers for nonconvex MINLP problems that can obtain local solutions, i.e., KNITRO v12.4.0 and BONMIN v1.8.8 [33], which is an open-source solver, using CBC v2.10.5 and IPOPT v3.12.13, and the global solvers BARON v21.1.13 [34], LINDO GLOBAL v13.0.4099.255 [35], COUENNE v0.5.8 [36], and SCIP v8.0.0 [37]. Among these global solvers, COUENNE and SCIP are open-source solvers. Moreover, GUROBI v9.5.0 [38], which is capable of solving nonconvex MIQCP problems, was used to solve the problem without considering losses. In all the tests, default settings were considered in all solvers, and the time limit was set to one hour.

Table 1 shows the dimensions of the problems. The number of integer variables is equal to the number of corridors in the system. The tests were carried out without considering losses in the system and with losses for the systems with generation rescheduling. Table 1 shows that the 6-bus Garver's system is the smallest problem considered, while the 93-bus Colombian system is the largest one. Moreover, when losses are considered, the problems become significantly more complex, as indicated by the number of nonzeros in the Jacobian and the Hessian when compared with the respective problems without losses.

Tables 2–5 present the main results, including the investment cost and computational time to find the solution in each test (displayed within brackets). An asterisk indicates that the solver terminated before the time limit of one hour, finding a locally (or globally) optimal solution. A boldface value indicates that the solver found the best-known solution of the problem. NC indicates that the solver was not able to find an integer solution for the problem in the maximum allowed time. Table 2 presents the results for the TNEP problem without considering losses, using local and global solvers for nonconvex MINLP problems to solve the problem directly. The case without losses is obtained by letting the resistance $R_{ij} = 0$ at all corridors. Table 3 presents the results obtained by the local and global solvers for nonconvex MINLP

TABLE 1. Dimensions of the problems.

Problem characteristics	Without losses								With losses			
	6-bus Garver's with rescheduling	6-bus Garver's	IEEE 24-bus with rescheduling	IEEE 24-bus	46-bus Southern Brazilian with rescheduling	46-bus Southern Brazilian	93-bus Colombian with rescheduling	93-bus Colombian	6-bus Garver's with rescheduling	IEEE 24-bus with rescheduling	46-bus Southern Brazilian with rescheduling	93-bus Colombian with rescheduling
Number of variables	38	38	115	115	214	214	451	451	53	156	294	606
Bounded below only	0	0	0	0	0	0	0	0	15	41	79	155
Bounded below and above	18	18	51	51	91	91	204	204	18	51	91	204
Free	20	20	64	64	123	123	247	247	20	64	124	247
Number of integer variables	15	15	41	41	79	79	155	155	15	41	79	155
Number of constraints	51	51	147	147	280	280	558	558	66	188	362	713
Linear equalities	6	6	24	24	45	45	93	93	6	24	46	93
Quadratic equalities	15	15	41	41	79	79	155	155	15	41	79	155
General nonlinear equalities	0	0	0	0	0	0	0	0	15	41	79	155
Linear one-sided inequalities	30	30	82	82	156	156	310	310	30	82	158	310
Number of nonzeros in Jacobian	148	148	414	414	790	790	1593	1593	263	736	1424	2827
Number of nonzeros in Hessian	25	25	76	76	153	153	304	304	40	134	272	545

problems for the systems with generation rescheduling considering losses. Table 4 presents the results obtained by the proposed BB algorithm for the problem without considering losses. Finally, Table 5 presents the results obtained by the proposed BB algorithm for the problem considering losses.

The following sections will discuss the obtained results for each system in more detail.

A. 6-BUS GARVER'S SYSTEM

The 6-bus Garver's system was originally presented in [39]. Up to three new lines are allowed to be constructed in each corridor.

The global optimal solution for this problem without considering losses and considering generation rescheduling has an investment cost of MUS$ 110.000 with $n_{3-5} = 1$ and $n_{4-6} = 3$. Note that, in Table 2, the solvers KNITRO and BONMIN were not capable of finding the global optimal solution to the problem, whereas the solvers for global optimization were able to find the global solution. The solutions found by KNITRO and BONMIN are 18.18% and 121.82% worse than the optimal solution, respectively. The solutions were found by each solver in less than 1 s. On the other hand, by analyzing Table 4, it is possible to verify that the proposed BB algorithm was able to find the global optimal solution in up to 1 s with every solver used to solve the NLP subproblems.

For the problem without considering losses and without considering generation rescheduling, the global optimal solution has an investment cost of MUS$ 231.000 with $n_{2-6} = 3$, $n_{3-5} = 1$, $n_{4-6} = 2$, and $n_{5-6} = 1$. Table 2 shows that the solvers BONMIN and LINDO GLOBAL were incapable of finding the global optimal solution of the problem, whereas the other solvers were able to find the global solution in less than 1 s. The solution found by BONMIN is 3.03% worse than the optimal solution, and LINDO GLOBAL found a solution that is, in fact, infeasible. Table 4 shows that the proposed BB algorithm was able to find the global optimal solution in approximately 1 s with every solver used to solve the NLP subproblems.

When considering the electrical losses, the best-known solution for the problem has an investment cost of MUS$ 130.000 with $n_{2-3} = 1$, $n_{3-5} = 1$, and $n_{4-6} = 3$. Table 3 shows that the solvers KNITRO and LINDO GLOBAL were incapable of finding the best-known solution of the problem, whereas the other solvers were able to find the best-known solution in no more than 4 s. The solution found by KNITRO and LINDO GLOBAL is 7.69% worse than the best-known solution. Table 5 shows that the proposed BB algorithm was able to find the best-known solution with every NLP solver in no more than 3 s, except for MINOS, whose solution is 30.77% worse than the best-known solution.

B. IEEE 24-BUS SYSTEM

The topology of the IEEE 24-bus system can be found in [40]. Up to five new lines are allowed to be constructed in each corridor.

The global optimal solution for this problem without considering losses and considering generation rescheduling has an investment cost of MUS$ 152.000 with $n_{6-10} = 1$, $n_{7-8} = 2$, $n_{10-12} = 1$, and $n_{14-16} = 1$. Table 2 shows that the solver LINDO GLOBAL found an infeasible solution to the problem, whereas the other solvers were able to find the global solution in no more than 5 s. The results shown in Table 4 indicate that the proposed BB algorithm was able to find the global optimal solution with every solver used to solve the NLP subproblems in no more than 3 s.

The global optimal solution for the problem without considering losses and without considering generation

TABLE 2. Investment costs for the systems obtained by solving the problem without losses using local/global solvers for MINLP (MUS$).

System	Best-known	Solver						
		KNITRO	BONMIN	BARON	LINDO GLOBAL	COUENNE	SCIP	GUROBI
6-bus Garver's with rescheduling	110.000	130.000 (<1 s)*	244.000 (1 s)*	**110.000** (<1 s)*	**110.000** (<1 s)*	**110.000** (<1 s)*	**110.000** (<1 s)*	**110.000** (<1 s)*
6-bus Garver's	231.000	**231.000** (<1 s)*	238.000 (<1 s)*	**231.000** (<1 s)*	Infeasible*	**231.000** (<1 s)*	**231.000** (<1 s)*	**231.000** (<1 s)*
IEEE 24-bus with rescheduling	152.000	**152.000** (<1 s)*	**152.000** (2 s)*	**152.000** (4 s)*	Infeasible*	**152.000** (5 s)*	**152.000** (5 s)*	**152.000** (<1 s)*
IEEE 24-bus	370.000	**370.000** (<1 s)*	**370.000** (3 s)*	**370.000** (218 s)	Infeasible*	**370.000** (51 s)*	383.000 (2 s)	**370.000** (<1 s)*
46-bus Southern Brazilian with rescheduling	72.870	**72.870** (8 s)*	**72.870** (115 s)*	**72.870** (62 s)*	Infeasible*	**72.870** (41 s)*	**72.870** (7 s)*	**72.870** (2 s)*
46-bus Southern Brazilian	154.420	157.304 (11 s)*	**154.420** (311 s)*	164.880 (1258 s)	250.143 (3321 s)	168.314 (40 s)	**154.420** (26 s)*	**154.420** (372 s)*
93-bus Colombian with rescheduling	189.650	**189.650** (1 s)*	**189.650** (96 s)*	**189.650** (31 s)	Infeasible	196.373 (11 s)*	NC	404.120 (1 s)
93-bus Colombian	562.417	**562.417** (31 s)*	**562.417** (179 s)*	735.180 (93 s)	4,512.213 (22 s)	NC	NC	651.576 (709 s)

TABLE 3. Investment costs for the systems obtained by solving the problem with losses using local/global solvers for MINLP (MUS$).

System	Best-known	Solver					
		KNITRO	BONMIN	BARON	LINDO GLOBAL	COUENNE	SCIP
6-bus Garver's with rescheduling	130.000	140.000 (<1 s)*	**130.000** (<1 s)*	**130.000** (4 s)*	140.000 (<1 s)*	**130.000** (1 s)*	**130.000** (<1 s)*
IEEE 24-bus with rescheduling	188.000	203.000 (<1 s)*	**188.000** (10 s)*	**188.000** (362 s)*	**188.000** (12 s)*	**188.000** (43 s)*	**188.000** (137 s)*
46-bus Southern Brazilian with rescheduling	75.895	102.821 (<1 s)*	94.004 (128 s)*	**75.895** (3589 s)*	129.055 (58 s)*	**75.895** (255 s)*	**75.895** (1053 s)*
93-bus Colombian with rescheduling	202.920	**202.920** (4 s)*	204.528 (47 s)*	781.747 (2850 s)	370.250 (121 s)	235.610 (45 s)	NC

TABLE 4. Investment costs for the systems obtained by solving the problem without losses using the proposed BB algorithm (MUS$).

System	Best-known	Solver			
		IPOPT	KNITRO	LOQO	MINOS
6-bus Garver's with rescheduling	110.000	**110.000** (<1 s)*	**110.000** (1 s)*	**110.000** (<1 s)*	**110.000** (1 s)*
6-bus Garver's	231.000	**231.000** (1 s)*	**231.000** (1 s)*	**231.000** (1 s)*	**231.000** (1 s)*
IEEE 24-bus with rescheduling	152.000	**152.000** (3 s)*	**152.000** (2 s)*	**152.000** (1 s)*	**152.000** (1 s)*
IEEE 24-bus	370.000	**370.000** (10 s)*	**370.000** (5 s)*	**370.000** (3 s)*	**370.000** (4 s)*
46-bus Southern Brazilian with rescheduling	72.870	**72.870** (43 s)*	**72.870** (86 s)*	**72.870** (20 s)*	105.317 (3 s)*
46-bus Southern Brazilian	154.420	**154.420** (459 s)*	**154.420** (241 s)*	**154.420** (163 s)*	185.774 (4 s)*
93-bus Colombian with rescheduling	189.650	**189.650** (20 s)*	**189.650** (5 s)*	**189.650** (4 s)*	**189.650** (9 s)*
93-bus Colombian	562.417	572.061 (1982 s)	572.061 (663 s)	**562.417** (3292 s)	579.411 (1545 s)

TABLE 5. Investment costs for the systems obtained by solving the problem with losses using the proposed BB algorithm (MUS$).

System	Best-known	Solver			
		IPOPT	KNITRO	LOQO	MINOS
6-bus Garver's with rescheduling	130.000	**130.000** (3 s)*	**130.000** (2 s)*	**130.000** (1 s)*	170.000 (<1 s)
IEEE 24-bus with rescheduling	188.000	**188.000** (36 s)*	**188.000** (18 s)*	**188.000** (4 s)*	Infeasible*
46-bus Southern Brazilian with rescheduling	75.895	**75.895** (24 s)*	**75.895** (22 s)*	**75.895** (2 s)*	Infeasible*
93-bus Colombian with rescheduling	202.920	**202.920** (78 s)*	**202.920** (102 s)*	**202.920** (10 s)*	Infeasible*

rescheduling has an investment cost of MUS$ 370.000 with $n_{3-24} = 1$, $n_{6-10} = 1$, $n_{7-8} = 1$, $n_{10-12} = 1$, $n_{14-16} = 1$, $n_{15-24} = 1$, $n_{16-17} = 2$, and $n_{17-18} = 2$. Table 2 shows that the solver LINDO GLOBAL, again, converged to an infeasible solution, SCIP found a solution that is 3.51% worse than the optimal solution, while the other solvers were able to find the global solution in no more than 218 s. Table 4 shows that the proposed BB algorithm was able to find the global optimal solution in less than 10 s with every solver used to solve the NLP subproblems.

When considering losses in the system, the best-known solution for the problem has an investment cost of MUS$ 188.000 with $n_{6-10} = 1$, $n_{7-8} = 2$, $n_{10-12} = 1$, $n_{14-16} = 1$, and $n_{16-17} = 1$. The results shown in Table 3 indicate that, except for KNITRO, the solvers were able to find the best-known solution to the problem in no more than 362 s. The solution found by KNITRO is 7.98% worse than the best-known solution. Table 5 shows that the proposed BB algorithm was able to find the best-known solution with every NLP solver in less than 36 s, except when using MINOS, for which the BB algorithm incorrectly indicated that the problem was infeasible.

C. 46-BUS SOUTHERN BRAZILIAN SYSTEM

The topology of the 46-bus Southern Brazilian system can be found in [11]. Up to five new lines are allowed to be constructed in each corridor.

The global optimal solution for this problem without considering losses and considering generation rescheduling has an investment cost of MUS$ 72.870 with $n_{2-5} = 1$, $n_{13-20} = 1, n_{20-23} = 1, n_{20-21} = 2, n_{42-43} = 1, n_{46-6} = 1$, and $n_{5-6} = 2$. Table 2 shows that only the solver LINDO GLOBAL was incapable of finding the global optimal solution to the problem, converging to an infeasible solution. The other solvers were capable of finding the optimal solution in less than 115 s. Table 4 shows that the proposed BB algorithm was able to find the global optimal solution with every solver used to solve the NLP subproblems in less than 86 s, except for MINOS, which found a solution that is 44.53% more expensive than the optimal solution.

For the problem without considering losses and without considering generation rescheduling, the global optimal solution has an investment cost of MUS$ 154.420 with $n_{20-21} = 1, n_{42-43} = 2, n_{46-6} = 1, n_{19-25} = 1, n_{31-32} = 1$, $n_{28-30} = 1$, $n_{26-29} = 3$, $n_{24-25} = 2$, $n_{29-30} = 2$, and $n_{5-6} = 2$. Table 2 shows that only the solver BONMIN was capable of finding the global optimal solution of the problem in 311 s, whereas the other solvers were unable to find the global solution. The solutions found by KNITRO, BARON, LINDO GLOBAL, and COUENNE are, respectively, 1.87%, 6.77%, 61.99%, and 9.00% worse than the optimal solution. Table 4 shows that the proposed BB algorithm was able to find the global optimal solution with every solver used to solve the NLP subproblems in less than 459 s, except MINOS, whose solution is 20.30% more expensive than the optimal solution.

When considering the electrical losses, the best-known solution for the problem has an investment cost of MUS$ 75.895 with $n_{18-20} = 1$, $n_{20-23} = 1$, $n_{20-21} = 2$, $n_{42-43} = 1$, $n_{46-6} = 1$, and $n_{5-6} = 2$. Table 3 shows that only the solvers BARON and COUENNE were capable of finding the best-known solution of the problem in less than 3589 s. The solutions found by KNITRO, BONMIN, and LINDO GLOBAL are, respectively, 35.48%, 23.86%, and 70.04% more expensive than the best-known solution. Table 5 shows that the proposed BB algorithm was able to find the best-known solution with every NLP solver in less than 24 s, except when using MINOS, for which the BB algorithm incorrectly indicated that the problem was infeasible.

D. 93-BUS COLOMBIAN SYSTEM

The initial topology for the 93-bus Colombian system is presented in Fig. 6; the continuous lines indicate existing transmission lines, and the dotted lines indicate corridors where new lines can be constructed. Up to four new lines are allowed to be constructed in each corridor.

The global optimal solution for this problem without considering losses and considering generation rescheduling has an investment cost of MUS$ 189.650 with $n_{27-29} = 1$, $n_{62-73} = 1$, $n_{19-82} = 1$, $n_{82-85} = 1$, and $n_{68-86} = 1$. Table 2 shows that the solvers KNITRO, BONMIN, and BARON were able to find the global optimal solution in no more than 96 s, whereas LINDO GLOBAL converged to an infeasible solution, SCIP was not capable of finding a feasible solution, and COUENNE and GUROBI, respectively, found solutions that are 3.54% and 113.18% more expensive than the optimal solution. The results shown in Table 4 indicate that the proposed BB algorithm was able to find the global optimal solution with every solver used to solve the NLP subproblems in no more than 20 s.

For the problem without considering losses and without considering generation rescheduling, the global optimal solution has an investment cost of MUS$ 562.417 with $n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-65} = 1$, $n_{30-72} = 1$, $n_{55-57} = 1$, $n_{55-84} = 1$, $n_{56-57} = 1$, $n_{55-62} = 1$, $n_{27-64} = 1$, $n_{27-29} = 1$, $n_{50-54} = 1$, $n_{62-73} = 1$, $n_{54-56} = 1$, $n_{72-73} = 1$, $n_{19-82} = 2$, $n_{82-85} = 1$, and $n_{68-86} = 1$. Table 2 shows that only the solvers KNITRO and BONMIN were capable of finding the global optimal solution to the problem in less than 179 s. The solutions found by BARON, LINDO GLOBAL, and GUROBI are, respectively, 30.72%, 702.29%, and 15.85% more expensive than the optimal solution, while COUENNE and SCIP were unable to find an integer solution for the problem within the time limit. By analyzing Table 4, it is possible to verify that the proposed BB algorithm was only able to find the global optimal solution to the problem using the solver LOQO to solve the NLP subproblems in 3292 s. The solution obtained by the proposed BB algorithm with the solvers IPOPT and KNITRO is only 1.71% more expensive than the optimal solution, while the solution obtained with the solver MINOS is 3.02% more expensive than the optimal solution.

When considering losses in the system, the best-known solution for the problem has an investment cost of MUS$ 202.920 with $n_{27-29} = 1$, $n_{62-73} = 1$, $n_{19-82} = 2$, $n_{82-85} = 1$, and $n_{68-86} = 1$. The results shown in Table 3 indicate that only KNITRO was able to find the best-known solution to the problem in 4 s. The solutions found by BONMIN, BARON, LINDO GLOBAL, and COUENNE are, respectively, 0.79%, 285.25%, 82.46%, and 16.11% worse than the best-known solution. SCIP was unable to find an integer solution for the problem within the time limit. Table 5 shows that the proposed BB algorithm was able to find the best-known solution with every NLP solver in less than 102 s, except when using MINOS, for which the BB algorithm incorrectly indicated that the problem was infeasible.

E. DISCUSSION

Based on the results shown in Table 2 and Table 3, it can be verified that no solver for MINLP problems was able to obtain the optimal solution for every instance of the problem considered. The solvers KNITRO and BONMIN

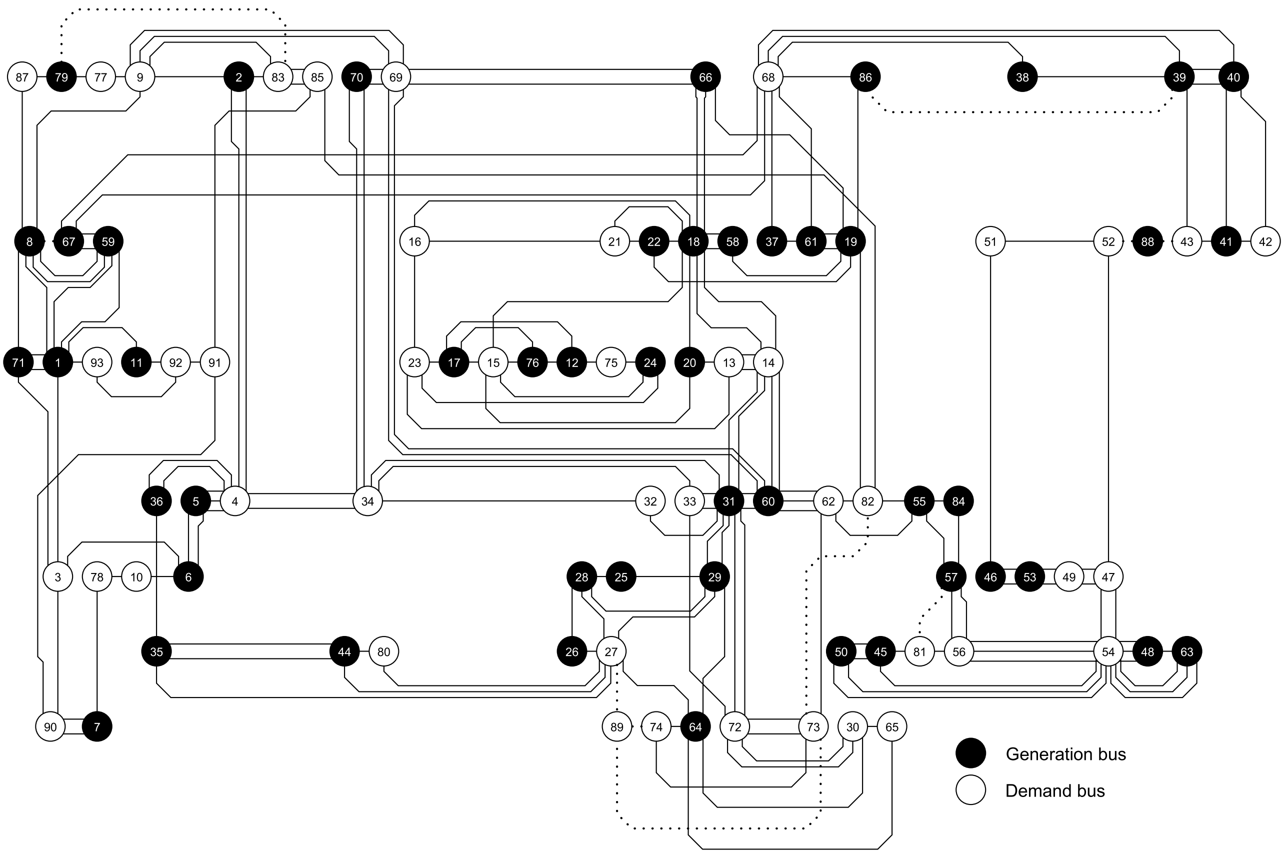


FIGURE 6. Initial topology of the 93-bus Colombian system.

were able to find the best-known solution in 7 and 8 out of 12 instances, respectively. The global solvers BARON, LINDO GLOBAL, COUENNE, and SCIP found the best-known solutions for 9, 2, 8, and 8 out of 12 instances, respectively. Moreover, GUROBI found the best-known solutions for 6 out of 8 instances of the problem without considering losses. Moreover, note that GUROBI cannot solve the TNEP problem with losses, since it is not a MIQCP problem. It should be noted that the global solvers are more efficient at solving smaller instances of the problem.

The proposed BB algorithm for nonconvex MINLP problems was capable of finding the best-known solutions of all instances considered when the solver LOQO was used to solve the NLP subproblems. Moreover, with the solvers IPOPT and KNITRO, it was incapable of finding the best-known solution in only one instance. With the solver MINOS, the proposed BB was only capable of finding the best-known solution for five instances. Note that, for the 93-bus Colombian system without considering losses and generation rescheduling, the BB algorithm with the solvers IPOPT, KNITRO, and MINOS was terminated due to the maximum computational time allowed of 3600 s. Therefore, it may be possible to achieve the best-known solutions by tuning the parameters of the algorithm, such as the maximum computational time.

An alternative to improve the performance of the proposed BB algorithm to solve large instances is to use heuristics to try to find initial high-quality solutions to the problem, as done by most of the solvers, to improve the solutions at the nodes of the BB tree and to define the next separation variable and the next subproblem to be solved.

VII. CONCLUSION

This work presented and discussed the difficulties and ineffectiveness of the direct use of the traditional branch and bound (BB) algorithm for solving nonconvex mixed-integer nonlinear programming (MINLP) problems and proposed a modified version of the BB for solving this category of problems.

The proposed implementation of the BB algorithm included strategies to deal with nonlinear programming (NLP) subproblems, to detect the infeasibility of an NLP subproblem, and to treat the nonconvexity of the problem, as well as adequate fathoming rules.

The BB algorithm was tested to solve the transmission network expansion planning (TNEP) problem, which is a difficult nonconvex MINLP problem. Four systems were considered in the tests, with and without losses and generation rescheduling. Four optimization solvers were used to solve the NLP subproblems. Moreover, for comparison purposes, the problems were also directly solved using two local and five global optimization solvers for nonconvex MINLP problems.

The results indicate that the proposed BB algorithm presents competitive results for the TNEP problem when compared with commercial and open-source solvers for nonconvex MINLP problems, especially when the solver LOQO is used to solve the NLP subproblems since it was possible to obtain the best-known solutions for all instances.

Future works will develop improvements for the basic BB algorithm presented in this work, including heuristic approaches for obtaining high-quality feasible solutions to the problem and determining the next subproblem to be solved and the division variable. Moreover, the TNEP problem considering the full ac model will be solved.

REFERENCES

[1] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, Jul. 1960.

[2] M. R. Kılınç and N. V. Sahinidis, "Exploiting integrality in the global optimization of mixed-integer nonlinear programming problems with BARON," *Optim. Methods Softw.*, vol. 33, no. 3, pp. 540–562, May 2018.

[3] M. Mahdavi, C. Sabillon Antunez, M. Ajalli, and R. Romero, "Transmission expansion planning: Literature review and classification," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3129–3140, Sep. 2019.

[4] R. Hemmati, R. Hooshmand, and A. Khodabakhshian, "Comprehensive review of generation and transmission expansion planning," *IET Gener., Transmiss. Distrib.*, vol. 7, no. 9, pp. 955–964, Sep. 2013.

[5] N. G. Ude, H. Yskandar, and R. C. Graham, "A comprehensive state-of-the-art survey on the transmission network expansion planning optimization algorithms," *IEEE Access*, vol. 7, pp. 123158–123181, 2019.

[6] N. Alguacil, A. L. Motto, and A. J. Conejo, "Transmission expansion planning: A mixed-integer LP approach," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1070–1077, Aug. 2003.

[7] L. Bahiense, G. C. Oliveira, M. Pereira, and S. Granville, "A mixed integer disjunctive model for transmission network expansion," *IEEE Trans. Power Syst.*, vol. 16, no. 3, pp. 560–565, Aug. 2001.

[8] S. Binato, M. V. F. Pereira, and S. Granville, "A new benders decomposition approach to solve power transmission network design problems," *IEEE Trans. Power Syst.*, vol. 16, no. 2, pp. 235–240, May 2001.

[9] M. Majidi-Qadikolai and R. Baldick, "A generalized decomposition framework for large-scale transmission expansion planning," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1635–1649, Mar. 2018.

[10] S. Hong, H. Cheng, and P. Zeng, "N-K constrained composite generation and transmission expansion planning with interval load," *IEEE Access*, vol. 5, pp. 2779–2789, 2017.

[11] S. Haffner, A. Monticelli, A. Garcia, J. Mantovani, and R. Romero, "Branch and bound algorithm for transmission system expansion planning using a transportation model," *IEE Proc. Gener., Transmiss. Distrib.*, vol. 147, no. 3, pp. 149–156, May 2000.

[12] M. Rider, A. Garcia, and R. Romero, "Transmission system expansion planning by a branch-and-bound algorithm," *IET Gener. Transmiss. Distrib.*, vol. 2, no. 1, pp. 90–99, Jan. 2008.

[13] M. A. J. Delgado, M. Pourakbari-Kasmaei, and M. J. Rider, "A modified branch and bound algorithm to solve the transmission expansion planning problem," in *Proc. 13th Int. Conf. Environ. Electr. Eng. (EEEIC)*, Nov. 2013, pp. 234–238.

[14] F. W. Glover and G. A. Kochenberger, *Handbook of Metaheuristics*. Boston, MA, USA: Kluwer, 2003.

[15] P. Murugan, "Modified particle swarm optimisation with a novel initialisation for finding optimal solution to the transmission expansion planning problem," *IET Gener., Transmiss. Distrib.*, vol. 6, no. 11, pp. 1132–1142, Nov. 2012.

[16] M. C. da Rocha and J. T. Saraiva, "A discrete evolutionary PSO based approach to the multiyear transmission expansion planning problem considering demand uncertainties," *Int. J. Electr. Power Energy Syst.*, vol. 45, no. 1, pp. 427–442, Feb. 2013.

[17] A. Verma, B. K. Panigrahi, and P. R. Bijwe, "Harmony search algorithm for transmission network expansion planning," *IET Gener., Transmiss. Distrib.*, vol. 4, no. 6, pp. 663–673, Jun. 2010.

[18] A. Rastgou and J. Moshtagh, "Improved harmony search algorithm for transmission expansion planning with adequacy–security considerations in the deregulated power system," *Int. J. Electr. Power Energy Syst.*, vol. 60, pp. 153–164, Sep. 2014.

[19] M. Mahdavi, A. Kimiyaghalam, H. H. Alhelou, M. S. Javadi, A. Ashouri, and J. P. S. Catalao, "Transmission expansion planning considering power losses, expansion of substations and uncertainty in fuel price using discrete artificial bee colony algorithm," *IEEE Access*, vol. 9, pp. 135983–135995, 2021.

[20] S. M. L. Silva, L. T. Faria, R. Romero, and J. F. Franco, "Efficient multistart with path relinking search strategy for transmission system expansion planning," *IEEE Access*, vol. 9, pp. 153213–153225, 2021.

[21] N. Gupta, M. Khosravy, N. Patel, and T. Senjyu, "A bi-level evolutionary optimization for coordinated transmission expansion planning," *IEEE Access*, vol. 6, pp. 48455–48477, 2018.

[22] M. J. Rider, A. V. Garcia, and R. Romero, "Power system transmission network expansion planning using AC model," *IET Gener., Transmiss. Distrib.*, vol. 1, no. 5, pp. 731–742, Sep. 2007.

[23] P. V. Gomes and J. T. Saraiva, "A two-stage strategy for security-constrained ac dynamic transmission expansion planning," *Electr. Power Syst. Res.*, vol. 180, pp. 1–10, Mar. 2020.

[24] Z. Luburic, H. Pandzic, and M. Carrion, "Transmission expansion planning model considering battery energy storage, TCSC and lines using AC OPF," *IEEE Access*, vol. 8, pp. 203429–203439, 2020.

[25] R. H. Byrd, J. Nocedal, and R. A. Waltz, "Knitro: An integrated package for nonlinear optimization," in *Large-Scale Nonlinear Optimization*, G. Di Pillo and M. Roma, Eds. Boston, MA, USA: Springer, 2006, pp. 35–59.

[26] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, Mar. 2006.

[27] R. J. Vanderbei, "LOQO: An interior point code for quadratic programming," *Optim. Methods Softw.*, vol. 11, nos. 1–4, pp. 451–484, Jan. 1999.

[28] B. A. Murtagh and M. A. Saunders, "MINOS 5.51 user's guide," Stanford Univ., Stanford, CA, USA, Tech. Rep. SOL 83-20R, 2003.

[29] J.-M. Gauthier and G. Ribière, "Experiments in mixed-integer linear programming using pseudo-costs," *Math. Program.*, vol. 12, no. 1, pp. 26–47, Dec. 1977.

[30] J. T. Linderoth and M. W. Savelsbergh, "A computational study of search strategies for mixed integer programming," *Informs J. Comput.*, vol. 11, no. 2, pp. 173–187, May 1999.

[31] R. Fourer, D. M. Gay, and B. W. Kernighan, *AMPL: A Modeling Language for Mathematical Programming*, 2nd ed. Duxbury, MA, USA: Thomson, 2003.

[32] (2021). *LaPSEE Power System Test Cases Repository*. Accessed: Oct. 1, 2021. [Online]. Available: http://www.feis.unesp.br/#!/lapsee

[33] P. Bonami, L. T. Biegler, A. R. Conn, G. Cornuéjols, I. E. Grossmann, C. D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter, "An algorithmic framework for convex mixed integer nonlinear programs," *Discrete Optim.*, vol. 5, no. 2, pp. 186–204, May 2008.

[34] N. Sahinidis, "BARON user manual," Optim. Firm, Atlanta, GA, USA, Jan. 2021.

[35] *LINDO SYSTEMS INC*. Accessed: Dec. 26, 2021. [Online]. Available: https://www.lindo.com/

[36] P. Belotti, "Couenne: A user's manual," Clemson Univ., Clemson, SC, USA, Tech. Rep., 2009.

[37] S. Vigerske and A. Gleixner, "SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework," *Optim. Methods Softw.*, vol. 33, no. 3, pp. 563–593, May 2018.

[38] (2022). Gurobi Optimization. *Gurobi Optimizer Reference Manual*. Accessed: Mar. 20, 2022. [Online]. Available: http://www.gurobi.com

[39] L. Garver, "Transmission network estimation using linear programming," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 7, pp. 1688–1697, Sep. 1970.

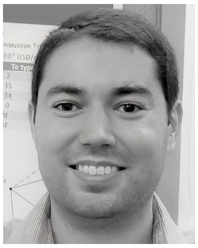
[40] R. Fang and D. J. Hill, "A new strategy for transmission expansion in competitive electricity markets," *IEEE Trans. Power Syst.*, vol. 18, no. 1, pp. 374–380, Feb. 2003.

REINALDO T. ZOPPEI received the Licentiate degree in sciences (physics) from the Federal University of Mato Grosso, Mato Grosso, Brazil, in 1986, and the M.S. degree in applied physics from the University of São Paulo, São Paulo, Brazil, in 1999. He is currently pursuing the Ph.D. degree in electrical engineering with São Paulo State University, Ilha Solteira, Brazil.

He is also a Professor with the Federal University of Rondonópolis, Mato Grosso. His research interest includes the development of optimization methods for power systems.

MARCOS A. J. DELGADO received the Licentiate and M.S. degrees in mathematics from São Paulo State University, São José do Rio Preto, Brazil, in 2001 and 2004, respectively, and the Ph.D. degree in electrical engineering from São Paulo State University, Ilha Solteira, Brazil, in 2015.

He is currently a Professor with the Federal University of Rondonópolis, Mato Grosso, Brazil. His research interests include the development of optimization models and methods for power systems.



LEONARDO H. MACEDO (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from São Paulo State University, Ilha Solteira, Brazil, in 2012, 2015, and 2019, respectively.

From 2016 to 2017, he was a Visiting Student with the University of Washington, Seattle, WA, USA, and from 2019 to 2020, he was a Postdoctoral Researcher with the Universidad de Castilla-La Mancha, Ciudad Real, Spain. He is currently a Postdoctoral Researcher with São Paulo State University. His current research interests include the development of methods for the optimization, planning, and control of electrical power systems.

MARCOS J. RIDER (Senior Member, IEEE) received the B.Sc. (Hons.) and P.E. degrees in electrical engineering from the National University of Engineering, Lima, Peru, in 1999 and 2000, respectively, the M.Sc. degree in electrical engineering from the Federal University of Maranhão, São Luís, Brazil, in 2002, and the Ph.D. degree in electrical engineering from the University of Campinas (UNICAMP), Campinas, Brazil, in 2006.

From 2010 to 2015, he was a Professor with São Paulo State University, Ilha Solteira, Brazil. He is currently a Professor with the Department of Systems and Energy, UNICAMP. His research interests include the development of methodologies for the optimization, planning and control of electrical power systems, and applications of artificial intelligence in power systems.

RUBÉN ROMERO (Senior Member, IEEE) received the B.S. and P.E. degrees in electrical engineering from the National University of Engineering, Lima, Peru, in 1978 and 1984, respectively, and the M.S. and Ph.D. degrees in electrical engineering from the University of Campinas, Campinas, Brazil, in 1990 and 1993, respectively.

He is currently a Full Professor in electrical engineering with São Paulo State University, Ilha Solteira, Brazil. His research interests include methods for the optimization, planning and control of electrical power systems, applications of artificial intelligence in power systems, and operations research.