UM ESTUDO DE UM PROBLEMA DE PROGRAMAÇÃO EM DOIS NÍVEIS: O PROBLEMA DE ESTRATÉGIA DE PREÇO SOB INCERTEZA EM MERCADO DE ENERGIA

Wagner Pimentel

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: Márcia Helena Costa Fampa

Rio de Janeiro
Dezembro de 2014

UM ESTUDO DE UM PROBLEMA DE PROGRAMAÇÃO EM DOIS NÍVEIS: O PROBLEMA DE ESTRATÉGIA DE PREÇO SOB INCERTEZA EM MERCADO DE ENERGIA

Wagner Pimentel

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Márcia Helena Costa Fampa, Ph.D.

Prof. Nelson Maculan Filho, Ph.D.

Prof. Fernanda Maria Pereira Raupp, Ph.D.

Prof. Jose Andre de Moura Brito, Ph.D.

Prof. Olinto César Bassi de Araújo, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
DEZEMBRO DE 2014

Pimentel, Wagner

Um estudo de um Problema de Programação em Dois Níveis: o Problema de Estratégia de Preço sob Incerteza em Mercado de Energia/Wagner Pimentel. – Rio de Janeiro: UFRJ/COPPE, 2014.

XIV, 88 p.: il.; 29, 7cm.

Orientador: Márcia Helena Costa Fampa

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2014.

Referências Bibliográficas: p. 84 – 88.

1. Bilevel Problem. 2. Problem of Price Strategy. 3. Genetic Algoritms. 4. Semidefinite Programming. I. Fampa, Márcia Helena Costa. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Dedico esta conquista a minha família: Andrea de Azevedo Cardoso, Rodrigo de Azevedo Cardoso Pimentel e Anne de Azevedo Cardoso Pimentel. Dedico aos meus pais Walter Barros Pimentel e Marilda Pimentel pela educação e ensinamentos. Dedico, também, a todas as pessoas que passaram pela minha vida e de alguma forma contribuíram para a minha formação.

Agradecimentos

Gostaria de agradecer primeiramente a Deus por me proporcionar mais uma conquista. Agradeço ao PESC e às instituições COPPE, UFRJ e CEFET/RJ. Agradeço com muito carinho a minha orientadora Márcia Helena Costa Fampa que me conduziu com muita sabedoria e dedicação no desenvolvimento desta pesquisa. A todos os professores da linha de pesquisa de otimização pelos ensinamentos.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

UM ESTUDO DE UM PROBLEMA DE PROGRAMAÇÃO EM DOIS NÍVEIS: O PROBLEMA DE ESTRATÉGIA DE PREÇO SOB INCERTEZA EM MERCADO DE ENERGIA

Wagner Pimentel

Dezembro/2014

Orientador: Márcia Helena Costa Fampa

Programa: Engenharia de Sistemas e Computação

Apresentamos neste trabalho um estudo do problema de estratégia de preço sob incerteza em um mercado de energia. Na primeira parte deste estudo, cujo objetivo é produzir limites inferiores para as instâncias do problema, foi desenvolvido um algoritmo genético a partir do modelo de programação em dois níveis do referido problema. Apresentamos uma reformulação do problema de estratégia de preço como um problema quadrático com restrições quadráticas. Aplicamos a técnica de relaxação semidefinida e desenvolvemos um algoritmo de plano de corte SDP para o modelo relaxado do problema, reforçado por cortes derivados do produto de restrições do problema, com o objetivo de determinar fortes limites superiores. Por fim, aplicamos a técnica de relaxação linear estendida e desenvolvemos outro algoritmo de plano de corte para o modelo relaxado do problema, reforçado por cortes derivados do produto de restrições do problema e por cortes derivados da decomposição espectral da matriz solução da relaxação, objetivando determinar fortes limites superiores a um reduzido custo computacional.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

APPROACHES FOR SOLVING A BILEVEL PROBLEM: THE STRATEGIC PRICING PROBLEM UNDER UNCERTAINTY IN ENERGY MARKET

Wagner Pimentel

December/2014

Advisor: Márcia Helena Costa Fampa

Department: Systems Engineering and Computer Science

In this work we study the problem of strategic pricing under uncertainty in an electricity market. In the first part of the study presented, where the goal is to produce lower bounds for instances of the problem, was developed a genetic algorithm from the bilevel programming model of the problem. We also present a reformulation of the strategic pricing problem as a quadratic problem with quadratic constraints. We then consider a semidefinite programming (SDP) relaxation of the problem and develop a cutting plane SDP algorithm for the relaxation, strengthened by cuts derived from the product of the constraints of the problem, aiming at determining strong upper bounds. Finally, we consider an extended linear relaxation of the problem and develop another cutting plane algorithm for the relaxation, enhanced by cuts derived from the product of the problem constraints and cuts derived from the spectral decomposition of the solution matrix of the relaxation, aiming at determining strong upper bounds at a low computational effort.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

AE	Algoritmo Evolucionário, p. 27
AG	Algoritmo Genético, p. 1
ANNEL	Agência Reguladora de Energia Elétrica, p. 14
APC1	Algoritmo de Plano de Corte RLE, p. 2
ASDP	Algoritmo de Plano de Corte SDP, p. 2
BMO	Boletim Mensal de Operação do Sistema Integrado Nacional, p. 14
BRKGA	Algoritmo Genético de Chaves Aleatórias Viciadas (*Biased Random Key Genetic Algorithms*), p. 83
B	Geração de Biomassa, p. 14
CEEE	Companhia Estadual de Energia do Rio Grande do Sul, p. 22
CESP	Companhia Energética de São Paulo, p. 19
DPR	Desvio Percentual Relativo, p. 40
H	Geração Hidrelétrica, p. 14
LP	Problema de Programação Linear (*Linear Programming*), p. 11
MILP	Problema de Programação Inteira Mista (*Mixed Integer Linear Programming*), p. 1

MIQCP	Problema de Programação Inteira Mista com Restrições Quadráticas (Mixed Integer Quadratically Constrained Programs), p. 50
N	Geração Nuclear, p. 14
ONS	Operador Nacional do Sistema Brasileiro de Energia Elétrica, p. 14
PEP	Problema de Estratégia de Preço, p. 1
PPDN	Problema de Programação em Dois Níveis, p. 1
QCQP	Problema Quadrático com Restrições Quadráticas (*Quadratically constrained quadratic program*), p. 1
RLC	Relaxação Linear Contínua do Modelo MILP, p. 2
RLE	Relaxação Linear Estendida, p. 2
RLT	Técnica de Reformulação por Linearização (*Reformulation Linearization Techniques*), p. 2
SDP	Problema de Programação Semidefinida (*Semidefinite Programming*), p. 1
T	Geração Termoelétrica, p. 14

Capítulo 1

Introdução

Apresentamos neste trabalho um estudo do problema de estratégia de preço sob incerteza em um mercado de energia (PEP). O mercado de venda de energia por atacado trabalha da seguinte forma: todos os geradores de energia ofertam, livremente, preços por unidade de energia e a sua capacidade de produção disponível. O operador do sistema despacha os geradores em ordem crescente de preço até que a demanda do sistema seja alcançada. Os geradores despachados são pagos pelo preço da unidade despachada de maior preço, preço *spot* do sistema. Isto corresponde ao bem conhecido formato de leilão de preço uniforme, que é geralmente adotado nos mercados de eletricidade e será considerado neste trabalho. Neste mercado, a remuneração econômica, ou o lucro, de cada gerador de energia depende da sua habilidade de administrar submissões de ofertas de preço e quantidade de energia elétrica. Este problema pode ser visto como um problema de programação em dois níveis, com dois atores: os geradores, interessado em maximizar o seu lucro, e o operador do sistema, interessado em minimizar o custo de produção associado ao mercado de energia. Este problema é estocástico e não-convexo e devido a sua dificuldade existe uma busca intensiva por algoritmos eficientes para a sua solução.

Na primeira parte deste estudo, realizamos uma revisão bibliográfica do PEP e da aplicação de metaheurística a problemas de programação em dois níveis (PPDN) com o objetivo de adquirir mais conhecimento e experiência para o início do trabalho. Com base na formulação do problema de estratégia de preço como um problema de programação em dois níveis, desenvolvemos um algoritmo genético (AG). Experimentos numéricos em instâncias com configuração derivada do sistema brasileiro de energia demonstram a qualidade dos resultados obtidos com a metaheurística proposta. A opção pela utilização de metaheurística como proposta de solução para o PEP foi feita devido a dificuldade da utilização de um algoritmo exato aplicado a uma formulação de programação linear inteira mista (MILP), na solução de instâncias reais derivadas do sistema brasileiro de energia, com elevado número de cenários. Na sequência desta pesquisa, realizamos uma reformulação do PEP como um problema quadrático com restrições quadráticas (QCQP) não-convexo e efetuamos um estudo da aplicação de relaxação semidefinida (SDP) ao QCQP. Apresentamos um algoritmo de plano de corte SDP (ASDP) para o modelo relaxado do QCQP, com o objetivo de determinar fortes limites superiores para as instâncias do PEP. Utilizamos restrições obtidas da técnica de reformulação por linearização (RLTs) para reforçar as relaxações. Experimentos numéricos em instâncias reais com configuração derivada do subsistema da região sul brasileira demonstram a qualidade dos resultados obtidos com o ASDP, quando comparados com os resultados obtidos pela solução da relaxação linear contínua (RLC) do MILP associado ao PEP. Estes resultados evidenciam a qualidade da aplicação de relaxação semidefinida combinada com as restrições RLTs na obtenção de fortes limites superiores para um QCQP. Por fim, utilizando a reformulação do PEP como um QCQP não-convexo, aplicamos a técnica de relaxação linear estendida (RLE) ao QCQP e apresentamos um algoritmo de plano de corte RLE (APC1) para o modelo linear relaxado do QCQP com o objetivo de determinar, da mesma forma, fortes limites superiores para as instâncias do PEP e ser uma alternativa eficiente, computacionalmente, à técnica de programação semidefinida. Utilizamos restrições RLTs junto com as restrições derivadas da decomposição espectral da matriz solução da relaxação do QCQP, objetivando reforçar os limites superiores para as instâncias do PEP a um baixo custo computacional. Utilizamos, também, a técnica de cortes esparsos derivados da decomposição espectral para agilizar o processo de reotimização do APC1. Experimentos numéricos em instâncias com configuração derivada do subsistema de energia da região sul do Brasil demonstram a qualidade dos resultados obtidos com o APC1 quando comparados com os resultados obtidos pela solução da RLC do modelo MILP associado ao problema, apresentando ainda um reduzido custo computacional em comparação com o ASDP.

Capítulo 2

Parte I: O Problema de Estratégia de Preço como um Problema de Programação em Dois Níveis

2.1 Revisão Bibliográfica do Problema de Estratégia de Preço

Nesta seção apresentamos uma revisão bibliográfica do PEP cujo objetivo é apresentar as pequisas relacionadas ao referido problema. Inicialmente, uma revisão sobre modelos de programação matemática para o PEP pode ser encontrada em [1], em que destacamos os seguintes trabalhos que dividem os modelos em: problema de programação linear com restrições de complementaridade, MILP e PPDN. RAMOS et al. [2], apresentam uma formulação não-linear para o PEP com restrição de equilíbrio. CONEJO e PRIESTO [3], e CONEJO et al. [4] propõem alguns procedimentos heurísticos e HOBBS [5] usa modelos lineares com restrição de complementaridade para o problema. HOBBS e HELMAN [6] apresentam uma revisão completa de aplicações para os modelos baseados em restrição de complementaridade para mercados de energia, e BUSHNELL [7] apresenta uma aplicação de um modelo com restrições de complementaridade para sistema hidrotermal, onde os agentes atuam como jogadores estratégicos.

MILP tem sido utilizado para modelar o problema, por exemplo, em [8], onde uma função não-convexa de demanda residual permite o cálculo da oferta estratégica ótima em um ambiente com barramento simples, sem rede de transmissão, como ilustrado no exemplo derivado do sistema espanhol. BAÍLLO et al. [9] também apresentam uma função de demanda residual para um modelo MILP, mas consideram incerteza e estabelecem cenários da funções de demanda residual para a sessão de mercado do dia seguinte.

PPDN também tem sido usado no contexto de mercado de energia elétrica para caracterizar sua estrutura hierárquica. O modelo de PPDN pode ser dividido em dois estágios: o primeiro estágio, denominado líder, escolhe sua posição ótima e, no segundo estágio, denominado seguidor, o decisor otimiza sua função objetivo dado uma determinada posição do líder. Em [10] e [11], o PEP foi abordado como um PPDN aninhado, onde o gerador maximiza o seu bem-estar sujeito a uma solução que maximiza o bem-estar total social baseado em todas as ofertas do mercado. Ambas as abordagens consideram a dinâmica associada dos agentes do mercado, que é modelada por procedimentos iterativos que buscam por um equilíbrio de $Nash$ [12].

Em [13], os autores abordam uma versão mono-estágio de um PPDN onde as ofertas dos competidores são modeladas por meio de cenários. O problema é formulado como um classe especial do PPDN conhecido como problema de taxação [14] e então reformulado como um MILP. Um procedimento heurístico para o problema é também apresentado neste artigo. Utilizaremos este MILP, na sequência deste trabalho, para gerar a solução ótima ou a melhor solução viável conhecida para as instâncias do PEP. Em [15] uma formulação MILP é também apresentada para o problema.

2.2 Problema de Programação em Dois Níveis

Um PPDN consiste em um problema de otimização, denominado problema líder, que é restrito por um outro problema de otimização, denominado problema seguidor. O PPDN surge quando dois tomadores de decisão independentes possuem caráter não cooperativo [16], ou seja, cada tomador de decisão procura otimizar o seu objetivo baseado na tomada de decisão do seu oponente. O líder e o seguidor jogam um *Stackelberg duopoly game* [17]. Em [17], um modelo genérico do PPDN é representado como:

$$(\text{BP}) \begin{cases} \text{Min}_{y \in Y} & \varphi(x(y), y), & \text{(Líder)} \\ s.a & \psi(x(y), y) \leq 0, & \\ onde & x(y) = arg\min_{x \in X} f(x, y), & \text{(Seguidor)} \\ s.a & g(x, y) \leq 0. & \end{cases} \tag{2.1}$$

$Y \subset \mathbb{R}^m$ e $X \subset \mathbb{R}^n$ são conjuntos fechados e representam o espaço de busca do problema líder e seguidor, respectivamente. $\psi : X \times Y \rightarrow \mathbb{R}^p$ e $g : X \times Y \rightarrow \mathbb{R}^q$ são funções vetoriais que representam as restrições dos problemas. $\varphi : X \times Y \rightarrow R$ e $f : X \times Y \rightarrow R$ são funções de valores reais que representam as funções objetivo dos problemas. O conjunto $S = \{(x, y) : x \in X, y \in Y, \psi(x, y) \leq 0, g(x, y) \leq 0\}$ é o conjunto restrição do BP. Para um determinado $y \in Y$, o conjunto $X(y) = \{x \in X : g(x, y) \leq 0\}$ é o conjunto viável do problema seguidor. O conjunto $R(y) = \{x \in X : x \in arg\min_{w \in X(y)} f(w, y)\}$ é chamado de conjunto reação racional do BP, para determinado valor de $y \in Y$. O conjunto viável do BP é $F = \{(x, y) \in S : x \in R(y)\}$. Um ponto viável $(x^*, y^*) \in F$ é um *Stackelberg equilibrium* (com o primeiro jogador como sendo o líder) se $\varphi(x^*, y^*) \leq \varphi(x, y)$ para todo $(x, y) \in F$.

Em [18, 19] foi provado que o problema de programação linear em dois níveis é um problema NP-*Hard*. Entende-se como problema de programação linear em dois níveis um PPDN em que tanto o problema líder quanto o problema seguidor são lineares. Para um maior entendimento desta classificação, *ODUGUWA* e *ROY* [20] fizeram uma descrição de cinco classes de PPDN. O PPDN utilizado neste trabalho, que modela o PEP, pertence a uma das classes descritas em [20], sendo o problema líder correspondente a um problema de programação não-linear e o problema seguidor correspondente a um problema de programação linear.

2.3 Problema de Estratégia de Preço em Mercado de Energia

No PEP, o tradicional planejamento de operação baseado em otimização centralizada é substituído por um procedimento descentralizado, baseado no funcionamento do mercado. Neste problema os agentes privados, geradores, competem por contratos para venda de energia para empresas distribuidoras e consumidores livres. Os geradores podem livremente fazer suas ofertas de preço para produção de energia elétrica. Baseado nas ofertas dos geradores, as unidades de energia são então carregadas (despachadas) em ordem crescente de oferta de preço até que a demanda do sistema seja atendida. Todos os geradores despachados recebem um preço por unidade de energia equivalente ao preço do último gerador despachado (maior preço ofertado), que corresponde ao custo marginal ou preço *spot* do sistema [21].

Uma das características básicas do processo desregulado é a criação de um mercado de venda de energia por atacado, onde todas as ofertas para compra de energia e as transações de venda são colocadas. De um modo simples, o mercado de venda de energia por atacado trabalha da seguinte forma:

-Todos os geradores ofertam, livremente, preços para produção de energia, tipicamente, oferta de preço-quantidade em um período, para o próximo dia.

-O operador do sistema despacha as unidades de geração em ordem crescente de preço até que a demanda do sistema seja alcançada. Os geradores despachados são pagos pelo preço da unidade despachada de maior preço, preço *spot* do sistema. Isto corresponde ao bem conhecido formato de leilão de preço uniforme, que é geralmente adotado nos mercados de eletricidade e será considerado neste trabalho.

A existência de um despacho baseado em oferta tem motivado ou produzido alguns desafios técnicos que têm sido amplamente discutidos e estudados na literatura [3, 22–24].

No mercado desregulado de energia, o sistema de despacho e o preço *spot* dependem da oferta de preço e da quantidade de energia de cada agente gerador, ou seja, estes agentes assumem muito mais riscos e tornam-se altamente responsáveis por suas decisões. Por outro lado, o operador do sistema estará sempre interessado em analisar as estratégias com o objetivo de prevenir abusos de mercado.

Portanto, no mercado de eletricidade desregulado, os geradores submetem um conjunto de preços por geração, por período, e a capacidade disponível para o próximo dia. Baseado nesses dados de entrada, sobre o horário de pico de carga, o operador do sistema realiza o seguinte despacho econômico a cada período [13]:

$$(\text{DE})\begin{cases} \text{Min}_{g_j} & \sum_{j\in J} \lambda_j g_j, & (\text{variável dual}) & \\ \text{s. a} & \sum_{j\in J} g_j = d, & (\pi_d) & \\ & g_j \leq \bar{g}_j, & (\pi_{g_j}) & j\in J, \\ & g_j \geq 0, & & j \in J, \end{cases} \tag{2.2}$$

onde os dados de entrada d, λ_j e $\bar{g}_j$ representam, respectivamente, demanda do sistema (MWh), oferta de preço ($R\$/MWh$) e a capacidade de geração (MWh) ofertada do gerador j. As variáveis g_j representam a produção de energia do gerador j (MWh). O valor ótimo da variável dual π_d é considerado o preço *spot* do sistema. O lucro de cada gerador j, em cada período, corresponde à expressão: $(\pi_d - c_j)g_j$, para cada $j \in J$, onde c_j representa o custo operacional unitário. Observe que o custo operacional para um determinado gerador pode ser diferente da sua oferta de preço.

O lucro líquido total de uma empresa geradora de energia E, empresa ofertante, é dado por:

$$\sum_{j\in E}(\pi_d - c_j)g_j,$$

onde E é também utilizado para denotar um conjunto de índices associados aos geradores pertencente à empresa ofertante ($E \subset J$).

Uma empresa geradora de energia E representa uma empresa pública, ou privada, produtora de energia que possua algumas unidades de geração própria. A empresa E tem por objetivo determinar um conjunto de ofertas de preço $\lambda_E = \{\lambda_j, j \in E\}$ e quantidade de energia $\bar{g}_E = \{\bar{g}_j, j \in E\}$ que maximize o seu lucro líquido total.

2.3.1 Problema de Oferta Ótima sob Incerteza

Nós consideramos o esquema de *Bertrand* para o PEP, onde a quantidade ofertada por cada gerador da empresa E é fixada como $\bar{g}_j$. O problema consiste em determinar a oferta de preço da empresa que produz o lucro máximo. A complexidade deste problema é aumentada pelo fato de que o cálculo de π_d e g_j, no problema de despacho econômico (2.2), depende do conhecimento do vetor de preço de todas as empresas, bem como das disponibilidades de geração e do valor da demanda do sistema. Contudo, estes dados não estão disponíveis para nenhuma empresa no momento da sua oferta, na prática. Portanto, a oferta estratégica tem que levar em conta a incerteza associada a esses valores [13]. A abordagem utilizada para tratar com a incerteza sobre os dados do problema, neste trabalho, define um conjunto de cenários para os agentes concorrentes e busca maximizar o lucro da empresa sobre todos os cenários. Neste caso, as ofertas dos geradores concorrentes e a demanda do sistema são incertos, e são representados por um conjunto S de cenários indexados por s, que ocorrem com probabilidade exógena p_s, (s=1,...,|S|). A formulação em dois níveis para este problema é dado por:

$$
(\text{PPDN}) \left\{
\begin{array}{llll}
\text{Max}_{\lambda_E} & \sum\limits_{s \in S} p_s \sum\limits_{j \in E} (\pi_d^s - c_j) g_j^s, & & \\
\text{s. a} & & & \\
& \text{Min}_{g_j^s} \;\; \sum\limits_{s \in S} \sum\limits_{j \in E} \lambda_j g_j^s + \sum\limits_{j \in J \setminus E} \bar{\lambda}_j^s g_j^s, & & \\
& \text{s. a} \;\; \sum\limits_{j \in J} g_j^s = d^s, & & s \in S, \\
& \qquad 0 \leq g_j^s \leq \bar{g}_j, & j \in E, & s \in S, \\
& \qquad 0 \leq g_j^s \leq \bar{g}_j^s, & j \in J \setminus E, & s \in S.
\end{array}
\right. \tag{2.3}
$$

onde o problema líder representa o interesse da empresa E, maximizar o lucro esperado, e o problema seguidor representa o interesse do operador do sistema, minimizar o custo operacional do sistema. A empresa ofertante é classificada como líder do PPDN e controla as variáveis λ_j, para $j \in E$, enquanto o operador do sistema é classificado como seguidor do PPDN e controla as variáveis g_j^s para $j \in J$, $s \in S$.

Uma vez que as variáveis do líder são determinadas ou fixadas, o problema de programação linear (LP), o problema seguidor, pode ser resolvido com o auxílio de qualquer resolvedor de otimização. Devido a sua estrutura, este LP pode ser resolvido separadamente para cada cenário. Cada subproblema pode ser resolvido, para cada um dos cenários, por um procedimento polinomial que descreveremos no capítulo (3). Algumas abordagens diferentes foram apresentadas na literatura para resolver o PEP. Em FAMPA *et al.* [13], o PEP foi formulado como uma classe especial do PPDN, conhecida como o problema de taxação [14]. Uma reformulação do PEP como um problema de programação matemática com restrição de equilíbrio foi apresentada em [13]. Esta reformulação foi obtida do modelo (2.3), substituindo o problema seguidor por suas condições de otimalidade:

$$
(\mathrm{QP}) \left\{
\begin{array}{lll}
\mathrm{Max}_{\lambda_E} \quad \sum\limits_{s \in S} p_s \sum\limits_{j \in E} (\pi_d^s - c_j) g_j^s & & \\
\text{s. a} & & \\
\sum\limits_{j \in J} g_j^s = d^s, & & s \in S, \\
0 \leq g_j^s \leq \bar{g}_j, & j \in E, & s \in S, \\
0 \leq g_j^s \leq \bar{g}_j^s, & j \in J \setminus E, & s \in S, \\
\pi_d^s + \pi_{g_j}^s - \lambda_j \leq 0, & j \in E, & s \in S, \\
\pi_d^s + \pi_{g_j}^s \leq \bar{\lambda}_j^s, & j \in J \setminus E, & s \in S, \\
\pi_{g_j}^s \leq 0, & j \in J, & s \in S, \\
\sum\limits_{s \in S} \Big(\sum\limits_{j \in E} \lambda_j g_j^s + \sum\limits_{j \in J \setminus E} \bar{\lambda}_j^s g_j^s - d^s \pi_d^s - \sum\limits_{j \in E} \bar{g}_j \pi_{g_j}^s - \sum\limits_{j \in J \setminus E} \bar{g}_j^s \pi_{g_j}^s \Big) = 0 & &
\end{array}
\right. \tag{2.4}
$$

Uma formulação de programação linear inteira mista (MILP) foi, também, apresentada em [13] utilizando o modelo (2.4) como referência. Das condições de complementaridade da formulação (2.4), temos que:

$$
\pi_d^s g_j^s = \lambda_j g_j^s - \pi_{g_j}^s \bar{g}_j,
$$

para $j \in E$ e $s \in S$. Uma condição necessária e suficiente para que o PEP seja limitado é que as ofertas dos competidores sejam capazes de atender a demanda do sistema. Neste caso, as variáveis λ_j, $j \in E$, podem ser limitadas superiormente pela máxima oferta dos agentes concorrentes, em todos os cenários, ou seja,

$$\lambda_j \leq \bar{\lambda} := \max_{j \in J \setminus E, s \in S} \{\bar{\lambda}_j^s\}.$$

Cada variável, λ_j, $j \in E$ no modelo (2.4), foi substituída por sua, respectiva, decomposição binária $\lambda_j = \sum_{l=0}^{L} 2^l z_j^l$, onde $z_j^l \in \{0,1\}$ e $L := \lfloor \log_2(\bar{\lambda}) \rfloor$. Seja g_j^{ls} igual a $g_j^s z_j^l$ e assim a formulação (2.4) foi reescrita como um MILP:

$$\text{MILP} \begin{cases} \text{Max} \sum_{s \in S} p_s \sum_{j \in E} (\sum_{l \in \bar{L}} 2^l g_j^{ls} - \pi_{g_j}^s \bar{g}_j - c_j g_j^s), & \\ \text{s. a} & \\ \sum_{j \in J} g_j^s = d^s, & s \in S, \\ 0 \leq g_j^s \leq \bar{g}_j, & s \in S,\ j \in E, \\ 0 \leq g_j^s \leq \bar{g}_j^s, & s \in S,\ j \in J \setminus E, \\ \pi_d^s + \pi_{g_j}^s - \sum_{l=0}^{L} 2^l z_j^l \leq 0, & s \in S,\ j \in E,\ l \in L, \\ \pi_d^s + \pi_{g_j}^s \leq \bar{\lambda}_j^s, & s \in S,\ j \in J \setminus E, \\ \pi_{g_j}^s \leq 0, & s \in S,\ j \in J, \\ -\bar{g}_j z_j^l \leq g_j^{ls} \leq \bar{g}_j z_j^l, & s \in S,\ j \in E,\ l \in L, \\ -\bar{g}_j (1 - z_j^l) \leq g_j^{ls} - g_j^s \leq \bar{g}_j (1 - z_j^l), & s \in S,\ j \in E,\ l \in L, \\ z_j^l \in \{0,1\}, & j \in E,\ l \in L, \\ \sum_{s \in S} (\sum_{j \in E} (\sum_{l \in L} 2^l g_j^{ls} - \bar{g}_j \pi_{g_j}^s) + \sum_{j \in J \setminus E} (\bar{\lambda}_j^s g_j^s - \bar{g}_j^s \pi_{g_j}^s) - d^s \pi_d^s) = 0. & \end{cases} \tag{2.5}$$

Este modelo apresenta um total de $|E||L| + |S||E||J| + |S|(2|J|+1)$ variáveis e $1 + |S| + 4(|S||J|) + 4(|S||E||L+1|)$ restrições.

Em [15], uma formulação MILP é também apresentada para o problema e a solução exata é apresentada para algumas pequenas instâncias do problema. De qualquer maneira, devido ao tamanho das instâncias em uma aplicação real, esta abordagem pode não ser adequada devido ao tamanho do modelo. Este fato motivou a proposta de solução aproximada por metaheurística apresentada no capítulo (3) deste trabalho. Portanto, utilizaremos este modelo MILP na sequência deste texto para gerar a solução ótima ou a melhor solução viável conhecida para as instâncias do PEP.

2.4 Geração das Instâncias do Sistema Integrado e do Subsistema Sudeste

Nesta seção descreveremos o estudo realizado para a geração das instâncias consideradas nos experimentos numéricos. Iniciamos com uma pesquisa criteriosa no site do Operador Nacional do Sistema Brasileiro (ONS) (http://www.ons.org.br) e no site da Agencia Reguladora de Energia Elétrica (ANNEL), (http://www.aneel.gov.br) com o objetivo de levantar as informações necessárias para gerar as instâncias com configuração derivada do sistema elétrico brasileiro. Os dados iniciais deste trabalho foram obtidos no Boletim Mensal de Operação (BMO), disponíveis para consulta, no site do ONS. Consideramos apenas os dados relacionados à última semana de cada mês de 2008. Selecionamos apenas os geradores despachados naquele ano, ou seja, 178 geradores que de fato contribuíram para o atendimento da demanda nacional do sistema, Tabela 2.1. A capacidade de geração de cada gerador $j \in J$ foi fixada no maior valor da amostra de geração no período considerado. Denote por G_j a capacidade de geração amostral de cada gerador. G_j será utilizada para gerar aleatóriamente a capacidade de geração dos agentes concorrentes ($\bar{g}_j^s$, $j \in J \setminus E$, $s \in S$).

A capacidade instalada para as instâncias é de 73 GW, considerando todo o sistema nacional. Quatro tipos de geração de energia foram identificados no BMO, em 2008: geração hidrelétrica (H), termoelétrica (T), nuclear (N) e de biomassa (B). A geração hidrelétrica composta por 126 usinas contribui com 87% da capacidade instalada, a geração térmica, com 48 usinas, colabora com 10% da capacidade do sistema e o complemento da capacidade instalada, 3%, é realizado por 2 usinas nucleares e 2 usinas de biomassa.

O sistema de geração brasileiro é composto por quatro subsistemas, a saber: subsistema sul, sudeste, norte e nordeste. O conjunto de instâncias de teste, também, inclui instâncias menores que consideram apenas o subsistema do sudeste. Este subsistema conta com 114 usinas que contribuem com 60% da capacidade instalada do sistema integrado nacional, apresentando uma capacidade instalada de 44GW. A geração hidrelétrica com 92 usinas contribui com 85% da capacidade instalada. A geração termoelétrica é representada por 18 usinas e colabora com 12% da capacidade do subsistema. O restante da produção de energia que corresponde a 3% da capacidade instalada é representado, mais uma vez, por 2 usinas nucleares e 2 usinas de biomassa. Este subsistema possui as mesmas características do sistema integrado nacional com respeito à distribuição da matriz energética. As usinas que não pertencem ao subsistema sudeste estão prefixadas por um asterístico ($*$).

Tabela 2.1: Sistema Integrado Nacional em 2008

Usinas $j \in E$	Cap. G_j	Tipo	Usinas $j \in E$	Cap. G_j	Tipo
RJUSAN01	520	N	*JACUI	177	H
RJUSAN02	1352	N	JAURU	77	H
CAMARGOS	37	H	*LAJEADO	888	H
EMBORCACAO	561	H	MANSO	169	H
IGARAPE	121	T	MASCARENHAS	154	H
IRAPE-US	194	H	NO.FLUMINENSE	804	T
ITUTINGA-US	49	H	OURINHOS	39	H
JAGUARA-US	352	H	*P.DO CAVALO	67	H
MIRANDA	271	H	*P.MEDICI	216	T
NOVA PONTE	402	H	*PASSO REAL	99	H
SALTO GRANDE CS	62	H	*PECEM	15	T
TRES MARIAS	321	H	*PEIXE ANGICAL	357	H
ILHA SOLTEIRA	**2557**	**H**	*PETROLINA	130	T
JAGUARI	**22**	**H**	PHCEB	22	H
JUPIA	**1261**	**H**	PHCEE	27	H
P. PRIMAVERA	**1300**	**H**	PHCHF	33	H

Continua na próxima página

Tabela 2.1: Sistema Integrado Nacional em 2008

Usinas $j \in E$	Cap. G_j	Tipo	Usinas $j \in E$	Cap. G_j	Tipo
PARAIBUNA	**67**	**H**	PHCLG	9	H
TRES IRMAOS	**529**	**H**	PHCMG	77	H
*U. A. SALES	237	H	PHCMT	278	H
*U.SOBRADINHO	507	H	PHCOP	114	H
*UB.ESPERANCA	220	H	PHCPL	75	H
*US. L.GONZAGA	1142	H	PHCRJ	92	H
*USINA PA-I	173	H	PHCSC	60	H
*USINA PA-II	358	H	PHEME	42	H
*USINA PA-III	450	H	PHERS	90	H
*USINA PA-IV	1401	H	PHESC	155	H
*USINA XINGO	2769	H	PICADA	50	H
*BENTO MUNHOZ	1113	H	PIE-RP	28	B
*SALTO CAXIAS	1057	H	PIRAJU	59	H
*US. FIGUEIRA	14	T	PIRATININGA	165	T
*USI PAR.SOUZA	234	H	PORTO ESTRELA	112	H
CAPIVARA	561	H	QUEIMADO	88	H
CHAVANTES	296	H	RISOLETANEVES	140	H
JURUMIRIM	77	H	ROSAL	49	H
ROSANA	255	H	SA CARVALHO	68	H
SALTO GRANDE	83	H	*SAO JERONIMO	8	T
TAQUARUCU	303	H	SAO SIMAO-US	1668	H
*ALTOS	15	T	SOBRAGI	60	H
*ARACATI	11	T	*STA. CLARA PR	120	H
*BATURITE	11	T	STA.CLARA-MG	60	H

Continua na próxima página

Tabela 2.1: Sistema Integrado Nacional em 2008

Usinas $j \in E$	Cap. G_j	Tipo	Usinas $j \in E$	Cap. G_j	Tipo
*CAMPO MAIOR	13	T	*TERMOCABO	49	T
*CAUCAIA	15	T	*U.BARRAGRANDE	663	H
*CRATO	13	T	*U.C NOVOS	880	H
*IGUATU	15	T	*U.D.FRANCISCA	124	H
*JAGUARARI	101	T	*U.ITA	1446	H
*JUAZEIRO NORT	15	T	*U.MACHADINHO	1156	H
*MARAMBAIA	13	T	U.MIMOSO	27	H
*NAZARIA	13	T	*U.MONTE CLARO	124	H
CAMPOS	30	T	*U.QUEB.QUEIXO	120	H
CORUMBA	376	H	*U.URUGUAIANA	279	T
FUNIL	174	H	*US. TUCURUI	6849	H
FURNAS	846	H	*UT PERNAMBUCO	217	T
ITUMBIARA	1394	H	*UT. FORTALEZA	103	T
L.C.BARRETO	602	H	UTE SOL	168	T
M. MORAES	364	H	VOLTA GRANDE	300	H
MARIMBONDO	1199	H	XAVANTES	34	T
P. COLOMBIA	302	H	B.L.SOBRINHO	289	T
SANTA CRUZ	160	T	*BAHIA I	31	T
UHE S.DA MESA	877	H	CELSO FURTADO	131	T
ITAIPU 50 HZ	5522	H	FER.GASPARIAN	261	T
ITAIPU 60 HZ	6039	H	JUIZ DE FORA	84	T
FONTES NOVA	132	H	L.C.PRESTES	142	T
ILHA POMBOS	174	H	MACAE MERCHAN	834	T
NILO PECANHA	337	H	*ROMULOALMEIDA	125	T

Continua na próxima página

Tabela 2.1: Sistema Integrado Nacional em 2008

Usinas $j \in E$	Cap. G_j	Tipo	Usinas $j \in E$	Cap. G_j	Tipo
PEREIRA PASSO	52	H	*SEPE TIARAJU	160	T
SANTA BRANCA	30	H	*U.ARAUCARIA	465	T
AIMORES	195	H	*US.CAMACARI	330	T
AMADORAGUIAR1	227	H	UT.AUR.CHAVES	226	T
AMADORAGUIAR2	197	H	UTE CUIABA	72	T
C. DOURADA	535	H	AGUA VERMELHA	1289	H
CANASTRA	29	H	BARIRI	135	H
CANOAS I	80	H	BARRA BONITA	128	H
CANOAS II	68	H	CACONDE	74	H
COCAL	28	B	E.CUNHA	95	H
CORUMBA IV	129	H	IBITINGA	123	H
*CURUA-UNA	30	H	LIMOEIRO	21	H
DAIA	34	T	N.AVANHANDAVA	273	H
ESPORA	21	H	PROMISSAO	208	H
*FUNDAO	120	H	CANA BRAVA	376	H
FUNIL GRANDE	180	H	PONTE PEDRA	176	H
GOV.L.BRIZOLA	905	T	*U. ALEGRETE	40	T
GUAPORE	54	H	U. W.ARJONA	72	T
GUILMA AMORIM	132	H	*U.CHARQUEADAS	54	T
H.BORDEN EXT	68	H	*U.JLACERDA-A	207	T
H.BORDEN SUB	47	H	*U.JLACERDA-B	243	T
IGARAPAVA	168	H	*U.JLACERDA-C	325	T
*ITAPEBI US	222	H	*U.PASSO FUNDO	207	H
*ITAUBA	344	H	*U.S.OSORIO	901	H

Continua na próxima página

Tabela 2.1: Sistema Integrado Nacional em 2008

Usinas	Cap.	Tipo	Usinas	Cap.	Tipo
$j \in E$	G_j		$j \in E$	G_j	
ITIQUIRA I	156	H	*U.S.SANTIAGO	1332	H

Em todos os problemas teste consideramos a Companhia Energética de São Paulo (CESP) como o agente ofertante. A CESP controla 6 usinas hidrelétricas totalizando 5736 MW de capacidade instalada que corresponde a 7.8% da capacidade do sistema integrado nacional e 13% da capacidade do subsistema do sudeste. A Tabela 2.2 traz o nome, a capacidade de geração amostral (em MW) de cada usina controlada pela CESP. Portanto, temos para todas as instâncias $|E| = 6$, referente às usinas hidrelétricas sob o controle da CESP, e $|J| = 178$ ou $|J| = 114$, referindo-se à totalidade dos geradores que compõem o sistema integrado nacional e o subsistema sudeste, respectivamente. Consideramos diferentes números de cenários variando entre 5 e 70. Para todas as instâncias, as capacidades de geração dos concorrentes $\bar{g}_j^s$, $j \in J \setminus E$, $s \in S$, são selecionadas randomicamente no intervalo $[0.9G_j, G_j]$.

Tabela 2.2: Sistema de Geração da CESP em 2008

Usinas	Cap.
$j \in E$	G_j
ILHA SOLTEIRA	2557
JAGUARI	22
JUPIÁ	1261
PORTO PRIMAVERA	1300
PARAIBUNA	67
TRÊS IRMÃOS	529

Os custos operacionais $c_j, j \in E$ (em R$/MWh) são selecionados aleatoriamente no intervalo $[0.9C_t, 1.1C_t]$, onde C_t representa o custo de produção associado a cada tipo de geração de energia, $t \in [1, 2, 3, 4]$. A Tabela 2.3 apresenta o custo de produção de energia elétrica no Brasil, em 2008, associado a cada tipo de geração considerado nesta pesquisa. Esta informação está disponível para pesquisa no site da ANNEL.

Tabela 2.3: Custo de Produção de Energia em 2008

Tipo de Geração	Custo de Produção C_t
Hidrelétrica	118.40
Termoelétrica	330.11
Nuclear	138.75
Biomassa	101.75

As ofertas de preço $\bar{\lambda}_j^s$, $j \in J \setminus E$, $s \in S$ são selecionadas randomicamentes em $[1.1C_t, 1.5C_t]$, $t \in [1, 2, 3, 4]$, garantindo que a oferta de uma unidade de geração seja maior do que o seu custo operacional. Finalmente, as demandas d^s, $\forall s \in S$ são selecionadas randomicamente em $[0.8\bar{G}^s, \bar{G}^s]$, onde $\bar{G}^s$ corresponde à soma da capacidade de geração de todos os concorrentes, no cenário $s \in S$. Observe que esta seleção de demanda garante que os problemas sejam limitados, uma vez que os concorrentes podem sempre satisfazer a demanda sem a participação do ofertante, CESP.

Na Figura 2.1 apresentamos a descrição do procedimento de geração de p_s que define a probabilidade de ocorrência de cada cenário $s \in S$.

Entrada: O número de cenários, $|S|$.
Defina as variáveis: fator, probRel, somaProb e $\alpha \in [0,1)$
somaProb:=0
fator:=1
probRel:=1.0/$|S|$
para $(s = 1, \ldots, |S| - 1)$ **faça**
    p_s :=probRel + α(fator/(5$|S|$)
    somaProb:=somaProb + p_s
    fator:=(-1)fator
$p_{|S|}$:= 1.0 - somaProb
Saída: $p_s, \forall s \in S$

Figura 2.1: Procedimento para Geração das Probabilidades dos Cenários

Na Tabela 2.4 apresentamos um exemplo das probabilidades geradas para uma instância com cinco cenários (|S|=5).

Tabela 2.4: Geração de Probabilidade

Cenário $s \in S$	Probabilidade p_s
1	0.2146337917123161
2	0.1605381599929197
3	0.2222109831099499
4	0.1872490355049551
5	0.2153680296798591

Todos os sorteios utilizaram uma distribuição uniforme no processo de geração de instâncias. Cinco instâncias foram geradas para cada combinação de valores para $|J|$, $|E|$ e $|S|$, totalizando 75 instâncias de teste, Tabela 3.1 da seção (3.3) deste trabalho.

2.5 Geração das Instâncias do Subsistema Sul

Assim como na seção anterior, os dados de entrada para as instâncias de teste são relacionados ao ano de 2008 e estão disponíveis para consulta no site do ONS e no site da ANEEL. Consideramos as informações disponíveis no BMO referentes à última semana de cada mês de 2008 e utilizamos apenas 10 usinas que contribuíram para o atendimento da demanda do subsistema sul do Brasil, pertencentes ao estado do Rio Grande do Sul. O subsistema sul brasileiro de energia contém um total de 28 usinas correspondendo a 16% da capacidade nacional instalada, apresentando uma capacidade instalada de 11 GW. A geração hidrelétrica corresponde a 68% da capacidade instalada do subsistema e conta com 19 usinas. A geração termoelétrica contribui com 32% da capacidade instalada, com 9 usinas.

As 10 usinas selecionadas correspondem a 26% da capacidade instalada do subsistema sul com 2940 MW. A geração hidrelétrica (H) corresponde a 92% da capacidade instalada desta seleção e conta com 8 usinas. A geração termoelétrica (T) contribui com 8% da capacidade instalada, com 2 usinas.

Em todas as instâncias de teste relacionadas a este subsistema consideramos a Companhia Estadual de Energia do Rio Grande do Sul (CEEE) como agente ofertante. A CEEE controla 4 usinas hidrelétricas totalizando 596 MW de capacidade instalada, que corresponde a 20% da capacidade dos 10 agentes selecionados do subsistema sul.

A Tabela 2.5 mostra o nome, a capacidade de geração amostral (em MW) e o tipo das usinas selecionadas. As usinas da CEEE estão destacadas em negrito. Todos os sorteios utilizaram uma distribuição uniforme no processo de geração de instâncias. Cinco instâncias foram geradas para cada combinação de $|J|$,$|E|$ e $|S|$. Geramos cinco instâncias com $|J| = 8$ e $|E| = 2$, ou $|J| = 9$ e $|E| = 3$, ou $|J| = 10$ e $|E| = 4$, respectivamente, onde apenas as duas primeiras usinas da Tabela 2.5, ou as três primeiras, ou todas as usinas da CEEE foram consideradas. Consideramos, também, diferentes números de cenários, $|S| = 2$, $|S| = 3$, ou $|S| = 4$, totalizando 45 instâncias de teste, Tabela 4.1 da seção (4.6) deste trabalho.

Tabela 2.5: Subsistema Sul em 2008 (Selecionado)

Usinas $j \in E$	Cap. G_j	Tipo
ITAÚBA	**344**	**H**
DONA FRANCISCA	**124**	**H**
PASSO REAL	**99**	**H**
CANASTRA	**29**	**H**
P.MEDICI	216	T
SAO JERONIMO	8	T
U.MONTE CLARO	124	H
U.MACHADINHO	1156	H
JACUI	177	H
U.BARRAGRANDE	663	H

2.6 Exemplo: Instância: 08,02,02 do Subsistema Sul

Apresentamos nesta seção um exemplo de uma das instâncias geradas para o subsistema sul brasileiro, sendo a CEEE adotada como o agente ofertante. Observe que nesta instância as usinas Passo Real e Canastra não foram consideradas.

J 8

E 2

S 2

d 2159.5 1818.5

p 0.53 0.47

c 108.0 113.0

$\bar{g}$ 344.0 124.0

$\bar{g}^1$ 197.0 7.0 119.0 1069.0 169.0 617.0

$\bar{g}^2$ 197.0 7.0 121.0 1098.0 166.0 659.0

$\bar{\lambda}^1$ 410.0 364.0 139.0 149.0 168.0 155.0

$\bar{\lambda}^2$ 492.0 463.0 171.0 148.0 163.0 154.0

$max\bar{\lambda}$ 492.0

L 8

2.7 Solução MILP para a Instância: 08,02,02 do Subsistema Sul

Apresentamos nesta seção a solução obtida com o solver $CPLEX$ aplicado ao modelo MILP (2.5) para a referida instância do subsistema sul brasileiro. Tabela 2.6 apresenta a solução MILP ($z^* = 30655.9434$ e $\lambda = (410, 112)$) ordenada por oferta e com o gerador marginal marcado com uma seta. A CEEE determinou o gerador marginal apenas no primeiro cenário.

Tabela 2.6: Solução MILP da Instância: 08,02,02

Usinas (s=1)	$\bar{g}_j$	g_j	λ_j	π_d^1
U.D.FRANCISCA	124	124	112	
U.MONTE CLARO	119	119	139	
U.MACHADINHO	1069	1069	149	
U.BARRAGRANDE	617	617	155	
JACUI	169	169	168	
SAO JERONIMO	7	7	364	
ITAÚBA	344	54,5	**410**	$<\!-$
P.MEDICI	197	0	410	
$\sum g_j$		2159,5		
Usinas (s=2)	$\bar{g}_j$	g_j	λ_j	π_d^2
U.D.FRANCISCA	124	124	112	
U.MACHADINHO	1098	1098	148	
U.BARRAGRANDE	659	596,5	**154**	$<\!-$
JACUI	166	0	163	
U.MONTE CLARO	121	0	171	
ITAÚBA	344	0	410	
SAO JERONIMO	7	0	463	
P.MEDICI	197	0	492	
$\sum g_j$		1818,5		

Capítulo 3

Parte II: Algoritmo Genético Aplicado ao Problema de Estratégia de Preço

3.1 Revisão Bibliográfica da Aplicação de Metaheurística em PPDN

Nesta seção apresentamos uma revisão bibliográfica cujo objetivo é analisar o que tem sido feito em pesquisa relativo ao estudo da aplicação de metaheurísticas a PPDN, justificando o modelo e a metodologia adotados nesta tese. $WANG$ *et al.* [25] apresentaram um especial PPDN de minimização não-linear, onde a função objetivo e as restrições do problema líder são não-diferenciáveis e não-convexas e a função objetivo e as restrições do problema seguidor são diferenciáveis e convexas. Este problema foi transformado em um problema de otimização não-linear equivalente substituindo o problema seguidor por suas condições de otimalidade. Para resolver o problema equivalente, efetivamente, foi construído um problema específico de otimização bi-objetivo, utilizando como conceitos fundamentais: conjunto dominante e otimalidade de *Pareto*. Para resolver este problema específico um algoritmo evolucionário (AE) foi proposto, auxiliado por um novo método de manipulação de restrições lineares e não-lineares. O esquema de manipulação de restrições lineares faz com que todos os indivíduos da população do AE satisfaçam todas as restrições lineares do problema, exatamente. Já o esquema de manipulação de restrições não-lineares faz com que todos os indivíduos satisfaçam todas as restrições não-lineares do problema, aproximadamente, mantendo satisfeitas as restrições lineares. Os operadores de cruzamento e mutação fazem uso deste método de manipulação de restrições, fazendo com que as soluções geradas sejam melhoradas a cada iteração. Provou-se a convergência global do AE, utilizando como conceito principal a medida de *Lebesgue*. Uma característica distintiva do algoritmo é que ele pode ser usado para manipular PPDN não-lineares, com a função objetivo do líder não-diferenciável, porém contínua. KUO e HAN [26] aplicaram programação linear em dois níveis para *supply chain management* e desenvolveram um método híbrido eficiente baseado em AG e otimização por nuvens de partículas. $MARINAKIS$ *et al.* [17] propuseram uma formulação do problema de roteamento de veículos como um PPDN, baseado na formulação de $FISHER$ e $JAIKUMAR$ [27], onde as restrições do problema de roteamento de veículos podem ser separadas em dois conjuntos. O primeiro conjunto de restrições corresponde às restrições do problema de atribuição generalizado, assegurando que cada rota inicia e termina em um único depósito, que todo cliente será atendido por um único veículo e que a carga associada a um veículo não exceda a sua capacidade. O segundo conjunto de restrições corresponde às restrições de um problema do caixeiro viajante, associado a todos os clientes de um determinado veículo. Assim, esta formulação proporciona a idéia de que o problema de roteamento de veículos pode ser formulado como um PPDN. No primeiro nível, no problema líder, o decisor atribui clientes a veículos verificando a viabilidade das rotas construídas, ou seja, as restrições de capacidade dos veículos, sem levar em conta a sequência na qual os veículos vão visitar os clientes. No segundo nível, no problema seguidor, o tomador de decisão busca as rotas ótimas dessas atribuições. Uma vez que o custo de cada rota foi calculado no segundo nível, o tomador de decisão do primeiro nível, estima as melhores atribuições. Com base nesta formulação, um AG foi proposto. Utilizaremos esta metodologia como referência para o desenvolvimento da metaheurística para a solução do PEP, ou seja, utilizaremos um AG como proposta de solução do problema modelado como um PPDN. No primeiro nível do algoritmo proposto, um AG será utilizado para manter uma população das mais promissoras ofertas de preço do agente ofertante, empresa E. No segundo nível do algoritmo, o problema de despacho econômico será resolvido, de forma independente, para cada indivíduo da população do AG.

3.2 Algoritmo Genético

Algoritmos genéticos são métodos de otimização e busca inspirados nos mecanismos de evolução de populações de seres vivos. Foram propostos por $HOLLAND$ [28] e popularizados por um dos seus alunos, $GOLDBERG$ [29]. Estes algoritmos seguem o princípio da seleção natural e sobrevivência do mais apto. O primeiro passo para construção de um AG consiste em definir o conjunto de genes que representam as características do problema em questão. Se o problema possui n variáveis independentes estas passam a ser os n genes de um indivíduo, representando uma possível solução para o problema de otimização. Um indivíduo é representado por n variáveis dispostas em sequência, ou seja, um vetor de variáveis do problema ou um ponto do espaço de busca do problema. Após a modelagem do indivíduo devemos definir o tamanho da população de indivíduos do AG e a forma com que esta será inicializada. A inicialização pode ser feita de forma aleatória ou utilizando algum conhecimento do problema. Neste último caso podemos utilizar heurísticas construtivas para formar a população inicial do AG. Durante o processo evolutivo cada indivíduos sofre avaliação por uma função objetivo, obtendo um valor denominado aptidão que está associado à qualidade da solução. No processo evolutivo os indivíduos mais aptos tendem a sobreviver e os menos aptos tendem a ser descartados, por meio de um operador de seleção ou método de seleção. Os indivíduos selecionados serão expostos ao meio em que vivem e poderão sofrer a ação de dois operadores genéticos, o operador de cruzamento e o operador de mutação. O operador de cruzamento combina características dos indivíduos selecionados com o objetivo de produzir novos indivíduos para a próxima geração. O objetivo deste operador é explorar o espaço de busca do problema. Já o operador de mutação faz pequenas modificações, pontuais, cujo objetivo é inserir novas características na população. O operador de mutação permite que o algoritmo possa sair ou passar por ótimos locais do espaço de busca do problema. Após a aplicação dos operadores genéticos uma nova geração é determinada. De posse da nova geração o processo é repetido até que um critério de parada seja satisfeito. Em otimização um critério de parada pode ser: o tempo de execução, a convergência do AG, o número de gerações do AG, ou qualquer combinação que possa ser criada em função do conhecimento do problema. Desta forma o AG simula a evolução de uma população de possíveis soluções do problema e retorna a melhor solução ao final.

3.2.1 Fundamentação da Proposta de Solução

Algoritmos genéticos têm sido amplamente utilizados na solução de problemas de otimização combinatória [30]. Provou-se teoricamente e empiricamente que AGs fornecem uma busca robusta em espaços complexos [29]. Alguns trabalhos recentes, também, aplicam AG para solução de PPDN, [17, 26]. Esta é a primeira vez em que um AG está sendo aplicado ao PEP. Em função dos detalhes da aplicação e a não disponibilidade das instâncias utilizadas em [13], não foi possível apresentar comparações deste trabalho com outros relatados na literatura. Sendo assim, analisaremos a qualidade do método proposto comparando as soluções obtidas com a melhor solução, MILP, conhecida para as instâncias geradas para o PEP, como apresentado em [13, 31, 32].

Consideramos o uso da técnica de metaheurística para a solução prática do PEP como uma das principais contribuições deste trabalho. Esta abordagem de solução baseada em metaheurística foi motivada pelo fato de que as instâncias com dados reais, derivadas do sistema brasileiro de energia, tornam-se intratáveis com o método exato, quando a dimensão do problema aumenta. A dimensão do PEP aumenta com o número de cenários e um número razoável de cenários é sempre necessário para garantir a confiabilidade dos resultados.

O uso de AG foi motivado por seus recentes sucessos aplicados em PPDN. O algoritmo proposto nos permitirá a produção de soluções de boa qualidade e também nos permitirá responder as questões naturais que surgem com a competição no mercado de energia, tais como:
- O que acontecerão com os preços? Aumentarão ou diminuirão?
- Como os preços podem influenciar o mercado de energia?

3.2.2 Procedimento de Solução do Problema Seguidor do PPDN

Uma vez que as variáveis do líder são determinadas ou fixadas por um indivíduo do AG, o problema de programação linear (LP) ou o problema seguidor pode ser resolvido separadamente para cada cenário. Cada subproblema pode ser resolvido, para cada um dos cenários, pelo procedimento polinomial apresentado na Figura 3.1. Este procedimento realiza o despacho econômico ou resolve o problema seguidor do PPDN e retorna a produção de energia de todos os geradores e o preço *spot* em todos os cenários, que contribuirão para o cálculo da aptidão associada ao indivíduo em questão.

Neste procedimento, para satisfazer o problema líder e maximizar o lucro da empresa ofertante, se uma oferta de um gerador em E, modelada em cada indivíduo da população, for igual a oferta de um gerador concorrente, não pertencente a E, então o gerador em E deve ter prioridade no despacho econômico. Além disso, se dois geradores em E ofertam o mesmo preço, o gerador com o menor custo operacional deve ser despachado primeiro ou ter prioridade. Para garantir essas duas condições no procedimento da Figura 3.1 e realizar o despacho econômico de modo mais eficiente, criaremos duas listas simplesmente encadeada, LSE1 e LSE2, em ordem crescente de oferta de preço, concatenando-as e gerando uma terceira, LSE3, com todas as ofertas de preço ordenadas.

A LSE1, ao final do processo de criação, será composta por todas as ofertas de preço e capacidade de geração de energia dos geradores em E, em ordem crescente de oferta de preço e de custo operacional de geração. Este passo de construção terá complexidade $\Theta(|E|^2)$, no pior caso. As inserções nesta lista ocorrerão sempre em ordem decrescente de custo operacional garantindo que: se duas ofertas de preço são iguais, a oferta do gerador de menor custo operacional será inserida em um posição anterior a outra oferta. O pior caso deste passo de criação ocorrerá quando todas as ofertas de preço estão, dispostas, em ordem inversa aos custos operacionais, ou seja, a menor oferta de preço está associada ao gerador de maior custo operacional, a segunda menor oferta está associada ao gerador com o segundo maior custo operacional, e assim sucessivamente, sendo que a maior oferta está associada ao gerador de menor custo operacional. Neste caso particular as inserções nesta lista ocorrerão sempre ao final, o pior caso possível. Já o melhor caso ocorrerá quando os custos operacionais e as oferta estiverem dispostas na mesma ordem, neste caso as inserções ocorrerão sempre no início, $\Theta(|E|)$. Consideramos uma pré-ordenação dos custos operacionais, em ordem decrescente, no início deste procedimento.

Entrada: Ofertas de preço, $\tilde{\lambda}_j$, $\forall j \in E$ e $\tilde{\lambda}_j^s$, $\forall j \in J \setminus E$, $s \in S$. Capacidades de geração, $\tilde{g}_j$, $\forall j \in E$ e $\tilde{g}_j^s$, $\forall j \in J \setminus E$, $s \in S$. Demanda $d^s, \forall s \in S$. A ordem dos custos operacionais dos geradores em E. A ordem das ofertas de preço dos geradores em $J \setminus E$, para todos os cenários em S.

para $(s \in S)$ **faça**
- Sejam $\tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \ldots \leq \tilde{\lambda}_{|J|}$ as ofertas de preço ordenada dos geradores em J, no cenário s, e $\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_{|J|}$ as correspondentes capacidades de geração ofertadas, após o processo de concatenação.
- Seja k o máximo índice tal que $\sum_{j=1}^{k} \tilde{g}_j < d^s$.
- **para** $(j = 1, \ldots, k)$ **faça**
  - $g_j^s := \tilde{g}_j$.
- $g_{k+1}^s := d^s - \sum_{j=1}^{k} \tilde{g}_j$.
- **para** $(j = k+2, \ldots, |J|)$ **faça**
  - $g_j^s := 0$.
- $\pi_d^s := \tilde{\lambda}_{k+1}$.

Saída: g_j^s, $\forall j \in J$, $s \in S$, $\pi_d^s, \forall s \in S$

Figura 3.1: Solução Polinomial do Problema Seguidor

Por outro lado, a LSE2 será composta por todas as ofertas de preço e capacidade de geração de todos os geradores concorrentes de $J \setminus E$, por cenário em S. A complexidade deste passo de criação será $\Theta(|J \setminus E|)$, por cenário, no pior caso, pois as ofertas serão inseridas em ordem decrescente de oferta de preço e sempre ocorrerão no início desta lista. Consideramos, também, que as ofertas dos concorrente estejam pré-ordenadas, em ordem decrescente, no início deste procedimento.

No processo de concatenação iremos comparar duas a duas as ofertas de preço das duas listas e, se as ofertas forem iguais, a oferta de preço da LSE1 será ligada à

LSE3, garantindo que os geradores de E tenham prioridade no despacho econômico. A complexidade, no pior caso, desta concatenação será $\Theta(|J|)$. Sendo assim, a complexidade de cada subproblema, para cada cenário, apresentado na Figura 3.1, será dada por: $\Theta(|E|^2 + |J \setminus E| + |J|)$, ou seja, será igual a: $\Theta(\max\{|E|^2, |J|\})$. Portanto, a complexidade do procedimento que soluciona o problema seguidor será igual a $\Theta(max\{|E|^2, |S||J|\})$

3.2.3 População Inicial

Definimos um indivíduo para o AG como um vetor de $|E|$ componentes, onde a j-ésima componente corresponde à oferta de preço λ_j do gerador $j \in E$.

É fácil verificar que existe sempre uma solução ótima para o PEP, onde todos os geradores pertencentes à empresa E ofertam o mesmo preço de um gerador não pertencente a E, em algum cenário. Portanto, denotamos por $\mathcal{L}$ o conjunto de todos os possíveis valores para cada componente do indivíduo do AG, inicialmente definimos $\mathcal{L} = \{\bar{\lambda}_j^s, j \in J \setminus E, s \in S\}$. Para melhorar a eficiência do algoritmo podemos ainda eliminar algumas ofertas de preço deste conjunto. Considere λ_{max}^s como o máximo valor em $\mathcal{L}$ que pode ser atribuído a uma oferta de preço de cada gerador da empresa E, tal que na solução ótima do problema de despacho econômico (2.2), para o cenário s, todo gerador de E gere toda sua capacidade. Agora, seja $\lambda_{MIN} = min\{\lambda_{max}^s, s \in S\}$, observe que podemos eliminar de $\mathcal{L}$ todos as ofertas de preço que são menores do que λ_{MIN}, sem risco de eliminar a solução ótima do espaço de busca. Além do mais, considere λ_{min}^s como o mínimo valor em $\mathcal{L}$ que pode ser associado a uma oferta de preço de um gerador da empresa E de maneira que nenhum deles seja despachado na solução ótima do problema seguidor (2.3), para o cenário s. Considere ainda $\lambda_{MAX} = max\{\lambda_{min}^s, s \in S\}$, logo podemos eliminar de $\mathcal{L}$ todos as ofertas de preço que são maiores ou iguais a λ_{MAX}. Portanto, na sequência deste texto consideremos o conjunto $\mathcal{L}$ de todas os possíveis valores associados às ofertas de preço dos geradores da empresa E, definido como:

$$\mathcal{L} = \{\bar{\lambda}_j^s, j \in J \setminus E, s \in S | \lambda_{MIN} \leq \bar{\lambda}_j^s < \lambda_{MAX}\}. \tag{3.1}$$

Considere, ainda, três diferentes tipos de solução para gerar a população inicial do algoritmo:

A solução do tipo1 é baseada na idéia de inicializar a população com as soluções onde a empresa E gera toda a sua capacidade em um determinado cenário s, selecionado aleatoriamente. Consideramos $\lambda_j = \lambda_{max}^s$ para todo $j \in E$, e portanto temos que $g_j^s = \bar{g}_j$ para todo $j \in E$.

A solução do tipo2 corresponde ao caso onde nenhum gerador pertencente à empresa E é despachado em um determinado cenário s, selecionado aleatoriamente. Consideramos $\lambda_j = \lambda^s_{min}$ para todo $j \in E$, e portanto temos que $g^s_j = 0$ para todo $j \in E$. Admitindo que os geradores não pertencente a E, geradores concorrentes, possam sempre atender a demanda em todos os cenários, o procedimento sempre irá gerar uma solução viável.

Entrada: Um cenário selecionado aleatoriamente s.
Seja $\mathcal{L}^s = \{\bar{\lambda}^s_j \in \mathcal{L} | \lambda^s_{max} \leq \bar{\lambda}^s_j < \lambda^s_{min}\}$.
Considere as ofertas de preço de todos os geradores da empresa E iguais as ofertas de preço em $\mathcal{L}^s$. Calcule o correspondente lucro da empresa E resolvendo o problema de despacho econômico (2.2) para o cenário s.
Determine como oferta de preço dos geradores da empresa E, a oferta de preço que retorne o lucro máximo de E, no cenário s.
Saída: $\lambda_j, \forall j \in E$

Figura 3.2: Procedimento para Gerar a Solução do Tipo3

Por último, a solução do tipo3 é baseada no procedimento polinomial, Figura 3.2, para resolver o PEP para um cenário s, selecionado aleatoriamente. Em [13] foi provado que existe uma solução ótima para o PEP onde todos os geradores pertencentes à empresa E ofertam o mesmo preço, se $|S| = 1$.

Note que podemos criar $|S|$ indivíduos distintos de cada tipo apresentado acima. Portanto, trabalharemos com uma população inicial com $4|S|$ indivíduos gerados da seguinte forma: $|S|$ indivíduos de cada tipo abordado anteriormente, mais $|S|$ indivíduos gerados aleatoriamente.

3.2.4 Operadores Genéticos

Os operadores genéticos contribuem para o sucesso do processo evolutivo de um AG. Os operadores genéticos utilizados neste trabalho foram idealizados da maneira mais simples possível, priorizando a perfomance do AG, objetivando o alcance do maior número de gerações, em um dado período de tempo. O operador de seleção implementado no AG foi a seleção por torneio de três indivíduos, o cruzamento de dois pontos foi escolhido como operador de cruzamento e o operador de mutação foi idealizado para permitir a inclusão de novas características nos indivíduos da população.

Operador de Seleção

Um operador de seleção em um AG seleciona os indivíduos mais aptos da população corrente para participar do processo evolutivo. Para este problema, um indivíduo é mais apto quanto maior for o valor do lucro associado a sua oferta. Optamos por utilizar a seleção por torneio dado a sua facilidade de implementação e eficácia na seleção dos melhores indivíduos. A seleção por torneio sorteia n indivíduos da população corrente e seleciona, dentre estes, o indivíduo mais apto para participar da fase de cruzamento. Optamos por utilizar $n = 3$, neste operador de seleção, por ser um valor que proporciona uma diversidade na escolha do indivíduo e ao mesmo tempo proporciona uma eficiência no desempenho do operador.

Operador de Cruzamento

Após a seleção dos indivíduos da população corrente, aplicaremos o operador de cruzamento de dois pontos para cada par de indivíduos selecionados. O cruzamento de dois pontos troca características presentes no par de indivíduos (pai1 e pai2). Neste tipo de cruzamento geramos duas posições aleatórias p_1 e p_2, e as partes dos indivíduos compreendidas entre p_1 e p_2 são trocadas entre si, gerando o filho1 e o filho2, respectivamente. Este processo é aplicado para |TAMPOP - ELITE| indivíduos da população gerando a mesma quantidade de filhos que preservam as características genéticas de seus pais. O percentual de 10% para o tamanho do conjunto elite foi escolhido após a fase de refino de parâmetros.

Operador de Mutação

O operador de mutação em um AG é o operador que permite que novas características sejam adicionadas à população corrente. Neste operador um indivíduo e um de seus genes são selecionados aleatoriamente para receber um novo valor. Em otimização global este operador permite que o processo evolutivo saia de ótimos locais que são comuns em problemas não-convexos multimodais. No operador de mutação implementado o gene selecionado corresponde a um dos geradores da empresa ofertante e receberá um novo valor correspondente a uma oferta de preço válida, ou seja, uma valor pertencente a $\mathcal{L}$, como definido em (3.1). Garantimos neste processo de atualização que a nova oferta, para o gerador selecionado, seja superior ao seu custo operacional. A taxa de mutação utilizada neste AG foi de 10%, definida mediante à fase de refino de parâmetros e justificada pela não-convexidade do PEP e pela ocorrência de vários pontos de máximos locais. Neste operador selecionamos 10% do número total de genes da população, ou seja, 10% de (|E|TAMPOP) genes serão selecionados e atualizados a cada geração do AG. Por exemplo, supondo TAMPOP=100 e |E|=6 teremos na população um total de 600 genes e portanto 60 deles sofrerão a mutação. No processo de mutação sortearemos dois números aleatórios a_1 e a_2, com $a_1 \in [0, 599]$ e $a_2 \in [0, 5]$. Supondo, ainda, $a_1 = 435$ e $a_2 = 3$, então a quarta posição, derivada de a_2, do trigésimo sexto indivíduo da população, derivado do resto da divisão de a_1 por TAMPOP, sofrerá mutação. Este processo é repetido até que todos os 60 genes sofram mutação.

3.2.5 Refino de Parâmetros

Nesta seção descrevemos o estudo de refino de parâmetros do AG. O refino de parâmetros consiste da realização de uma sequência de experimentos que busca identificar quais são os valores de parâmetros que produzem os melhores resultados. Utilizamos o desenho experimental fatorial completo como estratégia de refino de parâmetros. Este tipo de estratégia considera todas as possíveis combinações de valores associados aos parâmetros, [33]. Optamos por realizar os testes de refino de parâmetros apenas para os parâmetros: tamanho do conjunto elite e taxa de mutação do AG. Consideramos como possíveis valores para o tamanho do conjunto elite os percentuais de: 5%, 10% e 15% do tamanho da população e os percentuais de: 5%, 10%, 15%, 20% e 25% para a taxa de mutação, totalizando 15 possíveis combinações de parâmetros por experimento. Em cada experimento efetuamos dez execuções do AG para cada combinação de parâmetros, totalizando 150 execuções por instância. Ao todo, geramos 30 instâncias para esta fase de afinação de parâmetros, duas para cada tamanho, compatíveis com as instâncias da seção de resultados (3.3). Portanto, foram realizados 4500 execuções ao todo nesta investigação de refino de parâmetros. O critério de parada utilizado no AG foi o tempo de execução igual a 120 segundos.

Para que a comparação fosse possível, os resultados das execuções foram transformados em desvio percentual relativo (DPR):

$$DPR(\%) = 100 \times \frac{Z^*_{exec} - \bar{Z}_{exec}}{Z^*_{exec}}. \tag{3.2}$$

onde $\bar{Z}_{exec}$ corresponde à melhor solução oriunda de uma execução do AG e Z^*_{exec} corresponde à melhor solução obtida dentre todas às geradas no experimento por instância, para todas as combinações de parâmetros. Os valores dos DRPs entre diferentes instâncias são considerados comparáveis e devido a essa propriedade, os DRPs são usados como métrica de performance há bastante tempo, [34]. Em *DONG et al.* [35], o DRP foi utilizado para comparar a performance de alguns métodos heurísticos. Classificamos como melhor combinação de parâmetros aquela que apresentou a menor média geral de DPRs para todas as instâncias consideradas.

O melhor resultado foi alcançado com o tamanho do conjunto elite igual a 10% do tamanho da população e a taxa de mutação igual a 10% de (|E|TAMPOP).

3.2.6 O Algoritmo Genético Proposto

Nesta seção um algoritmo genético, Figura 3.3, é proposto para resolver o PPDN. No primeiro nível do algoritmo proposto um AG é utilizado para manter uma população das mais promissoras ofertas de preços para os geradores da empresa ofertante E. No segundo nível do algoritmo proposto o problema seguidor (2.3) é resolvido, independentemente, para cada solução do primeiro nível utilizando o procedimento polinomial, Figura 3.1.

```
Crie a população inicial com TAMPOP = 4|S| indivíduos como descrito na
seção (3.2.3).
Avalie a aptidão de cada indivíduo resolvendo o problema seguidor de (2.3)
utilizando o procedimento apresentado na Figura 3.1.
Ordene os indivíduos em ordem crescente de aptidão.
enquanto (critério de parada não é satisfeito) faça
    Selecione dois indivíduos (pais) da população corrente usando seleção por
    torneio de tamanho igual a 3.
    Aplique o cruzamento de dois pontos ao par de indivíduos selecionados.
    Repita os dois passos anteriores até que 90% de TAMPOP indivíduos
    sejam gerados.
    Aplique o operador de mutação selecionando randomicamente 10% do
    número total de genes da população para serem alterados.
    Avalie e ordene os indivíduos em ordem crescente de aptidão.
    Copie 10% dos melhores indivíduos da população corrente (conjunto
    ELITE) para a próxima população.
    Copie 90% dos indivíduos de TAMPOP, gerados após o processo de
    mutação, para a próxima população.
    Atualize a população.
Saída: λ_j, ∀j ∈ E, g_j^s, ∀j ∈ J, s ∈ S, π_d^s, ∀s ∈ S
```

Figura 3.3: Descrição Geral do AG

3.3 Resultados Numéricos do AG

Neste capítulo apresentamos os resultados computacionais obtidos com o AG proposto quando aplicado às instâncias do PEP, com configuração derivada do sistema brasileiro de energia. O algoritmo foi implementado na linguagem de programação C e compilado com gcc (GNU *COMPILE* C). Todas as execuções foram realizadas em uma máquina com a seguinte configuração: *Precision WorkStation* T7400, *Intel*(R) *Xeon*(R) CPU X5472@3GHz, 16GB, Ubuntu 9.10. O resolvedor CPLEX, v12.5 [36], foi utilizado para obter a solução ótima, ou a melhor solução, das instâncias considerando a formulação MILP do problema, (2.5). Em todas as execuções do CPLEX, limitamos o tempo de processamento em 6 horas (ou 21600 segundos) para obtenção da melhor solução viável para as instâncias do PEP, melhor solução MILP.

Tabela 3.1: Instâncias Geradas para o AG e Resultados do MILP

Instância	Solução do MILP				
$Inst_{\|J\|,\|E\|,\|S\|}$	$\#Var.$	$\#Restr.$	$DPR_{(6h.)}$	$DPR_{(20m.)}$	$Tempo$
Inst114,6,10	2884	6731	-	-	**154**
Inst114,6,15	4299	10096	-	1,04%	1323
Inst114,6,20	5714	13461	-	1,89%	2600
Inst114,6,25	7129	16826	-	1,00%	2792
Inst114,6,30	8544	20191	-	4,13%	6521
Inst114,6,40	11374	26921	0,68%	7,06%	16266
Inst114,6,50	14204	33651	0,82%	8,92%	**21600**
Inst114,6,60	17034	40381	3,37%	14,54%	20579
Inst114,6,70	19864	47111	1,49%	15,37%	**21600**
Inst178,6,05	2109	4646	-	-	**547**
Inst178,6,10	4164	9291	-	-	**392**
Inst178,6,15	6219	13936	-	1,49%	2913
Inst178,6,20	8274	18581	-	2,29%	7962
Inst178,6,25	10329	23226	0,24%	2,70%	16567
Inst178,6,30	12384	27871	1,01%	3,80%	16320

A Tabela 3.1 apresenta a média dos resultados obtidos para cada grupo de instâncias utilizando o modelo MILP. Na linha correspondente à instância Inst$|J|, |E|, |S|$, apresentamos a média dos valores para as cinco instâncias com $|J|$ geradores no total, $|E|$ deles pertencendo à empresa CESP, e $|S|$ cenários. Nas colunas da Tabela

3.1 apresentamos, além do número de variáveis e restrições, a média dos *gaps* de dualidade do CPLEX, ou dos $DPRs$, em seis horas ($DPR_{6h.}$), a média dos *gaps* de dualidade, ou dos $DPRs$, em 20 minutos ($DPR_{20m.}$) e a média dos tempos de execução (Tempo) em segundos. Para três das instâncias consideradas o resolvedor obteve a solução ótima em menos de 20 minutos (Tempo $<$ 1200) e para duas delas o CPLEX não obteve a solução ótima em 6 horas (Tempo = 21600), caracterizando-as como instâncias de fácil solução e instâncias de difícil solução, respectivamente, garantindo uma boa diversidade para o experimento. Para sinalizar que o resolvedor encontrou a solução ótima para todas as instâncias (DPR nulo) no período investigado, utilizamos um traço (-), na respectiva coluna.

Tabela 3.2: Resultados dos Testes Computacionais com o AG

Instância	Melhor Solução do AG			
$Inst\|J\|,\|E\|,\|S\|$	DPR_{AG}	$Tempo$	$Ger.$	$DPR_{pop.}$
Inst114,6,10	0,01% (0,01)	7,78 (9,32)	280 (332)	8,03% (1,21)
Inst114,6,15	0,00% (0,01)	16,51 (17,15)	275 (292)	7,25% (0,28)
Inst114,6,20	0,00% (0,00)	8,07 (7,74)	73 (77)	3,75% (0,00)
Inst114,6,25	0,01% (0,02)	25,26 (18,78)	124 (87)	6,19% (1,05)
Inst114,6,30	0,01% (0,02)	24,53 (21,58)	101 (94)	7,41% (1,15)
Inst114,6,40	0,03% (0,04)	33,26 (27,35)	69 (61)	1,93% (0,00)
Inst114,6,50	0,01% (0,02)	22,87 (12,88)	27 (20)	5,24% (0,52)
Inst114,6,60	0,00% (0,01)	35,74 (17,37)	32 (19)	3,73% (0,84)
Inst114,6,70	-0,03%(0,02)	34,43 (13,73)	21 (12)	6,24% (1,66)
Inst178,6,05	0,00% (0,00)	12,49 (16,69)	773 (983)	1,22% (0,08)
Inst178,6,10	0,06% (0,11)	12,48 (16,50)	226 (312)	6,07% (2,05)
Inst178,6,15	0,01% (0,01)	31,55 (32,04)	261 (274)	4,08% (0,52)
Inst178,6,20	0,00% (0,01)	24,80 (19,76)	106 (91)	2,17% (0,00)
Inst178,6,25	0,01% (0,02)	33,02 (29,11)	87 (84)	1,36% (0,00)
Inst178,6,30	0,03% (0,04)	33,38 (25,54)	59 (50)	1,92% (0,07)

Na Tabela 3.2 apresentamos os resultados de dez execuções do AG para cada instância gerada. Para cada instância a média das dez melhores soluções é calculada, bem como, o DPR entre esta média e a melhor solução MILP (DPR_{AG}), como em (3.2). Apresentamos na segunda coluna a média e o desvio padrão, entre parênteses, dos $DPRs$ para cada grupo de instâncias. Na última coluna apresentamos uma estatística similar à segunda coluna, relacionando a média e o desvio padrão dos $DPRs$ entre a melhor solução gerada na população inicial do AG e a melhor solução

MILP ($DPR_{pop.}$). Apresentamos ainda, nas demais colunas, a média e o desvio padrão do tempo, em segundos, bem como a média e o desvio padrão do número de gerações necessário para o AG encontrar a melhor solução. O critério de parada utilizado foi o limite de tempo igual a 120 segundos para a execução do AG. O principal objetivo na Tabela 3.2 é analisar a convergência da proposta de solução e a sua performance, comparando os resultados obtidos com a metaheurística e a melhor solução MILP conhecida para as instâncias, Tabela 3.1. Notamos na Tabela 3.2 que o AG foi capaz de melhorar o limite inferior para um dos grupos de instâncias (Inst114,6,70), o que corrobora com o objetivo da solução heurística que ganha relevância quando aplicado a grandes instâncias. Observamos, também, que o AG apresenta um comportamento robusto em todas as instâncias visto que a média dos $DPRs$ e o desvio padrão são bem baixos. Outro aspecto que devemos considerar é a performance da proposta refletida pelo reduzido tempo computacional necessário para obter a melhor solução se comparado com o tempo computacional gasto pelo método exato, Tabela 3.1. Para finalizarmos, cabe ainda analisar a qualidade dos indivíduos gerados na população inicial deste algoritmo, contribuindo para a eficácia da proposta.

Em [37], o autor mostra que a estratégia ótima de um agente ofertante é ofertar o seu custo operacional. No entanto, é sabido que esta estratégia só é ótima se o agente ofertante não tem poder de mercado, ou seja, se o ofertante não é capaz de influenciar o preço *spot*. Caso contrário, se o agente ofertante têm poder de mercado, ele estará interessado em desenvolver uma oferta estratégica. Esta discussão motivou a análise e detalhamento dos resultados para uma instância particular, comparando o preço *spot*, a geração e o lucro líquido da CESP, considerando as duas estratégias.

Nas Tabelas 3.3, 3.4 e 3.5 apresentamos os resultados para a instância $\text{Inst}178, 6, 10$ que foi selecionada do conjunto de instâncias de teste. Concluímos da solução da instância considerada que o lucro líquido esperado para a CESP com a oferta estratégica de preço foi de R$ 630780.00, contra R$ 389310.00 com a oferta baseada no custo operacional, um ganho de 62%. A Tabela 3.3 compara as ofertas de preço geradas com a oferta estratégica (AG) e as ofertas baseadas no custo operacional (ambos em R$/MWh) para todas as usinas controladas pela CESP. Todas as ofertas de preço obtidas com o AG são maiores do que o custo operacional associado, $c_j, j \in E$, apresentados na coluna λ_{c_j}.

Tabela 3.3: Ofertas Estratégicas de Preço (AG)

Usina	λ_j	λ_{c_j}
ILHA SOLTEIRA	366	123
JAGUARI	168	126
JUPIÁ	148	118
PORTO PRIMAVERA	363	126
PARAIBUNA	141	118
TRÊS IRMÃOS	363	117

A Tabela 3.4 compara o preço *spot* (em R$/MWh), a geração total das usinas controladas pela CESP (em MW), e o lucro líquido (em R$) da CESP em cada um dos dez cenários, considerando as duas estratégias. É possível observar que o preço *spot* em cada cenário com a oferta estratégica foi maior do que o preço *spot* obtido com a estratégia baseada em custo para nove dos dez cenários. Comparando o preço *spot* e a geração total das usinas nas duas estratégias, nota-se que a CESP tem o gerador marginal, gerador que determina o valor do preço *spot*, em sete cenários com a oferta estratégica o que fornece um menor nível de geração do que o observado com a oferta baseada em custo operacional. No cenário 4, não existe diferença entre as estratégias com relação ao preço *spot*, o total de geração e o lucro líquido da CESP. A empresa não teve gerador marginal neste cenário. A razão disto é que o preço *spot* (370) excedeu a oferta de preço da usina Ilha Solteira (366), a maior oferta dentre todas as usinas controladas pela CESP, Tabela 3.3. Como consequência, a oferta estratégia de preço não teve efeito sobre o lucro líquido. Além do mais, o lucro líquido nos cenários 5 e 6 foi menor para a oferta estratégica pois o preço *spot* nestes cenários foi menor do que a oferta de preço das usinas Ilha Solteira, Porto Primavera e Três Irmãos, portanto, estas usinas não foram despachadas e perderam a oportunidade de gerar lucro. Entretanto, esta perda foi compensada pelo alto ganho nos outros cenários, conduzindo para um ganho total de 62%.

Tabela 3.4: Oferta Estratégica, (OE), *versus* Oferta Baseada em Custo, (OC)

Cenário	Preço *Spot*		Geração		Lucro líquido	
	OE	OC	OE	OC	OE	OC
1	363.00	174.00	1874	5736	459355.00	298384.00
2	363.00	176.00	3062	5736	740960.50	309856.00
3	366.00	175.00	3240	5736	793046.50	304120.00
4	370.00	370.00	5736	5736	1422640.00	1422640.00
5	173.00	167.00	1350	5736	74074.00	258232.00
6	193.00	173.00	1350	5736	101074.00	292648.00
7	363.00	175.00	2816	5736	682658.50	304120.00
8	366.00	177.00	3316	5736	811514.50	315592.00
9	366.00	175.00	3838	5736	938360.50	304120.00
10	363.00	173.00	1648	5736	403759.00	292648.00

Finalmente, a Tabela 3.5 mostra a geração (em MW) das usinas Ilha Solteira, Porto Primavera e Três Irmãos em cada cenário. Os valores em negrito indicam a geração em cada cenário das usinas marginais. A geração das outras três usinas controladas pela CESP foi omitida porque essas usinas sempre geram toda sua capacidade.

Tabela 3.5: Geração das Usinas Controladas pela CESP

Cenário	I. Solteira	P. Primavera	T. Irmãos
1	0	0	**524**
2	0	**1183**	529
3	**61**	1300	529
4	2557	1300	529
5	0	0	0
6	0	0	0
7	0	**937**	529
8	**137**	1300	529
9	**659**	1300	529
10	0	0	**298**

Comparando a produção das usinas com a sua capacidade de geração, Tabela 2.2, é possível concluir que Ilha Solteira determinou o preço *spot* nos cenários 3, 8, e 9, que Porto Primavera estabeleceu o preço *spot* nos cenários 2 e 7, e que Três Irmãos determinou o preço *spot* nos cenários 1 e 10.

3.4 Conclusão

Neste capítulo apresentamos uma proposta pioneira para a solução aproximada do PEP, formulado como um PPDN, via AG . As variáveis controladas pelo problema líder são determinadas pelo AG. Para cada indivíduo do AG, o problema seguidor do PPDN, um problema de programação linear, é resolvido exatamente por um algoritmo polinomial. Apresentamos os resultados obtidos com a aplicação do AG para as instâncias do problema, com configuração derivada do sistema brasileiro de energia, e comparamos os resultados com a melhor solução conhecida para as instâncias. Os resultados mostram que o AG é bastante robusto e obteve soluções de boa qualidade para todas as instâncias consideradas. Finalmente, apresentamos um estudo de caso, baseado na análise da solução de uma instância particular do conjunto de instâncias de teste. Dos resultados obtidos, vimos claramente, que a CESP foi capaz de influenciar o preço *spot* do sistema. Estes resultados endossam o interesse dos agentes ofertantes em desenvolver estratégias para otimizar o seu lucro, bem como diminuir a sua quantidade ofertada e, ou aumentar a sua oferta de preço. Por outro lado, o regulador do sistema sempre estará interessado em analisar tal esquema de oferta estratégica como o objetivo de prevenir abusos de mercado. Os dois pontos de vista sustentam a relevância da solução prática do PEP nos mercados de energia. Consideramos o uso da técnica de metaheurística para a solução do PEP como uma das principais contribuições desta pesquisa. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho junto ao Simpósio Brasileiro de Pesquisa Operacional do ano 2012, obtendo o prêmio "GAPSO Best Paper Award for the CLAIO/SBPO - Professor Roberto Diéguez Galvão"atribuído ao melhor artigo, [31], além de uma publicação no periódico *International Transactions in Operational Research* - *ITOR*, [32].

Capítulo 4

Parte III: Relaxação Semidefinida Aplicada ao Problema de Estratégia de Preço

4.1 Revisão Bibliográfica da Técnica de Relaxação Convexa

Nesta seção apresentamos uma revisão bibliográfica cujo objetivo é analisar as pesquisas relacionadas à aplicação de relaxação semidefinida a QCQP não-convexo e à aplicação de desigualdades válidas para reforçar a relaxação SDP. Buscamos com esta pesquisa obter conhecimento necessário para modelar o PEP como um QCQP não-convexo e aplicar relaxação semidefinida para obter fortes limites superiores para as instâncias do PEP modeladas como um QCQP não-convexo. Em [38], $ANSTREICHER$ considerou uma relaxação para um QCQP não-convexo baseada em programação SDP e na técnica RLT. Mostrou-se que o uso de restrições SDP junto com a técnica RLT produzem limites que são melhores do que os limites produzidos com a aplicação isolada de umas das técnicas, porém a um elevado custo computacional. $BURER$ e $VANDENBUSSCHE$, em [39], estudaram um problema de maximização não-convexo de programação quadrática com restrições lineares. Foi colocado que as técnicas existentes de otimização global para um problema de programação quadrática não-convexo geram um número infinito de nós na árvore de *branch-and-bound* (B&B), e que uma questão em aberto de interesse teórico é: como desenvolver um algoritmo de B&B finito para um problema de programação quadrática não-convexo? Os autores informaram que, uma maneira de garantir a exploração de um número finito de nós da árvore de decisão é forçar que a condição $Karush - Kuhn - Tucker$ (KKT) de primeira ordem seja satisfeita nas ramificações. Nesse sentido, foi proposta a utilização de relaxação SDP em cada nó da ramificação, garantindo uma ramificação KKT finita, na solução do problema. Nesta mesma linha de pesquisa, FAMPA *et al.* [40] apresentaram uma relaxação SDP para um problema de programação linear com restrições de equilíbrio, problema de maximização, e aplicaram um algoritmo B&B para obter a solução ótima global do mesmo. Os autores propuseram a utilização de relaxação SDP para gerar fortes limites para os nós da árvore de B&B aliada a uma redução do número de nós processados. Também é mostrado que o relaxação SDP é mais forte do que a relaxação LP, mesmo no nó raiz da árvore de B&B. $SAXENA$ *et al.* [41] abordaram o problema de gerar fortes relaxações convexas do problema de programação inteira mista com restrições quadráticas (MIQCP). Estes problemas são muito difíceis, porque eles combinam dois tipos de não-convexidades: variáveis inteiras e restrições quadráticas não-convexas. Para produzir fortes relaxações para um MIQCP os autores usaram a técnica de programação disjuntiva e a metodologia *lift-and-project*. Em particular, foram propostos novos métodos para gerar desigualdades válidas a partir da equação $X = xx^T$. A restrição não-convexa $X - xx^T \preceq 0$ foi utilizada para gerar disjunções de dois tipos: O primeiro tipo de disjunção é diretamente derivado dos autovetores da matriz $X - xx^T$ associados aos autovalores positivos. Já o segundo tipo de disjunção é obtido através da combinação de vários autovetores para minimizar a largura das disjunções. Utilizaram, também, a restrição SDP convexa $X - xx^T \succeq 0$ para gerar cortes quadráticos convexos. Essas duas abordagens foram combinadas em um algoritmo de plano de corte. Em outro artigo, $SAXENA$ *et al.* [42] informaram que uma maneira comum de produzir uma relaxação convexa de um MIQCP é reformular o problema em um espaço de dimensão superior através da introdução de variáveis X_{ij} para representar cada um dos produtos $x_i x_j$ de variáveis que aparecem na forma quadrática. Informaram, também, que uma vantagem deste tipo de relaxação estendida é que ela pode ser reforçada de forma eficiente usando a restrição SDP, convexa, $X - xx^T \succeq 0$ e programação disjuntiva. Por outro lado, os autores colocam que o principal inconveniente de tal formulação é seu tamanho, mesmo para problemas com um número moderado variáveis. Esse artigo estudou métodos para a construção de relaxação de baixa dimensão para o MIQCP que captam a força da formulação SDP e, para fazer isso, foi usada a técnica de projeção pioneira no âmbito da metodologia de *lift-and-project*. *BAO et al.* [43] colocaram que na interseção de otimização não-linear e combinatória, a programação quadrática tem atraído significativo interesse ao longo das últimas décadas e que uma variedade de relaxações para o QCQP pode ser formulada com SDP. Os autores apresentaram uma comparação sistemática de relaxação SDP para QCQP.

4.2 Relaxação Semidefinida

Nesta seção, propomos uma reformulação do PEP como um QCQP não-convexo e investigamos a aplicação de relaxação semidefinida para obter fortes limites superiores para o problema. A técnica de relaxação SDP de QCQP tem sido estudada por um número crescente de pesquisadores, inspirados pelos trabalhos pioneiros [44–46]. As pesquisas neste campo estão muito ativas como podemos observar nos trabalhos recentes de [38–42, 47] e no artigo de [43].

Embora relaxação SDP seja muito eficaz em gerar fortes limites para um QCQP, é bem sabido que o custo computacional exigido para resolver a relaxação pode ser considerável, especialmente quando o tamanho do modelo relaxado torna-se grande, devido à inclusão de desigualdades válidas. O desafio é, portanto, conseguir um bom compromisso entre o tamanho do modelo relaxado SDP e a qualidade do limite obtido. Nesta seção propomos um algoritmo de plano de corte com o objetivo de obter fortes limites superiores para as instâncias do PEP.

Notação

Neste capítulo, $\mathbb{R}^n$ refere-se ao espaço Euclideano n-dimensional, $e_i \in \mathbb{R}^n$ representa um vetor canônico com a i-ésima componente unitária, S^n é o conjunto das matrizes $(n \times n)$ simétricas, S^n_+ é o conjunto das matrizes $(n \times n)$ simétricas definidas positivas, $\mathbb{R}^{1+n}$ e S^{1+n}_+ são usados para denotar os espaços $\mathbb{R}^n$ e S^n_+ com uma adicional 0-ésima entrada ou adicional 0-ésima linha e coluna, pré-fixadas. Dadas duas matrizes simétricas $(n \times n)$, X, Y, utilizaremos $X \bullet Y = \text{trace}(X^T Y) = \sum_{i=1,j=1}^{n} X_{ij} Y_{ij}$ e usaremos $X \succeq 0$ para denotar que a matriz X é semidefinida positiva.

4.3 Formulação QCQP do Problema de Estratégia de Preço sob Incerteza

O objetivo desta seção é apresentar a formulação do PEP como um QCQP não-convexo. Para esta formulação consideremos o modelo (2.4). Por simplicidade e sem perda de generalidade consideremos $|S| = 1$. Assim, podemos reescreve-lo como:

$$
\begin{array}{lll}
\mathrm{Max}_{\lambda_E} & \sum_{j \in E} (\pi_d - c_j) g_j & \\
\text{s. a} & & \\
\sum_{j \in J} g_j = d, & & \\
0 \leq g_j \leq \bar{g}_j, & j \in E, & \\
0 \leq g_j \leq \bar{g}_j, & j \in J \setminus E, & \\
\pi_d + \pi_{g_j} - \lambda_j \leq 0, & j \in E, & \\
\pi_d + \pi_{g_j} \leq \bar{\lambda}_j, & j \in J \setminus E, & \\
\pi_{g_j} \leq 0, & j \in J, & \\
\sum_{j \in E} \lambda_j g_j + \sum_{j \in J \setminus E} \bar{\lambda}_j g_j - d\pi_d - \sum_{j \in E} \bar{g}_j \pi_{g_j} - \sum_{j \in J \setminus E} \bar{g}_j \pi_{g_j} = 0. & &
\end{array}
\tag{4.1}
$$

Considere um problema quadrático não-convexo, com restrições quadráticas formulado como:

$$
(\mathrm{QCQP}) \left\{
\begin{array}{lll}
\mathrm{Max} & x^T Q_0 x + 2q_0^T x + r_0 & \\
\text{s. a} & x^T Q_j x + 2q_j^T x + r_j \leq 0, & j = 1, \ldots, m_q \\
 & p_j^T x = v_j, & j = 1, \ldots, m_{l_e} \\
 & b_j \leq a_j^T x \leq c_j, & j = 1, \ldots, m_{l_1} \\
 & \alpha_j^T x \leq \gamma_j, & j = 1, \ldots, m_{l_2} \\
 & \beta_j \leq \delta_j^T x, & j = 1, \ldots, m_{l_3}
\end{array}
\right.
\tag{4.2}
$$

onde $Q_j \in S^n$ (Q_j indefinida), $q_j \in \mathbb{R}^n$, $r_j \in \mathbb{R}$, para $j = 0, \ldots, m_q$, $p_j \in \mathbb{R}^n$, $v_j \in \mathbb{R}$, para $j = 1, \ldots, m_{l_e}$, $a_j \in \mathbb{R}^n$, $b_j, c_j \in \mathbb{R}$, para $j = 1, \ldots, m_{l_1}$, $\alpha_j \in \mathbb{R}^n$, $\gamma_j \in \mathbb{R}$, para $j = 1, \ldots, m_{l_2}$, $\delta_j \in \mathbb{R}^n$, $\beta_j \in \mathbb{R}$, para $j = 1, \ldots, m_{l_3}$.

Sejam $\lambda_E = (\lambda_j)$, $g_E = (g_j)$ e $\pi_{g_E} = (\pi_{g_j})$ com $j \in E$, $g_{\bar{E}} = (g_j)$ e $\pi_{g_{\bar{E}}} = (\pi_{g_j})$ com $j \in J \setminus E$ e π_d. Defina $n = 2|J| + |E| + 1$ e $x \in \mathbb{R}^n$, tal que:

$$x = (\lambda_E; g_E; g_{\bar{E}}; \pi_d; \pi_{g_E}; \pi_{g_{\bar{E}}}).$$

Seja $c_E = (c_j)$, $j \in E$ e consideremos e um vetor com todas as componentes iguais a 1. Suponha que:

$$\begin{aligned}
&r_0 = 0,\\
&q_0 = \tfrac{1}{2}(0; -c_E; 0; 0; 0; 0),\\
&Q_0 = \frac{1}{2}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & e^T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},
\end{aligned}$$

então podemos representar a função objetivo $\sum\limits_{j \in E} (\pi_d - c_j) g_j$ da formulação (4.1) na forma quadrática $x^T Q_0 x + 2 q_0^T x + r_0$ como definido em (4.2).

Da mesma forma, sejam $\bar{g}_E = (\bar{g}_j)$ com $j \in E$ e $\bar{\lambda}_{\bar{E}} = (\bar{\lambda}_j)$, $\bar{g}_{\bar{E}} = (\bar{g}_j)$ com

$j \in J \setminus E$. Considere I_E uma matriz identidade de dimensão E.

$$
\begin{aligned}
r_1 &= 0, \\
q_1 &= \tfrac{1}{2}(0; 0; \bar{\lambda}_{\bar{E}}; -d; -\bar{g}_E; -\bar{g}_{\bar{E}}), \\
Q_1 &= \frac{1}{2}\begin{pmatrix}
0 & I_E & 0 & 0 & 0 & 0 \\
I_E & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\end{aligned}
$$

então podemos representar a restrição de complementaridade $\sum_{j \in E} \lambda_j g_j + \sum_{j \in J \setminus E} \bar{\lambda}_j g_j - d\pi_d - \sum_{j \in E} \bar{g}_j \pi_{g_j} - \sum_{j \in J \setminus E} \bar{g}_j \pi_{g_j}$ da formulação (4.1) na forma quadrática $x^T Q_1 x + 2q_1^T x + r_1$ como definido em (4.2).

Consideremos as definições abaixo:

$$
\begin{aligned}
& p = (0; e; 0; 0; 0; 0), && \\
& v = d, && \\
& a_j = (0; e_j; 0; 0; 0; 0), && j \in E, \\
& a_j = (0; 0; e_j; 0; 0; 0), && j \in J \setminus E, \\
& b_j = 0, && j \in J, \\
& c_j = \bar{g}_j, && j \in J, \\
& \alpha_j = (-e_j; 0; 0; 1; -e_j; 0), && j \in E, \\
& \alpha_j = (0; 0; 0; 1; 0; -e_j), && j \in J \setminus E, \\
& \gamma_j = 0, && j \in E, \\
& \gamma_j = \bar{\lambda}_j, && j \in J \setminus E, \\
& \delta_j = (0; 0; 0; 0; e_j; 0), && j \in E, \\
& \delta_j = (0; 0; 0; 0; 0; e_j), && j \in J \setminus E, \\
& \beta_j = 0, && j \in J,
\end{aligned}
$$

consideremos $m_q = m_{l_e} = 1$ e $m_{l_1} = m_{l_2} = m_{l_3} = |J|$. Portanto, diante do que foi apresentado, a formulação definida em (4.1) pode ser representada na forma quadrática, como definido em (4.2).

Em um problema com $|S| > 1$ teríamos $n = |E| + |S|(2|J|+1)$ variáveis originais, onde apenas as variáveis $\lambda_E = (\lambda_j), j \in E$ não dependem de $s \in S$, teríamos ainda $|S| + 4|S||J| + 1$ restrições associadas à formulação original.

4.4 Relaxação SDP aplicada ao QCQP

Uma abordagem padrão para obter a relaxação convexa do QCQP não-convexo (4.2) é introduzir a variável $X \in S_+^n$ na formulação, produzindo a seguinte reformulação em um espaço superior para o problema:

$$
(\text{QCQP'}) \begin{cases} \text{Max} & S_0 \bullet Y \\ \text{s. a} & S_j \bullet Y \leq 0, & j = 1, \ldots, m_q \\ & \quad p_j^T x = v_j, & j = 1, \ldots, m_{l_e} \\ & b_j \leq a_j^T x \leq c_j, & j = 1, \ldots, m_{l_1} \\ & \quad \alpha_j^T x \leq \gamma_j, & j = 1, \ldots, m_{l_2} \\ & \beta_j \leq \delta_j^T x, & j = 1, \ldots, m_{l_3} \\ & Y = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix}, (X = xx^T) \end{cases}
$$

onde $S_j = \begin{pmatrix} r_j & q_j^T \\ q_j & Q_j \end{pmatrix}$, $j = 0, \ldots, m_q$.

A única restrição não-convexa no QCQP' é a última restrição, impondo que Y seja uma matriz semidefinida positiva, rank-1, com $Y_{00} = 1$. Assim, uma relaxação convexa do QCQP' é então dada pelo seguinte problema SDP, obtido por relaxar a restrição rank-1.

$$
(\text{SDP}) \begin{cases} \text{Max} & S_0 \bullet Y \\ \text{s. a} & S_j \bullet Y \leq 0, & j = 1, \ldots, m_q \\ & \quad p_j^T x = v_j, & j = 1, \ldots, m_{l_e} \\ & b_j \leq a_j^T x \leq c_j, & j = 1, \ldots, m_{l_1} \\ & \quad \alpha_j^T x \leq \gamma_j, & j = 1, \ldots, m_{l_2} \\ & \beta_j \leq \delta_j^T x, & j = 1, \ldots, m_{l_3} \\ & S_{m_q+1} \bullet Y = 1, \\ & Y \succeq 0, (X - xx^T \succeq 0) \end{cases}
$$

onde $S_{m_q+1} = e_0 e_0^T$ e $e_0 \in \mathbb{R}^{1+n}$.

Limitantes SDP

Para garantir que a relaxação SDP seja limitada, impomos os seguintes limites superiores à diagonal de Y

$$Y_{ii} \leq \max\{l_i^2, u_i^2\},\ i = 1, \ldots, n, \tag{4.3}$$

onde l_i e u_i são, respectivamente, limites inferior e superior para x_i. Incluímos, também, em SDP as restrições de não-negatividade sobre x_i, dado por:

$$Y_{0j} \geq 0,\ j = 1, \ldots, n, \tag{4.4}$$

Restrições Lineares

Utilizando a idéia introduzida em [48], multiplicamos as restrições lineares originais por cada uma das variáveis do problema para gerar restrições quadráticas válidas reforçando a igualdade entre X_{ij} e $x_i x_j$ para $i, j = 1, \ldots, n$.

Considerando o primeiro tipo de restrições de desigualdade lineares no QCQP, dado por:

$$b \leq a^T x \leq c, \tag{4.5}$$

derivamos a desigualdade convexa quadrática válida:

$$(a^T x - b)(a^T x - c) \leq 0 \Leftrightarrow x^T a a^T x - (b + c) a^T x + bc \leq 0. \tag{4.6}$$

Considerando agora o segundo tipo de restrição de desigualdade:

$$\alpha^T x \leq \gamma \tag{4.7}$$

gerando as desigualdades quadráticas válidas:

$$(\alpha^T x - \gamma)x_i \leq 0 \quad \forall x_i \geq 0. \tag{4.8}$$

Para o terceiro tipo de restrição de desigualdade linear

$$\delta^T x \geq \beta, \tag{4.9}$$

produzimos as desigualdades quadráticas válidas

$$(\delta^T x - \beta)x_i \geq 0 \quad \forall x_i \geq 0. \tag{4.10}$$

Finalmente, para as restrições de igualdades

$$p^T x = v \tag{4.11}$$

derivamos as igualdades quadráticas válidas

$$(p^T x - v)x_i = 0,$$

para cada variável x_i no problema e incluímos na relaxação as duas restrições

$$p^T x = v, \quad \text{e} \quad (p^T x - v)x_i = 0, \forall i = 1, \ldots, n. \tag{4.12}$$

Desigualdades RLT

Para reforçar a relaxação SDP consideramos, também, a bem conhecida técnica RLT, [48–50], ou mais especificamente as seguintes desigualdades bilineares válidas:

$$
\begin{aligned}
(x_i - u_i)(x_j - l_j) &\leq 0 \\
(x_i - l_i)(x_j - u_j) &\leq 0 \\
(x_i - l_i)(x_j - l_j) &\geq 0 \\
(u_i - x_i)(u_j - x_j) &\geq 0
\end{aligned}
$$

gerando as conhecidas desigualdades RLT, dadas por:

$$
\begin{aligned}
X_{ij} - l_j x_i - u_i x_j + l_j u_i &\leq 0 \\
X_{ij} - l_i x_j - u_j x_i + l_i u_j &\leq 0 \\
X_{ij} - l_j x_i - l_i x_j + l_i l_j &\geq 0 \\
X_{ij} - u_j x_i - u_i x_j + u_i u_j &\geq 0.
\end{aligned}
\tag{4.13}
$$

Naturalmente, as restrições limitantes (4.3) estão contidas neste conjunto de desigualdades RLTs.

Observe que todas as desigualdades quadráticas válidas são introduzidas na relaxação SDP como uma restrição linear sobre Y, dada por $S \bullet Y \leq 0$, onde $S = \begin{pmatrix} r & q^T \\ q & Q \end{pmatrix}$, com o apropriado vetor q, r e a submatriz Q. Para a desigualdade quadrática convexa (4.6), por exemplo, temos que $S = \begin{pmatrix} bc & -\frac{1}{2}(b+c)a^T \\ -\frac{1}{2}(b+c)a & aa^T \end{pmatrix}$. Além do mais, neste caso, uma vez que $Q = aa^T$ é semidefinida positiva, a solução projetada x que está contida na primeira linha e coluna da solução, Y, da relaxação semidefinida, necessariamente satisfaz a restrição quadrática (4.6). De fato, considerando X uma sub-matriz de Y obtida quando eliminamos a primeira linha e a primeira coluna de Y, ou seja,

$$Y = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix}, \tag{4.14}$$

então, desde que $Y \succeq 0$, temos que $X - xx^T \succeq 0$. Portanto, $Q \succeq 0$ implica que $Q \bullet (X - xx^T) \geq 0$ e $x^T Q x + 2q^T x + r \leq Q \bullet X + 2q^T x + r = S \bullet Y \leq 0$.

4.5 Algoritmo de Plano de Corte SDP

Baseado nos resultados apresentados na literatura para outras aplicações de relaxação SDP, concluímos que podemos obter limites justos usando fortes relaxações SDP do PEP, porém a um elevado custo computacional como identificado em [51]. Este fato motivou a elaboração de um algoritmo de plano de corte (ASDP) para o QCQP. Portanto, nesta seção apresentamos um ASDP com o objetivo de obter limite superior apertado para a relaxação SDP e ao mesmo tempo reduzir o custo computacional associado. Um ASDP parte de uma relaxação inicial SDP e através de um processo iterativo adiciona restrições válidas à relaxação inicial com o objetivo de obter limites mais justos para o problema. O ASDP seleciona a cada iteração quais restrições devem ser adicionadas à formulação SDP. Esta seleção é obtida através da solução do problema de separação linear (4.15), que possibilita a identificação da restrição mais violada pela solução da relaxação SDP corrente:

$$(\mathrm{S}_{\tilde{X},\tilde{x}}) \begin{cases} \text{Max} & \sum\limits_{i=1}^{n}(S_i \bullet \tilde{Y})\alpha_i \\ \text{s. a} & \sum\limits_{i=1}^{n}\alpha_i \leq 1 \end{cases} \tag{4.15}$$

onde $\tilde{Y}$ é a solução da relaxação SDP e $(S_i \bullet Y) \leq 0 \in P_S$. Seja $S_k \bullet Y \leq 0$ a restrição mais violada de P_S, então a solução ótima, p^* de $(\mathrm{S}_{\tilde{X},\tilde{x}})$ é caracterizada por $\alpha_k = 1$ e $\alpha_i = 0$ para todo $i \neq k$, $i = 1, \ldots, n$. Se vp^* é o valor de p^* então $vp^* = S_k \bullet \tilde{Y}$. Desta maneira, podemos selecionar a restrição mais violada do conjunto P_S para participar da relaxação SDP.

No ASDP proposto, Figura 4.1, partiremos da relação inicial SDP, definida anteriormente na seção (4.4), com as restrições limitantes (4.3 e 4.4) inclusas, além das restrições lineares originais do problema (4.5, 4.7 e 4.9 e 4.11). Esta relaxação inicial apresenta um total de $0.5n(n+3)$ variáveis (x_i, X_{ij}), onde $n = |E| + |S|(2|J|+1)$ e $2 + 3|E| + |S|(7|J|+4)$ são restrições associadas à formulação original, QCQP', mais $n(2n + |S|(|J|+1)+2)$ restricões selecionáveis, as quais definem o conjunto que denotamos por P_S, constituído por $2n(n+1)$ retrições RLTs e $n|S|(|J|+1)$

Entrada: ϵ, $MAXCUT$ e P_S

Seja (SDP) a relaxação semidefinida de (QCQP-L);
Seja $(\tilde{x}, \tilde{Y})$ a solução ótima de (SDP);
Seja $nviol$ o número de restrições RLTs violadas por $\tilde{Y}$;
enquanto ($nviol > 0$) **faça**
  se ($nviol \leq MAXCUT$) **então**
    Adicione todas as restrições RLTs violadas à (SDP);
  senão
    Adicione as $MAXCUT$ restrições RLTs mais violadas à (SDP);
  Seja $(\tilde{x}, \tilde{Y})$ a solução ótima de (SDP);
  Seja $nviol$ o número de restrições RLTs violadas por $\tilde{Y}$;

Saída: O valor da solução ótima da (SDP)

Figura 4.1: Algoritmo de Plano de Corte ($ASDP$)

restrições adicionais (4.8, 4.10 e 4.12). A constante ϵ utilizada no algoritmo assume um valor baixo, próximo de zero, que limita a violação mínima para o ASDP.

4.6 Resultados Numéricos do Algoritmo de Plano de Corte SDP

Consideramos neste experimento a relaxação SDP da formulação QCQP (2.4) do PEP. Desenvolvemos o ASDP, Figura 4.1, com o objetivo de encontrar fortes limites superiores para as instâncias do problema. A Tabela 4.1 apresenta as instâncias geradas, o número de variáveis da formulação QCQP e da formulação estendida, as restrições da formulação QCQP e o número de restrições selecionáveis, P_S, associados à relaxação SDP.

Tabela 4.1: Instâncias Geradas para o ASDP

Instância	Dimensão			
$Inst_{\|J\|,\|E\|,\|S\|}$	$\#Var.(x_i)$	$\#Var.(x_i, X_{ij})$	$\#Restr.$	P_S
Inst08,02,02	36	702	128	3312
Inst08,02,03	53	1484	188	7155
Inst08,02,04	70	2555	248	12460
Inst09,03,02	41	902	145	4264
Inst09,03,03	60	1890	212	9120
Inst09,03,04	79	3239	279	15800
Inst10,04,02	46	1127	162	5336
Inst10,04,03	67	2345	236	11323
Inst10,04,04	88	4004	310	19536

O principal objetivo desse experimento numérico é comparar os resultados da aplicação do ASDP com os resultados da solução da relaxação contínua (RLC) do modelo MILP. Analisaremos, ainda, o impacto do ASDP sobre a qualidade dos limites superiores encontrados e sobre o tempo computacional requerido na solução das relaxações. A cada iteração do ASDP, $MAXCUT = \lfloor |P_S|/100 \rfloor$ restrições mais violadas são inseridas à relaxação corrente. Utilizamos o valor de $\epsilon = 0.001$ como limite para violação mínima no ASDP.

O código foi implementado na linguagem de programação C e compilado com gcc (GNU *COMPILE* C). Todas as execuções foram realizadas em uma máquina com a seguinte configuração: *Precision WorkStation* T7400, *Intel*(R) *Xeon*(R) CPU X5472@3GHz, 8GB, Ubuntu 9.10. As soluções SDPs foram obtidas com o solver

CSDP 6.1.1, [52]. O solver CPLEX, v12.5 [36] foi utilizado para extrair a solução MILP e RLC.

A Tabela 4.2 apresenta o resultado da relaxação RLC aplicada às instâncias geradas para este estudo. Apresentamos o número de variáveis e o número de restrições associados ao MILP, a média dos $DPRs$ entre a solução da relaxação RLC e a solução MILP do PEP (DPR_{RLC}), bem como o custo computacional médio em segundos. O desvio percentual relativo, DPR utilizado nesta análise é dado por:

$$DPR(\%) = 100 \times \frac{Z^*_{Relax.} - Z^*_{MILP}}{Z^*_{MILP}}. \tag{4.16}$$

onde $Z^*_{Relax.}$ corresponde à solução ótima de uma das relaxações e Z^*_{MILP} corresponde à solução ótima MILP.

Tabela 4.2: Resultados da RLC do MILP

Instância	Solução da RLC			
$Inst_{\|J\|,\|E\|,\|S\|}$	$\#Var.$	$\#Restr.$	DPR_{RLC}	$Tempo$
Inst08,02,02	88	211	40,75%	0,13
Inst08,02,03	123	316	20,68%	0,04
Inst08,02,04	158	421	49,33%	0,05
Inst09,03,02	119	291	40,49%	0,04
Inst09,03,03	165	436	30,40%	0,05
Inst09,03,04	211	581	48,94%	0,05
Inst10,04,02	150	371	51,88%	0,05
Inst10,04,03	207	556	34,52%	0,05
Inst10,04,04	264	741	49,74%	0,06

A Tabela 4.3 apresenta o resultado da aplicação do ASDP nas instâncias consideradas neste experimento, onde cada linha representa a média dos resultados da aplicação do ASDP para cinco instâncias de mesmo tamanho do QCQP. A primeira coluna da Tabela 4.3 traz o nome das instâncias do experimento, a segunda coluna exibe o DPR entre a solução do ASDP e a solução ótima do MILP (DPR_{ASDP}), a terceira coluna apresenta o DPR entre a solução do ASDP e a solução da relaxação RLC, $DPR_{ASDP-RLC}$. Finalmente, a quarta coluna apresenta o número de iteraçoes do ASDP, a quinta coluna apresenta o tempo total da execução do ASDP, expresso em segundos e a sexta coluna apresenta o percentual médio de restrições

RLTs inseridas pelo ASDP para cada grupo de instâncias. Este percentual é dado por:

$$Perc.(\%) = 100 \times \frac{Iter. \times MAXCUT}{|P_S|}. \tag{4.17}$$

Tabela 4.3: Análise da Aplicação do ASDP

Inst|J|,|E|,|S|	DPR_{ASDP}	$DPR_{ASDP-RLC}$	$Iter.$	$Tempo$	$Perc.$
Inst08,02,02	0,99%	-27,93%	20	64,14	19,80%
Inst08,02,03	0,01%	-16,86%	13	195,03	12,80%
Inst08,02,04	0,86%	-30,68%	24	4842,36	24,00%
Inst09,03,02	1,24%	-27,27%	26	266,97	26,40%
Inst09,03,03	4,09%	-18,70%	21	970,62	21,40%
Inst09,03,04	1,83%	-30,84%	23	5248,35	22,60%
Inst10,04,02	3,17%	-29,86%	30	613,92	29,60%
Inst10,04,03	1,95%	-21,83%	21	1617,48	21,40%
Inst10,04,04	1,52%	-30,99%	18	5061,39	17,80%

Os resultados da Tabela 4.3 mostram que o ASDP encontrou fortes limites superiores para todos os grupos de instâncias, segunda coluna da tabela. Observando a terceira coluna concluímos que em todas as instâncias o ASDP conseguiu ser melhor do que a relaxação RLC, apresentando um ganho máximo de $30,99\%$, mostrando a eficácia da metodologia, porém o ASDP apresentou um elevado custo computacional mesmo inserindo no máximo um percentual médio de $29,60\%$ do número total de restrições selecionáveis $|P_S|$.

4.7 Conclusão

Neste capítulo apresentamos uma formulação matemática do PEP como um QCQP não-convexo e investigamos a aplicação de relaxação SDP para calcular fortes limites superiores para o problema. Realizamos a aplicação de um ASDP com o objetivo de determinar forte relaxação SDP a um custo computacional reduzido se comparado com o custo computacional necessário para resolver a relaxação derivada da adição de todas as restrições de PS. Verificamos a eficiência do ASDP na obtenção de fortes limites para o problema. Notamos ainda que o ASDP superou com folga a RLC em todas as instâncias, obtendo um ganho médio de $26,11\%$, representado na terceira coluna da tabela, porém apresentando um elevado custo computacional. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho no Simpósio Brasileiro de Pesquisa Operacional do ano 2013, [51].

Capítulo 5

Parte IV: Relaxação Linear Estendida Aplicada ao Problema de Estratégia de Preço

5.1 Revisão Bibliográfica da Técnica de Relaxação Linear Estendida

Nesta seção apresentamos a revisão bibliográfica da técnica de relaxação linear estendida, RLE. A principal referência associada à metodologia abordada neste capítulo é [53]. Em [53], o autor trata da aplicação de um APC para determinar fortes limites superiores para QCQPs. A RLE é uma aproximação exterior da relaxação SDP do QCQP, relaxando a restrição $Y \succeq 0$, reforçadas por desigualdades válidas que são adicionadas a uma relaxação inicial, iterativamente, por meio de um APC, apertando ou aproximando o limite superior do limite da relaxação SDP. Estas desigualdades válidas são derivadas da técnica RLT e de cortes esparsos que surgem da decomposição espectral da matriz, Y, solução de relaxação estendida. Nesse artigo foi proposto um procedimento de geração de cortes esparsos a partir de cortes densos SDP, visto que cortes densos acarretam um custo computacional extra na solução da RLE.

5.2 Relaxações Linear Estendida

Neste capítulo consideramos uma reformulação do PEP como um QCQP não-convexo e investigamos a aplicação de LP na construção de RLE para o problema proposto, baseado em uma relaxação semidefinida. Embora a relaxação semidefinida seja bastante efetiva na geração de fortes limites para QCQP, isto requer um considerável custo computacional, especialmente quando o tamanho do modelo relaxado torna-se, demasiadamente, grande devido a inclusão de desigualdades válidas. Para contornar esta dificuldade, a aproximação LP estendida a partir de uma relaxação SDP tem sido aplicada na literatura. Por exemplo, [53] investigou RLE de relaxação SDP com objetivo de capturar a força da relaxação SDP com a performance dos resolvedores LP para obter fortes limites superiores para as intâncias do QCQP, a um baixo custo computacional se comparado com a técnica SDP.

Neste capítulo aplicaremos RLE reforçada por restrições RLTs e também por cortes SDPs derivados da decomposição espectral da matrix, Y, solução da relaxação RLE, que aproximam a restrição semidefinida positica $Y \succeq 0$.

5.3 Aproximação Linear Estendida de Relaxação SDP

Uma RLE pode ser obtida de uma relaxação SDP, substituindo a última restrição, $Y \succeq 0$, por $Y = Y^T$, ou seja, impondo somente a simetria da matriz Y. Assim, obtemos a seguinte formulação linear estendida:

$$
(\text{QCQP-L}) \left\{
\begin{array}{lll}
\text{maximize} & S_0 \bullet Y & \\
\text{subject to} & S_j \bullet Y \leq 0, & j = 1, \ldots, m_q \\
& \quad p_j^T x = \theta_j, & j = 1, \ldots, m_{l_e} \\
& \quad \alpha_j^T x \leq \gamma_j, & j = 1, \ldots, m_{l_1} \\
& \beta_j \leq \delta_j^T x, & j = 1, \ldots, m_{l_2} \\
& S_{m_q+1} \bullet Y = 1, & \\
& Y = Y^T, &
\end{array}
\right.
$$

onde o QCQP-L é uma RLE em x e Y com $0.5n(n+3)$ variáveis e o mesmo número de restrições do modelo SDP, como definido na seção (4.4). Observe que o valor ótimo do QCQP-L, sem as restrições que ligam os valores de x e Y, é usualmente um limite superior fraco para a relaxação SDP. Propomos a adição das restrições RLTs, como definido na seção (4.4), e dos cortes SDPs derivados da decomposição espectral da matriz Y, solução da relaxação, para reforçar o limites superiores produzidos com esta metodologia.

5.3.1 Cortes SDPs

Os cortes SDPs considerados nesta pesquisa são baseados no fato da matriz Y ser semidefinida positiva, se e somente se:

$$v^T Y v \geq 0, \forall v \in R^{n+1} \tag{5.1}$$

Estes cortes SDPs serão utilizados no APC proposto e serão detalhados na próxima seção.

5.4 Gerando Cortes Esparsos

Podemos reformular a relaxação SDP como uma relaxação linear estendida a seguir:

$$
\text{(SDP-L)} \begin{cases}
\text{maximize} & S_0 \bullet Y & \\
\text{subject to} & S_j \bullet Y \leq 0, & j = 1, \ldots, m_q \\
& \quad p_j^T x = \theta_j, & j = 1, \ldots, m_{l_e} \\
& \quad \alpha_j^T x \leq \gamma_j, & j = 1, \ldots, m_{l_1} \\
& \beta_j \leq \delta_j^T x, & j = 1, \ldots, m_{l_2} \\
& S_{m_q+1} \bullet Y = 1, & \\
& Y = Y^T, & \\
& v^T Y v \geq 0, \forall v \in R^{n+1}, &
\end{cases}
$$

seja $\tilde{Y}$ um ponto arbitrário do espaço de busca das variáveis Y. A decomposição espectral de $\tilde{Y}$ é utilizada para decidir se $\tilde{Y}$ está, ou não, no cone das matrizes semidefinidas positivas. Sempre será possível aplicar tal decomposição para uma matriz simétrica, $Y = Y^T$, [54]. Sejam λ_k e v_k, para $k = 1, 2, ..., n$, o autovalor e o correspondente autovetor ortonormal de $\tilde{Y}$, respectivamente. Assuma sem perda de generalidade que $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ e seja $t \in \{0, ..., n\}$ tal que, $\lambda_t \leq 0 \leq \lambda_{t+1}$. Se $t = 0$, então todos os autovalores são não-negativos e $\tilde{Y}$ é semidefinida positiva. Por outro lado, se $t > 0$ então $v_k^T \tilde{Y} v_k = \lambda_k < 0$, para $k = 1, ..., t$ e portanto, o corte válido $v_k^T Y v_k \geq 0$ é violado por $\tilde{Y}$ e pode ser adicionado à relaxação, eliminado esta solução do conjunto viável do problema.

Em [53] os autores afirmam que este procedimento possui duas desvantagens, a saber: primeiro, apenas um corte é gerado de cada autovetor v_k para $k = 1, \ldots, t$, enquanto que a decomposição espectral requer um não desprezível custo computacional. Segundo, os cortes são usualmente muitos densos, ou seja, a maior parte das entradas em $v_k v_k^T$ são não-zero. Os autores afirmam que cortes densos não são bons para serem utilizados em abordagens de plano de corte, pois eles tornam mais lento o processo de reotimização da RLE.

Para evitar essas desvantagens, eles propõem um procedimento para geração de cortes esparsos para os autovetores. A idéia básica para computar um simples corte esparso é iniciar com o vetor $w = v_k$, para $k = 1, \ldots, t$, e iterativamente fixar algumas componentes de w em zero, garantindo que $w^T \tilde{Y} w$ permaneça suficientemente negativo. Se as componentes zeradas forem consideradas em uma ordem aleatória, alguns cortes novos podem ser obtidos de um simples autovetor v_k.

O algoritmo para gerar corte esparso de um dado autovetor v_k é reproduzido diretamente de [53] pelo procedimento *SparseCut* mostrado na Figura 5.1. O algoritmo recebe como entradas o autovetor v_k, a matriz $\tilde{Y}$, e dois números entre 0 e 1, pct_{NZ} e pct_{VIOL}, que controlam o percentual máximo de entradas não-zero no vetor final e a violação mínima requerida para o correspondente corte, respectivamente. No procedimento, o parâmetro $length[v_k]$ corresponde ao tamanho do vetor v_k.

SparseCut(v_k,$\tilde{Y}$, pct_{NZ} e pct_{VIOL})
min_{VIOL}=$(-v_k^T \tilde{Y} v_k)\ pct_{VIOL}$;
$max_{NZ} = \lfloor length[v_k] pct_{NZ} \rfloor$;
w=v_k;
$perm$ = permutação aleatória de 1 até $length[w]$;
para $(i = 1, \ldots, length[w])$ **faça**
    $z = w$;
    $z[perm[i]] = 0$;
    se $(-z^T \tilde{Y} z > min_{VIOL})$ **então**
        $w = z$;
se (*número de entadas não-zero em* $w < max_{NZ}$) **então**
    return w;
senão
    return *null*;

Figura 5.1: Procedimento de Cortes Esparsos (*SparseCut*)

5.5 Algoritmo de Plano de Corte

Nesta seção propomos um APC para obter forte limite superior para o PEP. O algoritmo parte, inicialmente, da formulação QCQP-L e, iterativamente, adiciona as restrições RLTs, como definido na seção (4.4), e os cortes esparsos, como definido na Figura 5.1. Nesta pesquisa, propomos uma atualização dinâmica do parâmetro pct_{NZ} com o objetivo de permitir que cortes menos esparsos sejam produzidos e adicionados à relaxação durante o processo de convergência do algoritmo, melhorando os limites superiores obtidos.

Entrada: pct_{NZ}, pct_{VIOL}, λ_{MAX} e $MAXCUT$

Seja (RLE) a relaxação linear estendida (QCQP-L);
Seja $(\tilde{x}, \tilde{Y})$ a solução ótima de (RLE);
enquanto (*critério de parada*) **faça**
    Seja $nviol$ o número de restrições RLTs violadas por $\tilde{Y}$;
    se $(nviol > 0)$ **então**
        se $(nviol \leq MAXCUT)$ **então**
            Adicione todas as restrições RLTs violadas à (RLE);
        senão
            Adicione as $MAXCUT$ restrições RLTs mais violadas à (RLE);
    senão
        Sejam λ_k e v_k para $k = 1, \ldots, n$, respectivamente, os autovalores e os seus correspondentes autovetores ortonormais de $\tilde{Y}$, tal que $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$;
        $tot = 0$; $k = 1$;
        enquanto $(\lambda_k < \lambda_{MAX})$ **faça**
            $w_k = SparseCut(v_k, \tilde{Y}, pct_{NZ}, pct_{VIOL})$;
            se $(w_k \neq null)$ **então**
                Adicione as restrições $w_k^T Y w_k \geq 0$ à (RLE);
                $tot = tot + 1$;
            $k = k + 1$;
        se $(tot == 0)$ **então**
            $pct_{NZ} = 1.005 pct_{NZ}$;
    Seja $(\tilde{x}, \tilde{Y})$ a solução ótima de (RLE);

Saída: O valor da solução ótima da (RLE)

Figura 5.2: Algoritmo de Plano de Corte (APC1)

Na Figura 5.2 apresentamos o APC1. As entradas do APC1 são representadas pelos parâmetros: pct_{NZ}, pct_{VIOL}, λ_{MAX} e $MAXCUT$. Os dois primeiros parâmetros foram descritos na seção anterior e serão repassados ao procedimento $SparseCut$, descrito na Figura 5.1. O parâmetro $\lambda_{MAX} < 0$ corresponde ao máximo valor dos autovalores utilizados na geração dos cortes esparços, e o parâmetro $MAXCUT$ representa o número máximo de cortes adicionados à relaxação a cada iteração do APC1. O APC1 é dividido em duas fases. Na primeira fase (linhas 4-9) as restrições RLTs mais violadas são iterativamente adicionadas a formulação até que não exista mais restrições violadas pela solução da relaxação atual. Então, na segunda fase (linhas 11-20), a cada iteração do APC1, cortes esparsos gerados são adicionados e se nenhum corte esparso for gerado pelo procedimeto $SparseCut$ o valor do parâmetro pct_{NZ} será atualizado em 0.5% (linha 20), permitindo que na próxima iteração cortes menos esparsos sejam gerados e adicionados à relaxação corrente.

Como descrito na primeira fase do APC1, Figura 5.2, no máximo $MAXCUT$ cortes RLTs serão adicionados à relaxação a cada iteração (linhas 6-9) enquanto que na segunda fase do referido algoritmo todos os cortes esparsos gerados associados aos autovalores limitados por λ_{MAX} (linhas 14-18) serão, da mesma forma, adicionados à relaxação corrente. Os valores dos parâmetros utilizados no experimento numérico bem como o critério de parada utilizado (linha 3) serão descritos na próxima seção.

5.6 Resultados Numéricos do Algoritmo de Plano de Corte APC1

O principal objetivo com este experimento numérico é analisar a qualidade dos limites superiores obtidos com o APC1 para as intâncias QCQPs, Tabela 4.1, comparando os resultados da aplicação das metodologias.

O código foi implementado na linguagem de programação C e compilado com gcc (GNU *COMPILE* C). Todas as execuções foram realizadas em uma máquina com a seguinte configuração: *Precision WorkStation* T7400, *Intel*(R) *Xeon*(R) CPU X5472@3GHz, 8GB, Ubuntu 9.10. O resolvedor CPLEX v.12.5, [36], foi utilizado para obter a solução MILP e RLC e para resolver a RLE do APC1. Utilizamos a função *dsyev*, disponibilizada pela biblioteca LAPACK v.3.4.2., para calcular os autovalores e os autovetores ortonormais da matriz solução da relaxação, $\tilde{Y}$.

Utilizamos a convergência do APC1 como critério de parada (linha 3), adaptado de [53]. Considerando z_t com a solução da RLE na iteração t, o APC1 termina, apenas, se $t_{purif} \geq 50$ e $z_t \geq (1 - 0.0001).z_{t-50}$. Observe que um procedimento de purificação foi implementado para remover da RLE os cortes inativos a cada iteração t, se $t_{purif} \geq maxIterPurif$ e $z_t \geq (1 - 0.0001).z_{t-1}$. Para evitar uma convergência prematura, o parâmetro t_{purif} é reiniciado a cada purificação. Utilizamos $maxIterPurif = 40$, atualizando em 1% a cada purificação até o limite $maxIterPurif = 60$. Isto garante um número mínimo de iterações para o APC1, ou seja, o APC1 só pode ser finalizado se $maxIterPurif \geq 50$, assegurando que $t_{purif} \geq 50$ possa ocorrer. Na prática o APC1 permite que pelo menos 500 iterações sejam realizadas.

Neste experimento numérico, utilizamos $pct_{VIOL} = 0.95$, $pct_{NZ} = 0.4$, $\lambda_{MAX} = -0.5$, e $MAXCUT = \lfloor |P_S|/300 \rfloor$, Tabela 4.1.

A Tabela 5.1 compara os limites superiores produzidos pelo APC1, a solução ótima da relaxação RLC, os limites superiores produzidos com o algoritmo ASDP em relação à solução ótima MILP. Os resultados apresentados na Tabela 5.1 cor-

Tabela 5.1: Análise da Aplicação do APC1

Inst|J|,|E|,|S|	DPR_{RLC}	DPR_{ASDP}	DPR_{APC1}	$I._{APC1}$	$T._{APC1}$
Inst08,02,02	40,75%	0,99%	1,28%	853	20,63
Inst08,02,03	20,68%	0,01%	0,56%	346	26,02
Inst08,02,04	49,33%	0,86%	2,90%	452	270,11
Inst09,03,02	40,49%	1,24%	2,77%	811	27,76
Inst09,03,03	30,40%	4,09%	5,44%	559	68,62
Inst09,03,04	48,94%	1,83%	4,09%	560	618,49
Inst10,04,02	51,88%	3,17%	5,73%	649	34,19
Inst10,04,03	34,52%	1,95%	3,85%	460	145,42
Inst10,04,04	49,74%	1,52%	2,14%	457	990,60

respondem à média das melhores soluções do APC1 considerando dez execuções. Com a finalidade de comparar os resultados obtidos com as três metodologias, apresentamos a média dos $DPRs$ entre a solução ótima da relaxação RLC e a solução ótima MILP (DPR_{RLC}), a média dos $DPRs$ entre a solução do algoritmo ASDP e a solução ótima MILP (DPR_{ASDP}) e a média dos $DPRs$ entre a média APC1 e a solução ótima MILP (DPR_{APC1}). Apresentamos ainda o número médio de iterações e o tempo médio gasto APC1, em segundos.

Tabela 5.2: Comparação dos Limites Superiores entre o APC1 e o ASDP

Inst|J|,|E|,|S|	DPR_{APC1}	DPR_{ASDP}	$T._{APC1}$	$T._{ASDP}$
Inst08,02,02	1,28%	0,99%	20,63	64,14
Inst08,02,03	0,56%	0,01%	26,02	195,03
Inst08,02,04	2,90%	0,86%	270,11	4842,36
Inst09,03,02	2,77%	1,24%	27,76	266,97
Inst09,03,03	5,44%	4,09%	68,62	970,62
Inst09,03,04	4,09%	1,83%	618,49	5248,35
Inst10,04,02	5,73%	3,17%	34,19	613,92
Inst10,04,03	3,85%	1,95%	145,42	1617,48
Inst10,04,04	2,14%	1,52%	990,60	5061,39

A Tabela 5.2 compara os limites superiores produzidos pelo APC1 e pelo algoritmo ASDP. Com a finalidade de comparar os resultados obtidos com as duas metodologias, apresentamos o DPR entre a média das soluções do algoritmo APC1 e a solução ótima MILP (DPR_{APC1}). Apresentamos, também, o DPR entre a melhor solução do algoritmo ASDP e a solução ótima MILP (DPR_{ASDP}), além do tempo médio gasto pelas duas metodologias, em segundos.

Concluímos que o APC1 obteve resultados satisfatórios que lhe colocam como uma excelente alternativa na solução de QCQP não-covexos. O APC1 obteve um ganho médio de $25,11\%$ em relação à solução RLC, representado na segunda coluna da Tabela 5.1, e bons limites superiores se comparados com o ASDP representados na quarta coluna desta tabela, além de um reduzido custo computacional, Tabela 5.2. O ganho computacional médio é de $88,34\%$, como podemos observar nas duas útimas colunas desta tabela, ou seja, o custo computacional médio do APC1 corresponde a $11,66\%$ do custo computacional do ASDP.

A Tabela 5.3 compara os limites superiores produzidos pelo APC1 com os limites produzidos com o APC1 sem a atualização do parâmetro pct_{NZ} (ou APC2). Neste último caso o algoritmo torna-se similar ao proposto em [53]. Os resultados apresentados na Tabela 5.3 correspondem à média das melhores soluções do APC1 e do APC2 considerando dez execuções dos algoritmos. Com o objetivo de comparar os resultados obtidos pelos APCs, apresentamos o DPR entre as soluções ($DPR_{APC1-APC2}$). Apresentamos, também, o número médio de iterações e o tempo médio gasto pelas propostas, em segundos.

Tabela 5.3: Comparação dos Limites Superiores entre o APC1 e o APC2

Inst|J|,|E|,|S|	$DPR_{APC1-APC2}$	$I._{APC1}$	$T._{APC1}$	$I._{APC2}$	$T._{APC2}$
Inst08,02,02	-1,98%	853	20,63	456	5,17
Inst08,02,03	-0,14%	346	26,02	254	16,24
Inst08,02,04	0,00%	452	270,11	452	171,00
Inst09,03,02	-0,82%	811	27,76	457	9,06
Inst09,03,03	0,02%	559	68,62	559	60,05
Inst09,03,04	0,00%	560	618,49	560	431,74
Inst10,04,02	-0,15%	649	34,19	559	20,90
Inst10,04,03	0,00%	460	145,42	465	149,57
Inst10,04,04	0,00%	457	990,60	457	851,26

Concluímos que o APC1 obteve um ganho médio de $0,34\%$ em relação ao APC2 e que o APC1 realizou um custo computacional médio de 28% a mais em comparação com o APC2.

A Tabela 5.4 compara os limites superiores produzidos pelo APC1 com os limites produzidos com o APC1 denso (ou APC3), ou seja, sem utilizar o procedimento

SparseCut para produção de cortes esparsos. Os resultados apresentados na Tabela 5.4 correspondem à média das melhores soluções do APC1 considerando dez execuções do algoritmo APC1 e a solução do APC3. Com o objetivo de comparar os resultados obtidos pelos APC1 e APC3, calculamos para cada instância o DPR entre as soluções ($DPR_{APC1-APC3}$). Apresentamos, ainda, o número médio de iterações e o tempo médio gasto pelos algoritmos, em segundos.

Tabela 5.4: Comparação dos Limites Superiores entre o APC1 e o APC3

Inst|J|,|E|,|S|	$DPR_{APC1-APC3}$	$I._{APC1}$	$T._{APC1}$	$I._{APC3}$	$T._{APC3}$
Inst08,02,02	0,06%	853	20,63	788	35,82
Inst08,02,03	-0,14%	346	26,02	254	24,43
Inst08,02,04	0,00%	452	270,11	452	1722,75
Inst09,03,02	1,18%	811	27,76	1453	173,66
Inst09,03,03	0,58%	559	68,62	1172	1059,26
Inst09,03,04	0,00%	560	618,49	560	5741,85
Inst10,04,02	1,73%	649	34,19	1304	625,60
Inst10,04,03	0,00%	460	145,42	460	597,50
Inst10,04,04	0,00%	457	990,60	457	6641,64

Concluímos que o APC1 obteve resultados satisfatórios em relação ao APC3 apresentando um desvio médio de $0,38\%$, representado na segunda coluna da tabela e que o APC1 realizou um custo computacional médio equivalente a 13% do custo computacional do APC3. Isto reforça a necessidade da utilização do procedimento *SparseCut* na produção de cortes esparsos que são menos custosos para o processo de reotimização do APC.

5.7 Conclusão

Neste capítulo apresentamos um estudo da aplicação de um APC para resolver a RLE do PEP formulado como um QCQP não-convexo, reforçado pelas restrições RLTs e por cortes SDPs que aproximam passo a passo a variável matricial Y da matriz semidefinida positiva, $Y \succeq 0$, da relaxação SDP. Os resultados numéricos mostram que o APC1 obteve bons limites superiores para as instâncias do QCQP, porém a um baixo custo computacional em comparação com o ASDP. Portanto, fica demonstrado pelos resultados obtidos que a técnica de relaxação linear estendida surge com uma excelente alternativa para determinar fortes limites superiores para um QCQP não-convexo a um razoável custo computacional, se cortes esparsos forem utilizados. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho junto ao Simpósio Brasileiro de Pesquisa Operacional do ano 2014, [55].

Capítulo 6

Conclusão e Trabalhos Futuros

Apresentamos neste trabalho um estudo do problema de estratégia de preço sob incerteza em um mercado de energia (PEP). No capítulo (2) apresentamos inicialmente uma revisão bibliográfica do PEP, apresentamos em seguida um modelo genérico de um problema de programação em dois níveis (PPDN), introduzimos o PEP e sua formulação sob incerteza. Apresentamos uma reformulação do PEP como um problema de programação matemática com restrição de equilíbrio e finalmente apresentamos um modelo de programação inteira mista (MILP) associado ao PEP. Este último foi utilizado para gerar a melhor solução para as instâncias do PEP geradas como descrito no referido capítulo.

No capítulo (3) apresentamos um algoritmo genético (AG) para o PEP em um mercado de venda de energia por atacado, que foi formulado como um PPDN. Apresentamos os resultados obtidos com a aplicação do AG para as instâncias do problema, com configuração derivada do sistema brasileiro de energia, e comparamos os resultados com a melhor solução conhecida para as instâncias. Os resultados mostraram a robustez do AG que obteve soluções de boa qualidade para todas as instâncias consideradas. Finalmente apresentamos um estudo de caso, baseado na análise da solução de uma instância particular selecionada do conjunto de instâncias de teste. Dos resultados obtidos, vimos claramente, que a Companhia Energética de São Paulo (CESP) foi capaz de influenciar o preço *spot* do sistema. Estes resultados endossam o interesse dos agentes ofertantes em desenvolver estratégias para otimizar o seu lucro, bem como diminuir a sua quantidade ofertada e, ou aumentar a sua oferta de preço. Por outro lado, o regulador do sistema sempre estará interessado em analisar tal esquema de oferta estratégica como o objetivo de prevenir abusos de mercado. Os dois pontos de vista sustentam a relevância da solução prática do PEP nos mercados de energia. Consideramos o uso da técnica de metaheurística para a solução do PEP como uma das principais contribuições desta pesquisa. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho junto ao Simpósio Brasileiro de Pesquisa Operacional do ano 2012, obtendo o prêmio "GAPSO Best Paper Award for the CLAIO/SBPO - Professor Roberto Diéguez Galvão"atribuído ao melhor artigo, [31], além de uma publicação no periódico *International Transactions in Operational Research* - *ITOR*, [32].

No capítulo (4) apresentamos uma formulação matemática do PEP como um problema de programação quadrático com restrições quadráticas (QCQP) não-convexo e investigamos a aplicação de relaxação semidefinida (SDP) para calcular fortes limites superiores para o problema. Realizamos a aplicação de um algoritmo de plano de corte SDP (ASDP) com o objetivo de determinar forte relaxação SDP a um esforço computacional reduzido se comparado com o esforço necessário para resolver uma relaxação SDP reforçada pela adição de todas as restrições derivadas da técnica de reformulação por linearização (RLTs). Verificamos a eficiência do ASDP na obtenção de fortes limites para as instâncias do problema. Notamos ainda que o ASDP superou com folga a relaxação linear contínua (RLC) do modelo MILP em todas as instâncias, porém apresentando um elevado esforço computacional. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho junto ao Simpósio Brasileiro de Pesquisa Operacional do ano 2013, [51].

No capítulo (5) apresentamos um estudo da aplicação de um APC para resolver a relaxação linear estendida (RLE) do PEP formulado como um QCQP não-convexo, reforçado pelas restrições RLTs e por cortes espectrais esparsos que aproximam passo a passo a variável matricial Y da matriz semidefinida positiva, $Y \succeq 0$, da relaxação SDP. Os resultados numéricos mostraram que o APC obteve bons limites superiores para as instâncias do QCQP além de um reduzido custo computacional em comparação com o algoritmo ASDP. Portanto, fica demostrado pelos resultados obtidos que a técnica de relaxação linear estendida surge com uma excelente alternativa para determinar fortes limites superiores para as instâncias de um QCQP não-convexo a um razoável custo computacional, se cortes esparsos forem utilizados. Como resultado da pesquisa descrita neste capítulo, apresentamos um trabalho junto ao Simpósio Brasileiro de Pesquisa Operacional do ano 2014, [55]. Consideramos o uso da técnica de relaxação linear estendida aplicada ao PEP por meio de um APC como uma das principais contribuições desta pesquisa.

Como trabalhos futuros destacamos as seguintes propostas:

-Aplicar o BRKGA *framework*, [56], nas instâncias do PEP e confrontar os resultados com os resultados produzidos com o AG proposto neste trabalho.

-Aplicar o APC proposto neste trabalho para solucionar a RLE de outros problemas QCQPs.

-Desenvolver um algoritmo exato para solucionar o PEP aplicando a metaheurística e a RLE apresentada nesta pesquisa.

Referências Bibliográficas

[1] KWON, R. H., FRANCES, D. *Optimization-Based Bidding in Day-Ahead Electricity Auction Markets: A Review of Models for Power Producers*. In: Handbook of networks in power systems i, energy systems, a. sorokin et al. ed. , Springer-Verlag Berlin Heidelberg, 2012.

[2] RAMOS, A., VENTOSA, M., RIVIER, M. "Modeling Competition in Electric Energy Markets by Equilibrium Constraints", *Utilities Policy*, v. 7, n. 4, pp. 233–242, 1999.

[3] CONEJO, A. J., PRIETO, F. J. "Mathematical Programming and Electricity Markets", *Sociedad de Estatistica e Investigacion Operativa TOP*, v. 9, n. 1, pp. 1–53, 2001.

[4] CONEJO, A. J., CONTRERAS, J., ARROYO, J. M., et al. "Optimal Response of an Oligopolistic Generating Company to a Competitive Pool-Based Electric Power Market", *IEEE Transactions of Power Systems*, v. 17, n. 2, pp. 424–430, 2002.

[5] HOBBS, B. F. "Linear Complementarity Models of Nash-Cournot Competition in Bilateral and POOLCO Power Markets", *IEEE Transactions on Power Systems*, v. 6, n. 2, pp. 194–202, 2001.

[6] HOBBS, B. F., HELMAN, U. *Complementarity-Based Equilibrium Modeling for Electric Power Markets*. In: Modeling prices in competitive electricity markets, d. bunn ed. , J. Wiley, 2004.

[7] BUSHNELL, J. "A Mixed Complementarity Model of Hydrothermal Electricity Competition in the Western United States", *Operations Research*, v. 51, n. 1, pp. 80–93, 2003.

[8] DE LA TORRE, S., ARROYO, J. M., CONEJO, A. J., et al. "Price Maker Self-Scheduling in a Pool-Based Electricity Market: a Mixed-Integer LP Approach", *IEEE Transactions on Power Systems*, v. 17, n. 4, pp. 1037–1042, 2002.

[9] BAÃLLO, A., M. VETOSA, M. R., RAMOS, A. "Optimal Offering Strategies for Generation Companies Operating in Electricity Spot Markets", *IEEE Transactions on Power Systems*, v. 19, n. 2, pp. 745–753, 2004.

[10] WEBER, J. D., OVERBYE, T. J. "An Individual Welfare Maximization Algorithm for Electricity Markets", *IEEE Transactions on Power Systems*, v. 17, n. 3, pp. 590–596, 2002.

[11] HOBBS, B. F., METZLER, C. B., PANG, J. "Strategy Gaming Analysis for Electric Power Systems: an MPEC Approach", *IEEE Transactions on Power Systems*, v. 15, pp. 638–645, 2000.

[12] NASH, J. F. "Equilibrium Points in N-Person Games", *Proceedings of NAS*, v. 36, 1950.

[13] FAMPA, M., BARROSO, L. A., CANDAL, D., et al. "Bilevel Optimization applied to Strategic Pricing in Competitive Electricity Markets", *Computational Optimization and Applications*, v. 39, n. 2, pp. 121–142, 2008.

[14] LABBÉ, M., MARCOTTE, P., SAVARD, G. *On a class of bilevel programs, Nonlinear Optimization and Related Topics*. Kluwer Academic Publishers, 2000.

[15] PEREIRA, M. V., GRANVILLE, S., FAMPA, M., et al. "Strategic Bidding Under Uncertainty: A Binary Expansion Approach", *IEEE Transactions on Power Systems*, v. 20, n. 1, pp. 180–188, 2005.

[16] WANG, Y., LI, H., DANG, C. "A New Evolutionary Algorithm for a Class of Nonlinear Bilevel Programming Problems ans Its Global Convergence", *IFORMS Jornal on Computing*, v. 23, n. 4, pp. 68–629, 2011.

[17] MARINAKIS, Y., MIGDALAS, A., PARDALOS, P. M. "A New Bilevel Formulation for the Vehicle Routing Problem and a Solution Method Using a Genetic Algorithm", *Jornal Global Optimization*, v. 38, n. 4, pp. 55–580, 2007.

[18] BARD, J. "Some Properties of the Bilevel Programming", *Journal of Optimization Theory and Aplications*, v. 32, pp. 146–164, 1991.

[19] BEN-AYED, C., BLAIR, O. "Computational Difficulties of Bilevel Linear Programming", *Operations Research*, v. 38, pp. 556–560, 1990.

[20] ODUGUWA, V., ROY, R. "Bilevel Optimisation Using Genetic Algorithm", *IEEE International Conference on Artificial Intelligence Systems*, v. 25, n. 1, pp. 322–327, 2002.

[21] HUNT, S. *Making Competition Work in Electricity.* J. Wiley and Sons, 2003.

[22] KAHN, E. "Regulation by Simulation: the Role of Production Cost Models in Electricity Planning and Pricing", *Operations Research*, v. 3, n. 3, pp. 388–398, 1995.

[23] KAHN, E. "Numerical Techniques for Analyzing Market Power in Electricity", *The electricity Journal*, pp. 34–43, 1998.

[24] LINO, P., BARROSO, L. A., FAMPA, M., et al. "Bid-Based Dispatch of Hydrothermal Systems in Competitive Markets", *Annals of Operations Research*, v. 120, pp. 81–97, 2003.

[25] WANG, Y., JIAO, Y. C., LI, H. "An Evolutionary Algorithm for Solving Nonlinear Bilevel Programming Based on a New Constraint-Handling Scheme", *IEEE Transactions on Systems, Man, e Cybernetics-Part C: Applications and Reviews*, v. 35, n. 2, 2005.

[26] KUO, R. J., HAN, Y. S. "Using Hybrid Metaheuristic Approaches to Solve Bi-Level Linear Programming Problem for Supply Chain Management", *International Conference on Industrial Engineering and Engineering Management (IEEM)*, v. 25, n. 1, pp. 1154–1158, 2010.

[27] FISHER, M. L., JAIKUMAR, R. "A Generalized Assignment Heuristic for Vehicle Routing". In: *Proceedings of the International Workshop on Current and Future Directions in the Routing and Scheduling of Vehicles and Crews*, pp. 109–124, 1979.

[28] HOLLAND, J. H. *Adaptation im Natural and Artificial System.* University of Michigan Press, Ann Arbor, 1976.

[29] GOLDBERG, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison Wesley, Reading, 1989.

[30] PIMENTEL, W., FIGUEIREDO, C. M. H., CARVALHO, L. A. V. "Planejamento de Rotas Aéreas por Algoritmos Genéticos", *V Congresso Brasileiro de Redes Neurais (V CBRN)*, pp. 1–7, 2001.

[31] FAMPA, M., PIMENTEL, W. "A Genetic Algorithm to the Strategic Pricing Problem in Competitive Electricity Markets", *Proceedings of SBPO/CLAIO*, pp. 24–28, 2012.

[32] FAMPA, M., PIMENTEL, W. "An application of genetic algorithm to a bidding problem in electricity markets", *International Transations in Operational Research*, v. 22, n. 1, pp. 97–111, 2014.

[33] DOWDY, S., WEARDEN, S., CHILKO, D. *Statistics for Research*. Wiley, 2004.

[34] COSTA, W. E. *Approachs GRASP, VNS and PSO to Minimize the Total Flowtime Criterion in Flowshop Scheduling*. Tese de D.Sc., UFRN, Rio Grande do Norte, RN, Brasil, 2012.

[35] DONG, X., HUANG, H., CHEN, P. "An Iterated Local Search Algorithm for the Permutation Flowshop Problem with total Flowtime Criterion", *Computers and Operations Research*, v. 36, n. 5, pp. 1664–1669, 2009.

[36] GAY, D. M. *User's Manual for CPLEX*. IBM ILOG CPLEX v12.5, 2009.

[37] GROSS, G., FINLAY, D. "Generation Supply Bidding in Perfectly Competitive Electricity Markets", *Comput. Math. Org. Theory*, v. 6, pp. 83–98, 2000.

[38] ANSTREICHER, K. M. "Semidefinite Programming Versus the Reformulation-Linearization Technique for Nonconvex Quadratically Constrained Quadratic Programming", *Journal of Global Optimization*, v. 43, n. 23, pp. 471–484, 2009.

[39] BURER, S., VANDENBUSSCHE, D. "A Finite Branch-and-Bound Algorithm for Nonconvex Quadratic Programming via Semidefinite Relaxations", *Mathematical Programming*, v. 113, n. 2, pp. 259–282, 2008.

[40] FAMPA, M., MELO, W., MACULAN, N. "Semidefinite Relaxation for Linear Programs with Equilibrium Constraints", *International Transactions in Operational Research*, v. 20, pp. 201–212, 2013.

[41] SAXENA, A., BONAMI, P., LEE, J. "Convex Relaxations of Non-Convex Mixed Integer Quadratically Constrained Programs: Extended Formulations", *Mathematical Programming, Series B*, v. 124, n. 1-2, pp. 383–411, 2010.

[42] SAXENA, A., BONAMI, P., LEE, J. "Convex Relaxations of Non-Convex Mixed Integer Quadratically Constrained Programs: Projected Formulations", *Mathematical Programming, Series A*, v. 130, n. 2, pp. 359–413, 2011.

[43] BAO, X., SAHINIDIS, N., TAWARMALANI, M. "Semidefinite Relaxations for Quadratically Constrained Quadratic Programming: A Review and Comparisons", *Mathematical Programming*, v. 129, n. 1, pp. 129–157, 2011.

[44] LOVÁSZ, L. "On the Shannon Capacity of a Graph", *IEEE Transactions on Information Theory*, v. 25, n. 1, pp. 1–7, 1979.

[45] LOVÁSZ, L., SCHRIJVER, A. "Cones of matrices and set-functions and 0-1 optimization", *SIAM Journal on Optimization*, v. 1, pp. 166–190, 1991.

[46] GOEMANS, M. X., WILLIAMSON, D. P. "Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming", *Journal of ACM*, v. 42, n. 6, pp. 1115–1145, 1995.

[47] RENDL, F., RINALDI, G., WIEGELE, A. "Solving Max-Cut to Optimality by Intersecting Semidefinite and Polyhedral Relaxations", *Mathematical Programming, Series A*, v. 121, n. 2, pp. 307–335, 2010.

[48] SHERALI, H. D., TUNCBILEK, C. H. "A reformulation-convexification approach for solving nonconvex quadratic programming problems", *Journal of Global Optimization*, v. 7, n. 1, pp. 1–31, 1995.

[49] MCCORMICK, G. P. "Computability of Global Solutions to Factorable Nonconvex Programs: part I Convex Underestimating Problems", *Mathematical Programming*, v. 10, n. 1, pp. 147–175, 1976.

[50] SHERALI, H., ADAMS, W. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems.* Dordrecht, Kluwer Academic Publishers, 1999.

[51] FAMPA, M., PIMENTEL, W. "SDP relaxation for a strategic pricing bilevel problem in electricity markets", *Proceedings of SBPO/CLAIO*, 2013.

[52] BORCHERS, B. "CSDP, a C Library for Semidefinite Programming", *Optimization Methods & Software*, v. 11, n. 2, pp. 613–623, 1999.

[53] MARGOT, M., BELOTTI, P., QUALIZZA, A. "Linear Programming Relaxations of Quadratically Constrained Quadratic Programs", *Mathematics and its Applications*, v. 154, 2012.

[54] GOLUB, G. H., LOAN'S, C. F. V. *Matrix Computations.* JHU Press, 2013.

[55] FAMPA, M., PIMENTEL, W. "A cutting plane algorithm for bounding a strategic pricing problem in electricity markets", *Proceedings of SBPO/CLAIO*, 2014.

[56] TOSO, R. F., RESENDE, M. G. C. *A C++ application programming interface for bi- ased random-key genetic algorithms.* AT&T Labs Research, Florham Park, New Jersey, 2012.