

REINALDO TAKARA ZOPPEI

**ALGORITMO *BRANCH AND BOUND* PARA O PLANEJAMENTO DA
EXPANSÃO DE REDES DE TRANSMISSÃO USANDO MODELOS DE
PROGRAMAÇÃO NÃO LINEAR INTEIRA MISTA NÃO CONVEXOS**

Ilha Solteira

2023

REINALDO TAKARA ZOPPEI

**ALGORITMO *BRANCH AND BOUND* PARA O PLANEJAMENTO DA
EXPANSÃO DE REDES DE TRANSMISSÃO USANDO MODELOS DE
PROGRAMAÇÃO NÃO LINEAR INTEIRA MISTA NÃO CONVEXOS**

Tese apresentada à Faculdade de Engenharia da Universidade Estadual Paulista – UNESP, Câmpus de Ilha Solteira, como parte dos requisitos para obtenção do título de Doutor em Engenharia Elétrica.

Área de Conhecimento: Automação.

**Prof. Dr. Leonardo Henrique Faria
Macedo Possagnolo**
Orientador

Ilha Solteira

2023

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

Z88a Zoppei, Reinaldo Takara.
Algoritmo branch and bound para o planejamento da expansão de redes de transmissão usando modelos de programação não linear inteira mista não convexos / Reinaldo Takara Zoppei. -- Ilha Solteira: [s.n.], 2023
103 f. : il.

Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2023

Orientador: Leonardo Henrique Faria Macedo Possagnolo

Inclui bibliografia

1. Algoritmo branch and bound. 2. Otimização. 3. Planejamento da expansão de redes de transmissão. 4. Problemas de otimização não convexos. 5. Programação não linear inteira mista.

Impacto potencial desta pesquisa

A interpretação dos resultados do problema do planejamento da expansão da rede de transmissão define estratégias de ações futuras com grande importância do ponto de vista econômico e computacional. Para resolver esse problema, propomos uma técnica de solução para programação não linear inteira mista não convexa que apresenta melhorias, principalmente com questões de infactibilidade e convergência.

Potential impact of this research

The interpretation of the results of the transmission network expansion planning problem defines strategies for future actions with great importance from the economic and computational point of view. To solve this problem, we propose a solution technique for nonconvex mixed-integer nonlinear programming problems that presents improvements, mainly with issues of infeasibility and convergence.

CERTIFICADO DE APROVAÇÃO

TÍTULO DA TESE: ALGORITMO BRANCH AND BOUND PARA O PLANEJAMENTO DA EXPANSÃO DE REDES DE TRANSMISSÃO USANDO MODELOS DE PROGRAMAÇÃO NÃO LINEAR INTEIRA MISTA NÃO CONVEXOS

AUTOR: REINALDO TAKARA ZOPPEI

ORIENTADOR: LEONARDO HENRIQUE FARIA MACEDO POSSAGNOLO

Aprovado como parte das exigências para obtenção do Título de Doutor em Engenharia Elétrica, área:
Automação pela Comissão Examinadora:

Leonardo H. F. M. Possagno

Prof. Dr. LEONARDO HENRIQUE FARIA MACEDO POSSAGNOLO (Participação Presencial)
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira - UNESP

Jonatas Boas Leite

Prof. Dr. JONATAS BOAS LEITE (Participação Presencial)
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira - UNESP

Renzo

Dr. RENZO AMILCAR VARGAS PERALTA (Participação Presencial)
Departamento de Engenharia de Elétrica / Universidade Federal do ABC - UFABC

Lucas Teles de Faria

Prof. Dr. LUCAS TELES DE FARIA (Participação Presencial)
Departamento de Engenharia Elétrica / Faculdade de Engenharia e Ciências - UNESP

Matheus H. Marcolino

Prof. Dr. MATHEUS HENRIQUE MARCOLINO (Participação Virtual)
Instituto de Ciências Tecnológicas / Universidade Federal de Itajubá - UNIFEI

Ilha Solteira, 02 de junho de 2023

*Ao meu filho **Rafael**, que ele conquiste o sucesso em seus objetivos de vida e seja feliz com a sua esposa Ana Paula e filhas Ana Rafaela e Ana Gabriela.*

AGRADECIMENTOS

Ao Professor Dr. Leonardo Henrique Faria Macedo Possagnolo pela valiosa orientação, amizade, competência e dedicação.

Aos professores, funcionários e alunos do Programa de Pós-graduação em Engenharia Elétrica da FEIS/UNESP, em especial: Prof. Rubén Augusto Romero Lázaro, Prof. José Paulo Fernandes Garcia, Prof. Dionizio Paschoareli Júnior, Prof. Jean Marcos de Souza Ribeiro, Uiliam Nelson Lendzion Tomaz Alves, Matheus Henrique Marcolino, Julio Cezar Uzinski, Caio Enside e Gustavo Luizon.

À Universidade Estadual Paulista – UNESP, Câmpus de Ilha Solteira, pelo apoio através do “Programa de Pós-graduação em Engenharia Elétrica”.

Ao Professor Dr. Marcos André de Jesus Delgado do Departamento de Matemática da Universidade Federal de Rondonópolis pelo valioso apoio, amizade, incentivo, discussões e troca de conhecimentos.

Aos professores, funcionários e alunos da Universidade Federal de Rondonópolis, em especial: Prof.^a Sunamita Souza Silva, Prof. Eralci Moreira Therézio, Prof. Clayton Eduardo Lente Silva, Prof. João Eduardo Frederico, Prof. Rosevaldo de Oliveira, Prof. Dércio Braga Santos e Prof. Álvaro Moreira Neto.

À Universidade Federal de Rondonópolis – Mato Grosso, pelo apoio através do “Programa de Capacitação Docente”.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

RESUMO

O algoritmo *branch and bound* (BB) é amplamente utilizado para obter a solução global de problemas de programação linear inteira mista (PLIM). Por outro lado, quando esse algoritmo tradicional é usado diretamente para resolver problemas não convexos de programação não linear inteira mista (PNLIM), ele se torna ineficaz, principalmente devido à não linearidade e não convexidade da região factível do problema. Este trabalho apresenta as dificuldades e ineficácia do uso direto do algoritmo BB tradicional para a resolução de problemas PNLIM não convexos e propõe a formulação de um algoritmo BB eficiente para resolver esta categoria de problemas. O algoritmo BB proposto é formulado levando em consideração aspectos particulares de problemas de PNLIM não convexos, incluindo (i) como lidar com os subproblemas de programação não linear (PNL), (ii) como detectar a infactibilidade de um subproblema de PNL, (iii) como tratar a não convexidade do problema, e (iv) como definir as regras de sondagem. O algoritmo BB proposto é usado para resolver o problema de planejamento da expansão de redes de transmissão (PERT), um problema clássico de otimização de sistemas de potência, e seu desempenho é comparado com o desempenho de solucionadores comerciais de otimização para problemas de PNLIM. Os resultados obtidos para sistemas-teste com diferentes graus de complexidade indicam que o algoritmo BB proposto é eficaz para resolver o problema de PERT considerando o modelo CC com e sem considerar perdas e, também, considerando o modelo CA, apresentando desempenho igual ou melhor que os solucionadores comerciais de otimização.

Palavras-chave: Algoritmo *branch and bound*; Otimização; Planejamento da expansão de redes de transmissão; Problemas de otimização não convexos; Programação não linear inteira mista.

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 this traditional algorithm 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 proposed BB 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 test systems with different degrees of complexity indicate that the proposed BB algorithm is effective for solving the DC TNEP problem with and without considering losses and, also, considering the AC model, showing equal or better performance than off-the-shelf optimization solvers.

Keywords: Branch and bound algorithm; Mixed-integer nonlinear programming; Nonconvex optimization problems; Optimization. Transmission network expansion planning.

LISTA DE FIGURAS

FIGURA 1 - Ilustração da árvore BB após a divisão de P0.	25
FIGURA 2 - Ilustração da árvore BB após as divisões de P1 e P2.	26
FIGURA 3 - Região convexa de busca da solução e possíveis soluções de P.....	29
FIGURA 4 - Ponto A representando a solução de P0.	30
FIGURA 5 - Sub-regiões geradas com a variável de divisão x_1	31
FIGURA 6 - Sub-regiões geradas com a variável de divisão x_2	31
FIGURA 7 - Solução de P2 no ponto B.....	32
FIGURA 8 - Sub-regiões geradas com a criação dos subproblemas P3 e P4.....	32
FIGURA 9 - Soluções de P3 e P4 nos pontos C e D.	33
FIGURA 10 - Árvore branch and bound.	34
FIGURA 11 - Representação do caminho traçado pelo método BB.	34
FIGURA 12 - Representação gráfica da região não convexa de busca da solução de P.	36
FIGURA 13 - Representação gráfica da solução ótima local encontrada para P0.	37
FIGURA 14 - Representação da árvore BB após a primeira divisão.	37
FIGURA 15 - Representação da árvore BB para o 1º Caso.	38
FIGURA 16 - Representação da região de busca após a primeira divisão.	38
FIGURA 17 - Representação da região de busca após a segunda divisão.	38
FIGURA 18 - Representação da árvore BB para o 2º Caso.	39
FIGURA 19 - Representação da região de busca após a primeira divisão.	40
FIGURA 20 - Representação da região de busca após a segunda divisão.	40
FIGURA 21 - Representação da árvore BB após a segunda divisão.	41
FIGURA 22 - Representação da região de busca após a primeira divisão.	41
FIGURA 23 - Representação da região de busca após a segunda divisão.	41
FIGURA 24 - Representação da árvore BB após a segunda divisão.	42
FIGURA 25 - Representação da região de busca após a primeira divisão.	43
FIGURA 26 - Representação da região de busca após a segunda divisão.	43
FIGURA 27 - Representação da localização das soluções encontradas para cada um dos quatro casos.....	44
FIGURA 28 - Representação de uma região de busca não convexa.....	48
FIGURA 29 - Ilustração da restrição do balanço de potência.....	73
FIGURA 30 - Ilustração da restrição (50) com $X_{ij} = 1$	75
FIGURA 31 - Ilustração da restrição (51) com $G_{ij} = 1$	75
FIGURA 32 - Topologia inicial do sistema Garver de 6 barras.	83
FIGURA 33 - Topologia inicial do sistema IEEE de 24 barras.	85
FIGURA 34 - Topologia inicial do sistema Sul Brasileiro de 46 barras.....	87
FIGURA 35 - Topologia inicial do sistema Colombiano de 93 barras.	89

LISTA DE TABELAS

Tabela 1 - Quatro casos possíveis de resolução do problema de PNLIP com o uso do algoritmo BB definido para resolver problemas de PLIM	44
Tabela 2 - Dimensões dos problemas considerando o modelo CC	80
Tabela 3 - Resultados dos custos de investimentos dos sistemas sem perdas para o modelo CC usando solucionadores locais/globais (<i>MUS\$</i>)	81
Tabela 4 - Resultados dos custos de investimentos dos sistemas com perdas para o modelo CC usando solucionadores locais/globais (<i>MUS\$</i>)	82
Tabela 5 - Resultados dos custos de investimentos dos sistemas sem perdas para o modelo CC usando o algoritmo BB proposto (<i>MUS\$</i>)	82
Tabela 6 - Resultados dos custos de investimentos dos sistemas com perdas para o modelo CC usando o algoritmo BB proposto (<i>MUS\$</i>)	82
Tabela 7 - Dimensões dos problemas considerando o modelo CA	92
Tabela 8 - Resultados dos custos de investimentos dos sistemas para o modelo CA usando solucionadores locais/globais (<i>MUS\$</i>)	92
Tabela 9 - Resultados dos custos de investimentos dos sistemas para o modelo CA usando o algoritmo BB proposto (<i>MUS\$</i>)	92

LISTA DE ABREVIATURAS E SIGLAS

AMPL	<i>A mathematical programming language</i>
BB	<i>Branch and bound</i>
CA	Corrente alternada
CC	Corrente contínua
FIFO	<i>First in first out</i> – regra de seleção em largura de subproblemas
IEEE	Instituto de engenheiros elétricos e eletrônicos
LIFO	<i>Last in first out</i> – regra de seleção em profundidade de subproblemas
MUS\$	Milhões de dólares
PERT	Planejamento da expansão de redes de transmissão
PL	Programação linear
PLIP	Programação linear inteira pura
PLIM	Programação linear inteira mista
PNL	Programação não linear
PNLIP	Programação não linear inteira pura
PNLIM	Programação não linear inteira mista

LISTA DE SÍMBOLOS

Funções:

v Função objetivo do problema

Índices:

i, j Índices para as barras

ij, ji Índices para os caminhos de transmissão

ref Índice para a barra de referência

Conjuntos:

Ω_b Conjunto de barras do sistema

Ω_c Conjunto de todos os caminhos ij de transmissão formados entre duas barras vizinhas

Ω_c^0 Conjunto de caminhos ij com circuitos existentes na configuração base

Ω_c' Conjunto de caminhos ij sem circuitos existentes na configuração base

Parâmetros:

B_{ij} Susceptância no caminho ij

B_{ij}^{sh} Susceptância shunt no caminho ij

B_i^{sh} Susceptância shunt na barra i

C_{ij} Custo de construção de um novo circuito no caminho ij

D_i Demanda de potência ativa na barra de carga i

$\bar{F}_{ij}$ Fluxo máximo de potência ativa para cada circuito do caminho ij

G_{ij} Condutância de um circuito no caminho ij

$\underline{G}_i$ Mínima geração de potência ativa na barra i

$\bar{G}_i$ Máxima geração de potência ativa na barra i

$\bar{N}_{ij}$ Número máximo de novos circuitos que podem ser construídos no caminho ij

$\bar{N}_{ij}^1$ Número máximo de novos circuitos que podem ser construídos no caminho ij da configuração base

$\bar{N}_{ij}^2$ Número máximo de novos circuitos que podem ser construídos em um novo caminho ij que não pertence à configuração base

N_{ij}^0 Número de circuitos da configuração base existentes no caminho ij

Q_i Nível de demanda de potência reativa na barra i

$\underline{Q}_i$	Limite mínimo de geração de potência reativa na barra i
$\overline{Q}_i$	Limite máximo de geração de potência reativa na barra i
R_{ij}	Resistência de um circuito no caminho ij
$\underline{V}_i$	Limite mínimo da magnitude de tensão na barra i
$\overline{V}_i$	Limite máximo da magnitude de tensão na barra i
X_{ij}	Reatância de um circuito no caminho ij
θ_{ref}	Referência angular para o conjunto de barras
Λ	Parâmetro de penalização de custo devido ao corte de carga
ε	Fator de segurança

Variáveis contínuas:

f_{ij}	No modelo de transportes e no modelo CC, f_{ij} é o fluxo total de potência ativa de todos os circuitos (existentes e adicionados) no caminho ij pertencente ou não à configuração base e no modelo híbrido linear, f_{ij} é o fluxo total de potência ativa dos novos circuitos adicionados no caminho ij pertencente ou não à configuração base
f_{ij}^0	Fluxo total de potência ativa dos circuitos da configuração base existentes no caminho ij pertencente à configuração base
f_{ij}^1	No modelo híbrido não linear, é o fluxo total de potência ativa no caminho ij da configuração base
f_{ij}^2	No modelo híbrido não linear, é o fluxo total de potência ativa no caminho ij que não pertence à configuração base
f_{ij}^{de}	Fluxo total de potência ativa saindo da barra i
f_{ij}^{para}	Fluxo total de potência ativa entrando na barra i
g_i	Geração de potência ativa na barra de geração i
ℓ_{ij}	Perda total de potência ativa no caminho ij
q_i	Geração artificial de potência ativa na barra de carga i
q_i	Nível de geração de potência reativa na barra i
q_{ij}^{de}	Fluxo total de potência reativa saindo da barra i
q_{ij}^{para}	Fluxo total de potência reativa entrando na barra i
v_i	Magnitude de tensão na barra i
θ_i	Ângulo de fase da tensão na barra i

θ_{ij} Diferença do ângulo de fase da tensão entre as barras i e j

Variáveis inteiras:

n_{ij} Número de novos circuitos adicionados no caminho ij que pode ou não existir na configuração base

n_{ij}^1 Número de novos circuitos adicionados no caminho ij existente na configuração base

n_{ij}^2 Número de novos circuitos adicionados em um novo caminho ij não existente na configuração base

SUMÁRIO

1	INTRODUÇÃO	17
1.1	CONTRIBUIÇÕES.....	18
1.2	ESTRUTURA DO TRABALHO.....	18
2	ALGORITMO BRANCH AND BOUND	20
2.1	ALGORITMO BB TRADICIONAL PARA RESOLVER PROBLEMAS DE PLIM	23
2.1.1	Estrutura do algoritmo BB para resolver problemas de PLIM.....	27
2.1.2	Estudo de caso: Análise de um problema de PLIP.....	28
2.1.3	Estudo de caso: Análise de um problema de PNLIP não convexo	35
2.1.3.1	<i>Ineficácia do algoritmo BB em problemas de PNLIM não convexo</i>	<i>44</i>
2.2	ALGORITMO BB PROPOSTO PARA RESOLVER PROBLEMAS DE PNLIM NÃO CONVEXOS	46
2.2.1	Como lidar com os subproblemas de PNL.....	46
2.2.2	Como tratar a não convexidade do modelo	47
2.2.3	Regras de sondagem	49
2.2.4	Como tratar questões de infactibilidade e convergência	50
2.2.5	Seleção da variável de separação e o próximo subproblema a analisar	51
2.2.6	Estrutura do algoritmo BB proposto para resolver problemas de PNLIM	54
3	PLANEJAMENTO DA EXPANSÃO DE REDES DE TRANSMISSÃO	57
3.1	INTRODUÇÃO.....	57
3.2	MODELOS CLÁSSICOS DE PERT	60
3.2.1	Modelo de transportes	62
3.2.2	Modelos híbridos	64
3.2.2.1	<i>Modelo híbrido não linear</i>	<i>64</i>
3.2.2.2	<i>Modelo híbrido linear</i>	<i>67</i>
3.2.3	Modelo CC.....	69
3.3	MODELO CC PARA O PROBLEMA DE PERT CONSIDERANDO PERDAS	71
3.4	MODELO CA PARA O PROBLEMA DE PERT.....	75
4	TESTES E RESULTADOS.....	79
4.1	TESTES COM O MODELO CC.....	79
4.1.1	Sistema Garver de 6 barras	83
4.1.2	Sistema IEEE de 24 barras.....	84
4.1.3	Sistema Sul Brasileiro de 46 barras.....	86
4.1.4	Sistema Colombiano de 93 barras	88
4.1.5	Discussão.....	90
4.2	TESTES COM O MODELO CA.....	91

4.2.1	Sistema Garver de 6 barras	93
4.2.2	Sistema IEEE de 118 barras.....	93
4.2.3	Discussão.....	94
5	CONCLUSÃO.....	95
	REFERÊNCIAS	97
	APÊNDICE – TRABALHOS PUBLICADOS PELO AUTOR.....	103

1 INTRODUÇÃO

Neste trabalho é apresentado um algoritmo *branch and bound* (BB) para resolver problemas de programação não linear inteira mista (PNLIM) não convexos, com aplicação ao problema de planejamento da expansão de redes de transmissão (PERT) (ZOPPEI, DELGADO, *et al.*, 2022). Inicialmente, um exemplo simples com duas variáveis é usado para ilustrar alguns problemas encontrados quando o algoritmo BB apresentado na literatura apenas para problemas de programação linear inteira mista (PLIM) convexos é utilizado diretamente em problemas de PNLIM não convexos. A natureza não linear do problema de PNLIM, mesmo com apenas duas variáveis, juntamente com a não convexidade da região de busca torna o algoritmo BB deficiente, por isso, uma tentativa de aplicação direta apresenta falhas. A ausência da linearidade ou da convexidade tira a garantia de um ótimo global. Além disso, as ferramentas para a resolução dos subproblemas de programação não linear (PNL) dependem muito de características do problema e do tipo da região de busca a ser investigada. A resolução desse tipo de problema é de grande complexidade devido a sua natureza combinatorial e a existência de um grande número de ótimos locais em uma região de busca não convexa. Isso mostra uma necessidade de se adequar o algoritmo BB para contornar os problemas que surgem.

Nesse contexto, é proposto um algoritmo BB para resolver diretamente problemas de PNLIM onde a região de busca pode ser não convexa. O algoritmo é desenvolvido com base nos questionamentos motivados pelo citado exemplo simples de duas variáveis. Como consequência, foi necessário estabelecer estratégias abordando os seguintes temas: como tratar problemas de programação não linear (PNL); como detectar a infactibilidade de um subproblema de PNL; como tratar a não convexidade do problema e; como definir os testes de sondagem. Junto a esses assuntos provenientes da natureza dos problemas de PNLIM, também foram analisadas estratégias que melhoram a eficácia do algoritmo BB através da escolha da variável para separação e da escolha do próximo subproblema a ser resolvido.

O algoritmo BB proposto é testado na resolução de problemas de PERT considerando a operação representada pelo modelo não linear de corrente contínua (CC) com e sem perdas de transmissão. Esses problemas são caracterizados como problemas de PNLIM não convexos. Nesse processo, são utilizados quatro sistemas teste configurados com e sem reprogramação de geração: o sistema Garver de 6 barras, o

sistema IEEE de 24 barras, o sistema Sul Brasileiro de 46 barras e o sistema Colombiano de 93 barras. O desempenho do algoritmo BB proposto é comparado com o desempenho de solucionadores comerciais de otimização para problemas de PNLIM. No processo comparativo é mostrado que os resultados obtidos pelo algoritmo BB proposto apresentam desempenho igual ou melhor que os solucionadores comerciais utilizados. Também foram realizados testes com dois sistemas considerando o modelo de PERT de corrente alternada (CA).

1.1 CONTRIBUIÇÕES

As principais contribuições deste trabalho são (i) mostrar porque o algoritmo BB, conforme definido para o caso de PLIM, não é suficiente para a resolução de problemas de PNLIM não convexos, (ii) identificar os problemas causados pela não linearidade do problema e pela não convexidade da região de busca, (iii) apresentar estratégias para contornar cada problema identificado, (iv) aplicar o algoritmo proposto em sistemas-teste para resolver o problema de PERT com e sem perdas de transmissão usando o modelo CC com formulações não convexas, (v) realizar testes com o modelo CA e (vi) avaliar o desempenho de vários solucionadores locais e globais para a resolução do problema de PERT.

1.2 ESTRUTURA DO TRABALHO

Este documento está estruturado em cinco capítulos. O Capítulo 1, conforme já foi visto, contém um texto introdutório sobre o assunto tratado neste trabalho. O Capítulo 2 apresenta o algoritmo BB tradicional para resolver problemas de PLIM, na sequência, esse algoritmo é utilizado para resolver um problema não linear com região de busca não convexa tendo como objetivo a identificação de questões relacionadas com a sua ineficácia na resolução desse tipo de problema e, finalmente, apresenta as estratégias propostas para adaptar o algoritmo BB para resolver problemas de PNLIM não convexos. O Capítulo 3 apresenta e discute as principais questões relacionadas com as modelagens matemáticas mais utilizadas na formulação do problema de PERT e, finalmente, apresenta o modelo CC utilizado na formulação do problema de PERT sem e com perdas de transmissão resolvidos neste trabalho, além do modelo CA. O

Capítulo 4 apresenta os testes e resultados e, por último, o Capítulo 5 apresenta as conclusões deste trabalho.

2 ALGORITMO BRANCH AND BOUND

No campo da pesquisa operacional, há uma teoria consolidada para resolver os problemas de programação linear (PL) com métodos eficazes, e.g., *simplex* e método de ponto interior. Complementando essa teoria, o algoritmo *branch and bound* (BB), um algoritmo de otimização clássica, surgiu como uma poderosa ferramenta para a resolução de problemas de programação linear inteira mista (PLIM). Inicialmente, ele foi utilizado principalmente para resolver o problema do caixeiro viajante e, com a evolução computacional, essa técnica de resolução está sendo utilizada para resolver diversos problemas de otimização.

O algoritmo BB, inicialmente foi proposto por Land e Doig em 1960 como um método de busca de solução para problemas de PLIM (LAND e DOIG, 1960). A primeira ideia surgiu por volta de 1958 onde foi descrito apenas um esquema enumerativo. Eastman, provavelmente, desenvolveu o primeiro algoritmo BB (EASTMAN, 1958). Todavia Land e Doig foram os primeiros que propuseram o método BB como um algoritmo genérico para resolver problemas de PLIM. Por se tratar de um método geral de ampla abordagem, o algoritmo BB foi amplamente difundido, destacando-se como uma ferramenta eficiente para a resolução de problemas de PLIM. Land e Doig apresentaram um algoritmo numérico para a solução de um problema de programação linear em que todas as variáveis assumiam somente valores discretos; trata-se da primeira apresentação do algoritmo BB. No decorrer do processo de busca de solução, a variável era forçada a assumir um valor inteiro, diferentemente da divisão em dois subproblemas como atualmente é usado.

A primeira vez que o método foi chamado de BB foi em Little et al. em 1963, em um artigo sobre o problema do caixeiro viajante (LITTLE, G.MURTY, *et al.*, 1963). Nesse artigo, o termo *branch* é utilizado para a busca da menor opção minimizante de distância e/ou custo entre duas cidades e o termo *bound* para a sondagem das cidades já percorridas e das infactibilidades. Isso foi reconhecido no artigo de Lawler e Wood em 1966, que apresentaram um modelo de árvore BB onde investigaram as condições impostas aos novos subproblemas para decidir quando eliminar e quando resolver cada subproblema gerado na relaxação do problema original (LAWLER e WOOD, 1966).

Dakin em 1965, baseado no artigo de Land e Doig, apresentou um algoritmo conceitualmente similar, trazendo como contribuição a vantagem de uma maior

facilidade de implementação do algoritmo (DAKIN, 1965). Foi no artigo de Dakin que o algoritmo tomou sua forma atual em que para uma dada uma variável x_j , com $k < x_j < k + 1$ para k inteiro, a região de busca da solução pode ser dividida em duas regiões não sobrepostas $x_j < k$ e $x_j > k + 1$. Dessa forma, foi proposto o algoritmo somente com duas divisões para cada variável, diferentemente de Land e Doig, que chegaram a fazer seis divisões em um nó da árvore BB.

Posteriormente, em 1970, Mitten determinou as quatro regras básicas do algoritmo BB, a saber: *branching*, *bounding*, *selection* e *elimination*, que caracterizavam respectivamente a divisão do problema em dois subproblemas, a limitação, a regra de seleção para escolher o próximo nó a ser avaliado e a regra de eliminação para excluir os nós que não precisam ser analisados por não acrescentarem contribuição se forem resolvidos (MITTEN, 1970).

Na década de 1980 Balas e Toth fizeram uma revisão do estado da arte do algoritmo BB até aquele instante, destacando a aplicação do algoritmo para a solução do problema do caixeiro viajante (BALAS e TOTH, 1983).

Em 1995, Bruin fez uma nova revisão citando diversos autores que apresentaram alguma contribuição ao método BB, em que destacou novamente as quatro regras básicas de funcionamento do algoritmo BB: a regra *branch* para a decomposição dos nós, a regra *bound* para a limitação, a regra de seleção para escolher o próximo nó a ser avaliado e a regra de eliminação para a poda dos nós (BRUIN, KINDERVATER e TRIENEKENS, 1995).

O conceito de *pseudocustos* para problemas de PLIM foi introduzido por Benichou et al. em 1971. A ideia era selecionar a variável de separação que maximiza a degradação do valor da função objetivo do problema de PL relaxado. Esta técnica permite estimar a variação no valor da função objetivo de um PL por unidade de variação no valor de uma variável inteira e estimar um limitante inferior de cada subproblema (BENICHO, GAUTHIER, et al., 1971). Assim, é possível observar o decréscimo da função objetivo para identificar a variável mais promissora para ser dividida e o subproblema que deve ser escolhido para análise. Subproblemas que possuem uma menor estimativa têm grandes probabilidades de levarem às melhores soluções inteiras (GAUTHIER e RIBIERE, 1977).

É importante destacar que a utilização dos pseudocustos é ineficiente para casos em que há muitas variáveis para separação, pois o cálculo dos pseudocustos depende

de subproblemas subsequentes. Nesse sentido, uma observação importante foi apresentada em (BENICHOU, GAUTHIER, *et al.*, 1971) e, posteriormente, confirmada em (LINDEROTH e SAVELSBERGH, 1999), que os pseudocustos das variáveis inteiras em um sentido de determinado ramo permanecem constantes ao longo do ramo na árvore BB, com exceção de apenas alguns nós, ou seja, o pseudocusto calculado de uma variável pode ser utilizado ao longo da árvore de BB sem ter que voltar a calculá-lo em outros nós.

Alguns autores introduziram algumas variações no uso dos pseudocustos, como Gauthier e Ribiere em 1977 e Linderoth e Savelsbergh em 1999, que sugeriram a criação de um limite para o número de iterações destinadas ao cálculo de pseudocustos e, Glankwamdee em 2006 que estendeu a análise considerando os primeiros nós como nós *filhos* e os nós subsequentes como nós *netos* (GAUTHIER e RIBIERE, 1977), (LINDEROTH e SAVELSBERGH, 1999), (GLANKWAMDEE e LINDEROTH, 2006).

Entre os autores citados até aqui, vários deles usaram o algoritmo BB anexado a outras ferramentas, como o Lagrangeano, heurísticas e outras relaxações e têm conseguido êxito na resolução de um grande número de problemas práticos importantes.

No entanto, o foco deste trabalho está relacionado com o uso do algoritmo BB na resolução de problemas não lineares na engenharia elétrica. Nesse sentido, destacam-se autores que fizeram algum tipo de aplicação do algoritmo BB para resolver problemas de programação não linear inteira mista (PNLIM).

Gupta e Ravindran em 1985, investigaram a capacidade do algoritmo BB para resolver problemas de PNLIM com região de busca convexa. Foram testadas 27 estratégias do algoritmo BB envolvendo uma variação no uso dos pseudocustos e regras heurísticas (GUPTA e RAVINDRAN, 1985). Uma aplicação direta do algoritmo BB para resolver problemas de PNLIM com região de busca convexa, também foi feita em (BONAMI, LEE, *et al.*, 2011). Nesses casos, a convexidade dos problemas é uma condição que facilita a convergência do algoritmo para encontrar a melhor solução sem ajuste. Isso possibilitou que o algoritmo BB original, definido para a resolução de problemas PLIM, fosse usado diretamente com êxito.

Por outro lado, devido à robustez do algoritmo BB, vários pesquisadores (RIDER, 2006; RIDER, GARCIA e ROMERO, 2008; KILINÇ e SAHINIDIS, 2018) se interessaram em utilizá-lo para a resolução de problemas de programação não linear inteira

mista (PNLIM) com região de busca não convexa, onde utilizaram um solucionador de suporte para resolver os subproblemas de programação não linear (PNL). Os trabalhos desenvolvidos mostraram que o algoritmo BB apresenta falhas ao resolver problemas de PNLIM não convexos. Foram apresentadas algumas estratégias para tentar superar as dificuldades decorrentes da natureza não linear do problema e da convergência para soluções de boa qualidade. Cabe observar que ainda não existe uma proposta de criação de um algoritmo BB geral para a resolução de problemas de PNLIM com região de busca não convexa.

Nesse sentido, como contribuição, este trabalho apresenta uma proposta de criação de um algoritmo BB para a resolução de problemas de PNLIM com região de busca não convexa e função objetivo linear. Inicialmente é apresentado o algoritmo BB tradicional para resolver problemas de PLIM. Em seguida, é realizada uma discussão sobre as questões da ineficácia do algoritmo BB tradicional na resolução de problemas de PNLIM não convexos. Finalmente, são apresentadas as estratégias propostas para adaptar o algoritmo BB para resolver problemas de PNLIM não convexos com função objetivo linear. Para dar credibilidade ao algoritmo BB proposto, ele é utilizado para resolver o problema de PERT usando os modelos CC e CA com formulações não convexas e os resultados obtidos são comparados com os existentes na literatura.

É interessante destacar que em trabalhos futuros, novos estudos poderão definir uma proposta para a criação de um algoritmo BB mais geral, onde possa ser possível efetuar a resolução de problemas de PNLIM com região de busca não convexa e função objetivo não linear.

2.1 ALGORITMO BB TRADICIONAL PARA RESOLVER PROBLEMAS DE PLIM

A técnica de solução utilizada no algoritmo BB tradicional (HAFFNER, 2000) para problemas de PLIM é descrita a seguir. Sem perda de generalidade, considera-se o caso de minimização (o caso de maximização é análogo) de um problema de PLIM como apresentado em (1).

$$(P) \quad \begin{cases} \text{minimizar } v(\mathbf{x}) \\ \text{sujeito a:} \end{cases} \quad (1)$$

$$\left\{ \begin{array}{l} \mathbf{x} \in C \end{array} \right.$$

onde v é uma função linear, $\mathbf{x} = (x_j)_{j \in I}$ é um vetor onde alguns componentes $j \in I$ devem ser valores inteiros, I é um conjunto de índices das variáveis e C é um conjunto convexo.

A resolução do problema P é uma tarefa difícil devido à condição complicante de integralidade de algumas variáveis x_j . Assim, a primeira estratégia do algoritmo BB é empregada para diminuir a complexidade do problema original, ou seja, é retirada a condição complicante de integralidade das variáveis inteiras do problema de PLIM, P , transformando-o em um subproblema de PL representado por P_0 . Uma consequência importante obtida com essa estratégia é que os subproblemas de PL têm a característica de obter soluções ótimas que são localizadas na fronteira (em um ponto extremo) da região convexa. No caso da solução de P_0 satisfazer às condições de integralidade do problema P , então, tem-se a solução ótima global do problema PLIM original P e o processo termina. No entanto, o que geralmente ocorre em problemas reais é a solução de P_0 ser obtida com algumas variáveis inteiras com valores não inteiros. Nesse caso, inicia-se a busca de uma melhor solução inteira mista a partir da solução de P_0 . Nesse contexto, o subproblema P_0 é dividido em subproblemas mais restritos de acordo com alguns critérios que são definidos no passo *branch*. O termo *branch* está relacionado com a filosofia do *dividir para conquistar*, ou seja, é uma estratégia utilizada pelo algoritmo para selecionar entre as variáveis pertencentes à solução do subproblema P_0 , uma variável inteira x_j com valor corrente não inteiro. A variável x_j selecionada é nomeada *variável de divisão* e pode ser escrita na forma $x_j = \lfloor x_j \rfloor + f_j$, em que $\lfloor x_j \rfloor$ é o maior valor inteiro contido em x_j , isto é, via truncamento é o maior inteiro menor que x_j e f_j representa a parte fracionária de x_j , ou seja, a parte truncada. A partir dessa seleção, são criados dois novos subproblemas de PL designados por P_1 e P_2 , que são representados como mostrado em (2) e (3).

$$(P_1) \left\{ \begin{array}{l} (P_0) \\ x_j \leq \lfloor x_j \rfloor \end{array} \right. \quad (2)$$

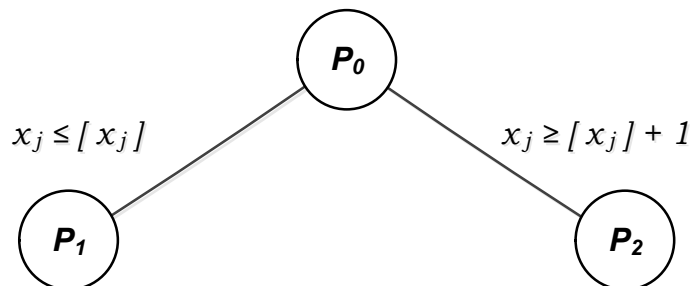
e

$$(P_2) \left\{ \begin{array}{l} (P_0) \end{array} \right. \quad (3)$$

$$\left\lfloor \begin{array}{l} x_j \geq \lfloor x_j \rfloor + 1 \end{array} \right.$$

Sendo assim, a árvore BB pode ser representada conforme mostrado na Figura 1. Nesse processo, é importante verificar que no passo *branch* inicial ocorre a divisão da região factível de P_0 em três sub-regiões sendo a sub-região 1 representada por $x_j \leq \lfloor x_j \rfloor$; a sub-região 2 representada por $x_j \geq \lfloor x_j \rfloor + 1$; e a sub-região 3 representada por $\lfloor x_j \rfloor < x_j < \lfloor x_j \rfloor + 1$. As sub-regiões 1 e 2 geram os subproblemas P_1 e P_2 com a finalidade de reduzir a região factível de P_0 sem eliminar possíveis soluções inteiras factíveis do problema original P e a sub-região 3 é descartada porque ela não possui solução inteira que contribua para a otimização do problema. A escolha da variável inteira x_j com valor corrente não inteiro como variável de separação é de fundamental importância, pois essa escolha geralmente altera o tamanho da árvore BB. Além da escolha da variável de separação x_j , também é preciso estabelecer uma estratégia para seleccionar a prioridade na resolução dos subproblemas P_1 e P_2 , pois o tipo de estratégia utilizada pode influenciar no tamanho final da árvore BB (BENICHO, GAUTHIER, *et al.*, 1971), (LINDEROTH e SAVELSBERGH, 1999).

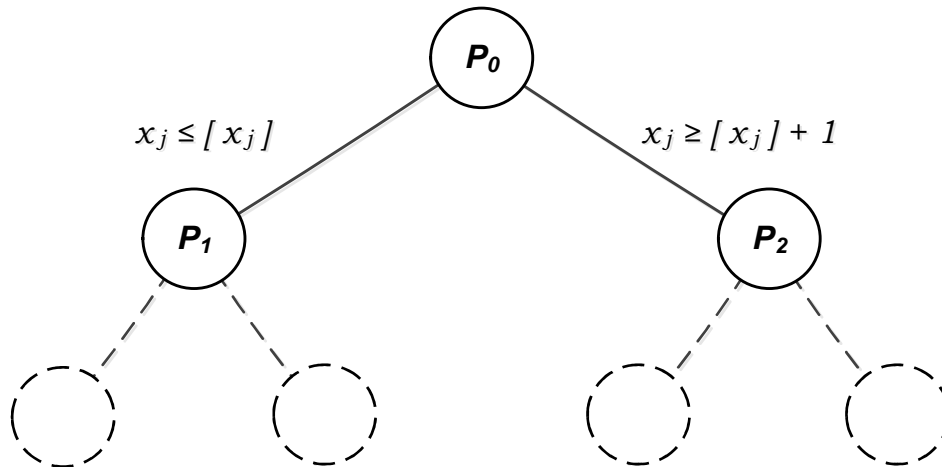
FIGURA 1 - Ilustração da árvore BB após a divisão de P_0 .



Fonte: Adaptado de (HAFFNER, 2000).

Após a primeira divisão do subproblema inicial P_0 em dois novos subproblemas, P_1 e P_2 e resolvendo-os, caso não seja encontrada a solução do problema original P , o processo continua com uma nova subdivisão tanto de P_1 como de P_2 , cada um gerando mais dois novos subproblemas, conforme pode ser visto na Figura 2.

FIGURA 2 - Ilustração da árvore BB após as divisões de P_1 e P_2 .



Fonte: Adaptado de (HAFFNER, 2000).

A cada iteração, o nó gerador é nomeado *nó pai* e cada um dos nós gerados são nomeados *nós filhos*. Cada subproblema representado pelo nó filho está sujeito a todas as restrições do subproblema representado pelo nó pai acrescido de mais uma restrição que o originou no passo *branch*. Em consequência disso, devido à linearidade do problema, a solução de cada subproblema pai é um limitante inferior para a solução de cada subproblema filho.

O termo *bound* é utilizado para se referir à aplicação das regras de sondagem nos subproblemas ainda não resolvidos. As regras de sondagem são utilizadas para eliminar um subproblema, ou seja, é o passo em que ocorre a poda dos nós. O algoritmo finaliza o processo quando não há mais subproblemas para sondar. Um subproblema P_k é sondado nos seguintes casos:

- i. Quando a solução v_k do subproblema P_k for infactível;
- ii. Quando for encontrada uma solução v^* do problema de PLIM original, o subproblema que produziu esta solução é eliminado. A solução v^* é chamada de incumbente e deve ser atualizada sempre que uma melhor solução incumbente for encontrada, no caso de minimização uma solução incumbente melhor é menor que v^* ;
- iii. Quando a solução v_k for maior que a solução incumbente v^* , pois no caso de minimização, a linearidade do problema impõe que qualquer solução proveniente de mais subdivisões deste nó será maior que v_k , ou seja, v_k é um limite inferior para todas as soluções de seus subproblemas descendentes.

O processo utilizado na técnica de solução do algoritmo BB tradicional para problemas de PLIM termina quando não há mais subproblemas para resolver. A solução incumbente v^* atualizada, se existir, é a solução do problema inicial P .

2.1.1 Estrutura do algoritmo BB para resolver problemas de PLIM

A estrutura do algoritmo BB para resolver problemas de PLIM é formada pelos seguintes passos.

1. Inicialização: Faça o valor da solução incumbente $v^* \leftarrow \infty$; inicialize o contador de subproblemas $k \leftarrow 0$; inicialize a lista L de subproblemas candidatos P_k com a solução v_0 do subproblema inicial P_0 . Se a solução de P_0 for inteira, então faça $v^* \leftarrow v_0$ e remova P_0 de L . Vá para o Passo 2;
2. Teste de convergência: Verifique a lista L . Se $L = \emptyset$, então o processo terminou e a solução incumbente atual v^* é a solução ótima para o problema de PLIM original. Se $L \neq \emptyset$, vá para o Passo 3;
3. Seleção do subproblema candidato: Utilize uma estratégia pré-definida para selecionar e retirar um subproblema candidato P_k da lista L . Resolva e armazene a sua solução ótima v_k como limitante inferior para todos os seus descendentes. Vá para o Passo 4;
4. Passo *bound*: Um subproblema candidato P_k é sondado quando satisfaz uma das seguintes condições:
 - a. Se a solução v_k do subproblema P_k é infactível;
 - b. Se a solução v_k for maior que a solução incumbente corrente v^* , ou seja, se $v_k > v^*$;
 - c. Se a solução v_k do subproblema P_k também for uma solução para o problema de PLIM original P , ou seja, se as condições de integralidade forem atendidas. No caso de $v_k < v^*$, então a solução incumbente é atualizada, $v^* \leftarrow v_k$, e o teste 4(b) é aplicado novamente, ou seja, a lista L deve ser revisada para que todos os nós que apresentem $v_k \geq v^*$ sejam sondados e k deve ser atualizado.

Após esse passo, se o subproblema candidato P_k foi sondado, retorne ao Passo 2; caso contrário, vá para o Passo 5;

5. Passo *branch*: Da solução do subproblema P_k , utilize uma estratégia pré-definida para selecionar uma variável inteira x_j com um valor atual não inteiro para separação e gerar dois novos subproblemas descendentes. Os dois novos subproblemas têm o mesmo conjunto de restrições do subproblema pai P_k acrescidos das restrições $x_j \leq \lfloor x_j \rfloor$ para um e $x_j \geq \lfloor x_j \rfloor + 1$ para o outro, em que $\lfloor x_j \rfloor$ é o maior inteiro menor que x_j . Adicione os dois novos subproblemas gerados à lista de candidatos L , faça $k \leftarrow k + 2$ e retorne ao Passo 3.

Esse processo iterativo se encerra quando não há mais subproblemas para son-
dar ou quando a solução do problema original P é encontrada, se existir. A próxima
seção apresenta um exemplo para ilustrar a metodologia utilizada pelo algoritmo BB
para resolver problemas de PLIM.

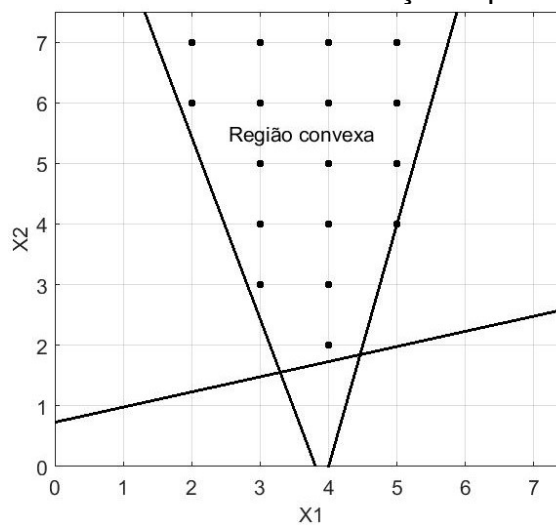
2.1.2 Estudo de caso: Análise de um problema de PLIP

A seguir, é apresentado um exemplo simples com duas variáveis para ilustrar a
estratégia utilizada pelo algoritmo BB em resolver problemas de PLIM. Para isso, se-
lecionamos o problema de programação linear inteira pura (PLIP), representado por
 P , conforme mostrado em (4). A resolução deste problema tem o objetivo de encontrar
o menor valor possível para a função objetivo $v = x_1 + 6x_2$, no entanto, a busca da
solução deve respeitar o conjunto de restrições formado pelas equações imediata-
mente abaixo da função objetivo.

$$\begin{array}{ll}
 (P) \left[\begin{array}{l}
 \text{minimizar } v = x_1 + 6x_2 \\
 \text{sujeito a:} \\
 x_2 + 3x_1 - 11,433 \geq 0 \\
 x_2 - 4x_1 + 16 \geq 0 \\
 x_2 - 0,25x_1 - 0,7297 \geq 0 \\
 x_1 \geq 0 \\
 x_2 \geq 0 \\
 x_1 \text{ e } x_2 \text{ inteiros}
 \end{array} \right. & \left. \begin{array}{l} \\ \\ \\ \\ \\ \\ \end{array} \right] (R)
 \end{array} \tag{4}$$

Esse problema pode ser representado graficamente (variável x_2 como função da variável x_1) conforme mostrado na Figura 3. Este gráfico mostra que a região R de busca da solução é uma região convexa. A região R é limitada pela reta decrescente à esquerda definida pela primeira restrição, pela reta crescente à direita definida pela segunda restrição, pela reta crescente mais abaixo definida pela terceira restrição, a quarta e a quinta restrições definem que a região R de busca da solução deve pertencer ao primeiro quadrante. Finalmente, a sexta e última restrição com x_1 e x_2 inteiros é representada pelos pontos destacados na cor preta localizados na parte interna da região R . Cada um desses pontos representa uma possível solução que o algoritmo BB poderá encontrar para o problema P de PLIP.

FIGURA 3 - Região convexa de busca da solução e possíveis soluções de P .



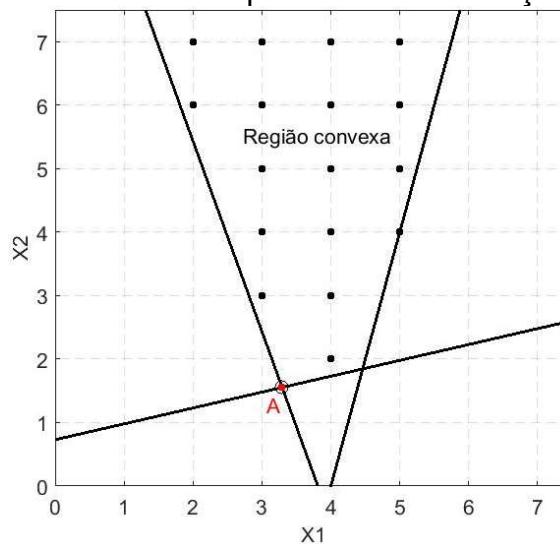
Fonte: Elaboração do próprio autor.

Para encontrar a solução do problema P , a primeira estratégia utilizada pelo algoritmo BB é relaxar a integralidade das variáveis inteiras do problema P transformando-o em um problema de PL, representado em (5) pelo subproblema P_0 . Esse procedimento não altera a região R de busca da solução, isto é, a região de busca da solução de P_0 continua sendo a mesma de P .

$$\begin{array}{l}
 \text{minimizar } v = x_1 + 6x_2 \\
 \text{sujeito a:} \\
 \left. \begin{array}{l}
 x_2 + 3x_1 - 11,433 \geq 0 \\
 x_2 - 4x_1 + 16 \geq 0 \\
 x_2 - 0,25x_1 - 0,7297 \geq 0 \\
 x_1 \geq 0 \\
 x_2 \geq 0
 \end{array} \right\} (R)
 \end{array} \quad (P_0) \quad (5)$$

Considerando que o solucionador KNITRO (BYRD, NOCEDAL e WALTZ, 2006) é utilizado como ferramenta de suporte para resolver os subproblemas de PL que surgem em cada nó da árvore BB, inicialmente o algoritmo BB encontra a solução ótima global $v_0 = 12,6115$ para a função objetivo de P_0 , com $x_1 = 3,29332$ e $x_2 = 1,55303$. Na Figura 4, o ponto A (3,29332; 1,55303) é utilizado para representar a solução de P_0 . Observe que o ponto A está localizado na fronteira da região de busca da solução.

FIGURA 4 - Ponto A representando a solução de P_0 .

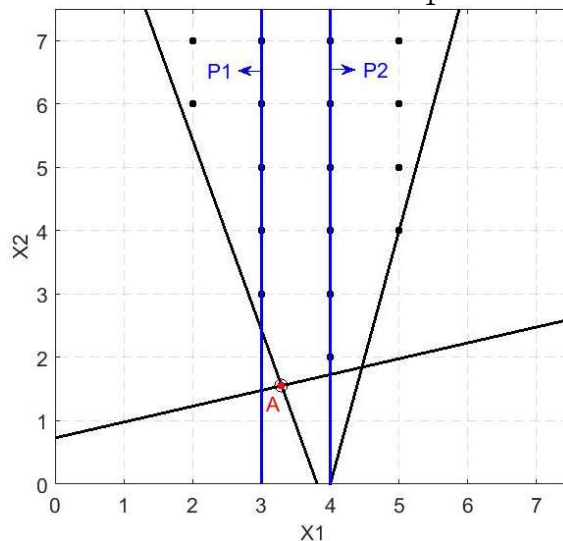


Fonte: Elaboração do próprio autor.

Como as variáveis inteiras x_1 e x_2 da solução de P_0 têm valores correntes não inteiros, então o próximo passo do método BB é a aplicação do passo *branch*. O passo *branch* é utilizado para selecionar uma das variáveis x_1 ou x_2 para realizar a separação da região factível de busca da solução gerando dois subproblemas descendentes de P_0 . Se a variável $x_1 = 3,29332$ for selecionada, então os subproblemas P_1 (P_0 com

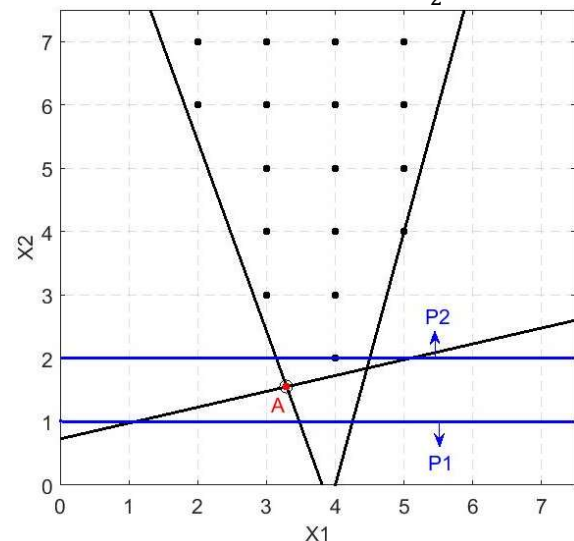
a restrição $x_1 \leq 3$) e P_2 (P_0 com a restrição $x_1 \geq 4$) são criados, nesse caso, as sub-regiões da região factível assumem a forma mostrada na Figura 5. No caso da variável $x_2 = 1,55303$ ser selecionada, então os subproblemas P_1 (P_0 com a restrição $x_2 \leq 1$) e P_2 (P_0 com a restrição $x_2 \geq 2$) são criados e as sub-regiões da região factível assumem a forma mostrada na Figura 6.

FIGURA 5 - Sub-regiões geradas com a variável de divisão x_1 .



Fonte: Elaboração do próprio autor.

FIGURA 6 - Sub-regiões geradas com a variável de divisão x_2 .



Fonte: Elaboração do próprio autor.

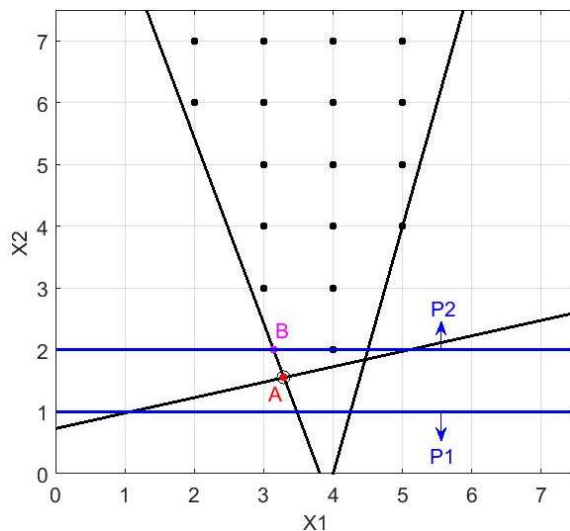
Essa estratégia do algoritmo BB divide a região R de busca da solução em três sub-regiões, sendo que uma delas é eliminada do problema e as outras duas são menores e, portanto, torna-se mais fácil de encontrar uma possível solução. Por exemplo, observe na Figura 5 que a região convexa R foi dividida em três sub-regiões: a sub-região do lado esquerdo, $x_1 \leq 3$, gerada pela criação do subproblema P_1 ; a sub-região do lado direito, $x_1 \geq 4$, gerada pela criação do subproblema P_2 e; a sub-região central, com $3 < x_1 < 4$, que não contém pares ordenados inteiros, por isso, ela é excluída da busca pela solução com x_1 e x_2 inteiros definida no problema inicial. Esse processo se repete sempre que uma variável inteira com valor corrente não inteiro é utilizada para criar dois novos subproblemas descendentes do subproblema pai.

A escolha de uma variável inadequada para divisão implica na necessidade de uma maior quantidade de memória para armazenar as informações dos subproblemas gerados e seus descendentes, principalmente em problemas de médio/grande porte. Não existe uma técnica sistemática para determinar qual variável de separação conduz mais rapidamente à solução, existem apenas algumas regras empíricas que

permitem identificar caminhos mais atrativos. Uma regra simples de ser utilizada é a que seleciona a variável de divisão a partir de uma ordem pré-determinada do maior para o menor valor dos coeficientes da função objetivo (HAFFNER, 2000).

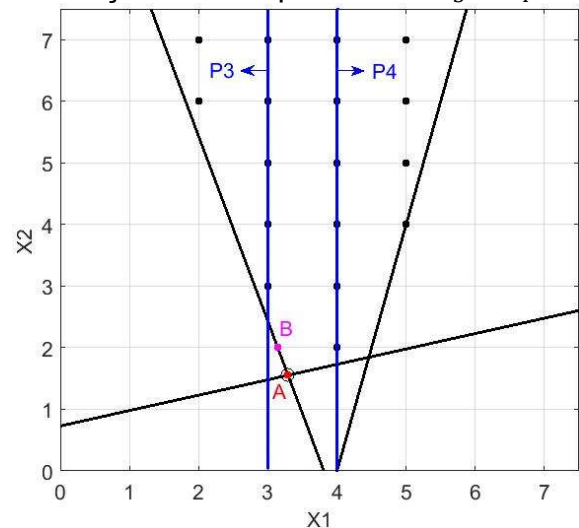
Nesse sentido, aplicando a citada regra simples, o algoritmo dá continuidade à resolução deste problema escolhendo a variável $x_2 = 1,55303$ para separação porque é ela a que possui o maior coeficiente na função objetivo do problema P . Sendo assim, os subproblemas P_1 (P_0 com a restrição $x_2 \leq 1$) e P_2 (P_0 com a restrição $x_2 \geq 2$) são criados conforme mostrado na Figura 6. Pode ser observado nessa figura que a solução do subproblema P_1 é infactível porque esse subproblema está localizado fora da região convexa de busca. A solução encontrada pelo solucionador KNITRO para o subproblema P_2 é $v_2 = 15,1443$ com $x_1 = 3,14433$ e $x_2 = 2,0$. A Figura 7 é um complemento da Figura 6 onde está ilustrado o ponto $B = (3,14433; 2,0)$ com as coordenadas x_1 e x_2 que representam a solução do subproblema P_2 . Observe que o ponto B está localizado na fronteira da sua respectiva sub-região factível.

FIGURA 7 - Solução de P_2 no ponto B.



Fonte: Elaboração do próprio autor.

FIGURA 8 - Sub-regiões geradas com a criação dos subproblemas P_3 e P_4 .

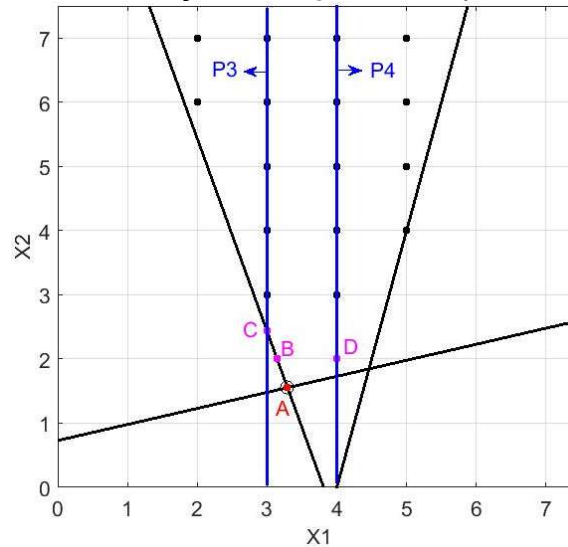


Fonte: Elaboração do próprio autor.

Na sequência, verifica-se que entre as variáveis $x_1 = 3,14433$ e $x_2 = 2,0$ da solução do subproblema P_2 somente a variável x_1 é inteira com valor corrente não inteiro. Portanto, ela é selecionada para que seja realizada uma nova separação da região factível. Então, os subproblemas P_3 (P_2 com a restrição $x_1 \leq 3$) e P_4 (P_2 com a restrição $x_1 \geq 4$) são criados. A Figura 8 mostra a representação gráfica das sub-regiões factíveis geradas com a criação dos subproblemas P_3 e P_4 . As soluções dos subproblemas

P_3 e P_4 são, respectivamente, $v_3 = 17,598$ com $x_1 = 3,0$ e $x_2 = 2,433$ e $v_4 = 16,0$ com $x_1 = 4,0$ e $x_2 = 2,0$. A Figura 9 é um complemento da Figura 8 onde estão ilustrados os pontos $C = (3,14433; 2,0)$ e $D = (4; 2)$ com as respectivas coordenadas x_1 e x_2 que representam as soluções dos subproblemas P_3 e P_4 . Observe que os pontos C e D estão localizados nas fronteiras das suas respectivas sub-regiões factíveis.

FIGURA 9 - Soluções de P_3 e P_4 nos pontos C e D .

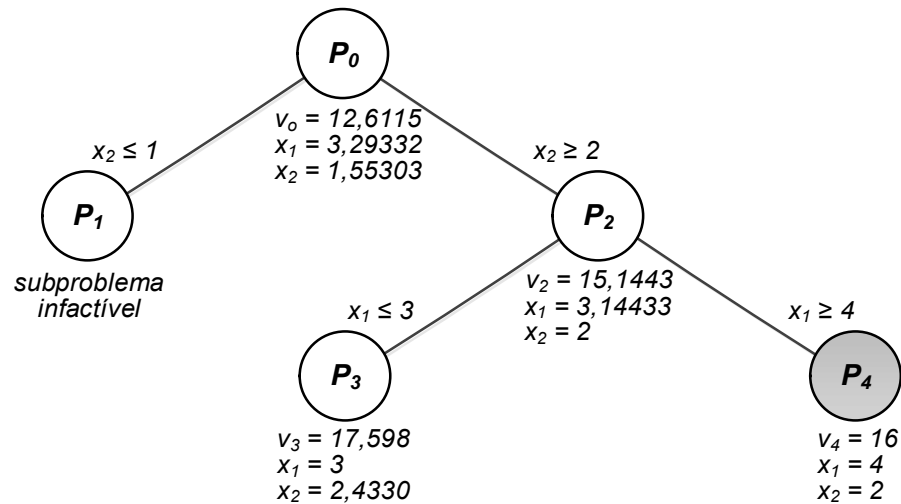


Fonte: Elaboração do próprio autor.

Nesta altura da resolução é interessante organizar todos os resultados obtidos em uma árvore BB, conforme mostrado na Figura 10. Pode ser observado nesta árvore BB que, devido a linearidade do problema, a solução v_0 do subproblema P_0 é uma limitante inferior para as soluções v_1 e v_2 de seus subproblemas descendentes, assim como, a solução v_2 do subproblema P_2 é uma limitante inferior para as soluções v_3 e v_4 de seus subproblemas descendentes. Como consequência, verifica-se que o passo *bound* pode eliminar os subproblemas P_1 e P_3 , ou seja, o subproblema P_1 é infactível (isso quer dizer que a sua solução está fora da região convexa de busca de solução factível) e o subproblema P_3 não precisa ser analisado porque a sua solução $v_3 = 17,598$ é maior que a solução $v_4 = 16$ encontrada pelo subproblema P_4 . A eliminação do subproblema P_3 é uma consequência da linearidade do problema, pois qualquer solução proveniente dos subproblemas descendentes de P_3 será maior que o valor encontrado na solução v_3 , ou seja, v_3 é um limite inferior para todas as demais soluções deste nó. Como a solução do subproblema P_4 encontrou valores inteiros para as variáveis inteiras x_1 e x_2 satisfazendo às condições de integralidade do problema

original P de PLIP e não há mais subproblemas para eliminar, então, a solução ótima global é representada pelo ponto $D = (4; 2)$ com valor mínimo de $v^* = 16$ para a função objetivo.

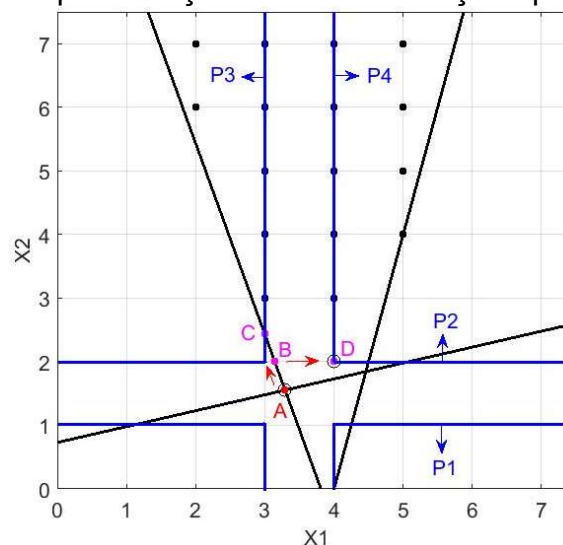
FIGURA 10 - Árvore branch and bound.



Fonte: Elaboração do próprio autor.

Na Figura 11 observa-se que a cada passo ao ser encontrada uma solução não inteira, o método BB impõe a condição de que essa solução esteja numa região infactível, forçando o retorno à factibilidade na busca de uma solução inteira para as variáveis inteiras com valor corrente não inteiro, de forma que, o caminho traçado pelo método é sempre de uma região infactível para uma região factível.

FIGURA 11 - Representação do caminho traçado pelo método BB.



Fonte: Elaboração do próprio autor.

A grande vantagem em se utilizar o algoritmo BB para resolver problemas de PLIM é a garantia de convergência para uma solução global, caso exista. Isso ocorre devido à linearidade do problema e a convexidade da região factível. Deve ser observado que essa garantia não é uma particularidade exclusiva do algoritmo BB, ela se deve às características do tipo de problema que está sendo resolvido (ROMERO, 1999).

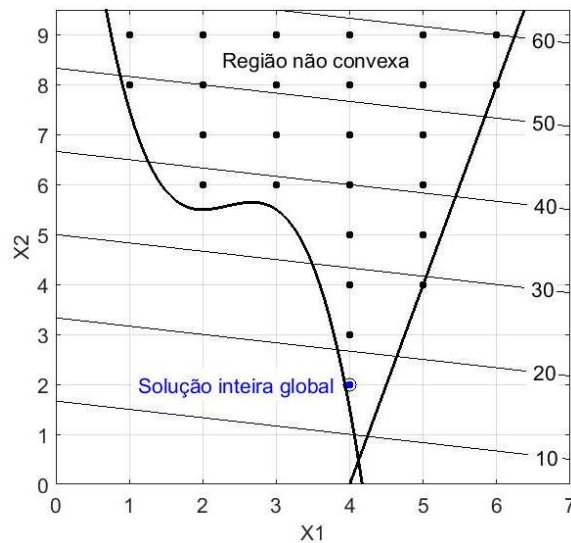
2.1.3 Estudo de caso: Análise de um problema de PNLIP não convexo

A seguir, novamente é apresentado um exemplo simples com duas variáveis para ilustrar que a estratégia utilizada pelo algoritmo BB formulado para resolver problemas de PLIM, na forma como foi definido na seção 2.1.1, apresenta falhas quando utilizado para resolver problemas não lineares com região de busca não convexa. Para isso, selecionamos o problema de programação não linear inteira pura (PNLIP) com região de busca não convexa, representado por P , conforme mostrado em (6).

$$(P) \left[\begin{array}{l} \text{minimizar } v = x_1 + 6x_2 \\ \text{sujeito a:} \\ (x_1 - 2)(x_1 - 2)(x_1 - 3) - 5.5 + x_2 \geq 0 \\ x_2 - 4x_1 + 16 \geq 0 \\ x_1 \geq 0 \\ x_2 \geq 0 \\ x_1 \text{ e } x_2 \text{ inteiros} \end{array} \right] (R) \quad (6)$$

A solução desse problema P pode ser visualizada graficamente conforme mostra a Figura 12. Nesta figura, verifica-se que as restrições impostas à solução desse problema delimitam uma região não convexa R de busca da solução. Dentro desta região estão representadas as localizações de todos os pontos que contém as possíveis soluções inteiras para as variáveis x_1 e x_2 . Também pode ser observada a existência de retas paralelas com seus respectivos valores representando algumas curvas de nível da função objetivo. Como o problema P de PNLIP é um problema de minimização, então as curvas de nível indicam que a solução inteira global desse problema está localizada no ponto $(x_1, x_2) = (4; 2)$ com valor mínimo de $v^* = 16$.

FIGURA 12 - Representação gráfica da região não convexa de busca da solução de P .



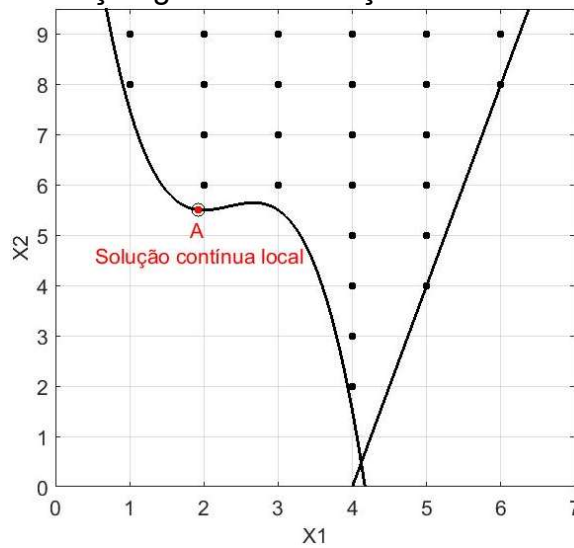
Fonte: Elaboração do próprio autor.

Diante dessas informações, a seguir é apresentada a resolução desse mesmo problema P de PNLIP utilizando o algoritmo BB na forma como foi definido na Seção 2.1.1. O objetivo que se pretende alcançar é analisar e argumentar sobre a possibilidade de utilizar o algoritmo BB (formulado para resolver problemas de PLIM) para resolver problemas de PNLIM não convexos. Como ferramenta de suporte, os solucionadores KNITRO (BYRD, NOCEDAL e WALTZ, 2006) e IPOPT (WÄCHTER e BIEGLER, 2006) são utilizados para resolver os subproblemas de PNL que surgem em cada nó da árvore BB.

No início do processo da resolução, o algoritmo BB relaxa a integralidade do problema P de PNLIP transformando-o em um subproblema P_0 de PNL. Os solucionadores KNITRO e IPOPT encontram a mesma solução ótima local para P_0 sendo $v = 34,9013$ com $x_1 = 1,92509$ e $x_2 = 5,50603$. Esse resultado está representado graficamente no ponto A da Figura 13.

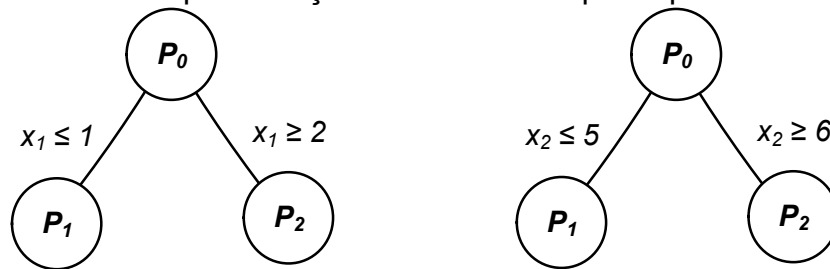
O passo *branch* pode ser iniciado com a escolha de qualquer uma das duas variáveis inteiras com valor corrente não inteiro $x_1 = 1,92509$ ou $x_2 = 5,50603$ para dividir e criar dois novos subproblemas. Para cada um dos dois casos relacionados com a escolha da variável x_1 ou da variável x_2 , as árvores BB assumem as formas mostradas na Figura 14.

FIGURA 13 - Representação gráfica da solução ótima local encontrada para P_0 .



Fonte: Elaboração do próprio autor.

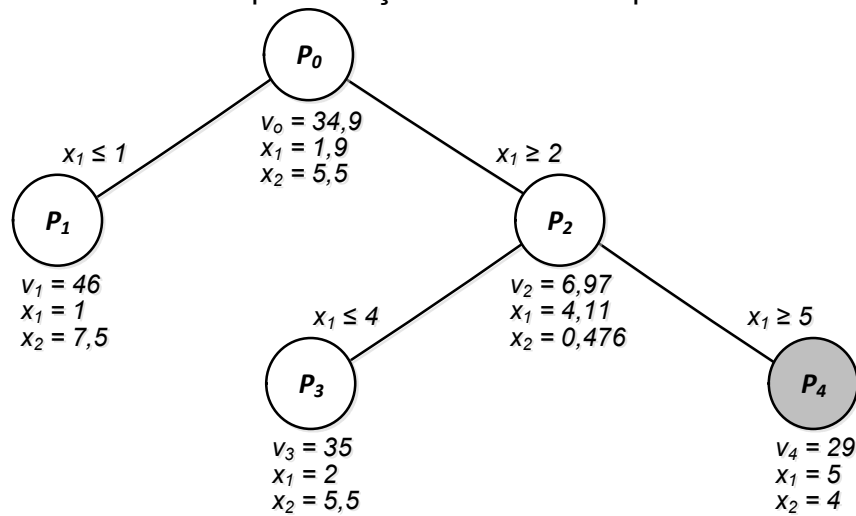
FIGURA 14 - Representação da árvore BB após a primeira divisão.



Fonte: Elaboração do próprio autor.

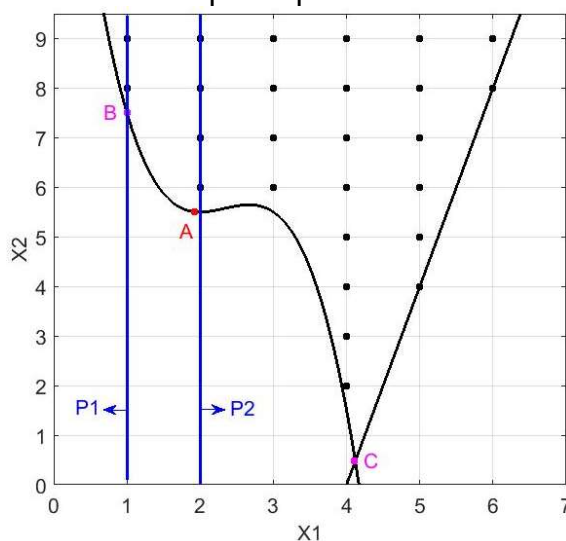
A resolução deste problema será apresentada em quatro casos possíveis para mostrar que a conclusão independe da escolha da primeira variável de divisão x_1 e x_2 ou dos solucionadores KNITRO e IPOPT utilizados para resolver os subproblemas de PNL. Após a escolha da primeira variável de divisão, o passo *branch* utilizará a estratégia de selecionar sempre a maior variável de divisão.

1º Caso: escolha de $x_1 \cong 1,9$ como variável de divisão e uso do solucionador KNITRO. Neste caso, o processo desenvolvido pelo algoritmo BB para resolver o problema P gerou a árvore BB mostrada na Figura 15. Observe que o passo *bound* eliminou os subproblemas P_1 e P_3 , pois v_1 e v_3 são soluções maiores que v_4 . Assim $v^* = v_4 = 29$ é a solução ótima no ponto $(x_1, x_2) = (5; 4)$ encontrada para o problema P .

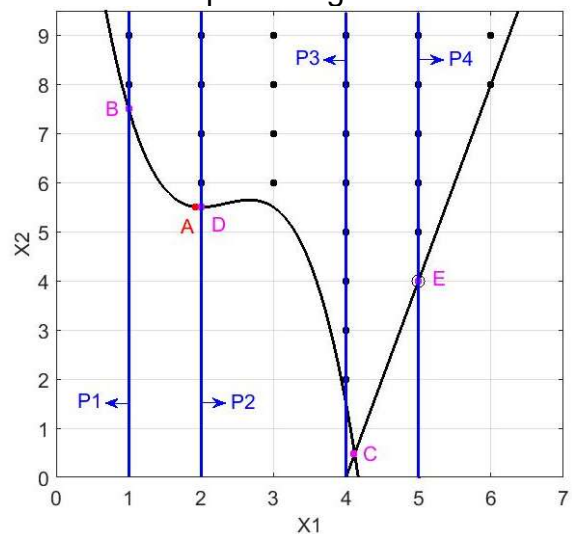
FIGURA 15 - Representação da árvore BB para o 1º Caso.

Fonte: Elaboração do próprio autor.

A Figura 16 mostra a representação geométrica da primeira partição na região factível, executada quando o passo *branch* selecionou a variável x_1 como variável de divisão no subproblema P_0 , gerando os subproblemas P_1 e P_2 . Nesta figura, o ponto A representa a solução $v_0 = 34,9$ no ponto $(x_1, x_2) = (1,9; 5,5)$ encontrada para o subproblema P_0 , o ponto B representa a solução $v_1 = 46$ no ponto $(x_1, x_2) = (1; 7,5)$ encontrada para o subproblema P_1 , e o ponto C representa a solução $v_2 = 6,97$ no ponto $(x_1, x_2) = (4,11; 0,476)$ encontrada para o subproblema P_2 .

FIGURA 16 - Representação da região de busca após a primeira divisão.

Fonte: Elaboração do próprio autor.

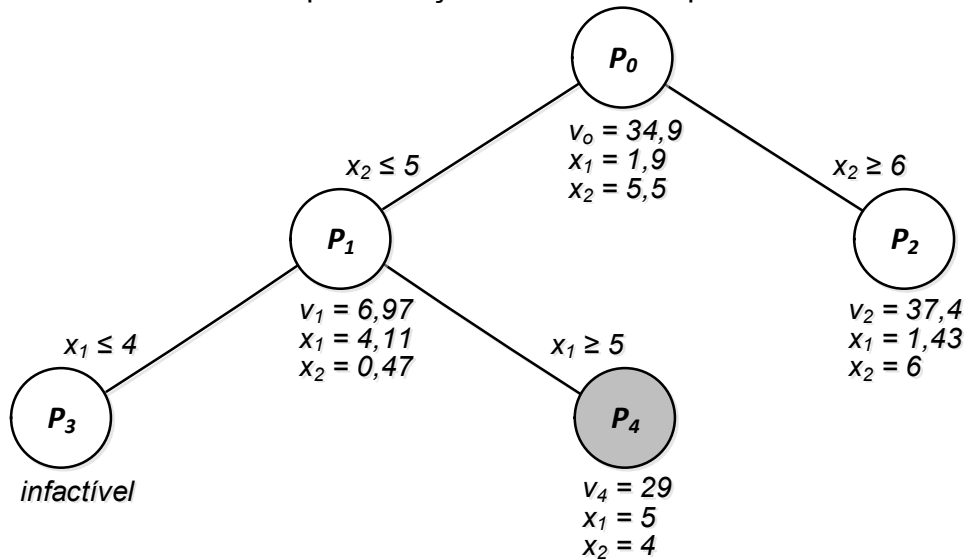
FIGURA 17 - Representação da região de busca após a segunda divisão.

Fonte: Elaboração do próprio autor.

A Figura 17 complementa a Figura 16 com a adição da representação geométrica da segunda partição na região factível, executada quando o passo *branch* selecionou a variável $x_1 = 4,11$ como variável de divisão no subproblema P_2 , gerando os subproblemas P_3 e P_4 . O ponto D representa a solução $v_3 = 35$ no ponto $(x_1, x_2) = (2; 5,5)$ encontrada para o subproblema P_3 e o ponto E representa a solução $v_4 = 29$ no ponto $(x_1, x_2) = (5; 4)$ encontrada para o subproblema P_4 . A solução $v_4 = v^*$ é a solução encontrada pelo algoritmo BB para o problema P no 1º Caso.

2º Caso: escolha de $x_2 \cong 5,5$ como variável de divisão e uso do solucionador KNITRO. Neste caso, o processo desenvolvido pelo algoritmo BB para resolver o problema P gerou a árvore BB mostrada na Figura 18. Observe que o passo *bound* eliminou o subproblema P_2 por possuir uma solução v_2 maior que v_4 e, também, eliminou o subproblema P_3 porque a sua solução foi encontrada numa posição localizada fora da região factível. Assim $v^* = v_4 = 29$ é a solução ótima no ponto $(x_1, x_2) = (5; 4)$ encontrada para o problema P .

FIGURA 18 - Representação da árvore BB para o 2º Caso.

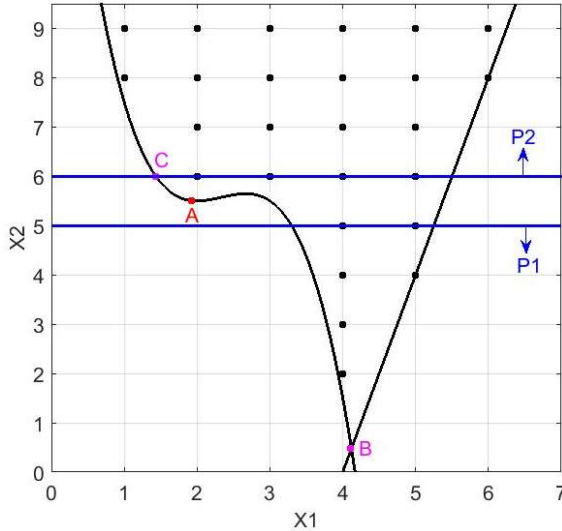


Fonte: Elaboração do próprio autor.

A Figura 19 mostra a representação geométrica da primeira partição na região factível, executada quando o passo *branch* selecionou a variável x_2 como variável de divisão no subproblema P_0 , gerando os subproblemas P_1 e P_2 . Nesta figura, o ponto A representa a solução $v_0 = 34,9$ no ponto $(x_1, x_2) = (1,9; 5,5)$ encontrada para o subproblema P_0 , o ponto B representa a solução $v_1 = 6,97$ no ponto $(x_1, x_2) = (4,11; 0,47)$,

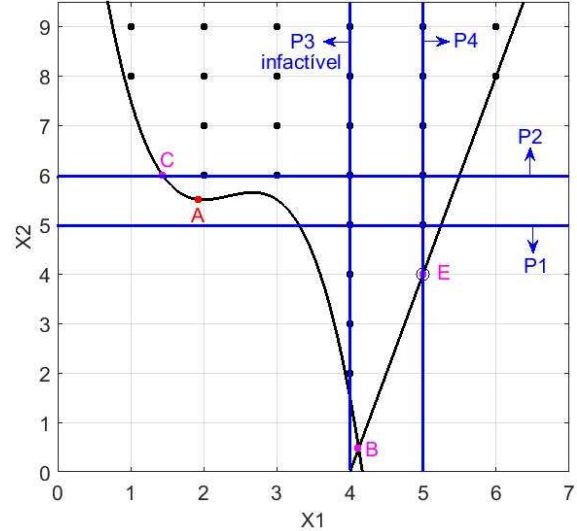
encontrada para o subproblema P_1 , e o ponto C representa a solução $v_2 = 37,4$ no ponto $(x_1, x_2) = (1,43; 6)$, encontrada para o subproblema P_2 .

FIGURA 19 - Representação da região de busca após a primeira divisão.



Fonte: Elaboração do próprio autor.

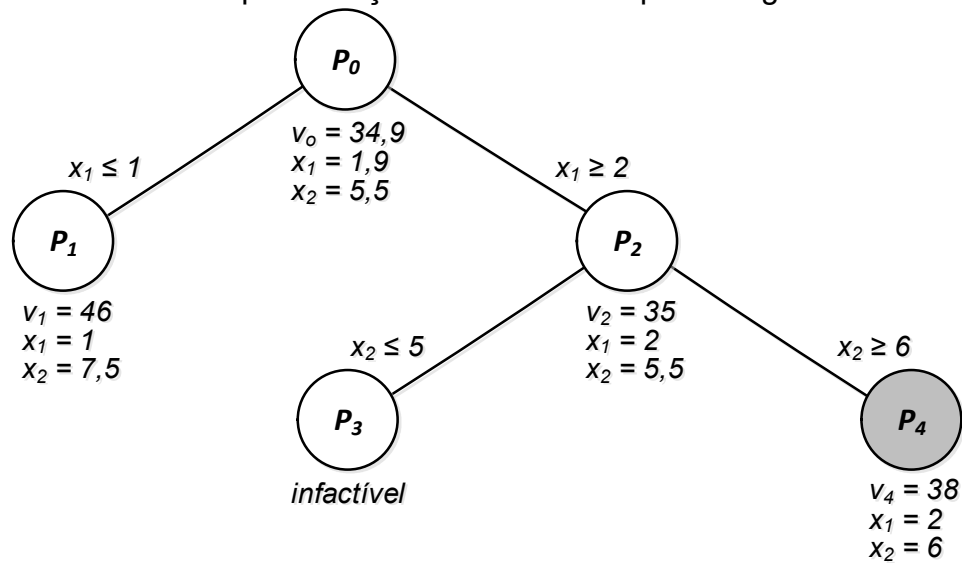
FIGURA 20 - Representação da região de busca após a segunda divisão.



Fonte: Elaboração do próprio autor.

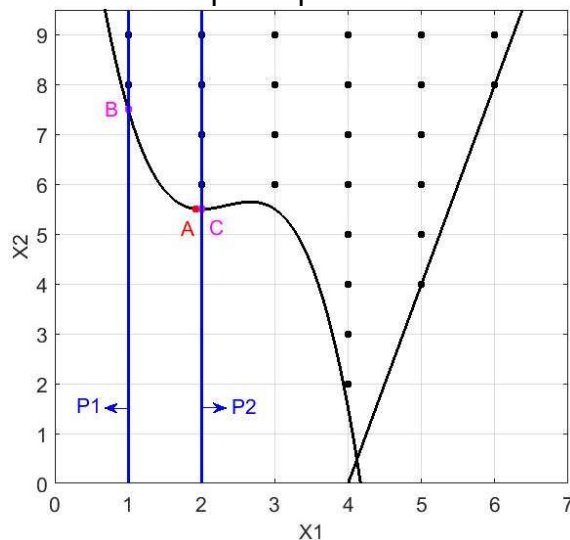
A Figura 20 complementa a Figura 19 com a adição da representação geométrica da segunda partição na região factível, executada quando o passo *branch* selecionou a variável $x_1 = 4,11$ como variável de divisão no subproblema P_1 , gerando os subproblemas P_3 e P_4 . O ponto D, representando a solução do subproblema P_3 , não é incluído nessa figura porque ele cai numa região infactível, e o ponto E representa a solução $v_4 = 29$ no ponto $(x_1, x_2) = (5; 4)$ encontrada para o subproblema P_4 . A solução $v_4 = v^*$ é a solução encontrada pelo algoritmo BB para o problema P no 2º Caso.

3º Caso: escolha de $x_1 \cong 1,9$ como variável de divisão e uso do solucionador IPOPT. Neste caso, o processo desenvolvido pelo algoritmo BB para resolver o problema P gerou a árvore BB mostrada na Figura 21. Observe que o passo *bound* eliminou o subproblema P_1 por possuir uma solução v_1 maior que v_4 e, também, eliminou o subproblema P_3 porque a sua solução foi encontrada numa posição localizada fora da região factível. Assim $v^* = v_4 = 38$ é a solução ótima no ponto $(x_1, x_2) = (2; 6)$ encontrada para o problema P .

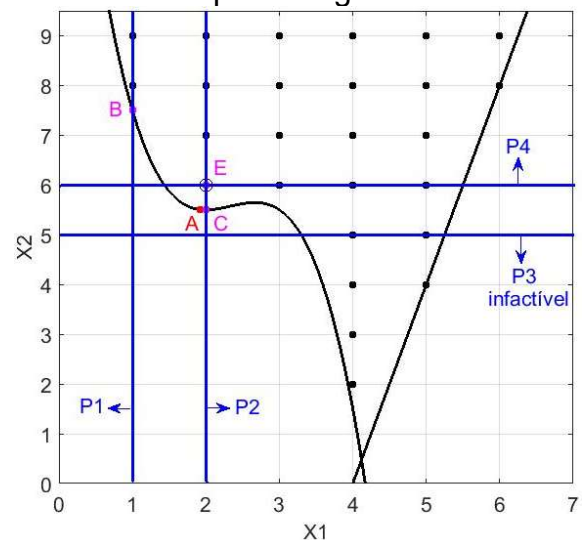
FIGURA 21 - Representação da árvore BB após a segunda divisão.

Fonte: Elaboração do próprio autor.

A Figura 22 mostra a representação geométrica da primeira partição na região factível, executada quando o passo *branch* selecionou a variável x_1 como variável de divisão no subproblema P_0 , gerando os subproblemas P_1 e P_2 . Nesta figura, o ponto A representa a solução $v_0 = 34,9$ no ponto $(x_1, x_2) = (1,9; 5,5)$ encontrada para o subproblema P_0 , o ponto B representa a solução $v_1 = 46$ no ponto $(x_1, x_2) = (1; 7,5)$, encontrada para o subproblema P_1 , e o ponto C representa a solução $v_2 = 35$ no ponto $(x_1, x_2) = (2; 5,5)$, encontrada para o subproblema P_2 .

FIGURA 22 - Representação da região de busca após a primeira divisão.

Fonte: Elaboração do próprio autor.

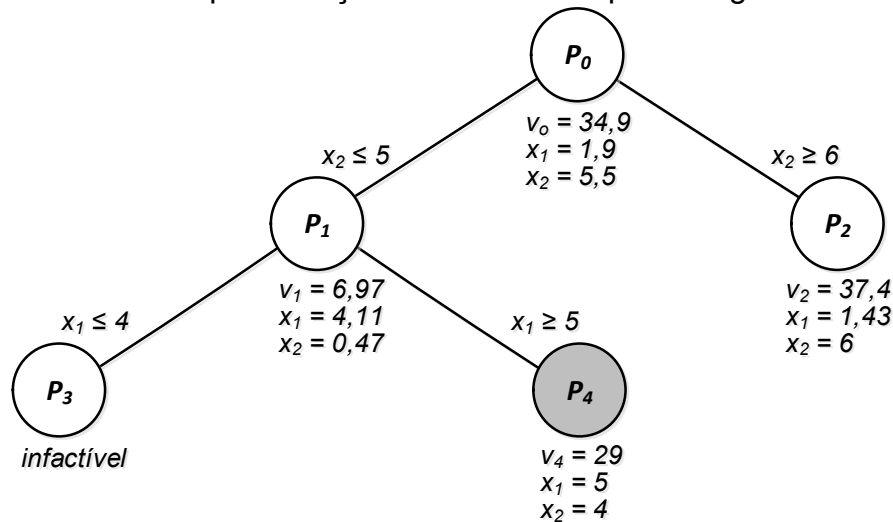
FIGURA 23 - Representação da região de busca após a segunda divisão.

Fonte: Elaboração do próprio autor.

A Figura 23 complementa a Figura 22 com a adição da representação geométrica da segunda partição na região factível, executada quando o passo *branch* seleccionou a variável $x_2 = 5,5$ como variável de divisão no subproblema P_2 , gerando os subproblemas P_3 e P_4 . O ponto D, representando a solução do subproblema P_3 , não é incluído nesta figura porque ele cai numa região infactível e o ponto E representa a solução $v_4 = 38$ no ponto $(x_1, x_2) = (2; 6)$, encontrada para o subproblema P_4 . A solução $v_4 = v^*$ é a solução encontrada pelo algoritmo BB para o problema P no 3º Caso.

4º Caso: escolha de $x_2 \cong 5,5$ como variável de divisão e uso do solucionador IPOPT. Neste caso, o processo desenvolvido pelo algoritmo BB para resolver o problema P gerou a árvore BB mostrada na Figura 24. Observe que o passo *bound* eliminou o subproblema P_2 por possuir uma solução v_2 maior que v_4 e, também, eliminou o subproblema P_3 porque a sua solução foi encontrada numa posição localizada fora da região factível. Assim $v^* = v_4 = 29$ é a solução ótima no ponto $(x_1, x_2) = (5; 4)$ encontrada para o problema P .

FIGURA 24 - Representação da árvore BB após a segunda divisão.

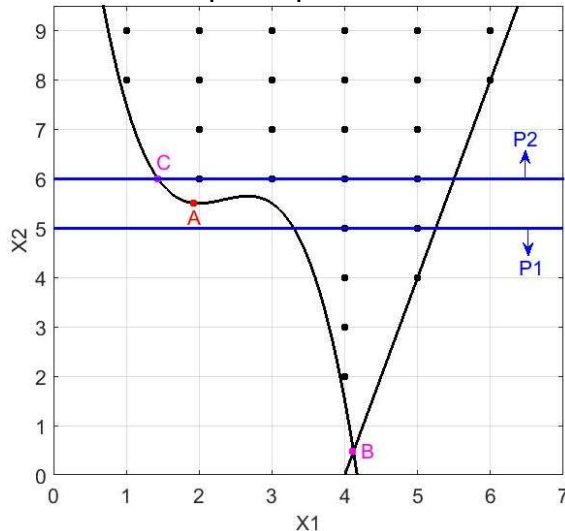


Fonte: Elaboração do próprio autor.

A Figura 25 mostra a representação geométrica da primeira partição na região factível, executada quando o passo *branch* seleccionou x_2 como variável de divisão no subproblema P_0 , gerando os subproblemas P_1 e P_2 . Nesta figura, o ponto A representa a solução $v_0 = 34,9$ no ponto $(x_1, x_2) = (1,9; 5,5)$, encontrada para o subproblema P_0 , o ponto B representa a solução $v_1 = 6,97$ no ponto $(x_1, x_2) = (4,11; 0,47)$ encontrada

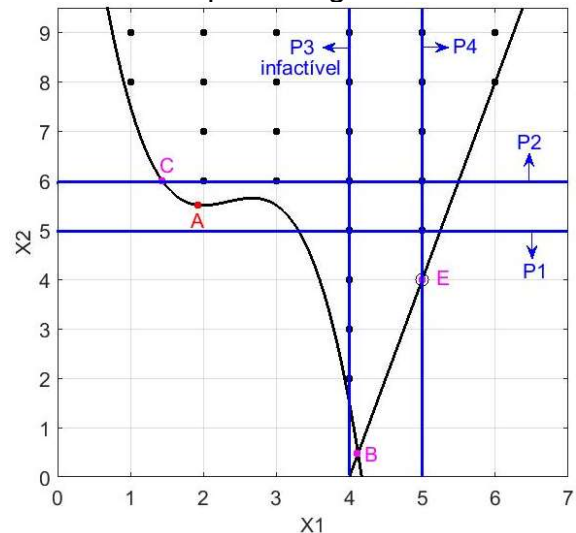
para o subproblema (P_1) e o ponto C representa a solução $v_2 = 37,4$ no ponto $(x_1, x_2) = (1,43; 6)$ encontrada para o subproblema (P_2).

FIGURA 25 - Representação da região de busca após a primeira divisão.



Fonte: Elaboração do próprio autor.

FIGURA 26 - Representação da região de busca após a segunda divisão.



Fonte: Elaboração do próprio autor.

A Figura 26 complementa a Figura 25 com a adição da representação geométrica da segunda partição na região factível, executada quando o passo *branch* selecionou $x_1 = 4,11$ como variável de divisão no subproblema P_1 , gerando os subproblemas P_3 e P_4 . O ponto D representando a solução do subproblema P_3 não é incluído nesta figura porque ele cai numa região infactível e o ponto E representa a solução $v_4 = 29$ no ponto $(x_1, x_2) = (5; 4)$, encontrada para o subproblema P_4 . A solução $v_4 = v^*$ é a solução encontrada pelo algoritmo BB para o problema P no 4º Caso.

Fazendo o fechamento dos quatro casos possíveis de resolução do problema P de PNLIP, com o uso do algoritmo BB definido para resolver problemas de PLIM, a Tabela 1, a seguir, mostra os resultados obtidos com (i) solução do subproblema inicial P_0 obtido relaxando a integralidade de P , (ii) escolha da variável de divisão (passo *branch*) e (iii) solução inteira final do problema P .

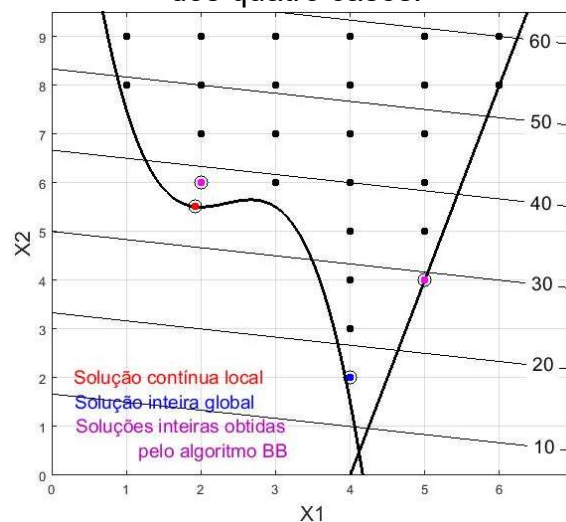
Os resultados obtidos nesses quatro casos mostram que o algoritmo BB não conseguiu obter a solução ótima global $v^* = 16$ no ponto $(x_1, x_2) = (4,2)$. Esses resultados estão mostrados graficamente na Figura 27. As curvas de nível nesta figura mostram claramente a existência de outros pontos ótimos locais de melhor qualidade do que aqueles que foram encontrados com o uso do algoritmo BB.

Tabela 1 - Quatro casos possíveis de resolução do problema de PNLIP com o uso do algoritmo BB definido para resolver problemas de PLIM

	KNITRO		IPOPT	
Solução do P_0	$v^* = 34,913$ $x_1 = 1,92509$ $x_2 = 5,5063$		$v^* = 34,913$ $x_1 = 1,92509$ $x_2 = 5,5063$	
Aplicando branch (e seguindo em profundidade)	Variável de divisão x_1	Variável de divisão x_2	Variável de divisão x_1	Variável de divisão x_2
Solução do P	$v^* = 29$ $x_1 = 5$ $x_2 = 4$	$v^* = 29$ $x_1 = 5$ $x_2 = 4$	$v^* = 38$ $x_1 = 2$ $x_2 = 6$	$v^* = 29$ $x_1 = 5$ $x_2 = 4$

Fonte: Elaboração do próprio autor.

FIGURA 27 - Representação da localização das soluções encontradas para cada um dos quatro casos.



Fonte: Elaboração do próprio autor.

Concluindo, este simples exemplo em duas dimensões mostra que, independentemente do uso do solucionador ou da escolha da variável de divisão, o algoritmo BB desenvolvido para resolver problemas de PLIM não produz o efeito desejado para resolver problemas de PNLIM não convexos.

A seguir, são destacados os problemas que provocam a deficiência do algoritmo.

2.1.3.1 Ineficácia do algoritmo BB em problemas de PNLIM não convexo

A natureza não linear do problema, mesmo com apenas duas variáveis como no exemplo anterior, juntamente com a não convexidade da região de busca, tornam o

algoritmo BB deficiente na forma como está definido para resolver problemas de PLIM. Por isso, uma tentativa de aplicação direta a problemas de PNLIM não convexos apresenta falhas. O algoritmo funciona, mas não alcança êxito.

Quando o modelo matemático é um problema de PNLIM não convexo, vários problemas surgem devido à ausência da linearidade ou da convexidade. As ferramentas de suporte utilizadas na resolução dos subproblemas de PNL dependem muito das características do problema e do tipo de região de busca a ser investigada. A não convexidade da região factível dificulta a convergência dos subproblemas de PNL para encontrar um ótimo global. Além disso, a otimização depende do ponto inicial utilizado e devido ao grande número de pontos ótimos locais nesse tipo de problema, frequentemente chega-se a pontos ótimos locais de baixa qualidade, isto é, ótimos locais que estão longe do ponto ótimo global conhecido. Assim, a necessidade de resolver os subproblemas de PNL em cada nó da árvore de BB é uma condição complicante, pois, de acordo com os testes de sondagem definidos para problemas de PLIM, o surgimento de um ótimo local no decorrer do processo pode induzir a eliminação de um nó importante da árvore e excluir um possível caminho que levaria a um ótimo de melhor qualidade. Essa eliminação equivocada pode ocorrer sempre que o algoritmo encontrar uma solução local de baixa qualidade. Diante do exposto, fica evidente a necessidade de adequar o método do algoritmo para contornar os problemas que surgem. Para isso, é necessário estabelecer estratégias de ação para lidar com temas sobre como:

1. Tratar subproblemas de PNL;
2. Manipular a não convexidade do problema;
3. Definir os testes de sondagem;
4. Detectar a infactibilidade e a convergência dos subproblemas de PNL.

Junto a esses assuntos provenientes da natureza dos problemas do tipo PNLIM, é indispensável analisar estratégias que melhorem a eficácia do algoritmo BB em temas relacionados sobre como:

5. Definir a escolha da variável de divisão para separar os subproblemas;
6. Definir a escolha do próximo subproblema a ser resolvido.

Na próxima seção são apresentadas soluções para cada questionamento levantado nesse tópico.

2.2 ALGORITMO BB PROPOSTO PARA RESOLVER PROBLEMAS DE PNLIM NÃO CONVEXOS

Para o algoritmo BB utilizado na resolução de problemas lineares se tornar eficiente na resolução de problemas de PNLIM é necessário estabelecer uma mudança de abordagem nas estratégias de ação do algoritmo, principalmente, para resolver os subproblemas de PNL que surgem em cada nó da árvore BB. Nesse processo existe a forte presença da não linearidade do problema e a não convexidade da região de busca. Nas próximas seções, são tratadas essas questões.

2.2.1 Como lidar com os subproblemas de PNL

Ao se trabalhar com problemas de PNLIM é necessário estabelecer uma ferramenta de apoio para a resolução dos subproblemas de PNL que surgem em cada nó da árvore BB. Neste trabalho, são utilizados os solucionadores KNITRO (BYRD, NOCEDAL e WALTZ, 2006), IPOPT (WÄCHTER e BIEGLER, 2006), LOQO (VANDERLEI, 1999) e MINOS (HUA ET AL, 2003). A decisão de utilizar esses quatro solucionadores ocorreu após testar alguns solucionadores disponíveis no mercado, sendo que esses foram os mais eficazes. É vantajoso para a pesquisa que haja mais de uma ferramenta para a resolução de cada problema de PNL, isso proporciona mais credibilidade aos resultados, além de uma comparação entre as quatro ferramentas. Nesse contexto, é interessante destacar as características dos quatro solucionadores que são utilizados para a resolução dos subproblemas de PNL.

O KNITRO é um *software* que busca soluções ótimas locais para a otimização em problemas de grande porte de programação não linear. Adicionalmente, apresenta uma opção que é utilizada para a resolução de problemas de PLIM e uma segunda opção para resolver problemas de PNLIM convexos. Neste trabalho, o solucionador KNITRO (i) será utilizado apenas para resolver os subproblemas de PNL e (ii) poderá ser usado casualmente para tentar resolver diretamente um problema de PNLIM não convexo apenas para a avaliação do desempenho do algoritmo BB proposto. O KNITRO oferece várias configurações como opções para abordar um problema de diferentes maneiras. As dificuldades apresentadas por cada tipo específico de problema determinam qual é a melhor forma de uso do solucionador, se o usuário não quiser se

preocupar com isso, existe um comando em que o próprio solucionador escolhe que tipo de ferramenta utilizar durante a busca da solução.

O IPOPT é um *software* de código aberto para a resolução de problemas não lineares, sendo possível otimizar problemas gerais de PNL de grande porte. Ao executar a otimização, ele utiliza um algoritmo de ponto interior primal-dual com um método de filtro de busca linear para assegurar a convergência global. Todavia, o solucionador IPOPT resolve apenas problemas de PNL, não tratando problemas inteiros ou mistos.

O solucionador LOQO é um *software* apropriado para lidar com uma variedade de problemas, incluindo lineares e não lineares, convexos e não convexos, restritos e irrestritos, desde que as funções definidoras sejam suaves (nos pontos avaliados pelo algoritmo). Ao executar a otimização, ele utiliza um algoritmo que se baseia no método de ponto interior aplicado a uma sequência de aproximações quadráticas. Na resolução de problemas não convexos, o solucionador LOQO itera a partir de um ponto de partida especificado e fornece uma solução localmente ótima, enquanto para problemas convexos ele encontra uma solução globalmente ótima.

O solucionador MINOS é um *software* especialmente eficiente na resolução de objetivos não lineares com restrições lineares ou quase lineares. Ele pode resolver problemas de otimização linear e não linear. Ao executar a otimização, ele utiliza um algoritmo que se baseia no método Lagrangiano linearmente restrito para problemas não lineares esparsos de grande porte.

2.2.2 Como tratar a não convexidade do modelo

Um dos fatores complicantes na resolução de problemas de PNLIM é a não convexidade da região factível por possuir muitos pontos ótimos locais. Isso prejudica o funcionamento do algoritmo BB tradicional, porque é possível que o algoritmo encontre um ponto ótimo local de baixa qualidade e, através dos testes de sondagem, sejam eliminados alguns nós promissores da árvore BB.

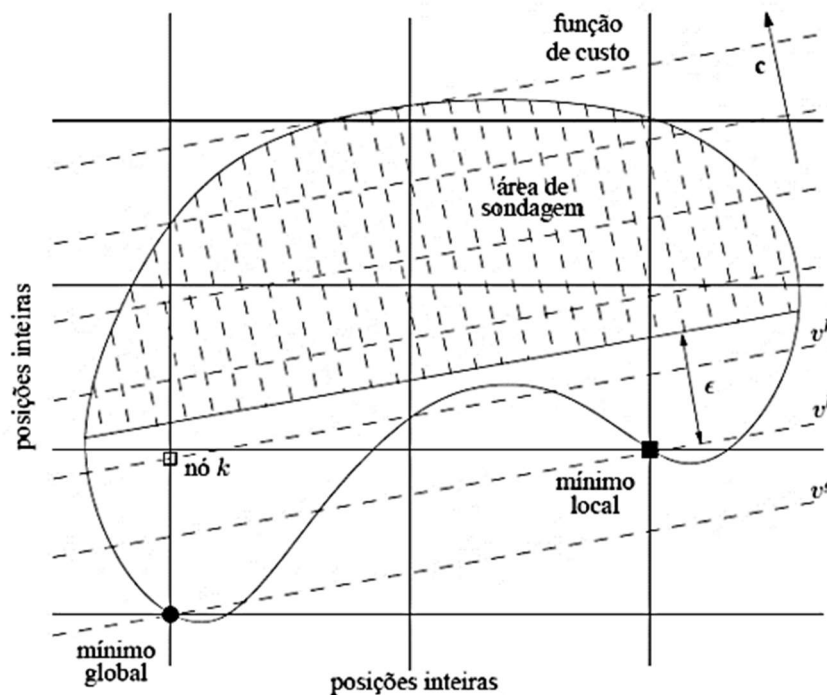
No caso linear onde existe região factível convexa, a função objetivo mantém um crescimento/decrescimento linear permitindo o controle sobre cada solução v_k encontrada na árvore BB. No caso de minimização, sempre a solução v_k do subproblema filho será maior do que a solução v_{k-1} do subproblema pai, ou seja, $v_k > v_{k-1}$. Isso

garante a funcionalidade dos testes de sondagem e a convergência para o ótimo global.

Para problemas de PNL com região de busca não convexa, como no caso do problema analisado no item 2.1.3, não é possível ter o controle de v_k em relação à v_{k-1} : pode ocorrer $v_k > v_{k-1}$ ou $v_k \leq v_{k-1}$. Ou seja, devido à não convexidade, a garantia que $v_k > v_{k-1}$ sempre ocorra é perdida, isso faz o algoritmo trabalhar sem a certeza de convergência. Mesmo quando a convergência ocorre, não é possível garantir que a solução encontrada seja a melhor possível.

Para contornar esse tipo de problema, um fator de segurança $\varepsilon > 0$ é acrescentado ao algoritmo, conforme apresentado em (RIDER, 2006). Uma representação geométrica da funcionalidade do fator de segurança ε é mostrada na Figura 28.

FIGURA 28 - Representação de uma região de busca não convexa.



Fonte: (RIDER, 2006).

Essa figura representa uma situação em que o algoritmo de busca encontra uma solução ótima local v^l . Se não for utilizado o fator de segurança ε , de acordo com o critério de sondagem estabelecido para o algoritmo BB linear usual, o nó k deve ser descartado, pois $v^k > v^l$. Essa eliminação traz como consequência a perda de um nó promissor que levaria ao valor ótimo. Nesse caso, a utilização do fator de segurança,

$\varepsilon > 0$, promove uma mudança na estratégia de sondagem do algoritmo BB, tal que, após um nó ser analisado e apresentar $v^k \geq v^*$, o algoritmo continua a preservar os nós descendentes que possuem uma solução v^k maior que v^* , até um limite $v^* + \varepsilon$. Isso pode evitar a perda de um nó promissor na busca de uma solução melhor. Portanto, é necessário redefinir a regra de sondagem do algoritmo BB, reescrevendo-a da seguinte forma: o subproblema k que tem solução $v^k > v^* + \varepsilon$ será descartado.

Observe que quando $\varepsilon = 0$ a nova regra de sondagem torna-se a mesma usada no algoritmo BB tradicional. Se ε for um número grande, o algoritmo BB terá uma chance maior de encontrar a solução ótima global do problema, mas o esforço computacional exigido também será maior, devido ao aumento no número de nós a serem analisados na árvore BB. Portanto, deve haver um equilíbrio na escolha do valor de ε .

Neste trabalho, para as regras de sondagem, foi definido um fator de segurança com valor $\varepsilon = 5\%$. Este valor de ε foi definido com base em testes experimentais e demonstrou ser conservador sem aumentar muito o tamanho da árvore BB e os tempos computacionais para resolver o problema.

2.2.3 Regras de sondagem

Em relação às regras de sondagem utilizadas no algoritmo BB tradicional para resolver problemas de PLIM, sua aplicação em problemas de PNLIM recebe uma modificação na regra de sondagem número 2 conforme justificativa recentemente apresentada no item 2.2.2. Sendo assim, nos subproblemas de PNL ainda não resolvidos na árvore BB, as regras de sondagem que eliminam a necessidade de analisar os nós descendentes de um nó pai são:

1. Um subproblema é sondado quando a sua solução v_k é infactível;
2. Para problemas de minimização, um subproblema é sondado quando possui uma solução $v_k > v^* + \varepsilon$;
3. Um subproblema é sondado quando o algoritmo encontra uma solução factível para o problema original de PNLIM não convexo.

2.2.4 Como tratar questões de infactibilidade e convergência

As questões relacionadas com a dificuldade em detectar a infactibilidade dos subproblemas de PNL que surgem na árvore BB e a incerteza da convergência do algoritmo para encontrar uma solução que seja a melhor possível, são questões que contribuem para a ineficiência na resolução de problemas de PNLIM não convexos quando é utilizado o algoritmo BB tradicional para problemas de PLIM.

Geralmente, muitos nós da árvore BB apresentam subproblemas infactíveis, principalmente em problemas com dimensões maiores, isso exige um alto esforço computacional dos solucionadores de otimização para detectá-los. A ideia é resolver subproblemas de PNL que sejam sempre factíveis e, para isso, é necessário obter transições mais suaves no espaço de busca do problema. Particularmente, para contornar os casos de infactibilidade dos subproblemas de PNL quando a função objetivo tem característica linear, é acrescentado na função objetivo e nas restrições um conjunto de variáveis artificiais não negativas (q_i) combinadas com um parâmetro de penalização positivo (Λ). O objetivo é inibir as possíveis infactibilidades ao forçar o algoritmo a rejeitar os subproblemas infactíveis zerando as variáveis artificiais. Não há como saber quando um subproblema é infactível, por isso, essa estratégia é utilizada para tentar contornar esse problema. Note que essa proposta garante uma solução factível trivial, com os valores das gerações artificiais complementando os valores das demandas das barras correspondentes.

Existe uma grande possibilidade para o algoritmo BB chegar a uma solução factível com a incorporação do parâmetro de penalização à função objetivo com característica linear, cujo objetivo é forçar o algoritmo a tentar zerar as variáveis artificiais associadas a esse parâmetro, mas para que isso ocorra é necessário definir um valor alto para o parâmetro de penalização. Não há um critério para saber qual é o valor ideal desse parâmetro, o que foi observado nos testes é que:

- i. Um parâmetro fixo pode ser ruim para um determinado sistema e bom para outro;
- ii. Um parâmetro pequeno pode não atingir o objetivo, pois nesse caso, a opção do algoritmo poderá ser a atribuição de valores positivos às variáveis artificiais;
- iii. Um parâmetro muito grande pode prejudicar a convergência, pois o algoritmo tentará a todo custo zerar as variáveis artificiais agregadas ao

parâmetro de penalização, prejudicando a convergência para uma solução de boa qualidade.

Tendo em vista a estruturação de um algoritmo BB para resolver problemas de PNLIM não convexos quando a função objetivo tem característica linear, é mais apropriado que o parâmetro de penalização seja definido em função do problema em questão. Portanto, é adequado que o valor definido para o parâmetro de penalização Λ seja determinado em função dos coeficientes da função objetivo de cada problema. Por exemplo, se $v(\mathbf{x}) = \sum_{j \in I} c_j x_j$, o parâmetro de penalização pode ser definido como $\Lambda := 2\|\mathbf{c}\|$, onde $\|\mathbf{c}\| := \sqrt{\sum_{j \in I} c_j^2}$ é a norma euclidiana dos coeficientes da função objetivo para um conjunto de índices I . Nessa forma, o parâmetro de penalização Λ proporciona maior flexibilidade para resolver diferentes sistemas, preserva a generalidade e retira a rigidez de um valor fixo para todos os problemas.

Em relação à convergência dos subproblemas de PNL, o solucionador é programado para resolver novamente o subproblema de PNL que não convergiu, reiniciando exatamente do ponto onde parou. São consideradas até três tentativas em cada nó. Não obtendo resultado favorável, o subproblema de PNL é considerado como não convergido, e o respectivo nó é descartado. Essa estratégia evita eliminar um nó quando o algoritmo BB não convergiu na primeira tentativa.

2.2.5 Seleção da variável de separação e o próximo subproblema a analisar

Não existem técnicas sistemáticas que identificam qual é a variável ótima de separação ou que determinam qual é o subproblema que conduz mais rápido à solução, mas existem algumas regras empíricas que podem ser utilizadas. Em sistemas onde o número de restrições e variáveis é grande, a forma pela qual são escolhidas as variáveis de separação e os subproblemas candidatos pode afetar completamente o desempenho do algoritmo BB. É muito grande o esforço dedicado na especialização de técnicas que possuam ferramentas mais eficientes para selecionar a variável de separação e o próximo subproblema a analisar.

Entre os vários métodos utilizados na seleção da variável de separação e do próximo subproblema a analisar, alguns estão fundamentados em uma sensibilidade denominada *pseudocusto*, descrita a seguir. O pseudocusto de uma variável inteira x_j com valor atual não inteiro é dado pela relação entre a degradação do valor ótimo da

função objetivo por unidade de variação da variável x_j quando ela é diminuída ou aumentada ao ser forçada a admitir um valor inteiro (GAUTHIER e RIBIERE, 1977; LINDEROTH e SAVELSBERGH, 1999; HAFFNER, 2000). Assim, para uma determinada variável inteira x_j com valor atual não inteiro, são definidos dois pseudocustos, um de redução, denotado por P_j^- , e outro de aumento, denotado por P_j^+ :

$$P_j^- = \frac{v_k^- - v_k}{f_j^k} \quad (7)$$

$$P_j^+ = \frac{v_k^+ - v_k}{(1 - f_j^k)} \quad (8)$$

em que v_k é o valor ótimo do subproblema k ; v_k^- é o valor ótimo do subproblema descendente de k obtido com a redução da variável x_j ; v_k^+ é o valor ótimo do subproblema descendente de k obtido com o aumento da variável x_j ; f_j^k é a parte fracionária de x_j^k definida como $f_j^k = x_j^k - \lfloor x_j^k \rfloor$, sendo $\lfloor x_j^k \rfloor$ o maior valor inteiro contido em x_j^k . Para cada uma das variáveis reais x_j^k de cada nó k da árvore são calculados os pseudocustos de redução P_j^- e de aumento P_j^+ . Para isso, é necessário resolver dois subproblemas, um para encontrar o valor v_k^- , da função objetivo do subproblema descendente de k obtido com a redução da variável x_j e o outro para encontrar o valor v_k^+ , da função objetivo do subproblema descendente de k obtido com o aumento da variável x_j .

Uma maneira de selecionar a variável de separação é através do método da *seleção pela maior degradação no valor da função objetivo*. O critério utilizado nesse método emprega pseudocustos, conforme mostrado em (9), e serve para obter uma estimativa da degradação provocada pela variável j do subproblema candidato do nó k .

$$\max_j \{ \min [P_j^- f_j^k; P_j^+ (1 - f_j^k)] \} \quad (9)$$

A execução desse critério pode ser realizada em três passos: (i) para cada uma das variáveis reais x_j^k de cada nó k da árvore BB, calcular os dois valores dados por $P_j^- f_j^k$ e $P_j^+ (1 - f_j^k)$; (ii) para cada uma das variáveis x_j^k , identificar os menores valores encontrados; (iii) a variável selecionada será aquela que tem o maior valor entre os menores valores encontrados para cada uma das variáveis x_j^k . Então, a variável selecionada será aquela cuja menor degradação no valor da função objetivo é máxima.

Após escolher a variável para separar e criar os novos subproblemas, a seleção do próximo subproblema a analisar, entre todos os candidatos, pode ser realizada através do método de *estimativas do valor da função objetivo* (v_{est}^k). Uma forma de calcular a estimativa v_{est}^k para cada um dos subproblemas candidatos utiliza pseudocustos e é definida em (10).

$$v_{est}^k = v_{inf}^k + \sum_{i \in I} \min[P_i^- \theta_i^k; P_i^+(1 - \theta_i^k)] \quad (10)$$

onde v_{inf}^k é o valor do limite inferior do subproblema candidato k e I é o conjunto das variáveis inteiras x_i^k com valor atual não inteiro do subproblema candidato k . Assim, a estimativa v_{est}^k para cada subproblema candidato é obtida através da soma do limite inferior do candidato com o somatório dos menores valores associados às alterações necessárias para que a variável x_i^k deixe seu valor original para assumir os valores inteiros vizinhos. O valor obtido para v_{est}^k busca estimar a melhor solução inteira que poderia ser obtida entre todos os subproblemas descendentes do candidato k . Então, o próximo subproblema candidato selecionado para analisar será aquele que apresentar a menor estimativa v_{est}^k no valor da função objetivo.

A grande desvantagem do uso das expressões (9) e (10), que empregam o uso dos pseudocustos, é a necessidade de resolver dois problemas de otimização para cada um dos subproblemas candidatos e armazenar as informações obtidas (solução e base ótima, para subproblemas de PL), para dar suporte à solução dos seus futuros descendentes, antes de fazer a seleção da variável de separação ou do próximo subproblema candidato. Em relação aos problemas armazenados, uma parte deles acaba sendo inútil pois, quando o valor da incumbente é atualizado é possível descartar alguns candidatos. Uma outra desvantagem, é que para problemas de PNL com região de busca não convexa não é possível ter o controle de v_k em relação à v_{k-1} : pode ocorrer $v_k > v_{k-1}$ ou $v_k \leq v_{k-1}$. Nesse caso, pode surgir pseudocustos negativos como resultado da não linearidade e da não convexidade do problema, sugerindo que a solução obtida para o subproblema k de PNL é apenas uma solução ótima local.

Alternativamente, é possível utilizar outros métodos para selecionar a variável de divisão e o próximo subproblema candidato a ser analisado. Outra maneira de selecionar a variável de divisão é a partir de uma ordem pré-determinada dos coeficientes da função objetivo, por exemplo: do maior para o menor. Para selecionar o próximo subproblema candidato a ser analisado também podem ser utilizadas as regras FIFO

(*first in first out*) e LIFO (*last in first out*). A regra FIFO consiste em um procedimento que realiza uma busca em largura na árvore BB, ou seja, consiste em tratar a lista dos subproblemas candidatos como sendo uma estrutura em que sempre é selecionado o primeiro nó que foi inserido e a regra LIFO consiste em um procedimento que realiza uma busca em profundidade na árvore BB, ou seja, consiste em tratar a lista dos subproblemas candidatos como sendo uma estrutura em que sempre é selecionado o último nó que foi inserido. A utilização desses métodos minimiza as necessidades de memória para armazenar as informações dos subproblemas candidatos.

Para evitar os inconvenientes relacionados ao cálculo dos pseudocustos, o algoritmo BB proposto emprega como estratégia de seleção da variável de divisão, os valores (do maior para o menor) dos coeficientes da função objetivo que são definidos no início do algoritmo e não mudam. Uma outra estratégia que o algoritmo BB proposto utiliza é a realização de uma busca em largura (regra FIFO) para determinar o próximo subproblema a ser analisado.

A seção a seguir, apresenta as estratégias propostas neste trabalho para obter uma versão do algoritmo BB que seja eficiente na resolução de problemas de PNLIM não convexos.

2.2.6 Estrutura do algoritmo BB proposto para resolver problemas de PNLIM

Com motivação baseada na deficiência do algoritmo BB tradicional ao lidar com problemas onde espaço de busca é não convexo, identificada ao analisar o estudo de caso apresentado na Seção 2.1.3, aqui é proposto um algoritmo BB adaptado para obter uma estrutura de operação que é capacitada para resolver problemas de PNLIM onde a região de busca pode ser não convexa. As estratégias de ação desse algoritmo proposto já foram discutidas em seções anteriores, mas é interessante que elas sejam pontuadas novamente de forma resumida.

Para a resolução dos subproblemas de PNL que surgem em cada nó da árvore BB são utilizados os solucionadores KNITRO, IPOPT, LOQO e MINOS, porque isso proporciona mais credibilidade aos resultados e permite uma comparação entre essas ferramentas de apoio. A não convexidade da região de busca faz o algoritmo trabalhar sem a certeza da convergência para uma solução ótima global e, para contornar esse problema, um fator de segurança $\varepsilon > 0$ é acrescentado ao algoritmo, tal que, o algoritmo continua a preservar os nós descendentes que possuem uma solução v^k maior

que v^* até um limite $v^* + \varepsilon$, isso pode evitar a perda de um nó promissor na busca de uma solução melhor. Para contornar os casos de infactibilidade dos subproblemas de PNL, o ajuste proposto é acrescentar na função objetivo com característica linear e nas restrições do problema de PNLIM original um conjunto de variáveis artificiais não negativas (q_i) combinadas com um parâmetro de penalização positivo (Λ), isso força o algoritmo a rejeitar os subproblemas infactíveis zerando as variáveis artificiais. Uma medida que atua em auxílio à convergência dos subproblemas de PNL é o solucionador ser programado para resolver novamente o subproblema de PNL que não convergiu, após três tentativas o respectivo nó é descartado, isso evita eliminar um nó quando o algoritmo não convergiu na primeira tentativa. Para evitar os inconvenientes relacionados ao cálculo e atualização dos pseudocustos, o algoritmo emprega como estratégia de seleção da variável de divisão os valores (do maior para o menor) dos coeficientes da função objetivo, que são definidos no início do algoritmo e não mudam. Também evitando o uso dos pseudocustos, o algoritmo realiza uma busca em largura (regra FIFO) para determinar o próximo subproblema a ser analisado.

Essas estratégias de ação procuram evitar a exigência de um alto esforço computacional por parte dos solucionadores de otimização. Isso foi pensado principalmente nos casos relacionados com a detecção da infactibilidade de um subproblema de PNL e no uso de pseudocustos para selecionar a variável de divisão ou o próximo subproblema candidato a analisar. Diante das condições pré-estabelecidas, a estrutura do algoritmo BB proposto para resolver problemas de PNLIM não convexos é formada pelos seguintes passos.

1. Inicialização: Faça o valor da solução incumbente $v^* \leftarrow \infty$; inicialize o contador de subproblemas $k \leftarrow 0$; inicialize a lista L de subproblemas candidatos (P_k) com a solução v_0 do problema de PNL correspondente P_0 . Se a solução de P_0 for inteira, então seja $v^* \leftarrow v_0$ e remova P_0 de L . Vá para o Passo 2.
2. Teste de convergência: Verifique a lista L ; se $L = \emptyset$, então o processo terminou e a solução incumbente atual v^* é a solução ótima para o problema PNLIM original. Caso contrário, vá para o Passo 3.
3. Seleção do subproblema candidato: Entre os subproblemas candidatos P_k pertencentes à lista L , selecione o primeiro que foi inserido nessa lista; resolva e armazene a sua solução ótima v_k . Vá para o Passo 4.

4. Passo *bound*: Um subproblema candidato P_k é sondado quando satisfaz uma das seguintes condições:

- (a) Se a solução v_k do subproblema P_k é infactível.
- (b) Se a solução v_k for maior que a atual solução incumbente v^* somada com o fator de segurança ε , ou seja, se $v_k > v^* + \varepsilon$.
- (c) Se a solução v_k do subproblema P_k também for uma solução para o problema de PNLIM original, ou seja, se as condições de integralidade forem atendidas. No caso de $v_k < v^*$, então a solução incumbente é atualizada, $v^* \leftarrow v_k$, e o teste 4(b) é aplicado novamente, ou seja, a lista L deve ser revisada para que todos os nós que apresentem $v_k \geq v^* + \varepsilon$ sejam sondados e k deve ser atualizado.

Após esse passo, se o subproblema candidato P_k foi sondado, retorne ao Passo 2; caso contrário, vá para o Passo 5.

5. Passo *branch*: Entre as variáveis inteiras x_j com valor atual não inteiro da solução do subproblema P_k , selecione aquela cujo coeficiente na função objetivo tem o maior valor para realizar a separação e adicione os dois novos subproblemas descendentes gerados à lista de candidatos L . Os dois novos subproblemas são gerados a partir do subproblema pai (P_k), um com a restrição $x_j \leq \lfloor x_j \rfloor$ incluída no conjunto de restrições do subproblema pai (P_k) e o outro com a restrição $x_j \geq \lfloor x_j \rfloor + 1$ incluída no conjunto de restrições do subproblema pai (P_k), onde $\lfloor x_j \rfloor$ é o maior inteiro menor que x_j . Após adicionar esses dois novos subproblemas à lista L , faça $k \leftarrow k + 2$ e retorne ao Passo 3.

Observação: Esse processo iterativo se encerra quando não há mais subproblemas para sondar ou quando a solução do problema original P é encontrada (caso exista). O algoritmo BB proposto é testado no problema do planejamento da expansão de redes de transmissão de energia elétrica.

3 PLANEJAMENTO DA EXPANSÃO DE REDES DE TRANSMISSÃO

3.1 INTRODUÇÃO

O tema relacionado com o planejamento da expansão da rede de transmissão (PERT) de energia elétrica é bastante conhecido por engenheiros, pesquisadores e especialistas em planejamento das empresas de energia elétrica em diferentes países. Ele se apresenta como um problema que é muito estudado e amplamente analisado sob diversos aspectos. A interpretação dos resultados obtidos na resolução deste problema possibilita a definição de estratégias que servem para direcionar as ações que são exigidas pelo sistema elétrico no futuro. Tais estratégias são de grande importância do ponto de vista econômico e computacional. O problema real apresenta uma rede de transmissão de um sistema elétrico com uma topologia atual, onde deve ser determinado um planejamento de expansão ótimo com a finalidade de atender um aumento esperado da demanda de carga dentro de um intervalo de tempo pré-definido de curto ou longo prazo, isto é, a sua solução tem como objetivo informar onde, quantas e quando novas linhas de transmissão devem ser construídas a um custo mínimo, garantindo a operação do sistema sem que haja cortes de carga (RIDER, 2006; MAHDAVI, ANTUNEZ, *et al.*, 2018; HEMMATI, HOOSHMAND e KHODABAKHSHIAN, 2013; UDE, YSKANDAR e GRAHAM, 2019). Este problema pode ser abordado considerando apenas um único estágio onde todos os investimentos são realizados em uma única etapa, chamado planejamento estático (FREITAS, MACEDO e ROMERO, 2019), ou quando o planejamento é dividido em vários estágios, designado planejamento multiestágio (FREITAS, MACEDO e ROMERO, 2020), (FREITAS, MACEDO e ROMERO, 2018).

O problema de PERT é um problema de otimização e, como todo problema de otimização matemática, sua solução é obtida através da implementação de dois processos consecutivos: a modelagem matemática e a técnica de solução. Em relação à modelagem matemática do problema de PERT, uma discussão é realizada na Seção 3.2. Quanto às técnicas de solução (SILVA, 2013), existem vários métodos que são usados e eles podem ser classificadas em dois grandes grupos que são os métodos exatos (técnicas exatas) e os métodos aproximados (técnicas heurísticas e metaheurísticas).

Entre os métodos que utilizam as técnicas exatas de otimização (VARGAS, 2018), podemos encontrar trabalhos que usam PLIM, PNLIM, decomposição de Benders e algoritmo BB. Essas técnicas fornecem resultados de boa qualidade. Como o problema de PERT possui variáveis inteiras é necessária a utilização de técnicas de programação inteira. Além disso, o problema de PERT tem a característica de apresentar um elevado número de soluções ótimas locais, como consequência isso exige um alto esforço computacional dos solucionadores (*solvers*) para obter a solução ótima global (ORIONDO, 2020), (MENESES, NASCIMENTO, *et al.*, 2020).

Uma estratégia utilizada para resolver o problema de PERT é através de uma formulação disjuntiva que converte o problema de PNLIM não convexo do modelo CC em um problema de PLIM que encontra a mesma solução original do problema. O modelo disjuntivo foi utilizado em (BAHIENSE, OLIVEIRA, *et al.*, 2001) para resolver o problema de PERT sem considerar as perdas de transmissão. A vantagem dessa formulação é que um problema de PLIM é mais fácil de resolver. No entanto, o modelo resultante encontra dificuldades para resolver sistemas de grande porte, pois a abordagem utilizada nessa formulação resulta em um maior espaço de busca e, também, promove um aumento significativo no número de restrições e variáveis contínuas e discretas.

Em (ALGUACIL, MOTTO e CONEJO, 2003) o modelo disjuntivo foi formulado com uma linearização por partes para as perdas. O modelo CC não considera as perdas de transmissão que podem não ser desprezíveis (ALGUACIL, MOTTO e CONEJO, 2003), por isso, é muito importante que elas sejam contabilizadas na formulação do problema de PERT. Outras aplicações do algoritmo BB, onde é usado o modelo linear disjuntivo para resolver o problema de PERT, podem ser encontradas nas referências (ASADA, CARRENO, *et al.*, 2005; ALGUACIL, ARROYO e CARRIÓN, 2010; GARCÉS, CONEJO, *et al.*, 2009).

Também tem sido utilizada a estratégia da decomposição matemática para resolver o problema de PERT. Uma estratégia de decomposição de Benders aplicada na resolução do modelo disjuntivo do problema de PERT é apresentada na referência (BINATO, PEREIRA e GRANVILLE, 2001). Uma estrutura de decomposição considerando geração renovável e cenários de contingência para o problema de PERT é proposta em (MAJIDI-QADIKOLAI e BALDICK, 2017). Na referência (HONG, CHENG e ZENG, 2017) é utilizada a decomposição de Benders para a expansão da geração do

problema de PERT considerando investimentos simultâneos em contingência e incerteza de carga.

Outra maneira de resolver o problema de PERT, que também é considerada neste trabalho, é usar um algoritmo BB para resolver o problema diretamente. As referências (HAFFNER, MONTICELLI, *et al.*, 2000; HAFFNER, MONTICELLI, *et al.*, 2001) aplicam o algoritmo BB para resolver o modelo de transportes do problema de PERT. Em (RIDER, GARCIA e ROMERO, 2008) é proposto um algoritmo BB usando o modelo CC com um método de pontos interiores para resolver o problema de PERT. Em oposição com este trabalho, a metodologia proposta na referência (RIDER, GARCIA e ROMERO, 2008) inviabiliza a escolha de diferentes solucionadores dos subproblemas de PNL que surgem durante o desenvolvimento do procedimento BB, pois utiliza um algoritmo de otimização baseado no método de ponto interior especializado para o problema. Ainda em comparação com a referência (RIDER, GARCIA e ROMERO, 2008), este trabalho apresenta outras melhorias, principalmente aquelas relacionadas com assuntos que tratam sobre questões de infactibilidade e de convergência. Em (DELGADO, POURAKBARI-KASMAEI e RIDER, 2013) é proposto um algoritmo BB usando o modelo CC para resolver o problema de PERT sem considerar as perdas que ocorrem na transmissão.

Também para resolver o problema de PERT, vários métodos aproximados de otimização (heurísticas e metaheurísticas) têm sido utilizados (REEVES, GLOVER e KOCHENBERGER, 2003). Entre eles, podemos encontrar trabalhos que usam o algoritmo de otimização de enxame de partículas (MURUGAN, 2012; DA ROCHA e SARAIVA, 2013), o algoritmo de busca harmônica (VERMA, PANIGRAHI e BIJWE, 2010; RASTGOU e MOSHTAGH, 2014), o algoritmo de colônia artificial de abelhas (MAHDAVI, ALHELOU, *et al.*, 2021), o algoritmo de partida múltipla com religação de caminho (SILVA, FARIA, *et al.*, 2021) e otimização evolutiva (GUPTA, KHOSRAVY, *et al.*, 2018). Os métodos aproximados têm como fundamento o raciocínio de que é possível obter uma solução de boa qualidade através de um processo iterativo de exploração na região factível de busca com a aceitação de algum critério que seja aplicado ao problema. São robustos, adaptáveis e de fácil implementação, por isso, são uma boa opção para abordar problemas complexos. Mas possuem algumas desvantagens, tais como: exigência de uma alta demanda computacional, ajuste de parâmetros e estabelecimento de uma condição de parada. Além disso, a característica multimodal (elevado número de ótimos locais) do problema de PERT leva a maioria dos métodos

aproximados a não fornecer a garantia de uma convergência para a solução ótima global e, em certas ocasiões, a solução ótima local é de baixa qualidade (ORIONDO, 2020; REEVES, GLOVER e KOCHENBERGER, 2003).

Uma proposta de utilização do modelo de operação CA no problema de PERT é apresentada em (RIDER, GARCIA e ROMERO, 2007). A resolução deste problema pode ser obtida principalmente através da utilização de heurísticas e metaheurísticas (GOMES e SARAIVA, 2020) e formulações convexas ou linearizadas (LUBURIĆ, PANDŽIĆ e CARRIÓN, 2020). A utilização da operação CA no problema de PERT ainda é difícil, porque as metaheurísticas não garantem que uma solução ótima possa ser encontrada e as formulações convexas ou linearizadas podem produzir resultados inviáveis.

3.2 MODELOS CLÁSSICOS DE PERT

O estado de operação de uma rede de transmissão de energia elétrica é descrito através de um conjunto de relações matemáticas que definem uma teoria denominada *fluxo de carga CA*, por isso, a modelagem matemática ideal para representar o problema de PERT teria que utilizar o modelo de fluxo de carga CA (ROMERO, 1999). No entanto, desde a introdução do problema de PERT por Garver em 1970 (GARVER, 1970), este modelo não é frequentemente utilizado devido à complexidade das suas relações matemáticas e, por isso, tem sido substituído por modelos matemáticos (SILVA, 2013; VARGAS, 2018; ORIONDO, 2020) que são versões relaxadas do modelo CA (MACEDO, MONTES, *et al.*, 2016), (MONTES, MACEDO, *et al.*, 2016), dentre os quais destacam-se: modelo de transportes, modelos híbridos linear (SCHIAVINATO, MACEDO e ROMERO, 2022) e não-linear, modelo de fluxo de carga CC (ou modelo CC) e, também, um modelo alternativo ao modelo CC que é chamado de modelo disjuntivo. Atualmente, muitos pesquisadores consideram que o modelo CC é o modelo ideal para representar o problema de PERT (ROMERO, 1999), e os principais motivos são:

- i. Geralmente, a topologia base do sistema elétrico é não conexa, isto é, existem conjuntos de barras isoladas da configuração principal. É difícil resolver este tipo de problema usando o modelo CA.
- ii. O planejamento do fornecimento da potência reativa é um problema de difícil convergência no modelo CA.

- iii. Em relação à distribuição dos fluxos de potência ativa, numerosos testes experimentais mostraram que os resultados obtidos usando o modelo CC apresentam resultados muito próximos aos encontrados quando o modelo de fluxo de carga CA é usado (RIDER, 2006).
- iv. Vários métodos de otimização que usam o modelo CC foram desenvolvidos de maneira adequada para resolver o problema de PERT (ROMERO, 1999; LATORRE, CRUZ, *et al.*, 2003; LEE, NG, *et al.*, 2006).

Não existem muitas publicações na literatura especializada em que seja usado o modelo CA. Isso ocorre, principalmente, devido às dificuldades encontradas em realizar o planejamento de reativos de forma simultânea com a expansão do sistema de transmissão e, também, devido ao fato de a topologia base ser desconexa e, às vezes, altamente ilhada (RIDER, 2006).

A modelagem matemática utilizada no modelo CC considera as duas leis de Kirchhoff, como consequência, o problema de PERT se torna um problema de PNLIM com região de busca não convexa. Os modelos de transportes e híbridos são versões relaxadas do modelo CC. A primeira lei de Kirchhoff (ou lei das correntes) aplicada no problema de PERT sem considerar as perdas de potência que ocorrem na transmissão, é baseada no princípio da conservação de carga, isto é, o somatório dos fluxos de potência ativa que entram numa barra do sistema deve ser igual ao somatório dos fluxos de potência ativa que saem dessa mesma barra. A aplicação da primeira lei de Kirchhoff considera que as unidades geradoras de potência ativa e os circuitos de transmissão devem operar dentro de limites que são especificados pelo problema. Para cumprir a segunda lei de Kirchhoff (ou lei das tensões), é necessário que cada laço do sistema respeite a condição de que $\sum_{i \in \Omega_{l_c}} (\theta_i - \theta_j) = 0$, onde Ω_{l_c} é o conjunto das barras terminais de cada caminho que pertence ao laço e $\theta_i - \theta_j$ é a defasagem angular entre as tensões das duas barras terminais de cada caminho ij pertencente ao respectivo laço. Um laço é um conjunto de caminhos que forma um percurso onde o ponto de partida coincide com o ponto de chegada, ou seja, é um percurso fechado.

No geral, a modelagem matemática do problema de otimização de PERT contempla uma função objetivo e um conjunto de restrições. O problema utiliza a função objetivo para buscar uma solução com o menor valor possível para o custo de investimento que será utilizado na construção de novos circuitos (linhas e/ou transformadores) de transmissão, no entanto, a busca da solução deve respeitar um conjunto de

restrições que é principalmente formado pelas equações básicas do fluxo de potência dos circuitos de transmissão e pelo número máximo de novos circuitos que são permitidos incluir na expansão da rede. A modelagem matemática não precisa ter uma estrutura rígida, dependendo da técnica de solução utilizada podem ocorrer a inclusão de incertezas em algum dos parâmetros e/ou a utilização de variáveis artificiais, entre outras. As relações matemáticas do modelo não devem atingir um nível de complexidade tal que não seja possível encontrar uma solução válida para o problema da vida real ou que dificulte a resolução do problema num tempo computacional aceitável (ORIONDO, 2020). No futuro, com a evolução para computadores mais rápidos e a apresentação de novas técnicas de solução, os modelos matemáticos atualmente julgados como complexos e de difícil solução, poderão ser restabelecidos e ajustados para resolver o problema de PERT (SILVA, 2013).

3.2.1 Modelo de transportes

O modelo de transportes foi proposto por GARVER em 1970 (GARVER, 1970; GALLEGO, ROMERO e MONTICELLI, 2000) e representou o início de uma sistematização de modelagens matemáticas utilizadas no problema de PERT. Esse modelo inclui na sua formulação matemática a primeira lei de Kirchhoff (lei das correntes) e ignora a segunda lei de Kirchhoff (lei das tensões). Nesse contexto, a formulação matemática do modelo de transportes assume a forma:

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij} \quad (11)$$

sujeito a:

$$\sum_{ji \in \Omega_c} f_{ji} - \sum_{ij \in \Omega_c} f_{ij} + g_i = D_i \quad \forall i \in \Omega_b \quad (12)$$

$$|f_{ij}| \leq (N_{ij}^o + n_{ij}) \bar{F}_{ij} \quad \forall ij \in \Omega_c \quad (13)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (14)$$

$$0 \leq n_{ij} \leq \bar{N}_{ij} \quad \forall ij \in \Omega_c \quad (15)$$

$$n_{ij} \text{ inteiro} \quad \forall ij \in \Omega_c \quad (16)$$

sendo que Ω_b é o conjunto de todas as barras do sistema; Ω_c é o conjunto de todos os caminhos ij de transmissão formados entre duas barras terminais i e j ; v é o custo total de investimento devido à construção de novos circuitos (linha e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; n_{ij} é o número de novos circuitos adicionados no caminho ij que pode ou não existir na configuração base; $\bar{N}_{ij}$ é o número máximo de novos circuitos que podem ser construídos no caminho ij ; N_{ij}^0 é o número de circuitos existentes no caminho ij da configuração base (ou inicial); f_{ij} é o fluxo total de potência ativa no caminho ij pertencente ou não à configuração base; $\bar{F}_{ij}$ é o fluxo máximo de potência ativa permitido para cada circuito do caminho ij ; g_i é o nível de geração de potência ativa na barra de geração i ; $\bar{G}_i$ é a máxima capacidade do gerador (de potência ativa) na barra i ; $\underline{G}_i$ é o limite mínimo de geração na barra i ; D_i é o nível de demanda na barra de carga i .

A função v em (11) representa a função objetivo que o problema deve minimizar, a finalidade é encontrar uma solução que forneça o menor valor possível de investimento para adicionar novos circuitos de transmissão na expansão da rede. A restrição (12) representa as equações da primeira lei de Kirchhoff, sendo uma para cada barra do sistema, onde f_{ji} é o fluxo de potência ativa entrando na barra i e f_{ij} é o fluxo de potência ativa saindo da barra i . A restrição (13) representa os limites de capacidade (fluxo total de potência ativa) de transmissão em cada caminho do sistema, sendo que os fluxos f_{ij} podem fluir nos dois sentidos (positivos ou negativos), por isso, a necessidade de incluir o valor absoluto nela. A restrição (14) representa o limite de geração de potência ativa nas barras que têm geradores conectados. A restrição (15) representa o número máximo de circuitos de transmissão que podem ser construídos em cada caminho ij . A restrição (16) define a integralidade da variável n_{ij} , ou seja, n_{ij} deve ser um número inteiro. esta restrição representa a maior fonte de complexidade do problema, principalmente para redes de transmissão de grande porte.

No modelo de transportes, a relação linear existente entre as variáveis e a integralidade da variável n_{ij} transforma o problema de PERT em um problema de PLIM, que é mais fácil de resolver do que um problema de PNLIM formulado no modelo CC. Além disso, a grande vantagem do modelo de transportes é que ele fornece uma

excelente flexibilidade para trabalhar com sistemas não conexos e/ou altamente ilhados. No entanto, a omissão da lei das tensões de Kirchhoff em sua formulação matemática pode, em algumas situações, fazer esse modelo encontrar uma solução ótima muito distante da correspondente solução ótima obtida pelo modelo CC (ROMERO, 1999; VARGAS, 2018; ORIONDO, 2020; SILVA, FARIA, *et al.*, 2021).

3.2.2 Modelos híbridos

Os modelos híbridos foram criados para melhorar os resultados alcançados pelo modelo de transportes e contornar as dificuldades do modelo CC. São utilizadas duas versões de modelos híbridos: a não linear e a linear.

3.2.2.1 *Modelo híbrido não linear*

Em (ORIONDO, 2020) o modelo híbrido não linear é caracterizado considerando que o número total de circuitos da rede de transmissão é dividido em três parcelas destacadas a seguir: N_{ij}^0 é o número de circuitos da configuração base existentes no caminho ij , n_{ij}^1 é o número de novos circuitos adicionados no caminho ij existente na configuração base e n_{ij}^2 é o número de novos circuitos adicionados em um novo caminho ij não existente na configuração base. Os circuitos contidos nas parcelas N_{ij}^0 e n_{ij}^1 devem satisfazer às duas leis de Kirchhoff e os circuitos contidos na parcela n_{ij}^2 devem satisfazer apenas a primeira lei de Kirchhoff.

Uma forma equivalente à apresentação das características do modelo híbrido não linear é dada em (ROMERO, 1999), onde consta o seguinte: (i) todas as barras do sistema precisam cumprir a primeira lei de Kirchhoff e (ii) somente os laços que inicialmente se encontravam na configuração base precisam cumprir a segunda lei de Kirchhoff. Os novos laços que possivelmente podem surgir, como resultado da adição de um novo circuito pertencente à parcela n_{ij}^2 , não precisam cumprir a segunda lei de Kirchhoff.

Conforme mostrado em (SILVA, 2013; ORIONDO, 2020; MIYASAKI, 2006), a formulação matemática do modelo híbrido não linear é a seguinte:

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij}^1 + \sum_{ij \in \Omega_c} C_{ij} n_{ij}^2 \quad (17)$$

sujeito a:

$$\sum_{ji \in \Omega_c} f_{ji}^2 - \sum_{ij \in \Omega_c} f_{ij}^2 + \sum_{ji \in \Omega_c} f_{ji}^1 - \sum_{ij \in \Omega_c} f_{ij}^1 + g_i = D_i \quad \forall i \in \Omega_b \quad (18)$$

$$f_{ij}^1 = (N_{ij}^0 + n_{ij}^1) (\theta_i - \theta_j) / X_{ij} \quad \forall ij \in \Omega_c^0 \quad (19)$$

$$|f_{ij}^1| \leq (n_{ij}^1 + N_{ij}^0) \bar{F}_{ij} \quad \forall ij \in \Omega_c^0 \quad (20)$$

$$|f_{ij}^2| \leq n_{ij}^2 \bar{F}_{ij} \quad \forall ij \in \Omega_c' \quad (21)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (22)$$

$$0 \leq n_{ij}^1 \leq \bar{N}_{ij}^1 \quad \forall ij \in \Omega_c^0 \quad (23)$$

$$0 \leq n_{ij}^2 \leq \bar{N}_{ij}^2 \quad \forall ij \in \Omega_c' \quad (24)$$

$$n_{ij}^1 \text{ inteiro} \quad \forall ij \in \Omega_c^0 \quad (25)$$

$$n_{ij}^2 \text{ inteiro} \quad \forall ij \in \Omega_c' \quad (26)$$

$$\theta_i \text{ irrestrito} \quad \forall i \in \Omega_b, i \neq ref \quad (27)$$

$$\theta_{ref} = 0 \quad (28)$$

sendo que Ω_b é o conjunto de barras do sistema; Ω_c^0 é o conjunto de caminhos ij com circuitos existentes na configuração base; Ω_c' é o conjunto de caminhos ij sem circuitos existentes na configuração base; $\Omega_c = \Omega_c^0 \cup \Omega_c'$ é o conjunto de todos os caminhos ij do sistema; v é o custo total de investimento devido à construção de novos circuitos (linhas e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; n_{ij}^1 é o número de novos circuitos adicionados no caminho ij existente na configuração base; n_{ij}^2 é o número de novos circuitos adicionados em um novo caminho ij não existente na configuração base; $\bar{N}_{ij}^1$ é o número máximo de novos circuitos que podem ser construídos no caminho ij da configuração base; $\bar{N}_{ij}^2$ é o número máximo de novos circuitos que podem ser construídos em um novo caminho ij que não pertence à configuração base; N_{ij}^0 é o número de circuitos da configuração base existentes no caminho ij ; f_{ij}^1 é o fluxo total de potência ativa no

caminho ij da configuração base; f_{ij}^2 é o fluxo total de potência ativa no caminho ij que não pertence à configuração base; $\bar{F}_{ij}$ é o fluxo máximo de potência ativa permitido para cada circuito do caminho ij ; X_{ij} é a reatância de um circuito no caminho ij ; g_i é o nível de geração de potência ativa na barra de geração i ; $\bar{G}_i$ é a máxima geração de potência ativa na barra i ; $\underline{G}_i$ é o limite mínimo de geração na barra i ; D_i é o nível de demanda na barra de carga i ; θ_i é o ângulo de fase da tensão na barra i ; θ_{ref} é a referência angular para o conjunto de barras.

Comparando com o modelo de transportes, o modelo híbrido não linear acrescenta uma modificação na aplicação da primeira lei de Kirchhoff conforme mostrado na restrição (18), e introduz as restrições (19), (20), (21) e (28) no problema.

Com relação à restrição (18), a estrutura da aplicação da primeira lei de Kirchhoff continua a mesma. As restrições (20) e (21) estão contempladas no modelo de Transportes, no entanto, aqui elas estão divididas em duas partes.

As restrições (19) e (28) estão relacionadas com a aplicação sistemática da segunda lei de Kirchhoff aos caminhos com linhas existentes. A restrição (19), definida pela equação $f_{ij}^1 = (N_{ij}^o + n_{ij}^1)(\theta_i - \theta_j)/X_{ij}$, é uma expressão da lei Ohm que relaciona os fluxos de potência ativa nos circuitos como função dos ângulos de fase das tensões de suas barras terminais. O termo não linear, $n_{ij}^1(\theta_i - \theta_j)$ e a integralidade das variáveis n_{ij}^1 e n_{ij}^2 representam as condições complicantes que transformam o modelo híbrido não linear em um problema de PNLIM, produzindo um nível de complexidade maior em relação ao modelo de transportes que é um problema de PLIM. A restrição (28) define uma referência angular para o sistema.

O modelo híbrido não linear trabalha de forma eficiente com as partes não conexas do sistema (ROMERO, 1999). Isso ocorre porque quando um novo circuito é instalado em um novo caminho que forma um novo laço que não existia na configuração base, o fluxo de potência ativa desse novo circuito é limitado apenas pela máxima capacidade de transmissão definida na restrição $|f_{ij}| \leq n_{ij} \bar{F}_{ij}$, sendo que, nesse caso, o mesmo não ocorre com o modelo CC, porque o fluxo de potência ativa do citado novo circuito também fica limitado pela máxima abertura angular definida na restrição $f_{ij} = n_{ij}(\theta_i - \theta_j)/X_{ij}$, que também é imposta pela segunda lei de Kirchhoff aos novos laços que não existiam na configuração base (ROMERO, 1999). Dessa forma, o modelo CC perde eficiência em trabalhar com sistemas não conexos, devido

a dependência que é adquirida ao ter que manipular a abertura angular entre as duas barras terminais do novo circuito.

Uma breve comparação entre as eficiências dos modelos de transporte, híbrido e CC para resolver o problema de PERT é feita a seguir. O modelo de transportes encontra soluções que é possível se localizar bastante distanciadas da solução ótima do modelo CC, mas, é muito mais fácil de resolver. O modelo híbrido encontra soluções que se localizam mais perto da solução ótima do modelo CC, porém, a sua resolução apresenta um nível de maior dificuldade. Ambos os modelos de transporte e híbrido apresentam bom desempenho em trabalhar com sistemas não conexos, enquanto o modelo CC pode apresentar sérias dificuldades. Esse tema está aberto a discussões sobre qual é o modelo mais adequado para utilização nas estratégias de obtenção de uma solução para o problema de PERT, no entanto, é evidente que os modelos relaxados têm um grande potencial para serem resolvidos com maior facilidade. Uma discussão mais detalhada sobre esse assunto pode ser encontrada em (ROMERO, 1999).

3.2.2.2 *Modelo híbrido linear*

O modelo híbrido não linear é pouco utilizado, pois apresenta dificuldades semelhantes às do modelo CC, nesse caso é preferível utilizar diretamente o modelo CC. Para contornar essa dificuldade, uma relaxação da não linearidade é aplicada no modelo híbrido não linear, com isso, uma nova formulação matemática denominada *modelo híbrido linear* é obtida. Como consequência, o problema de PERT se transforma em um problema de PLIM. As técnicas de solução usadas para resolver o modelo híbrido linear são as mesmas que as do modelo de transportes, assim sua aplicação se torna mais adequada porque ele oferece uma representação mais completa para o problema de PERT.

No modelo híbrido linear, todas as barras do sistema devem cumprir a primeira lei de Kirchhoff, em outras palavras, todos os novos circuitos adicionados em caminhos pertencentes ou não à configuração base devem satisfazer somente a primeira lei de Kirchhoff e, devem cumprir a segunda lei de Kirchhoff somente os laços existentes na configuração base considerando apenas os circuitos existentes neles, ou seja, todos os novos circuitos adicionados em laços existentes ou não existentes na configuração base não precisam cumprir a segunda lei de Kirchhoff (ROMERO, 1999;

SILVA, 2013; ORIONDO, 2020; MIASAKI, 2006). A formulação matemática assume a seguinte forma:

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij} \quad (29)$$

sujeito a:

$$\sum_{ji \in \Omega_c} f_{ji} - \sum_{ij \in \Omega_c} f_{ij} + \sum_{ji \in \Omega_c} f_{ji}^0 - \sum_{ij \in \Omega_c} f_{ij}^0 + g_i = D_i \quad \forall i \in \Omega_b \quad (30)$$

$$f_{ij}^0 = N_{ij}^0 (\theta_i - \theta_j) / X_{ij} \quad \forall ij \in \Omega_c^0 \quad (31)$$

$$|f_{ij}^0| \leq N_{ij}^0 \bar{F}_{ij} \quad \forall ij \in \Omega_c^0 \quad (32)$$

$$|f_{ij}| \leq n_{ij} \bar{F}_{ij} \quad \forall ij \in \Omega_c \quad (33)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (34)$$

$$0 \leq n_{ij} \leq \bar{N}_{ij} \quad \forall ij \in \Omega_c \quad (35)$$

$$n_{ij} \text{ inteiro} \quad \forall ij \in \Omega_c \quad (36)$$

$$\theta_i \text{ irrestrito} \quad \forall i \in \Omega_b, i \neq \text{ref} \quad (37)$$

$$\theta_{\text{ref}} = 0 \quad i = \text{ref} \quad (38)$$

sendo que Ω_b é o conjunto de barras do sistema; Ω_c^0 é o conjunto de caminhos ij com circuitos existentes na configuração base; Ω_c é o conjunto de todos os caminhos ij de transmissão formados entre quaisquer duas barras vizinhas; v é o custo total de investimento devido à construção de novos circuitos (linhas e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; N_{ij}^0 é o número de circuitos da configuração base existentes no caminho ij ; n_{ij} é o número de novos circuitos adicionados em um caminho ij que pertence ou não à configuração base; $\bar{N}_{ij}$ é o número máximo de novos circuitos que podem ser construídos em um caminho ij que pertence ou não à configuração base; f_{ij}^0 é o fluxo total de potência ativa dos circuitos da configuração base existentes no caminho ij pertencente à configuração base; f_{ij} é o fluxo total de potência ativa dos novos circuitos adicionados no caminho ij pertencente ou não à configuração base; $\bar{F}_{ij}$ é o fluxo

máximo de potência ativa permitido para cada circuito do caminho ij ; X_{ij} é a reatância do circuito no caminho ij pertencente à configuração base; g_i é o nível de geração de potência ativa na barra de geração i ; $\bar{G}_i$ é a máxima capacidade de geração de potência ativa na barra i ; $\underline{G}_i$ é o limite mínimo de geração na barra i ; D_i é o nível de demanda de potência ativa na barra de carga i ; θ_i é a magnitude do ângulo de fase da tensão na barra i ; θ_{ref} é a referência angular para o conjunto de barras.

Comparando com o modelo de transportes, o modelo híbrido linear acrescenta uma modificação na aplicação da primeira lei de Kirchhoff conforme mostrado na restrição (30) sendo que a estrutura da aplicação dessa lei continua a mesma e, ainda, introduz a aplicação da segunda lei de Kirchhoff somente nos circuitos pertencentes à configuração base conforme mostra a restrição (31).

É interessante observar que em um determinado caminho pertencente a um laço da configuração base, o fluxo de potência ativa de um circuito existente pode ser diferente do fluxo de potência ativa de um novo circuito adicionado nesse mesmo caminho (ROMERO, 1999), porque o primeiro está limitado pela máxima abertura angular ($f_{ij}^0 = N_{ij}^o (\theta_i - \theta_j) / X_{ij}$) e pela máxima capacidade de transmissão ($|f_{ij}^0| \leq N_{ij}^o \bar{F}_{ij}$), e o outro está apenas limitado pela máxima capacidade de transmissão ($|f_{ij}| \leq n_{ij} \bar{F}_{ij}$).

3.2.3 Modelo CC

Na referência (MONTICELLI, 1983) pode ser encontrada uma ampla discussão sobre o modelo CC. Esse modelo é considerado o modelo ideal para a análise do problema de PERT (ROMERO, 1999), habitualmente ele é utilizado nas pesquisas e por especialistas em planejamento das empresas de energia elétrica.

A formulação matemática do modelo CC considera as duas leis de Kirchhoff no sistema completo, isto é, todas as barras e todos os laços da configuração base e, também, todas as barras e todos os laços incluídos na expansão da rede de transmissão devem cumprir a primeira e a segunda lei de Kirchhoff, respectivamente (ROMERO, 1999; SILVA, 2013; ORIONDO, 2020; MIASAKI, 2006). O modelo CC tem o seguinte formato:

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij} \quad (39)$$

sujeito a:

$$\sum_{ji \in \Omega_c} f_{ji} - \sum_{ij \in \Omega_c} f_{ij} + g_i = D_i \quad \forall i \in \Omega_b \quad (40)$$

$$f_{ij} = (N_{ij}^0 + n_{ij}) (\theta_i - \theta_j) / X_{ij} \quad \forall ij \in \Omega_c \quad (41)$$

$$|f_{ij}| \leq (N_{ij}^0 + n_{ij}) \bar{F}_{ij} \quad \forall ij \in \Omega_c \quad (42)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (43)$$

$$0 \leq n_{ij} \leq \bar{N}_{ij} \quad \forall ij \in \Omega_c \quad (44)$$

$$n_{ij} \text{ inteiro} \quad \forall ij \in \Omega_c \quad (45)$$

$$\theta_i \text{ irrestrito} \quad \forall i \in \Omega_b, i \neq ref \quad (46)$$

$$\theta_{ref} = 0 \quad (47)$$

sendo que Ω_b é o conjunto de barras do sistema; Ω_c é o conjunto de todos os caminhos ij do sistema; v é o custo total de investimento devido à construção de novos circuitos (linhas e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; N_{ij}^0 é o número de circuitos da configuração base existentes no caminho ij ; n_{ij} é o número de novos circuitos adicionados em um caminho ij que pertence ou não à configuração base; $\bar{N}_{ij}$ é o número máximo permitido de adições de novos circuitos que podem ser construídos em um caminho ij que pertence ou não à configuração base; f_{ij} é o fluxo total de potência ativa de todos os circuitos (existentes e adicionados) no caminho ij que pertence ou não à configuração base; $\bar{F}_{ij}$ é o fluxo máximo de potência ativa permitido para cada circuito do caminho ij ; X_{ij} é a reatância do circuito no caminho ij ; g_i é o nível de geração de potência ativa na barra de geração i ; $\bar{G}_i$ é a máxima capacidade do gerador de potência ativa na barra i ; $\underline{G}_i$ é o limite mínimo de geração na barra i ; D_i é o nível de demanda de potência ativa na barra de carga i ; θ_i é o ângulo de fase da tensão na barra i ; θ_{ref} é a referência angular para o conjunto de barras.

Comparando com o modelo de transportes, o modelo CC introduz as restrições (41) e (47) no problema. Essas restrições estão relacionadas com a aplicação sistemática da segunda lei de Kirchhoff ao sistema.

A restrição $f_{ij} = (N_{ij}^o + n_{ij})(\theta_i - \theta_j)/X_{ij}$ é uma expressão que relaciona os fluxos de potência ativa nos circuitos como função dos ângulos de fase das tensões em suas barras terminais. O termo não linear, $n_{ij}(\theta_i - \theta_j)$, e a integralidade da variável n_{ij} representam as condições complicantes do problema. São elas que transformam o modelo CC em um problema de PNLIM.

Nesse tipo de problema costuma surgir um número muito grande de soluções ótimas locais, por isso, para haver compatibilidade com as ferramentas computacionais atualmente disponíveis, vários modelos relaxados e várias técnicas de otimização têm sido utilizadas (RIDER, 2006; ROMERO, 1999; MAHDAVI, ANTUNEZ, *et al.*, 2018; LATORRE, CRUZ, *et al.*, 2003; LEE, NG, *et al.*, 2006). Em relação à distribuição dos fluxos de potência ativa, numerosos testes experimentais mostram que os resultados obtidos usando o modelo CC apresentam resultados muito próximos aos obtidos usando o modelo de fluxo de carga CA.

3.3 MODELO CC PARA O PROBLEMA DE PERT CONSIDERANDO PERDAS

O modelo CC, apresentado na Seção 3.2.3, não considera as perdas de transmissão que podem não ser desprezíveis (ALGUACIL, MOTTO e CONEJO, 2003), por isso, é muito importante que elas sejam contabilizadas na formulação do problema de PERT. O modelo CC discutido nesta seção acrescenta considerações relacionadas com as perdas de potência ativa na rede de transmissão. A modelagem matemática apresentada nesta seção é utilizada para obter os resultados discutidos na Seção 4.

Pode ser considerado que as perdas de potência ativa na rede de transmissão correspondem a cargas distribuídas em todos os circuitos do sistema. Em sistemas de médio/grande porte as perdas que ocorrem na transmissão podem ser muito grandes quando comparadas com o nível de geração (RIDER, GARCIA e ROMERO, 2008), (DOMINGUEZ, MACEDO, *et al.*, 2016), (DOMINGUEZ, MACEDO, *et al.*, 2017). Na referência (MONTICELLI, 1983) pode ser encontrada uma discussão mais detalhada sobre esse assunto.

O problema de PERT considerando a operação CC da rede de transmissão com perdas de transmissão é um problema de PNLIM não convexo. Nesse caso, a modelagem matemática considera as duas leis de Kirchhoff no sistema completo conforme pode ser verificado na formulação matemática apresentada a seguir.

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij} + \Lambda \sum_{i \in \Omega_b} q_i \quad (48)$$

sujeito a:

$$\sum_{ji \in \Omega_c} \left(f_{ji} - \frac{\ell_{ji}}{2} \right) - \sum_{ij \in \Omega_c} \left(f_{ij} + \frac{\ell_{ij}}{2} \right) + g_i = D_i - q_i \quad \forall i \in \Omega_b \quad (49)$$

$$f_{ij} = (N_{ij}^o + n_{ij}) (\theta_i - \theta_j) / X_{ij} \quad \forall ij \in \Omega_c \quad (50)$$

$$\ell_{ij} = G_{ij} (N_{ij}^o + n_{ij}) (\theta_i - \theta_j)^2 \quad \forall ij \in \Omega_c \quad (51)$$

$$|f_{ij}| + \frac{\ell_{ij}}{2} \leq (N_{ij}^o + n_{ij}) \bar{F}_{ij} \quad \forall ij \in \Omega_c \quad (52)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (53)$$

$$0 \leq q_i \leq D_i \quad \forall i \in \Omega_b \quad (54)$$

$$0 \leq n_{ij} \leq \bar{N}_{ij} \quad \forall ij \in \Omega_c \quad (55)$$

$$n_{ij} \text{ inteiro} \quad \forall ij \in \Omega_c \quad (56)$$

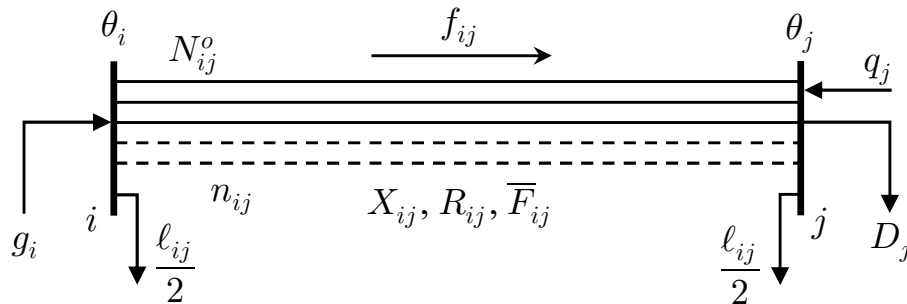
$$\theta_{ref} = 0 \quad (57)$$

sendo que Ω_b é o conjunto de todas as barras do sistema; Ω_c é o conjunto de todos os caminhos de transmissão formados entre duas barras vizinhas; v é o custo total de investimento devido à construção de novos circuitos (linhas e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; n_{ij} é o número de novos circuitos adicionados no caminho ij ; Λ é o parâmetro de penalização de custo devido à perdas de carga, sendo $\Lambda := 2\|\mathbf{c}\|$ onde $\|\mathbf{c}\| := \sqrt{\sum_{ij \in I} C_{ij}^2}$ é a norma euclidiana do vetor dos custos da construção de novos circuitos nos caminhos ij ; q_i é geração artificial de potência ativa na barra de carga i ; f_{ij} é o

fluxo total de potência ativa de todos os circuitos (existentes e adicionados) no caminho ij ; ℓ_{ij} perda total de potência ativa no caminho ij ; g_i é o nível de geração de potência ativa na barra de geração i ; D_i é o nível de demanda de potência ativa na barra de carga i ; N_{ij}^0 é o número de circuitos da configuração base (ou inicial) existentes no caminho ij ; $\theta_i - \theta_j$ é a defasagem angular entre as tensões das duas barras terminais do caminho ij ; X_{ij} é a reatância de um circuito no caminho ij ; G_{ij} é a condutância de um circuito no caminho ij ; $\bar{F}_{ij}$ é o fluxo máximo de potência ativa permitido para cada circuito do caminho ij ; $\bar{G}_i$ é a máxima geração de potência ativa na barra de geração i ; $\underline{G}_i$ é o limite mínimo de geração na barra i ; $\bar{N}_{ij}$ é o número máximo permitido de adições de novos circuitos no caminho ij ; θ_{ref} é a referência angular para o conjunto de barras.

A função v em (48) representa a função objetivo que o problema deve minimizar, o objetivo é encontrar uma solução que forneça o menor valor possível de investimento para adicionar novos circuitos de transmissão na expansão da rede. Nela estão incluídos o custo total de investimento da expansão do sistema de transmissão e o custo de corte de carga. O termo relacionado com o custo de corte de carga tem o objetivo de inibir as possíveis infactibilidades ao forçar o algoritmo a rejeitar os subproblemas de PNL infactíveis zerando as variáveis artificiais, ou seja, a intenção é evitar a ocorrência de subproblemas de PNL infactíveis.

FIGURA 29 - Ilustração da restrição do balanço de potência.



Fonte: Elaboração do próprio autor.

A restrição (49) está ilustrada na Figura 29, ela representa a equação do balanço de potência ativa definida pela primeira lei de Kirchhoff, sendo uma equação para cada barra do sistema, onde $f_{ji} - \ell_{ji}/2$ é o fluxo de potência ativa entrando na barra i e $f_{ij} + \ell_{ij}/2$ é o fluxo de potência ativa saindo na barra i . Observe, nessa figura, a

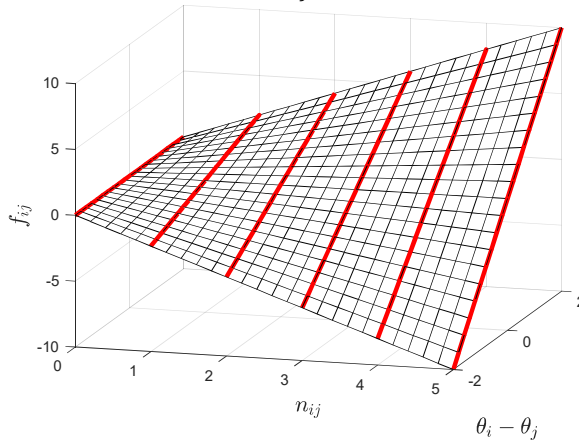
inclusão do efeito das perdas ativas em cada caminho ij com metade para cada uma de suas barras terminais e, também, inclui a geração artificial q_i a qual permite que o modelo seja factível em qualquer situação.

As restrições (50) e (57) estão relacionadas com a aplicação sistemática da segunda lei de Kirchhoff ao sistema. A restrição (51) é utilizada para se obter um conjunto de equações, sendo uma equação para cada caminho ij , que corresponde a uma estimativa das perdas de potência ativa no caminho ij em função da condutância de um circuito, do número de circuitos instalados no caminho ij e da diferença de fase do ângulo de tensão através do caminho ij . A condutância de um circuito é definida pela equação $G_{ij} = R_{ij} / (R_{ij}^2 + X_{ij}^2)$ onde R_{ij} é a resistência de um circuito no caminho ij . A restrição (52) é a capacidade de transmissão do fluxo de potência ativa de um caminho ij , sendo que os fluxos f_{ij} podem fluir nos dois sentidos, por isso, a necessidade de incluir o valor absoluto nela. A restrição (53) é a capacidade da geração de potência ativa das barras de geração e a restrição (54) é a capacidade de geração artificial das barras de carga. A restrição (55) limita o número máximo de novos circuitos que podem ser construídos no caminho ij . Finalmente, a restrição (56) representa a natureza inteira da variável de investimento n_{ij} .

No modelo apresentado, a função objetivo (48) é linear, assim como as restrições (49) e (52)–(57). No entanto, as restrições (50) e (51) são não lineares e não convexas, pois apresentam o produto de variáveis e variáveis quadradas. Além disso, observe que (50) é uma igualdade quadrática (com um termo bilinear), e quando as perdas não são consideradas no problema, o modelo se torna um problema de programação com restrições quadráticas inteira mista não convexa.

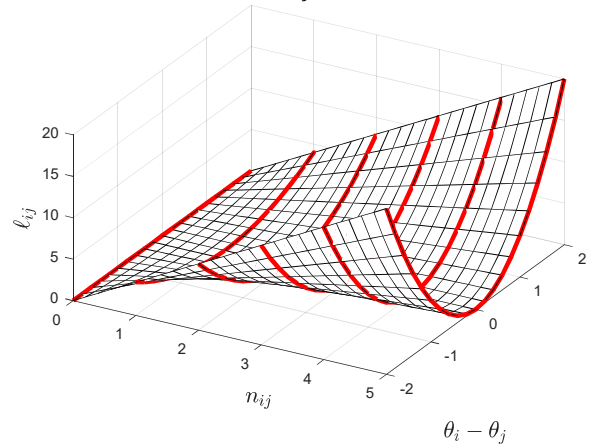
A Figura 30 representa a restrição (50) considerando f_{ij} como uma função de n_{ij} e $(\theta_i - \theta_j)$ para $X_{ij} = 1$, enquanto a Figura 31 representa a restrição (51) considerando ℓ_{ij} como uma função das mesmas variáveis para $G_{ij} = 1$. Nestas figuras, as curvas vermelhas representam os valores possíveis de f_{ij} e ℓ_{ij} quando n_{ij} é um número inteiro, enquanto toda a superfície de ambas as figuras representa os valores para f_{ij} and ℓ_{ij} quando a integralidade de n_{ij} é relaxada. Pode-se ver que ambas as restrições são não lineares e não convexas, mesmo quando a integralidade de n_{ij} é relaxada.

FIGURA 30 - Ilustração da restrição (50) com $X_{ij} = 1$.



Fonte: Elaboração do próprio autor.

FIGURA 31 - Ilustração da restrição (51) com $G_{ij} = 1$.



Fonte: Elaboração do próprio autor.

3.4 MODELO CA PARA O PROBLEMA DE PERT

O modelo matemático para o PERT usando o modelo CA é a formulação mais completa do problema (RIDER, 2006; SILVA, 2013; ORIONDO, 2020). Esse modelo é baseado nas relações do fluxo de carga CA apresentadas em (MONTICELLI, 1983). O modelo CA é definido como sendo uma extensão do modelo CC. Ele pode ser escrito como apresentado em (58)–(75)

$$\text{minimizar } v = \sum_{ij \in \Omega_c} C_{ij} n_{ij} + \Lambda \sum_{i \in \Omega_b} (q_i + w_i) \quad (58)$$

sujeito a:

$$g_i - \sum_{ij \in \Omega_c} f_{ij}^{de} - \sum_{ji \in \Omega_c} f_{ji}^{para} + q_i = D_i \quad \forall i \in \Omega_b \quad (59)$$

$$q_i - \sum_{ij \in \Omega_c} q_{ij}^{de} - \sum_{ji \in \Omega_c} q_{ji}^{para} + \mathcal{B}_i^{sh} v_i^2 + w_i = Q_i \quad \forall i \in \Omega_b \quad (60)$$

$$f_{ij}^{de} = (n_{ij} + N_{ij}^0) [G_{ij} v_i^2 - v_i v_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij})] \quad \forall ij \in \Omega_c \quad (61)$$

$$f_{ij}^{para} = (n_{ij} + N_{ij}^0) [G_{ij} v_j^2 - v_i v_j (G_{ij} \cos \theta_{ij} - B_{ij} \sin \theta_{ij})] \quad \forall ij \in \Omega_c \quad (62)$$

$$q_{ij}^{de} = (n_{ij} + N_{ij}^0) [-(B_{ij}^{sh} + B_{ij}) v_i^2 - v_i v_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij})] \quad \forall ij \in \Omega_c \quad (63)$$

$$q_{ij}^{para} = (n_{ij} + N_{ij}^0) [-(B_{ij}^{sh} + B_{ij}) v_j^2 + v_i v_j (G_{ij} \sin \theta_{ij} + B_{ij} \cos \theta_{ij})] \quad \forall ij \in \Omega_c \quad (64)$$

$$(f_{ij}^{de})^2 + (q_{ij}^{de})^2 \leq [(n_{ij} + N_{ij}^0)\bar{F}_{ij}]^2 \quad \forall ij \in \Omega_c \quad (65)$$

$$(f_{ij}^{para})^2 + (q_{ij}^{para})^2 \leq [(n_{ij} + N_{ij}^0)\bar{F}_{ij}]^2 \quad \forall ij \in \Omega_c \quad (66)$$

$$\underline{G}_i \leq g_i \leq \bar{G}_i \quad \forall i \in \Omega_b \quad (67)$$

$$\underline{Q}_i \leq q_i \leq \bar{Q}_i \quad \forall i \in \Omega_b \quad (68)$$

$$0 \leq q_i \leq D_i \quad \forall i \in \Omega_b \quad (69)$$

$$0 \leq w_i \leq Q_i \quad \forall i \in \Omega_b \quad (70)$$

$$\underline{V}_i \leq v_i \leq \bar{V}_i \quad \forall i \in \Omega_b \quad (71)$$

$$0 \leq n_{ij} \leq \bar{N}_{ij} \quad \forall ij \in \Omega_c \quad (72)$$

$$n_{ij} \text{ inteiro} \quad \forall ij \in \Omega_c \quad (73)$$

$$\theta_i \text{ irrestrito} \quad \forall i \in \Omega_b, i \neq ref \quad (74)$$

$$\theta_{ref} = 0 \quad (75)$$

no qual Ω_b é o conjunto de barras do sistema; Ω_c é o conjunto de todos os caminhos ij do sistema.

Os parâmetros são: v é o custo total de investimento devido à construção de novos circuitos (linhas e/ou transformadores) na rede de transmissão; C_{ij} é o custo da construção de um novo circuito no caminho ij ; N_{ij}^0 é o número de circuitos da configuração base existentes no caminho ij ; $\bar{N}_{ij}$ é o número máximo permitido de adições de novos circuitos que podem ser construídos em um caminho ij ; $\underline{G}_i$ e $\bar{G}_i$ são os limites mínimo e máximo de potência ativa gerada na barra i ; D_i é o nível de demanda de potência ativa na barra i ; Q_i é o nível de demanda de potência reativa na barra i ; $\underline{Q}_i$ e $\bar{Q}_i$ são os limites mínimo e máximo de geração de potência reativa na barra i ; $\bar{F}_{ij}$ é o limite máximo de fluxo de potência aparente no caminho ij ; $\underline{V}_i$ e $\bar{V}_i$ são os limites mínimo e máximo da magnitude de tensão na barra i ; G_{ij} é a condutância no caminho ij ; $B_{ij} = -X_{ij}/(R_{ij}^2 + X_{ij}^2)$ é a susceptância no caminho ij ; B_{ij}^{sh} é a susceptância shunt no caminho ij (se ij é um transformador $B_{ij}^{sh} = 0$, os taps fora do valor nominal dos transformadores não foram considerados e, neste caso as linhas de transmissão e os

transformadores tem um mesmo circuito equivalente); $\mathcal{B}_i^{sh}$ é a susceptância shunt na barra i ; θ_{ref} é a referência angular para o conjunto de barras.

As variáveis contínuas são: g_i é o nível de geração de potência ativa na barra i ; q_i é o nível de geração de potência reativa na barra i ; f_{ij}^{de} é o fluxo total de potência ativa saindo da barra i ; f_{ij}^{para} é o fluxo total de potência ativa entrando na barra i ; q_{ij}^{de} é o fluxo total de potência reativa saindo da barra i ; q_{ij}^{para} é fluxo total de potência reativa entrando na barra i ; v_i é a magnitude de tensão na barra i ; $\theta_{ij} = \theta_i - \theta_j$ é a diferença do ângulo de fase da tensão entre as barras i e j ; q_i é a geração artificial de potência ativa na barra i ; w_i é a geração artificial de potência reativa na barra i ; n_{ij} é o número de novos circuitos adicionados em um caminho ij .

Nesse modelo, a função objetivo v em (58) representa o investimento a ser minimizado no sistema devido à construção das novas linhas na rede de transmissão e inclui os custos de geração artificial. As restrições (59) e (60) representam as equações do balanço das potências ativa e reativa em cada barra do sistema, as quais são definidas pela primeira lei de Kirchhoff. As restrições (61)–(64) estão relacionadas com a aplicação sistemática da segunda lei de Kirchhoff ao sistema; elas representam os fluxos de potências ativa e reativa saindo e entrando na barra i . Essas restrições são altamente não lineares devido a existência de funções trigonométricas, multiplicação de variáveis contínuas e multiplicação de variáveis inteiras por contínuas. As restrições (65) e (66) representam os limites de capacidade de transmissão dos fluxos de potências ativa e reativa de um caminho ij , os quais são limitados pelo máximo fluxo de potência aparente $\bar{F}_{ij}$. As restrições (67) e (68) representam os limites superior e inferior da geração de potências ativa e reativa nas barras de geração. As restrições (69) e (70) são os limites de geração artificial de potência ativa e reativa na barra i . A restrição (71) representa os limites superior e inferior da magnitude de tensão em cada barra do sistema. A restrição (72) limita o número máximo de novos circuitos que podem ser construídos no caminho ij . A restrição (73) representa a natureza inteira da variável de investimento n_{ij} , a qual representa a maior fonte de complexidade do problema. Finalmente (74) indica que a variável θ_i é irrestrita e (75) define a referência angular do sistema elétrico.

A mesma estratégia utilizada em (48)–(57) para garantir soluções factíveis, i.e., considerar gerações artificiais nas barras, também foi considerada em (58)–(75), mas

nesse caso foram consideradas gerações artificiais de potência ativa e reativa, sendo ambas minimizadas na função objetivo.

4 TESTES E RESULTADOS

O algoritmo BB proposto para resolver problemas de PNLIM não convexos foi implementado na linguagem de modelagem matemática AMPL (FOURER, GAY e KERNIGHAN, 2003). O solucionador (*solver*) de código aberto IPOPT v3.12.13 (WÄCHTER e BIEGLER, 2006) e os solucionadores comerciais KNITRO v12.4.0 (BYRD, NOCEDAL e WALTZ, 2006), LOQO v7.03 (VANDERLEI, 1999) e MINOS v5.51 (HUA ET AL, 2003) foram usados, com configurações padrão, para resolver os subproblemas de PNL que surgem em cada nó da árvore BB.

Para as regras de sondagem, foi definido um fator de segurança com valor $\varepsilon = 5\%$. Este valor de ε foi definido com base em testes experimentais e demonstrou ser conservador sem aumentar muito o tamanho da árvore BB e os tempos computacionais para resolver o problema. Para outros problemas, pode ser adequado realizar uma análise para definir valores de ε mais adequados.

4.1 TESTES COM O MODELO CC

Os testes foram realizados utilizando o sistema Garver de 6 barras, o sistema IEEE de 24 barras, o sistema Sul Brasileiro de 46 barras e o sistema Colombiano de 93 barras considerando os seguintes casos: (i) sem perdas de potência ativa na transmissão e sem reprogramação de geração de potência ativa; (ii) sem perdas e com reprogramação de geração; (iii) com perdas e com reprogramação de geração.

Os casos *sem perdas* são obtidos considerando nulas as resistências em todos os caminhos, ou seja $R_{ij} = 0$. Nos casos *sem reprogramação de geração* os valores da geração nas barras de geração são fixos. Nos casos *com reprogramação de geração* os valores da geração nas barras de geração podem ser ajustados para um valor compreendido entre $\underline{g}_i \leq g_i \leq \overline{g}_i$. Observe que no caso *com perdas*, as perdas só podem ser consideradas nos casos *com reprogramação de geração*, sendo que a reprogramação é função do nível das perdas que ocorrem na transmissão.

Dados completos para todos os sistemas estão disponíveis em (LAPSEE). Todos os testes foram realizados em um computador com processador AMD Ryzen™ 9 5950X de 3,4 GHz e 32 GB de RAM. Deve-se destacar que o sistema Colombiano de 93 barras representa um caso de grande porte do problema de PERT. Observe que a

complexidade de um sistema para PERT não está diretamente relacionada ao número de barras. De fato, a complexidade do problema está muito mais relacionada ao nível de estresse da rede e ao número de linhas de transmissão que devem ser construídas para uma operação adequada da rede.

Para fins de comparação, o modelo de PNLIM (48)–(57) também foi resolvido diretamente usando solucionadores comerciais de otimização para problemas de PNLIM não convexos. Entre eles, foram utilizados os solucionadores que podem obter soluções locais KNITRO v12.4.0 e BONMIN v1.8.8 (BONAMI, BIEGLER, *et al.*, 2008). O solucionador BONMIN é um *solver* de código aberto que opera de forma associada com os solucionadores CBC v2.10.5 e IPOPT v3.12.13. Também foram utilizados os solucionadores globais BARON v21.1.13 (SAHINIDIS, 2015), LINDO GLOBAL v13.0.4099.255 (LINDO SYSTEMS INC.), COUENNE v0.5.8 (BELOTTI, 2009) e SCIP v8.0.0 (VIGERSKE e GLEIXNER, 2018). Entre esses solucionadores globais, COUENNE e SCIP são solucionadores de código aberto. Além disso, o GUROBI v9.5.0 (GUROBI OPTIMIZATION INC.), que é capaz de resolver problemas de programação quadrática não convexos, foi utilizado para resolver o problema sem considerar perdas. Em todos os testes, as configurações padrão foram consideradas em todos os solucionadores, e o tempo limite foi fixado em uma hora.

Tabela 2 - Dimensões dos problemas considerando o modelo CC

Características dos problemas	Sistemas sem perdas								Sistemas com perdas			
	Garver de 6 barras com reprogramação	Garver de 6 barras	IEEE de 24 barras com reprogramação	IEEE de 24 barras	Sul Brasileiro de 46 barras com reprogramação	Sul Brasileiro de 46 barras	Colombiano de 93 barras com reprogramação	Colombiano de 93 barras	Garver de 6 barras com reprogramação	IEEE de 24 barras com reprogramação	Sul Brasileiro de 46 barras com reprogramação	Colombiano de 93 barras com reprogramação
Número de variáveis	38	38	115	115	214	214	451	451	53	156	294	606
Limitadas abaixo somente	0	0	0	0	0	0	0	0	15	41	79	155
Limitadas abaixo e acima	18	18	51	51	91	91	204	204	18	51	91	204
Livres	20	20	64	64	123	123	247	247	20	64	124	247
Número de variáveis inteiras	15	15	41	41	79	79	155	155	15	41	79	155
Número de restrições	51	51	147	147	280	280	558	558	66	188	362	713
Igualdades lineares	6	6	24	24	45	45	93	93	6	24	46	93
Igualdades quadráticas	15	15	41	41	79	79	155	155	15	41	79	155
Igualdades não lineares gerais	0	0	0	0	0	0	0	0	15	41	79	155
Desigualdade linear unilateral	30	30	82	82	156	156	310	310	30	82	158	310
Números não zeros na Jacobiana	148	148	414	414	790	790	1593	1593	263	736	1424	2827
Números não zeros na Hessiana	25	25	76	76	153	153	304	304	40	134	272	545

Fonte: Elaboração do próprio autor.

A Tabela 2 mostra as dimensões dos problemas. O número de variáveis inteiras

é igual ao número de caminhos do sistema. Os testes foram realizados sem considerar perdas no sistema e com perdas para os sistemas com reprogramação de geração. A Tabela 2 mostra que o sistema Garver de 6 barras é o menor problema considerado, enquanto o sistema Colombiano de 93 barras é o maior. Além disso, quando as perdas são consideradas, os problemas tornam-se significativamente mais complexos, conforme indicado pelo número de não-zeros na Jacobiana e na Hessiana quando comparados com os respectivos problemas sem perdas. Os números não-zeros são definidos pelos parâmetros, variáveis contínuas e variáveis inteiras utilizados no modelo matemático. A matriz jacobiana é formada pelas derivadas parciais de primeira ordem. A matriz hessiana organiza todas as derivadas parciais de segunda ordem.

As Tabelas 3-6 apresentam os principais resultados, incluindo o custo de investimento e o tempo computacional (entre parênteses) para encontrar a solução em cada teste. Um asterisco indica que o solucionador encerrou antes do limite de tempo de uma hora, encontrando uma solução ótima local (ou global). Os tempos menores que 3600 s sem asterisco indicam o tempo que o solucionador demorou para encontrar a solução registrada, porém o limite de tempo foi utilizado para tentar encontrar uma solução melhor. Um valor em negrito indica que o solucionador encontrou a melhor solução conhecida do problema. A sigla NC indica que o solucionador não conseguiu encontrar uma solução inteira para o problema no tempo máximo permitido.

Tabela 3 - Resultados dos custos de investimentos dos sistemas sem perdas para o modelo CC usando solucionadores locais/globais (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador						
		KNITRO	BONMIN	BARON	LINDO GLOBAL	COUENNE	SCIP	GUROBI
Garver de 6 barras com reprogramação	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) *
Garver de 6 barras	231,000	231,000 (<1 s) *	238,000 (<1 s) *	231,000 (<1 s) *	infactível*	231,000 (<1 s) *	231,000 (<1 s) *	231,000 (<1 s) *
IEEE de 24 barras com reprogramação	152,000	152,000 (<1 s) *	152,000 (2 s) *	152,000 (4 s) *	infactível*	152,000 (5 s) *	152,000 (5 s) *	152,000 (<1 s) *
IEEE de 24 barras	370,000	370,000 (<1 s) *	370,000 (3 s) *	370,000 (218 s)	infactível*	370,000 (51 s) *	383,000 (2 s)	370,000 (<1 s) *
Sul do Brasil de 46 barras com reprogramação	72,870	72,870 (8 s) *	72,870 (115 s) *	72,870 (62 s) *	infactível*	72,870 (41 s) *	72,870 (7 s) *	72,870 (2 s) *
Sul do Brasil de 46 barras	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) *
Colombiano de 93 barras com reprogramação	189,650	189,650 (1 s) *	189,650 (96 s) *	189,650 (31 s)	Infactível*	196,373 (11 s) *	NC	404,120 (1 s)
Colombiano de 93 barras	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)

Fonte: Elaboração do próprio autor.

Tabela 4 - Resultados dos custos de investimentos dos sistemas com perdas para o modelo CC usando solucionadores locais/globais (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador					
		KNITRO	BONMIN	BARON	LINDO GLOBAL	COUENNE	SCIP
Garver de 6 barras com reprogramação	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 de 24 barras com reprogramação	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) *
Sul do Brasil de 46 barras com reprogramação	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) *
Colombiano de 93 barras com reprogramação	202,920	202,920 (4 s) *	204,528 (47 s) *	781,747 (2850 s)	370,250 (121 s)	235,610 (45 s)	NC

Fonte: Elaboração do próprio autor.

Tabela 5 - Resultados dos custos de investimentos dos sistemas sem perdas para o modelo CC usando o algoritmo BB proposto (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador			
		IPOPT	KNITRO	LOQO	MINOS
Garver de 6 barras com reprogramação	110,000	110,000 (<1 s) *	110,000 (1 s) *	110,000 (<1 s) *	110,000 (1 s) *
Garver de 6 barras	231,000	231,000 (1 s) *	231,000 (1 s) *	231,000 (1 s) *	231,000 (1 s) *
IEEE de 24 barras com reprogramação	152,000	152,000 (3 s) *	152,000 (2 s) *	152,000 (1 s) *	152,000 (1 s) *
IEEE de 24 barras	370,000	370,000 (10 s) *	370,000 (5 s) *	370,000 (3 s) *	370,000 (4 s) *
Sul do Brasil de 46 barras com reprogramação	72,870	72,870 (43 s) *	72,870 (86 s) *	72,870 (20 s) *	105,317 (3 s) *
Sul do Brasil de 46 barras	154,420	154,420 (459 s) *	154,420 (241 s) *	154,420 (163 s) *	185,774 (4 s) *
Colombiano de 93 barras com reprogramação	189,650	189,650 (20 s) *	189,650 (5 s) *	189,650 (4 s) *	189,650 (9 s) *
Colombiano de 93 barras	562,417	572,061 (1982 s)	572,061 (663 s)	562,417 (3292 s)	579,411 (1545 s)

Fonte: Elaboração do próprio autor.

Tabela 6 - Resultados dos custos de investimentos dos sistemas com perdas para o modelo CC usando o algoritmo BB proposto (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador			
		IPOPT	KNITRO	LOQO	MINOS
Garver de 6 barras com reprogramação	130,000	130,000 (3 s) *	130,000 (2 s) *	130,000 (1 s) *	170,000 (<1 s)
IEEE de 24 barras com reprogramação	188,000	188,000 (36 s) *	188,000 (18 s) *	188,000 (4 s) *	infatível*
Sul do Brasil de 46 barras com reprogramação	75,895	75,895 (24 s) *	75,895 (22 s) *	75,895 (2 s) *	infatível*
Colombiano de 93 barras com reprogramação	202,920	202,920 (78 s) *	202,920 (102 s) *	202,920 (10 s) *	infatível*

Fonte: Elaboração do próprio autor.

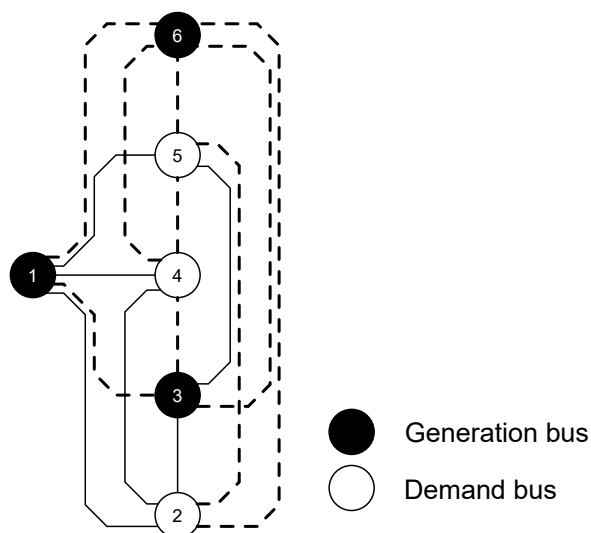
As Tabelas 3 e 5 apresentam, respectivamente, os resultados obtidos pelos solucionadores locais e globais e algoritmo BB proposto para problemas de PNLIM não convexos para resolver o problema de PERT considerando os seguintes dois casos: (i) sistemas sem perdas e sem reprogramação de geração e (ii) sistemas sem perdas e com reprogramação de geração. As Tabelas 4 e 6 apresentam, respectivamente, os resultados obtidos pelos solucionadores locais e globais e algoritmo BB proposto para problemas de PNLIM não convexos na resolução do problema de PERT considerando sistemas com perdas e com reprogramação de geração.

As seções a seguir apresentam uma discussão com mais detalhes sobre os resultados obtidos para cada sistema.

4.1.1 Sistema Garver de 6 barras

A topologia inicial para o sistema Garver de 6 barras é apresentada na Figura 32. As linhas contínuas indicam circuitos de transmissão existentes e as linhas pontilhadas indicam caminhos onde novos circuitos podem ser construídos. Até três novos circuitos podem ser construídos em cada caminho.

FIGURA 32 - Topologia inicial do sistema Garver de 6 barras.



Fonte: (GARVER, 1970)

Nas Tabelas 3 e 5, para o sistema Garver de 6 barras *sem considerar perdas de potência ativa na transmissão e considerando a reprogramação de geração de*

potência ativa a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 110,000 (cento e dez milhões de dólares) com a construção dos circuitos $n_{3-5} = 1$ e $n_{4-6} = 3$. A Tabela 3 mostra que, com exceção dos solucionadores KNITRO e BONMIN, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 1 s. As soluções encontradas pelo KNITRO e BONMIN são 18,18% e 121,82% piores que a solução ótima, respectivamente. Por outro lado, analisando a Tabela 5, é possível verificar que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em até 1 s.

Nas Tabelas 3 e 5, para o sistema Garver de 6 barras *sem considerar perdas e sem considerar a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 231,000 com a construção dos circuitos $n_{2-6} = 3$, $n_{3-5} = 1$, $n_{4-6} = 2$, e $n_{5-6} = 1$. A Tabela 3 mostra que, com exceção dos solucionadores BONMIN e LINDO GLOBAL, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 1 s. A solução encontrada pelo BONMIN é 3,03% pior que a solução ótima, e o LINDO GLOBAL encontrou uma solução que é infactível. Por outro lado, a Tabela 5 mostra que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 1s.

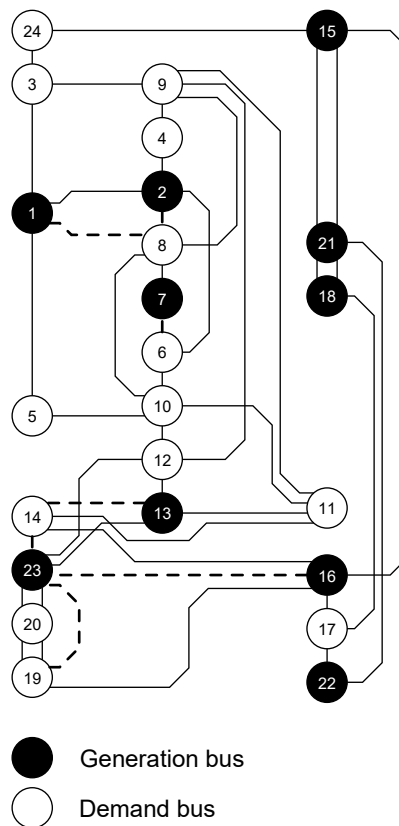
Nas Tabelas 4 e 6, para o sistema Garver de 6 barras *considerando as perdas e considerando a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 130,000 com a construção dos circuitos $n_{2-3} = 1$, $n_{3-5} = 1$, e $n_{4-6} = 3$. A Tabela 4 mostra que, com exceção dos solucionadores KNITRO e LINDO GLOBAL, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 4 s. As soluções encontradas pelo KNITRO e LINDO GLOBAL são 7,69% piores que a solução mais conhecida. Por outro lado, a Tabela 6 mostra que, com exceção do solucionador MINOS, o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 3 s. A solução encontrada pelo MINOS é 30,77% pior que a solução mais conhecida.

4.1.2 Sistema IEEE de 24 barras

A topologia inicial para o sistema IEEE de 24 barras é apresentada na Figura 33.

As linhas contínuas indicam circuitos de transmissão existentes e as linhas pontilhadas indicam caminhos onde novos circuitos podem ser construídos. Até cinco novos circuitos podem ser construídos em cada caminho.

FIGURA 33 - Topologia inicial do sistema IEEE de 24 barras.



Fonte: (FANG e HILL, 2003)

Nas Tabelas 3 e 5, para o sistema IEEE de 24 barras *sem considerar perdas de potência ativa na transmissão e considerando a reprogramação de geração de potência ativa* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 152,000 (cento e cinquenta e dois milhões de dólares) com a construção dos circuitos $n_{6-10} = 1$, $n_{7-8} = 2$, $n_{10-11} = 1$, e $n_{14-16} = 1$. A Tabela 3 mostra que, com exceção do solucionador LINDO GLOBAL, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 5 s. A solução encontrada pelo LINDO GLOBAL é infactível. Por outro lado, analisando a Tabela 5, é possível verificar que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em até 3 s.

Nas Tabelas 3 e 5, para o sistema IEEE de 24 barras *sem considerar perdas e sem considerar a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 370,000 com a construção dos circuitos $n_{3-24} = 1$, $n_{6-10} = 1$, $n_{7-8} = 1$, $n_{10-12} = 1$, $n_{14-16} = 1$, $n_{15-24} = 1$, $n_{16-1} = 2$, e $n_{17-1} = 2$. A Tabela 3 mostra que, com exceção dos solucionadores LINDO GLOBAL e SCIP, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 218 s. A solução encontrada pelo LINDO GLOBAL é infactível e o SCIP encontrou uma solução 3,51% pior que a solução ótima. Por outro lado, a Tabela 5 mostra que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 10 s.

Nas Tabelas 4 e 6, para o sistema IEEE de 24 barras *considerando as perdas e considerando a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 188,000 com a construção dos circuitos $n_{6-10} = 1$, $n_{7-8} = 2$, $n_{10-1} = 1$, $n_{14-1} = 1$, e $n_{16-1} = 1$. A Tabela 4 mostra que, com exceção do solucionador KNITRO, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 362 s. A solução encontrada pelo KNITRO é 7,98% pior que a solução mais conhecida. Por outro lado, a Tabela 6 mostra que, com exceção do solucionador MINOS, o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 36 s. A solução encontrada pelo MINOS é infactível.

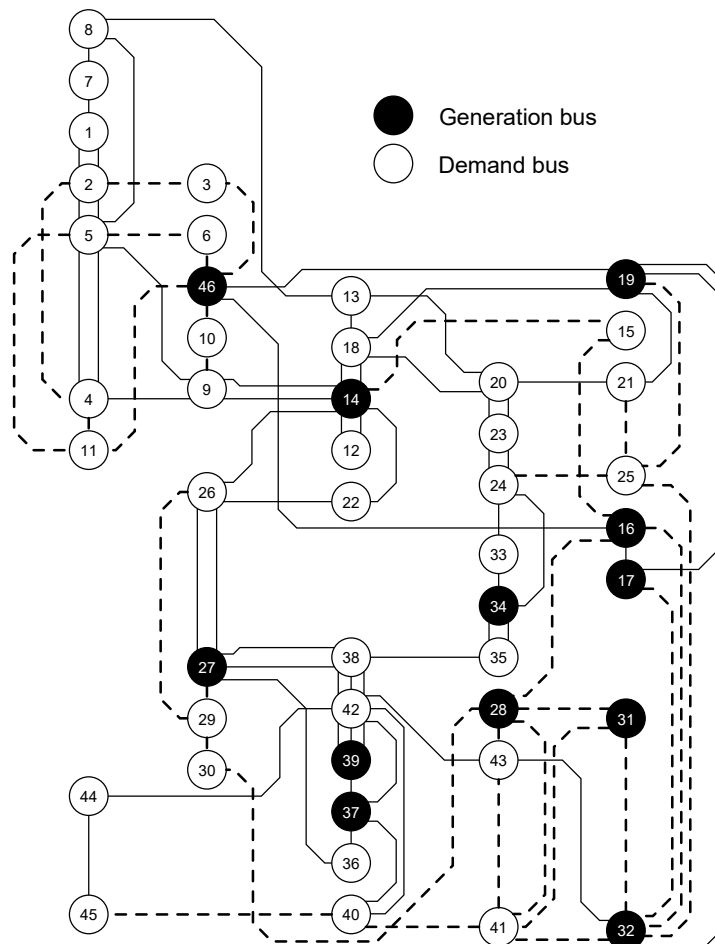
4.1.3 Sistema Sul Brasileiro de 46 barras

A topologia inicial para o sistema Sul Brasileiro de 46 barras é apresentada na Figura 34. As linhas contínuas indicam circuitos de transmissão existentes e as linhas pontilhadas indicam caminhos onde novos circuitos podem ser construídos. Até cinco novos circuitos podem ser construídos em cada caminho.

Nas Tabelas 3 e 5, para o sistema Sul Brasileiro de 46 barras *sem considerar perdas de potência ativa na transmissão e considerando a reprogramação de geração de potência ativa* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 72,870 (setenta e dois milhões, oitocentos e setenta mil dólares) com a construção dos circuitos $n_{2-5} = 1$, $n_{13} = 1$, $n_{20-23} = 1$, $n_{20-21} = 2$,

$n_{42-43} = 1$, $n_{46-6} = 1$, e $n_{5-6} = 2$. A Tabela 3 mostra que, com exceção do solucionador LINDO GLOBAL, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 115 s. A solução encontrada pelo LINDO GLOBAL é infactível. Por outro lado, analisando a Tabela 5, com exceção do solucionador MINOS, é possível verificar que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em até 86 s. A solução encontrada pelo MINOS é 44,53% mais cara que a solução ótima.

FIGURA 34 - Topologia inicial do sistema Sul Brasileiro de 46 barras.



Fonte: (HAFFNER, MONTICELLI, et al., 2000)

Nas Tabelas 3 e 5, para o sistema Sul Brasileiro de 46 barras *sem considerar perdas* e *sem considerar a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 154,420 com a construção dos circuitos $n_{20-2} = 1$, $n_{42-43} = 2$, $n_{46-6} = 1$, $n_{19-25} = 1$, $n_{31-32} = 1$, $n_{28-30} = 1$,

$n_{26-29} = 3$, $n_{24-2} = 2$, $n_{29-30} = 2$, e $n_{5-6} = 2$. A Tabela 3 mostra que, com exceção dos solucionadores KNITRO, BARON, LINDO GLOBAL e COUENNE, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em até 311 s. As soluções encontradas por KNITRO, BARON, LINDO GLOBAL e COUENNE são, respectivamente, 1,87%, 6,77%, 61,99% e 9,00% piores que a solução ótima. Por outro lado, a Tabela 5 mostra que, com exceção do solucionador MINOS, o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 459 s. A solução encontrada pelo MINOS é 20,30% mais cara que a solução ótima.

Nas Tabelas 4 e 6, para o sistema Sul Brasileiro de 46 barras *considerando as perdas e considerando a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 75,895 com a construção dos circuitos $n_{18-20} = 1$, $n_{20-23} = 1$, $n_{20-21} = 2$, $n_{42-} = 1$, $n_{46-6} = 1$, e $n_{5-6} = 2$. A Tabela 4 mostra que, com exceção dos solucionadores KNITRO, BONMIN e LINDO GLOBAL, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em menos de 3589 s. As soluções encontradas pelo KNITRO, BONMIN e LINDO GLOBAL são, respectivamente, 35,48%, 23,86% e 70,04% mais caras que a solução mais conhecida. Por outro lado, a Tabela 6 mostra que, com exceção do solucionador MINOS, o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em menos de 24 s. A solução encontrada pelo MINOS é infactível.

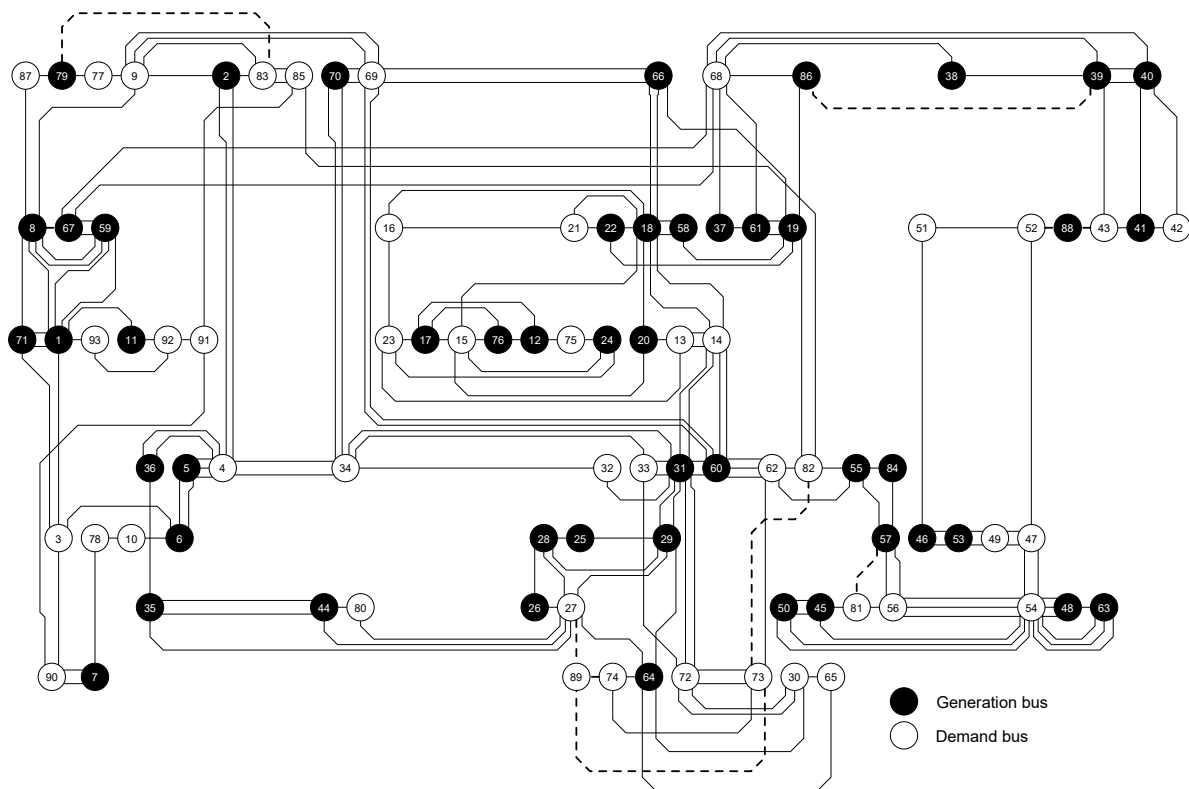
4.1.4 Sistema Colombiano de 93 barras

A topologia inicial para o sistema Colombiano de 93 barras é apresentada na Figura 35. As linhas contínuas indicam circuitos de transmissão existentes e as linhas pontilhadas indicam caminhos onde novos circuitos podem ser construídos. Até quatro novos circuitos podem ser construídos em cada caminho.

Nas Tabelas 3 e 5, para o sistema Colombiano de 93 barras *sem considerar perdas de potência ativa na transmissão e considerando a reprogramação de geração de potência ativa* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 189,650 (cento e oitenta e nove milhões, seiscentos e cinquenta mil dólares) com a construção dos circuitos $n_{27-29} = 1$, $n_{62-73} = 1$, $n_{19-82} = 1$, $n_{82-85} = 1$, e $n_{68-86} = 1$. A Tabela 3 mostra que, com exceção dos solucionadores

LINDO GLOBAL, COUENNE, SCIP e GUROBI, todos os demais solucionadores foram capazes de encontrar a solução ótima global do problema em até 96 s. O solucionador LINDO GLOBAL convergiu para uma solução infactível, o SCIP não foi capaz de encontrar uma solução viável, o COUENNE e o GUROBI encontraram, respectivamente, soluções 3,54% e 113,18% mais caras que a solução ótima. Por outro lado, a Tabela 5 mostra que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em não mais de 20 s.

FIGURA 35 - Topologia inicial do sistema Colombiano de 93 barras.



Fonte: Elaboração do próprio autor.

Nas Tabelas 3 e 5, para o sistema Colombiano de 93 barras *sem considerar perdas e sem considerar a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 562,417 com a construção dos circuitos $n_{43-88} = 2$, $n_{15-18} = 1$, $n_{30-6} = 1$, $n_{30-72} = 1$, $n_{55-5} = 1$, $n_{55-84} = 1$, $n_{56-5} = 1$, $n_{55-6} = 1$, $n_{27-64} = 1$, $n_{27-29} = 1$, $n_{50-5} = 1$, $n_{62-7} = 1$, $n_{54-5} = 1$, $n_{72-73} = 1$, $n_{19-} = 2$, $n_{82-85} = 1$, e $n_{68-8} = 1$. A Tabela 3 mostra que apenas os solvers KNITRO e BONMIN foram capazes de encontrar a solução ótima global do

problema em menos de 179 s. As soluções encontradas por BARON, LINDO GLOBAL e GUROBI são, respectivamente, 30,72%, 702,29% e 15,85% mais caras que a solução ótima, enquanto COUENNE e SCIP não conseguiram encontrar uma solução inteira para o problema dentro do prazo. Analisando a Tabela 5, é possível verificar que o algoritmo BB proposto só conseguiu encontrar a solução ótima global para o problema utilizando o solver LOQO para resolver os subproblemas de PNL em 3292 s. A solução obtida pelo algoritmo BB proposto com os solvers IPOPT e KNITRO é apenas 1,71% mais cara que a solução ótima, enquanto a solução obtida com o solver MINOS é 3,02% mais cara que a solução ótima.

Nas Tabelas 4 e 6, para o sistema Colombiano de 93 barras *considerando as perdas e considerando a reprogramação de geração* a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 202,920 com a construção dos circuitos $n_{27-29} = 1$, $n_{62-7} = 1$, $n_{19-82} = 2$, $n_{82-8} = 1$, and $n_{68-8} = 1$. Os resultados apresentados na Tabela 4 indicam que apenas o KNITRO foi capaz de encontrar a solução mais conhecida para o problema em 4 s. As soluções encontradas por BONMIN, BARON, LINDO GLOBAL e COUENNE são, respectivamente, 0,79%, 285,25%, 82,46% e 16,11% piores que a solução mais conhecida. O SCIP não conseguiu encontrar uma solução inteira para o problema dentro do prazo. A Tabela 6 mostra que o algoritmo BB proposto foi capaz de encontrar a solução mais conhecida com cada solucionador de PNL em menos de 102 s, exceto ao usar MINOS para o qual o algoritmo BB indicou incorretamente que o problema era infactível.

4.1.5 Discussão

Com base nos resultados apresentados na Tabela 3 e na Tabela 4, verifica-se que nenhum solucionador de problemas de PNLIM foi capaz de obter a solução ótima de todas as instâncias do problema considerado. Os solucionadores KNITRO e BONMIN foram capazes de encontrar a solução mais conhecida de 7 e 8 de 12 instâncias, respectivamente. Os solucionadores globais BARON, LINDO GLOBAL, COUENNE e SCIP encontraram as soluções mais conhecidas para 9, 2, 8 e 8 de 12 instâncias, respectivamente. Além disso, GUROBI encontrou as soluções mais conhecidas para 6 de 8 instâncias do problema sem considerar as perdas. Com exceção do solucionador LINDO GLOBAL, deve-se notar que os solucionadores globais são mais eficientes na resolução de instâncias menores do problema.

As Tabelas 5 e 6 mostram que o algoritmo BB proposto para problemas de PNLIM não convexos foi capaz de encontrar as melhores soluções conhecidas de todas as instâncias consideradas quando o solucionador LOQO foi usado para resolver os subproblemas de PNL. Além disso, com os solucionadores IPOPT e KNITRO, não foi possível encontrar a melhor solução conhecida de apenas uma instância. Com o solucionador MINOS, o algoritmo BB proposto só foi capaz de encontrar a melhor solução conhecida em 5 das 12 instâncias. Observe na Tabela 5 que, para o sistema Colombiano de 93 barras sem considerar perdas e sem reprogramação de geração, o algoritmo BB proposto utilizando os solucionadores IPOPT, KNITRO e MINOS foi encerrado devido ao tempo computacional máximo permitido de 3600 s. Portanto, pode ser possível obter as melhores soluções conhecidas ajustando os parâmetros do algoritmo, como o tempo computacional máximo.

Uma alternativa para melhorar o desempenho do algoritmo BB proposto para resolver grandes instâncias é usar heurísticas para tentar encontrar soluções iniciais de alta qualidade para o problema. Essa estratégia é utilizada em algumas técnicas de resolução para melhorar as soluções nos nós da árvore BB, e para definir a próxima variável de separação e o próximo subproblema a ser resolvido.

4.2 TESTES COM O MODELO CA

Testes adicionais com o algoritmo BB proposto foram realizados considerando o modelo CA do problema de PERT utilizando o sistema Garver de 6 barras e o sistema IEEE de 118 barras com o objetivo de se avaliar o potencial da proposta de resolver essa formulação altamente complexa.

A Tabela 7 mostra as dimensões dos problemas, a Tabela 8 apresenta os resultados obtidos pelos solucionadores locais e globais e a Tabela 9 apresenta os resultados obtidos pelo algoritmo BB proposto.

Nas Tabelas 8 e 9 constam o custo de investimento da expansão da rede de cada sistema e o tempo computacional (entre parênteses) para encontrar a solução em cada teste. Os tempos com um asterisco indicam que o solucionador encerrou antes do limite de tempo de uma hora encontrando uma solução ótima local (ou global). Os tempos menores que 3600 s sem asterisco indicam o tempo que o solucionador demorou para encontrar a solução registrada, porém o limite de tempo foi utilizado para tentar encontrar uma solução melhor. Um valor em negrito indica que o

solucionador encontrou a melhor solução conhecida do problema. A sigla NC indica que o solucionador não conseguiu encontrar uma solução inteira para o problema no tempo máximo permitido.

Tabela 7 - Dimensões dos problemas considerando o modelo CA

Características dos problemas	Sistema	
	Garver de 6 barras	IEEE de 118 barras
Número de variáveis	92	1082
Limitadas abaixo somente	0	0
Limitadas abaixo e acima	27	251
Livres	65	831
Número de variáveis inteiras	15	25
Número de restrições	102	1308
Igualdades lineares	12	220
Igualdades quadráticas	0	14
Igualdades não lineares gerais	60	716
Desigualdades quadráticas unilaterais	30	358
Números não zeros na Jacobiana	496	5254
Números não zeros na Hessiana	196	1894

Fonte: Elaboração do próprio autor.

Tabela 8 - Resultados dos custos de investimentos dos sistemas para o modelo CA usando solucionadores locais/globais (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador					
		KNITRO	BONMIN	BARON	LINDO GLOBAL	COUENNE	SCIP
Garver de 6 barras	180,000	180,000 (<1 s) *	180,000 (2 s) *	–	190,000 (1766 s)	NC	600,000 (16 s)
IEEE de 118 barras	25,920	25,920 (1 s) *	25,920 (7 s) *	–	37,953 (3365 s)	25,920 (216 s) *	NC

Fonte: Elaboração do próprio autor.

Tabela 9 - Resultados dos custos de investimentos dos sistemas para o modelo CA usando o algoritmo BB proposto (MUS\$)

Sistema	Melhor resultado conhecido	Solucionador			
		IPOPT	KNITRO	LOQO	MINOS
Garver de 6 barras	180,000	180,000 (1 s) *	180,000 (1 s) *	180,000 (1 s) *	–
IEEE de 118 barras	25,920	25,920 (22 s)	25,920 (2 s)	25,920 (7 s)	–

Fonte: Elaboração do próprio autor.

As seções a seguir apresentam uma discussão com mais detalhes sobre os resultados obtidos para cada sistema.

4.2.1 Sistema Garver de 6 barras

Nas Tabelas 8 e 9, para o sistema Garver de 6 barras considerando o modelo CA a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 180,000 (cento e oitenta milhões de dólares) com a construção dos circuitos $n_{1-5} = 1$, $n_{2-3} = 1$, $n_{2-6} = 1$, $n_{3-5} = 1$ e $n_{4-6} = 3$.

A Tabela 8 mostra que os solucionadores KNITRO e BONMIN foram capazes de encontrar a solução ótima global do problema em até 2 s. O BARON não foi capaz de resolver o problema pois não é capaz de resolver problemas com funções trigonométricas. Os solucionadores globais LINDO GLOBAL, COUENNE e SCIP não convergiram para a melhor solução conhecida no tempo máximo de 3600 s. Por outro lado, analisando a Tabela 9, é possível verificar que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em até 1 s, com exceção do solver MINOS, que não foi capaz de resolver os subproblemas de PNL.

4.2.2 Sistema IEEE de 118 barras

Nas Tabelas 8 e 9, para o sistema IEEE de 118 barras considerando o modelo CA a solução ótima global deste problema de PERT tem um custo de investimento de MUS\$ 25,920 (vinte e cinco milhões e novecentos e vinte mil dólares) com a construção dos circuitos $n_{8-9} = 1$, $n_{9-10} = 1$ e $n_{99-} = 1$.

A Tabela 8 mostra que os solucionadores KNITRO e BONMIN foram capazes de encontrar a solução ótima global do problema em até 7 s. O BARON não foi capaz de resolver o problema pois não é capaz de resolver problemas com funções trigonométricas. Os solucionadores globais LINDO GLOBAL e SCIP não convergiram para a melhor solução conhecida no tempo máximo de 3600 s enquanto o COUENNE encontrou a melhor solução conhecida em 216 s. Por outro lado, analisando a Tabela 9, é possível verificar que o algoritmo BB proposto foi capaz de encontrar a solução ótima global com todos os solucionadores utilizados para resolver os subproblemas de PNL em até 22 s, com exceção do solver MINOS, que novamente não foi capaz de resolver os subproblemas de PNL. Cabe observar aqui que o solver KNITRO apresentou a melhor performance nesse caso, pois quando foi utilizado, o algoritmo BB foi capaz de resolver o problema em apenas 2 s.

4.2.3 Discussão

Pode-se concluir que os resultados obtidos com o algoritmo BB para o problema de PERT usando o modelo CA são competitivos quando comparados com os resultados obtidos pelos solucionadores locais/globais quando utilizados para resolver o problema diretamente, principalmente quando o algoritmo BB utilizou o KNITRO para resolver os subproblemas de PNL.

Por outro lado, cabe comentar sobre a necessidade do desenvolvimento de instâncias mais complexas para serem testadas utilizando-se o modelo CA, pois a literatura apresenta apenas sistemas com grande número de barras e que necessitam de poucas linhas novas instaladas para uma operação adequada, sendo, portanto, sistemas de baixa ou média complexidade para o problema de PERT.

5 CONCLUSÃO

Esse trabalho apresentou e discutiu a ineficácia e as dificuldades do algoritmo branch and bound (BB) tradicional para resolver problemas de programação linear inteira mista (PLIM) quando utilizado para resolver problemas não convexos de programação não linear inteira mista (PNLIM) e propôs uma versão modificada desse algoritmo BB para resolver essa categoria de problemas.

A implementação proposta do algoritmo BB incluiu estratégias para lidar com subproblemas de programação não linear (PNL), detectar a infactibilidade de um subproblema de PNL e tratar a não convexidade do problema, bem como regras de sondagem adequadas.

O algoritmo BB foi testado para resolver o problema de planejamento de expansão da rede de transmissão (PERT), que é um problema de PNLIM não convexo de difícil resolução. Quatro sistemas foram utilizados nos testes (Garver de 6 barras, IEEE de 24 barras, Sul Brasileiro de 46 barras e Colombiano de 93 barras) considerando os casos com o modelo CC sem e com perdas de potência ativa durante a transmissão e sem e com reprogramação de geração de potência ativa. Quatro solucionadores de otimização (IPOPT, KNITRO, LOQO, MINOS) foram usados para resolver os subproblemas de PNL. Além disso, para fins de comparação, os problemas também foram resolvidos diretamente usando dois solucionadores de otimização local (KNITRO e BONMIN) e cinco globais (BARON, LINDO GLOBAL, COUENNE, SCIP e GUROBI) para problemas de PNLIM não convexos. Testes adicionais foram realizados considerando o modelo CA com os sistemas Garver de 6 barras e IEEE de 118 barras.

Os resultados indicam que o algoritmo BB proposto apresenta resultados competitivos para o problema de PERT quando comparado com solucionadores comerciais e de código aberto para problemas de PNLIM não convexos, principalmente quando o solver LOQO é utilizado para resolver os subproblemas de PNL, pois com ele foi possível obter as melhores soluções conhecidas para todas as instâncias, inclusive quando o modelo CA é utilizado.

Trabalhos futuros desenvolverão melhorias para o algoritmo BB proposto neste trabalho, incluindo abordagens heurísticas para obter soluções factíveis de alta qualidade para o problema e determinar a variável de divisão e o próximo subproblema a ser resolvido. Além disso, novos estudos poderão definir uma proposta para a criação

de um algoritmo BB mais geral, onde possa ser possível efetuar a resolução de problemas de PNLIM com região de busca não convexa e função objetivo não linear.

REFERÊNCIAS

- ALGUACIL, N.; ARROYO, J. M.; CARRIÓN, M. Transmission network expansion planning under deliberate outages. **International Journal of Electrical Power & Energy Systems**, London, v. 31, n. 9, p. 365-389, 2010.
- ALGUACIL, N.; MOTTO, A. L.; CONEJO, A. J. Transmission expansion planning: A mixed-integer LP approach. **IEEE Transactions on Power Systems**, Piscataway, v. 18, n. 3, p. 1070-1077, 2003.
- ASADA, E. N. et al. **A branch-and-bound algorithm for the multi-stage transmission expansion planning**. IEEE Power Engineering Society General Meeting. San Francisco: IEEE. 2005. p. 171-176.
- BAHIENSE, L. et al. A mixed integer disjunctive model for transmission network expansion. **IEEE Transactions on Power Systems**, Piscataway, v. 16, n. 3, p. 560-565, 2001.
- BALAS, E.; TOTH, P. **Branch and bound methods for the traveling salesman problem**. Carnegie-Mellon University. Pittsburgh, p. 71. 1983.
- BELOTTI, P. **Couenne**: a user's manual. Clemson: [s. n.], 2009.
- BENICHO, M. et al. Experiments in mixed integer linear programming. **Mathematical Programming**, Heidelberg, v. 1, p. 76-94, 1971.
- BINATO, S.; PEREIRA, M. V. F.; GRANVILLE, S. A new Benders decomposition approach to solve power transmission network design problems. **IEEE Transactions on Power Systems**, Piscataway, v. 16, n. 2, p. 235-240, 2001.
- BONAMI, P. et al. An algorithmic framework for convex mixed integer nonlinear programs. **Discrete optimization**, Amsterdam, v. 5, n. 2, p. 186-204, 2008.
- BONAMI, P. et al. More branch-and-bound experiments in convex nonlinear integer programming. **Optimization Online**, [S.l.], v. 91, n. 1, p. 1-33, 2011.
- BRUIN, A. D.; KINDERVATER, G. A. P.; TRIENEKENS, H. W. J. M. Asynchronous parallel branch and bound and anomalies. In: FERREIRA, A. . R. J. **International Workshop on Parallel Algorithms for Irregularly Structured Problems**. Berlin: Springer, 1995. p. 363-377.
- BYRD, R. H.; NOCEDAL, J.; WALTZ, R. A. Knitro: An integrated package for nonlinear optimization. in **Large-Scale Nonlinear Optimization**, G. Di Pillo and M. Roma, Eds. **Boston, MA, USA: Springer**, Boston, p. 35-59, 2006.
- DA ROCHA, M. C.; SARAIVA, J. T. A discrete evolutionary PSO based approach to the multiyear transmission expansion planning problem considering demand uncertainties. **International Journal of Electrical Power & Energy Systems**, London, v. 45, n. 1, p. 427-442, 2013.
- DAKIN, R. J. A tree-search algorithm for mixed integer programming problems. **The Computer Journal**, Oxford, v. 8, n. 3, p. 250-255, 1965.

DELGADO, M. A. J.; POURAKBARI-KASMAEI, M.; RIDER, M. J. **A modified branch and bound algorithm to solve the transmission expansion planning problem**. 13th International Conference on Environment And Electrical Engineering (EEEIC). Wroclaw: [s. n.]. 2013. p. 234-238.

DOMINGUEZ, A. H. et al. **Transmission network expansion planning considering HVAC/HVDC lines and technical losses**. 2016 IEEE PES Transmission & Distribution Conference and Exposition-Latin America (PES T&D-LA). Morelia: [s. n.]. 2016. p. 1-6.

DOMINGUEZ, A. H. et al. Multistage security-constrained HVAC/HVDC transmission expansion planning with a reduced search space. **IEEE Transactions on Power Systems**, Piscataway, v. 32, n. 6, p. 4805-4817, 2017.

EASTMAN, W. L. **Linear programming with pattern constraints**. Cambridge: Ph.D. dissertation, Harvard University, The Computation Laboratory, 1958.

FANG, R.; HILL, D. J. A new strategy for transmission expansion in competitive electricity markets. **IEEE Transactions on Power Systems**, Piscataway, v. 18, n. 1, p. 374-380, 2003.

FOURER, R.; GAY, D. M.; KERNIGHAN, B. W. **AMPL: A modeling language for mathematical programming**. Duxbury: 2nd ed. Thomson, 2003.

FREITAS, P. F. S.; MACEDO, L. H.; ROMERO, R. Uma estratégia para resolver o problema de planejamento da expansão de sistemas de transmissão multietápico com múltiplos cenários de geração. *In*: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 22., 2018, João Pessoa. **Anais [...]**. João Pessoa: [s. n.], 2018. p. 1-6. Disponível em: https://www.sba.org.br/open_journal_systems/index.php/cba/article/view/687. Acesso em: 20 Mar 2022.

FREITAS, P. F. S.; MACEDO, L. H.; ROMERO, R. A strategy for transmission network expansion planning considering multiple generation scenarios. **Electric Power Systems Research**, Amsterdam, v. 172, [s. n.], p. 22-31, 2019.

FREITAS, P. F. S.; MACEDO, L. H.; ROMERO, R. Flexible solution approach for multistage transmission network expansion planning with multiple generation scenarios. **Journal of Control, Automation and Electrical Systems**, Heidelberg, v. 31, [s. n.], p. 705-717, 2020.

GALLEGO, R. A.; ROMERO, R.; MONTICELLI, A. J. Tabu search algorithm for network synthesis. **IEEE Transactions on Power Systems**, Piscataway, v. 15, n. 2, p. 490-495, 2000.

GARCÉS, L. P. et al. A bilevel approach to transmission expansion planning within a market environment. **IEEE Transactions on Power Systems**, Piscataway, v. 24, n. 3, p. 1513-1522, 2009.

GARVER, L. L. Transmission network estimation using linear programming. **IEEE Transactions on Power Apparatus and Systems**, Piscataway, v. 89, n. 7, p. 1688-1697, 1970.

GAUTHIER, J. M.; RIBIERE, G. Experiments in mixed-integer linear programming using pseudo-costs. **Mathematical Programming**, Heidelberg, v. 12, n. 1, p. 76-94, 1977.

GLANKWAMDEE, W.; LINDEROTH, J. **Lookahead branching for mixed integer programming**. Lehigh University. Bethlehem, p. 41. 2006.

GOMES, P. V.; SARAIVA, J. T. A two-stage strategy for security-constrained AC dynamic transmission expansion planning. **Electric Power Systems Research**, Amsterdam, v. 180, [s. n.], p. 1-10, 2020.

GUPTA, N. et al. A bi-level evolutionary optimization for coordinated transmission expansion planning. **IEEE Access**, Piscataway, v. 6, [s. n.], p. 48455-48477, 2018.

GUPTA, O. K.; RAVINDRAN, A. Branch and bound experiments in convex nonlinear integer programming. **Management Science**, Catonsville, v. 31, n. 12, p. 1533-1546, 1985.

GUROBI OPTIMIZATION INC. Gurobi optimizer reference manual. Disponível em: <http://www.gurobi.com>. Acesso em: 20 Mar 2022.

HAFFNER, S. **O planejamento da expansão dos sistemas elétricos no contexto de um ambiente competitivo**. 2000. 167 f. Tese (Doutorado em Engenharia Elétrica) – Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2000. Disponível em: <http://repositorio.unicamp.br/Acervo/Detalhe/263038>. Acesso em: 20 Mar 2022.

HAFFNER, S. et al. Branch and bound algorithm for transmission system expansion planning using a transportation model. **IEE Proceedings-Generation, Transmission and Distribution**, Stevenage, v. 147, n. 3, p. 149-156, 2000.

HAFFNER, S. et al. Specialised branch-and-bound algorithm for transmission network expansion planning. **IEE Proceedings-Generation, Transmission and Distribution**, Stevenage, v. 148, n. 5, p. 482-488, 2001.

HEMMATI, R.; HOOSMAND, R. A.; KHODABAKHSHIAN, A. Comprehensive review of generation and transmission expansion planning. **IET Generation, Transmission & Distribution**, Stevenage, v. 7, n. 9, p. 955-964, 2013.

HONG, S.; CHENG, H.; ZENG, P. N-k constrained composite generation and transmission expansion planning with interval load. **IEEE Access**, Piscataway, v. 5, [s. n.], p. 2779-2789, 2017.

HUA ET AL, L. I. U. **MINOS 5.51 user's guide**. Stanford University. Stanford, p. 158. 2003.

KILINÇ, M. R.; SAHINIDIS, N. V. Exploiting integrality in the global optimization of mixed-integer nonlinear programming problems with BARON. **Optimization Methods and Software**, Abingdon, v. 33, n. 3, p. 540–562, 2018.

LAND, A. H.; DOIG, A. G. An automatic method of solving discrete programming problems. **Econometrica**, Chichester, v. 28, n. 3, p. 497–520, 1960.

LAPSEE. Power System Test Cases Repository. Disponível em: <http://www.feis.unesp.br/#!/lapsee>. Acesso em: 01 out. 2021.

LATORRE, G. et al. Classification of publications and models on transmission expansion planning. **IEEE Transactions on Power Systems**, Piscataway, v. 18, n. 2, p. 938-946, 2003.

LAWLER, E. L.; WOOD, D. E. Branch-and-bound methods: A survey. **Operations Research**, Catonsville, v. 14, n. 4, p. 699-719, 1966.

LEE, C. W. et al. **Transmission expansion planning from past to future**. 006 IEEE PES Power Systems Conference and Exposition. Atlanta: [s. n.]. 2006. p. 257-265.

LINDEROTH, J. T.; SAVELSBERGH, M. W. P. A computational study of search strategies for mixed integer programming. **Inform Journal on Computing**, Catonsville, v. 11, n. 2, p. 173-187, 1999.

LINDO SYSTEMS INC. Disponível em: <<https://www.lindo.com/>>. Acesso em: 26 Dec. 2021.

LITTLE, J. D. C. et al. An algorithm for travelling salesman problem. **Operations Research**, Catonsville, v. 11, [s. n.], p. 979-989, 1963.

LUBURIĆ, Z.; PANDŽIĆ, H.; CARRIÓN, M. Transmission expansion planning model considering battery energy storage, TCSC and LINEs using AC OPF. **IEEE Access**, Piscataway, v. 8, [s. n.], p. 203429-203439, 2020.

MACEDO, L. H. et al. MILP branch flow model for concurrent AC multistage transmission expansion and reactive power planning with security constraints. **IET Generation, Transmission & Distribution**, Stevenage, v. 10, n. 12, p. 3023-3032, 2016.

MAHDAVI, M. . K. A. et al. Transmission expansion planning considering power losses, expansion of substations and uncertainty in fuel price using discrete artificial bee colony algorithm. **IEEE Access**, Piscataway, v. 9, [s. n.], p. 135983-135995, 2021.

MAHDAVI, M. et al. Transmission expansion planning: literature review and classification. **IEEE Systems Journal**, Piscataway, v. 13, n. 3, p. 3129-3140, 2018.

MAJIDI-QADIKOLAI, M.; BALDICK, R. A generalized decomposition framework for large-scale transmission expansion planning. **IEEE Transactions on Power Systems**, Piscataway, v. 33, n. 2, p. 1635-1649, 2017.

MENESES, M. et al. Transmission network expansion planning considering line switching. **IEEE Access**, Piscataway, v. 8, [s. n.], p. 115148-115158, 2020.

MIASAKI, C. T. **Planejamento da expansão do sistema de transmissão de energia elétrica utilizando controladores FACTS**. 2006. 137 f. Tese (Doutorado em Engenharia Elétrica), Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista - UNESP, Ilha Solteira, 2006. Disponível em: <https://repositorio.unesp.br/handle/11449/100368>. Acesso em: 01 out. 2021.

MITTEN, L. G. Branch-and-bound methods: General formulation and properties. **Operations Research**, Catonsville, v. 18, [s. n.], p. 24-34, 1970.

MONTES, C. V. et al. Abordagem de PLIM para o planejamento da expansão de sistemas de transmissão usando o modelo AC. *In*: SIMPÓSIO BRASILEIRO DE SISTEMAS

- ELÉTRICOS, 6., 2016, Natal. **Anais [...]** Natal: UFRN; FUNPEC, 2016. p. 1-6. Disponível em:
https://www.researchgate.net/publication/305200905_Abordagem_de_PLIM_para_o_Planejamento_da_Expansao_de_Sistemas_de_Transmissao_Usando_o_Modelo_AC. Acesso em: 01 out. 2021.
- MONTICELLI, A. J. **Fluxo de carga em redes de energia elétrica**. São Paulo: Blucher, 1983.
- MURUGAN, P. Modified particle swarm optimisation with a novel initialisation for finding optimal solution to the transmission expansion planning problem. **IET Generation, Transmission & Distribution**, Stevenage, v. 6, n. 11, p. 1132-1142, 2012.
- ORIONDO, M. A. M. **Planejamento da expansão de sistemas de transmissão considerando o modelo CA e desligamento na operação de linhas de transmissão**. 2020. 243 f. Tese (Doutorado em Engenharia Elétrica), Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista - UNESP, Ilha Solteira, 2020. Disponível em:
<https://repositorio.unesp.br/handle/11449/193481>. Acesso em: 01 out. 2021.
- RASTGOU, A.; MOSHTAGH, J. Improved harmony search algorithm for transmission expansion planning with adequacy–security considerations in the deregulated power system. **International Journal of Electrical Power & Energy Systems**, London, v. 60, [s. n.], p. 153-164, 2014.
- REEVES, C.; GLOVER, F.; KOCHENBERGER. **Handbook of metaheuristics**. Dordrecht: Kluwer Academic Publishers, 2003. 55-82 p.
- RIDER, M. J. **Planejamento da expansão de sistemas de transmissão usando os modelos CC-CA e técnicas de programação não linear**. 235 f. 2006. Tese (Doutorado em Engenharia Elétrica), Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual DE Campinas - UNICAMP. Campinas, 2006. Disponível em:
<http://repositorio.unicamp.br/Acervo/Detalhe/365011>. Acesso em: 01 out. 2021.
- RIDER, M. J.; GARCIA, A. V.; ROMERO, R. Power system transmission network expansion planning using AC model. **IET Generation, Transmission & Distribution**, Stevenage, v. 1, n. 5, p. 731-742, 2007.
- RIDER, M. J.; GARCIA, A. V.; ROMERO, R. Transmission system expansion planning by a branch-and-bound algorithm. **IET Generation, Transmission & Distribution**, Stevenage, v. 2, n. 1, p. 90-99, 2008.
- ROMERO, R. A. **Planejamento a longo prazo da expansão de sistemas de transmissão de energia elétrica**. 1999. 137 f. Tese (Livre Docência), Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista - UNESP, Ilha Solteira, 1999.
- SAHINIDIS, N. **BARON user manual v. 15.6. 5**. The Optimization Firm. [S. l.], p. 30. 2015.
- SCHIAVINATO, M.; MACEDO, L. H.; ROMERO, R. **Análise crítica do modelo híbrido linear no planejamento da expansão de sistemas de transmissão**. XXIV Congresso Brasileiro de Automática. Fortaleza: [s. n.]. 2022. p. 1-7.

SILVA, E. F. **Planejamento estocástico da expansão da rede de transmissão de energia elétrica multiestágio considerando restrições de segurança**. 2013. 170 f. Tese (Doutorado em Engenharia Elétrica), Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista - UNESP, Ilha Solteira, 2013. Disponível em: <https://repositorio.unesp.br/handle/11449/100350?show=full>. Acesso em: 01 out. 2021.

SILVA, S. M. et al. Efficient multi-start with path relinking search strategy for transmission system expansion planning. **IEEE Access**, Piscataway, v. 9, [s. n.], p. 153213-153225, 2021.

UDE, N. G.; YSKANDAR, H.; GRAHAM, R. C. A comprehensive state-of-the-art survey on the transmission network expansion planning optimization algorithms. **IEEE Access**, Piscataway, v. 7, [s. n.], p. 123158-123181, 2019.

VANDERLEI, R. J. LOQO: an interior point code for quadratic programming. **Optimization Methods and Software**, Abingdon, v. 11, n. 1-4, p. 451-484, 1999.

VARGAS, L. M. E. **Specialized models for the long-term transmission network expansion planning problem**. 2018. 123 f. Tese (Doutorado em Engenharia Elétrica), Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista - UNESP, Ilha Solteira, 2018. Disponível em: <https://repositorio.unesp.br/handle/11449/180548>. Acesso em: 01 out. 2021.

VERMA, A.; PANIGRAHI, B. K.; BIJWE, P. R. Harmony search algorithm for transmission network expansion planning. **IET Generation, Transmission & Distribution**, Stevenage, v. 4, n. 6, p. 663-673, 2010.

VIGERSKE, S.; GLEIXNER, A. SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. **Optimization Methods and Software**, Abingdon, v. 33, n. 3, p. 563-593, 2018.

WÄCHTER, A.; BIEGLER, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. **Mathematical Programming**, Heidelberg, v. 106, n. 1, p. 25-57, 2006.

ZOPPEI, R. T. et al. A branch and bound algorithm for transmission network expansion planning using nonconvex mixed-integer nonlinear programming models. **IEEE Access**, Piscataway, v. 10, [s. n.], p. 39875-39888, 2022.

APÊNDICE – TRABALHOS PUBLICADOS PELO AUTOR

ZOPPEI, R. T.; DELGADO, M. A. J. **Causas da ineficácia do algoritmo branch and bound na resolução de problemas de programação não linear inteiro misto**, Anais do I Encontro Regional de Matemática Aplicada e Computacional – UFMS, Campo Grande, 2020.

ZOPPEI, R. T.; DELGADO, M. A. J.; MACEDO, L. H.; RIDER, M. J.; ROMERO R. A. branch and bound algorithm for transmission network expansion planning using non-convex mixed-integer nonlinear programming models. **IEEE Access**, Piscataway, v. 10, p. 39875-39888, 2022.