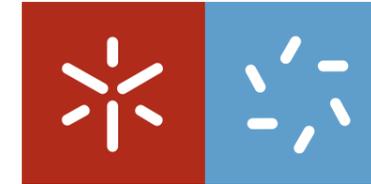


Universidade do Minho
Escola de Ciências

Helena Sofia Ferreira Rodrigues

**Problema de Optimização com
Restrições de Complementaridade:
uma aplicação ao mercado de energia eléctrica**



Universidade do Minho
Escola de Ciências

Helena Sofia Ferreira Rodrigues

**Problema de Optimização com
Restrições de Complementaridade:
uma aplicação ao mercado de energia eléctrica**

Tese de Mestrado em
Matemática Computacional

Trabalho efectuado sob orientação da
Doutora Maria Teresa Torres Monteiro

Outubro, 2005

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA TESE APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE;

Helena Sofia Ferreira Rodrigues

Agradecimentos

Para a elaboração deste trabalho, várias foram as pessoas que colaboraram, quer em termos científicos quer em termos pessoais.

O meu maior agradecimento vai para a minha orientadora - professora Teresa Monteiro - que, com a sua amizade e incansável disponibilidade, tornou este projecto possível. A sua energia e entusiasmo foram para mim uma fonte de inspiração e motivação.

Gostaria de agradecer ao Ismael Vaz, pela ajuda, sempre presente, nos problemas informáticos que me pareciam intransponíveis.

Não poderia deixar de agradecer ao Jorge e ao Diogo, por todas as suas colaborações relativamente aos conceitos de electricidade.

À Sandra, o meu muito obrigado pelas suas contribuições anglo-saxónicas.

Ao Vitor e a todos os meus amigos e familiares, que sempre foram acompanhando o meu trabalho e que, pacientemente, souberam entender todas as minhas desculpas por não passar mais tempo com eles.

Aos meus pais e irmão, a quem dedico este trabalho, por me encorajarem na realização dos meus sonhos.

A todos vós, o meu muito obrigada!

Resumo

Este trabalho tem dois objectivos - a análise do Problema de Optimização com Restrições de Complementaridade e a sua aplicação a um caso real.

O problema, conhecido na literatura por MPCC (*Mathematical Program with Complementarity Constraints*), insere-se na área da Optimização Não Linear e apresenta, para além das restrições não lineares, restrições de complementaridade. O conceito de complementaridade está relacionado com a noção de equilíbrio, daí a sua aplicabilidade na Engenharia e na Economia. É um problema de difícil resolução porque as usuais qualificações das restrições, necessárias para garantir a convergência dos algoritmos, falham em todos os pontos admissíveis. Têm sido desenvolvidos algoritmos específicos para a sua resolução mas necessitam de condições muito fortes para garantir eventual convergência. Estudos recentes provaram que a estacionaridade forte dum problema MPCC é equivalente às condições de optimalidade de primeira ordem do problema NLP equivalente (*NonLinear Program*). Com base neste resultado, uma reformulação NLP é usada na qual as restrições de complementaridade são substituídas por restrições não lineares de desigualdade equivalentes, podendo o problema MPCC ser resolvido por *solvers* NLP. Um conjunto de noventa e dois problemas codificados em AMPL é resolvido, na sua reformulação NLP equivalente, por quatro pacotes de *software* - LOQO, MATLAB, NPSOL e Filter SQP. É feita uma análise comparativa dos resultados.

O segundo objectivo deste trabalho consiste na formulação e codificação em AMPL de uma aplicação real, bem como a sua resolução usando alguns *solvers*. É estudado um problema de média dimensão relacionado com o mercado oligopolista de energia eléctrica, modelado como um jogo de Stackelberg.

Abstract

This work has two goals - the Mathematical Program with Complementarity Constraints analysis and its application to a real case.

The problem, MPCC, is part of the Nonlinear Optimization and presents nonlinear constraints as well as complementarity constraints. The complementarity concept is related to the equilibrium notion and so the MPCC is very common in Engineering and Economics areas. This problem is very difficult to solve because the usual constraints qualifications, necessary to guarantee the algorithms convergence, fail in all feasible points. Some specific algorithms to solve MPCC have been developed but they need strong assumptions to converge. Recent studies have proved that the strong stationarity of an MPCC is equivalent to the first order optimality conditions of the nonlinear program (NLP) equivalent. Using this result, an NLP reformulation is used in which the complementarity constraints are replaced by the equivalent nonlinear inequality constraints in such a way that the MPCC can be solved by NLP codes. A set of ninety two AMPL problems is solved, in its equivalent NLP reformulation, by four software packages - LOQO, MATLAB, NPSOL and Filter SQP. A results comparative analysis is made.

The second purpose of this work is the formulation and the AMPL codification of a real application as well as its resolution by NLP solvers. A medium scale problem related to electric energy oligopolist market, modelled as a Stackelberg game, is studied .

Conteúdo

Agradecimentos	i
Resumo	iii
Abstract	v
Índice de Tabelas	xi
Índice de Figuras	xiii
I Problema de Optimização com Restrições de Complementari- dade	1
1 Introdução	3
1.1 A complementaridade	4
1.2 Aplicações	5
1.3 Estrutura da tese	7
2 Definição do problema	9
2.1 Problema de optimização não linear	9
2.2 Formulação do MPCC	13
3 Técnicas de resolução	17
3.1 Estado da arte	17

3.2	Reformulação NLP	19
3.3	Estacionaridade forte do problema MPCC	20
4	<i>Solvers</i> NLP	23
5	Experiências computacionais	27
5.1	Detalhes de implementação	27
5.2	Análise de resultados	29
6	Conclusões e trabalho futuro	35
6.1	Conclusões	35
6.2	Trabalho Futuro	36
II	Aplicação ao mercado de energia eléctrica	39
1	Introdução	41
1.1	Motivação	41
1.2	Mercados de energia eléctrica	42
2	O problema	45
2.1	Optimização de dois níveis e o jogo de Stackelberg	45
2.2	Formulação do problema	46
2.2.1	Cenário de mercado com uma empresa líder	46
2.2.2	Formulação matemática	48
2.2.3	Dados do problema	53
3	Análise de resultados	55
	Bibliografia	59
	Apêndices	63

CONTEÚDO

ix

A Classificação dos problemas teste

65

B Análise detalhada dos resultados

67

C Dados do problema

71

D Código do problema

75

Índice de Tabelas

5.1	Número de falhas ocorridas por <i>solver</i>	30
5.2	Estatísticas relativas ao desempenho dos <i>solvers</i>	33
3.1	Principais características dos <i>solvers</i>	56
3.2	Resultados da função objectivo e variável do primeiro nível	56
C.1	Dados relativos aos nós de oferta (produtores)	71
C.2	Dados relativos aos nós de procura (consumidores)	72
C.3	Dados relativos aos arcos	73
C.4	Dados relativos aos ciclos considerados	73

Índice de Figuras

1.1	Contextualização do MPCC	4
2.1	Conjunto admissível para as variáveis de estado de um MPCC	13
5.1	Esquema de implementação dos <i>solvers</i>	28
5.2	Esquema de implementação do MATLAB	28
5.3	Perfis de desempenho relativo ao tempo de CPU	31
5.4	Perfis de desempenho relativo ao número de iterações	32
5.5	Número de iterações usadas pelos <i>solvers</i>	34
5.6	Cálculo de funções	34
5.7	Cálculo de gradientes	34
2.1	Curva da oferta (custo marginal) e curva da proposta	47
2.2	Esquema da rede de energia eléctrica	53

Parte I

Problema de Optimizaç o com Restriç es de Complementaridade

Capítulo 1

Introdução

Os Problemas de Optimização com Restrições de Complementaridade têm sido alvo de grande interesse, pois na sua formulação está subjacente a noção de equilíbrio - conceito presente em inúmeras situações da realidade.

Neste capítulo, é feita uma breve contextualização deste problema na Optimização, um enquadramento histórico do conceito de complementaridade e uma descrição sumária das possíveis áreas de aplicação. Por fim, apresenta-se a estrutura da tese.

O Problema de Optimização com Restrições de Complementaridade vai ser designado ao longo deste trabalho pela sigla MPCC (*Mathematical Program with Complementarity Constraints*). Este problema está enquadrado numa área mais ampla cuja importância é reconhecida na comunidade científica - a Optimização Não Linear. Dos vários ramos subjacentes, destaca-se o Problema de Optimização com Restrições de Equilíbrio (MPEC, *Mathematical Program with Equilibrium Constraints*). Este pode ser enunciado de duas formas distintas - consoante as suas restrições - originando, deste modo, dois tipos de problemas: Desigualdade Variacional ou Problema de Optimização com Restrições de Complementaridade. É sobre este último que o estudo desta dissertação se vai centrar. A Figura 1.1 esquematiza a interligação entre os vários conceitos acima referidos.



Figura 1.1: Contextualização do MPCC

1.1 A complementaridade

O significado da complementaridade é muito anterior à sua utilização na ciência, uma vez que surge naturalmente na realidade, pois está intrinsecamente relacionado com o conceito de equilíbrio.

Na otimização, o conceito de complementaridade foi introduzido por volta dos anos 30 [38]. Nessa altura os matemáticos foram confrontados com um problema que ainda hoje desempenha um papel muito importante na área da economia - o jogo de Stackelberg [63].

Em 1939, Karush na sua tese de mestrado não publicada e, mais tarde em 1951, Kuhn e Tucker [40] num trabalho totalmente independente apresentam as condições de optimalidade de primeira ordem - as conhecidas KKT - onde a complementaridade também está presente.

Nas últimas duas décadas, a colecção de problemas modelados como MPCC provenientes da economia e da engenharia aumentou consideravelmente. Segundo Ferris [20], vários foram os factores que contribuíram para este aumento, nomeadamente o desenvolvimento da capacidade dos computadores para dar resposta aos problemas propostos, quer em termos de rapidez, quer em termos de memória. Os melhoramentos consideráveis nos algoritmos e *software* que exploram os avanços significativos em técnicas de álgebra linear e otimização não linear, permitiram a resolução de problemas de grande dimensão e com um elevado nível de dificuldade. O aparecimento de sistemas modelares capazes de expressar

a condição de complementaridade como parte integrante da sintaxe do *solver* facilitaram o tratamento do problema MPCC. Também o crescente número de estudos apresentados acerca da existência, unicidade e computabilidade das suas soluções, foram contributos para o crescimento do número de problemas formulados como MPCC.

Hoje, resolver um MPCC, comparativamente com um típico problema de optimização não linear, é ainda uma tarefa bastante árdua. Porém, a sua extensiva aplicabilidade e estreita relação com outros ramos de pesquisa, são motivos para os investigadores continuarem a apostar neste tema.

1.2 Aplicações

Na optimização, o problema MPCC pode ter a sua origem na aplicação das condições de optimalidade KKT ou ainda ser o resultado, sob certas condições, do tratamento de um problema de optimização de dois níveis (*bilevel*).

Nalguns dos ramos da ciência, como a Engenharia, a Economia e a Ecologia, têm surgido situações reais passíveis de serem formuladas como MPCC. Muitas dessas aplicações estão enunciadas em [10, 21, 46]. Nesta secção serão apenas descritos sucintamente alguns exemplos, remetendo informações mais detalhadas para referências seleccionadas.

Na Engenharia, os MPCC têm sido utilizados para dar resposta a:

- problemas de mecânica de contacto, onde existe complementaridade entre a força e a distância entre dois corpos [23, 52]; a robótica é um dos subtemas onde mais se tem destacado o seu uso, nomeadamente no estudo de certos modelos de planeamento de movimento para as mãos de um robot encarregue de tarefas que envolvam contactos friccionais [50];
- problemas de obstáculo que consistem em encontrar uma posição de equilíbrio de uma membrana elástica sujeita a uma força vertical [55];
- problemas de lubrificação elastohidrodinâmica que estão relacionados com os efeitos

da deformação elástica e viscosidade de pressão sensitiva nas características operacionais de superfícies lubrificadas [22, 49];

- problemas na indústria com processos químicos onde se requer soluções para processos correntes de síntese, *design*, otimização e actividades de controlo; além disso, os processos químicos envolvem sistemas que são governados pelo equilíbrio químico [54];
- problemas de tráfego de rede que dizem respeito à análise do congestionamento do tráfego, ao transporte da informação na rede através de um conjunto de nós e um conjunto de arcos e à modificação de uma infraestrutura de transporte [48, 64].

Os MPCC ajudam a solucionar problemas na área da Economia tais como:

- teoria do equilíbrio geral onde, por um lado, o produtor tenta encontrar sectores de actividade e preços de mercadorias que lhe possibilite a obtenção do lucro máximo; por outro, está o consumidor que pretende adquirir bens a um preço mínimo de forma a satisfazer as suas necessidades. Entre os casos estudados destacam-se o equilíbrio de Walsarian, relacionado com o preço do bem e o excesso de oferta desse mesmo bem, e o modelo de equilíbrio do preço espacial (EPE) que descreve uma estrutura de mercado espacial e competitiva onde cada produtor está espacialmente separado dos outros, fazendo com que numa área circundante o preço de entrega do bem lhe seja mais compensatório [5, 11, 12, 62, 65];
- modelos da Teoria de Jogo onde os modelos económicos mais conhecidos são os jogos não cooperativos de Nash e de Stackelberg. O equilíbrio de Nash é um modelo de produção com m jogadores, cada qual com a sua estratégia e onde nenhum consegue aumentar os seus lucros através de modificações unilaterais ao nível de produção; cada um observa as acções dos restantes e, em seguida, reage optimamente, assumindo que as estratégias dos outros jogadores permanecem inalteradas. O equilíbrio de Nash acontece quando houver uma combinação de estratégias onde nenhum jogador tem um incentivo para mudar a sua acção. No modelo de Stakelberg existe um

jogador distinto - chamado líder - que antecipa as (re)acções dos restantes jogadores - denominados seguidores - e usa esse conhecimento para seleccionar a sua estratégia óptima; aqui a estratégia do seguidor é dependente da estratégia particular do líder; colectivamente, os seguidores comportam-se entre si como um jogo não cooperativo de Nash [18, 51, 53, 60, 66]. A optimização de dois níveis está intrinsecamente relacionada com o modelo de Stackelberg [46];

- modelação da competição nos mercados de energia eléctrica reestruturados; a electricidade gerada começa a ser uma actividade liberalizada onde as empresas assumem mais riscos, mas simultaneamente ficam com maior poder de decisão; uma vez que o preço já não se encontra tabelado, o objectivo deixa de ser a minimização dos custos passando a ser a maximização dos lucros [16, 17, 41, 61, 68].

No âmbito da Ecologia, as questões relacionadas com a emissão de gases que provocam o efeito de estufa também podem ser tratadas através da formulação MPCC. Com o Protocolo de Quioto e as recentes cimeiras internacionais relativas às alterações climáticas no globo terrestre, os países começam-se a aperceber que é necessário fazer um esforço concertado para reduzir a emissão de gases nocivos para o meio ambiente. Assim, estão a ser elaborados alianças e acordos comerciais entre países para negociarem entre si direitos de emissão. Os países desenvolvidos, uma vez que na sua generalidade ultrapassam já as suas quotas de emissão, vão tentando diminuir através de políticas ambientais e simultaneamente vão produzindo acordos comerciais com os países em desenvolvimento com o objectivo de negociar os direitos de emissão não utilizados por estes [8].

A procura de novas técnicas e algoritmos para resolver potencialmente problemas reais de grande dimensão, continua a ser uma área com intensa investigação.

1.3 Estrutura da tese

A tese está estruturada em duas partes. A Parte I, relativa ao primeiro objectivo deste trabalho, é constituída por seis capítulos. A Parte II, o estudo de uma aplicação real,

apenas contém três capítulos. Existem ainda quatro apêndices.

No Capítulo 2 é apresentada a formulação matemática de um problema de otimização não linear (NLP, *NonLinear Program*). São definidos alguns conceitos básicos de otimização e é apresentada a formulação matemática do problema MPCC. São ainda descritas algumas características pelas quais o MPCC é considerado, na otimização, um problema de difícil resolução.

O estudo de estratégias recentes para a resolução de um MPCC é descrito no Capítulo 3. Até ao presente momento, o desenvolvimento de algoritmos específicos ainda não produziu resultados satisfatórios em termos de rapidez e de robustez. São enunciados os resultados de estudos recentes que relacionam a estacionaridade forte do problema MPCC com as condições KKT do problema NLP. Baseado nesse resultado é apresentada uma versão equivalente NLP para resolver o problema MPCC.

No Capítulo 4 são descritos os *solvers* NLP utilizados na resolução de um conjunto de problemas existentes na literatura da especialidade. As experiências computacionais relativas à resolução de problemas MPCC na versão NLP equivalente, são analisadas no Capítulo 5. Os resultados mais significativos relativamente à eficiência e robustez dos *solvers* NLP são apresentados através de tabelas e gráficos. Os resultados mais detalhados assim como a classificação de cada problema utilizado, foram remetidos para os Apêndices A e B.

No Capítulo 6 são apresentadas as conclusões finais e apontadas algumas propostas para trabalho futuro.

A Parte II diz respeito ao estudo de um problema de um mercado oligopolista de energia eléctrica. No Capítulo 1, começa-se por explicar a razão pela qual foi escolhido este problema, mostrando a sua importância no panorama energético actual.

A sua formulação é apresentada no Capítulo 2. Inicia-se com uma exposição da relação entre o jogo de Stackelberg e a otimização de dois níveis e, de seguida, apresenta-se o problema sob um cenário da empresa líder de mercado. Os dados e a codificação em AMPL do problema encontram-se nos Apêndices C e D. No Capítulo 3, são apresentados e analisados os resultados obtidos através dos vários *solvers* usados na plataforma NEOS.

Capítulo 2

Definição do problema

Neste capítulo começa-se por apresentar a formulação geral de um problema de optimização não linear NLP bem como as condições de optimalidade de primeira e segunda ordem. Segue-se a definição do problema de optimização com restrições de complementaridade MPCC. São ainda descritas algumas das dificuldades na resolução deste tipo de problemas, nomeadamente a não satisfação das condições de optimalidade de primeira ordem e as consequências da sua violação.

2.1 Problema de optimização não linear

Para contextualizar o problema MPCC na área da optimização não linear vão definir-se algumas entidades e conceitos que serão necessários para o estudo deste problema. Estando inserido na Optimização Não Linear, torna-se pertinente a apresentação da formulação geral do problema de optimização não linear NLP.

Definição 2.1 *Problema NLP*

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.a.} \quad & c_i(x) = 0, i \in E \\ & c_i(x) \geq 0, i \in I \end{aligned} \tag{NLP}$$

em que f é a função objectivo ($f : \mathbb{R}^n \rightarrow \mathbb{R}$), $c_i, i \in E$ e $c_i, i \in I$ são o conjunto de restrições do tipo igualdade e desigualdade, respectivamente ($c_i : \mathbb{R}^n \rightarrow \mathbb{R}, i \in E \cup I$).

Os conjuntos E e I são conjuntos finitos de índices e as funções f e $c_i, i \in E \cup I$ são continuamente diferenciáveis até à segunda ordem.

Começa-se por definir a função Lagrangeana.

Definição 2.2 Função Lagrangeana

A função Lagrangeana do problema (NLP) é definida por:

$$\mathcal{L}(x, \lambda) = f(x) - \sum_{i \in E \cup I} \lambda_i c_i(x)$$

em que λ é o vector dos multiplicadores de Lagrange.

Em seguida apresentam-se as definições de região admissível e conjunto de restrições activas:

Definição 2.3 Região admissível

A região admissível Ω é o conjunto de pontos x que satisfaz todas as restrições, i.e.,

$$\Omega = \{x \mid c_i(x) = 0, i \in E; c_i(x) \geq 0, i \in I\}$$

Pode então reescrever-se o problema (NLP) como

$$\min_{x \in \Omega} f(x)$$

Definição 2.4 Conjunto de restrições activas

Sejam E e I os conjuntos de índices das restrições de igualdade e de desigualdade, respectivamente. Designa-se por conjunto das restrições activas o conjunto:

$$A(x) = \{c_i(x) \mid i \in E\} \cup \{c_i(x) \mid c_i(x) = 0 \wedge i \in I\} \quad (2.1)$$

É usual estabelecerem-se hipóteses, designadas por qualificações das restrições (*constraints qualification*), para garantir que não ocorrerá nenhum comportamento degenerado para o valor de x^* , sendo este a solução do problema em questão. Uma delas, talvez a mais usada na concepção de algoritmos, é a LICQ (*Linear Independence Constraint Qualification*).

Definição 2.5 LICQ

Dado um ponto x^* e um conjunto activo $A(x^*)$, a qualificação das restrições de independência linear é satisfeita se o conjunto dos gradientes das restrições activas

$$\{\nabla c_i(x^*), c_i \in A(x^*)\}$$

é linearmente independente.

Um ponto x que verifica a LICQ diz-se ponto regular, *i.e.*, a matriz $\{\nabla c_i(x), c_i \in A(x)\}$ tem característica completa (*full rank*). Note-se que se esta qualificação das restrições é satisfeita, nenhum dos gradientes das restrições activas pode ser zero.

Um generalização da LICQ, também muito utilizada nos algoritmos, é a qualificação das restrições MFCQ (*Mangasarian Fromovitz Constraint Qualification*):

Definição 2.6 MFCQ

Dados um ponto x^* e um conjunto activo $A(x^*)$, a qualificação das restrições de Mangasarian Fromovitz é satisfeita se existe um vector $w \in \mathbb{R}^n$ tal que:

$$\begin{aligned} \nabla c_i(x^*)^T w &> 0, \text{ para todo } c_i \in A(x^*) \wedge i \in I \\ \nabla c_i(x^*)^T w &= 0, \text{ para todo } i \in E \\ \nabla c_i(x^*), i \in E &\text{ são linearmente independentes} \end{aligned} \tag{2.2}$$

Note-se que a MFCQ é uma condição mais fraca do que a LICQ. Considere-se o seguinte lema que relaciona as qualificações das restrições LICQ e MFCQ:

Lema 2.1 *Seja x^* um ponto que satisfaz a LICQ. Então x^* também satisfaz a MFCQ.*

Demonstração.

Por hipótese, um ponto óptimo x^* satisfaz a LICQ. Então os gradientes das restrições activas $\{\nabla c_i(x^*), c_i \in A(x^*)\}$, são linearmente independentes, logo a terceira condição de (2.2) é imediatamente satisfeita. Uma vez que $\nabla c_i(x^*)$ representa uma matriz que tem

característica completa, *i.e.*, o ponto x^* é regular, então existe uma solução w para o sistema definido por:

$$\begin{cases} \nabla c_i(x^*)^T w = 1, & c_i \in A(x^*) \wedge i \in I \\ \nabla c_i(x^*)^T w = 0, & i \in E \end{cases}$$

Logo, pode-se considerar esta solução w para satisfazer as duas primeiras condições de (2.2). Desta forma, x^* satisfaz MFCQ. ■

Em seguida apresentam-se as condições de optimalidade para o problema (NLP). São designadas de primeira ordem porque dizem respeito às propriedades dos gradientes (primeiras derivadas) da função objectivo e das funções das restrições. Estas condições são conhecidas pelas condições KKT.

Definição 2.7 *Condições de primeira ordem*

Seja x^* é uma solução local de um problema de optimização não linear e ponto regular. Então existe um vector dos multiplicadores de Lagrange λ^* , com componentes λ_i^* , $i \in E \cup I$, tal que as seguintes condições são satisfeitas em (x^*, λ^*) :

$$\begin{aligned} \nabla_x \mathcal{L}(x^*, \lambda^*) &= \nabla f(x^*) - \sum_{i \in E \cup I} \nabla c_i(x^*) \lambda_i^* = 0 \\ c_i(x^*) &= 0, \quad i \in E \\ c_i(x^*) &\geq 0, \quad i \in I \\ \lambda_i^* &\geq 0, \quad i \in I \\ \lambda_i^* c_i(x^*) &= 0, \quad i \in E \cup I \end{aligned} \tag{2.3}$$

O ponto x^* que verifica as condições de primeira ordem diz-se ponto estacionário da Lagrangeana e o par (x^*, λ^*) que as verifica denomina-se par KKT.

As condições seguintes, de segunda ordem, examinam as segundas derivadas e dizem respeito à curvatura da função Lagrangeana.

Definição 2.8 *Condições suficientes de segunda ordem*

O ponto x^* , do par KKT (x^*, λ^*) , é um mínimo local se a matriz Hessiana da função Lagrangeana $\nabla_{xx}^2 \mathcal{L}(x^*, \lambda^*)$ for definida positiva no espaço nulo da matriz $\nabla c(x^*)^T$.

2.2 Formulação do MPCC

A presença de restrições de complementaridade é a característica que distingue um MPCC de um problema de otimização não linear - para além das restrições não lineares existentes num problema NLP, possui ainda restrições envolvendo o operador de complementaridade.

Definição 2.9 Problema MPCC

$$\begin{aligned}
 \min \quad & f(x) \\
 \text{s.a.} \quad & c_i(x) = 0, i \in E \\
 & c_i(x) \geq 0, i \in I \\
 & 0 \leq x_1 \perp x_2 \geq 0
 \end{aligned} \tag{MPCC}$$

onde $x = (x_0, x_1, x_2)$, com a variável de controlo $x_0 \in \mathbb{R}^n$ e as variáveis de estado $x_1, x_2 \in \mathbb{R}^p$; f é a função objectivo, $c_i, i \in E \cup I$ são os conjuntos das restrições de igualdade e desigualdade, respectivamente. Os conjuntos E e I são conjuntos finitos de índices. As restrições que envolvem a complementaridade são definidas com o operador \perp e exigem que o produto de duas quantidades não negativas seja zero, i.e., $x_{1i}x_{2i} = 0, i \in \{1..p\}$.

Tanto f como $c_i, i \in E \cup I$ são funções continuamente diferenciáveis até à segunda ordem. A Figura 2.1 representa o conjunto admissível para as variáveis de estado.

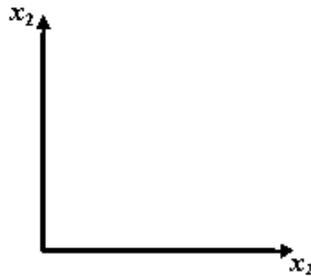


Figura 2.1: Conjunto admissível para as variáveis de estado de um MPCC

Num problema MPCC nem todas as desigualdades são estritamente satisfeitas em qualquer ponto admissível. Apresenta ainda a característica de ser não suave e, devido à sua natureza combinatoria relacionada com as restrições de complementaridade faz com que condições de optimalidade sejam complexas e difíceis de verificar.

Uma primeira abordagem para resolver um MPCC seria utilizar os típicos algoritmos não lineares existentes. Contudo, o comportamento da maioria destes algoritmos perto da solução é muito influenciado pela estrutura do conjunto admissível do problema [53]. Se forem satisfeitas certas qualificações das restrições, os algoritmos NLP comuns garantem a convergência. Infelizmente, o conjunto admissível do problema MPCC, é mal condicionado, uma vez que as qualificações das restrições mais conhecidas e utilizadas para provar a convergência dos algoritmos NLP não são satisfeitas em nenhum ponto que satisfaça as restrições de complementaridade [56, 58].

Considere-se o seguinte lema:

Lema 2.2 *Um problema MPCC viola a qualificação das restrições MFCQ em qualquer ponto admissível.*

Demonstração.

Considere-se $x = (x_0, x_1, x_2) \in \mathbb{R}^3$ (para $x \in \mathbb{R}^l$, para $l > 3$ a demonstração é análoga).

Considerando o problema (MPCC) pode definir-se $\mathcal{E} = \{c_i(x), i \in E; x_1 x_2 = 0\}$, o conjunto das restrições de igualdade, e $\mathcal{I} = \{c_i(x), i \in I; x_1 \geq 0; x_2 \geq 0\}$ o conjunto das restrições de desigualdade.

Por redução ao absurdo, suponha-se que o MPCC satisfaz a condição MFCQ num ponto ótimo x^* . Então existe um conjunto activo $\mathcal{A}(x^*)$ e existe um vector $w = (w_0, w_1, w_2) \in \mathbb{R}^3$ que satisfaz as condições de (2.2).

Pela terceira condição de MFCQ tem-se que $\nabla c_i(x^*) \neq 0, c_i(x) \in \mathcal{E}$. Em particular, para o caso da restrição de complementaridade tem-se que:

$$\begin{bmatrix} 0 \\ x_2 \\ x_1 \end{bmatrix} \neq 0 \implies x_1 \neq 0 \quad \text{ou} \quad x_2 \neq 0. \quad (2.4)$$

Existem então três casos a analisar:

Caso 1: $x_1 \neq 0$ e $x_2 = 0$

Da segunda condição de MFCQ e substituindo x_2 por 0, vem:

$$\begin{bmatrix} 0 & 0 & x_1 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} = 0 \implies w_2 = 0 \quad (2.5)$$

Por outro lado, da primeira condição de MFCQ e uma vez que $x_2 = 0$ é uma restrição activa do conjunto $\mathcal{A}(x^*)$ vem:

$$\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} > 0 \implies w_2 > 0 \quad (2.6)$$

Este resultado é absurdo pois (2.6) contradiz (2.5).

Caso 2: $x_1 = 0$ e $x_2 \neq 0$

Análise semelhante à do caso 1.

Caso 3: $x_1 \neq 0$ e $x_2 \neq 0$

Esta situação nunca pode acontecer, na medida em que deixa de ser um problema MPCC, pois deste modo não se verifica a restrição de complementaridade.

Conclusão: um MPCC viola sempre a qualificação das restrições MFCQ. ■

A restrição de complementaridade está na origem da falha de MFCQ. O facto do problema MPCC violar a qualificação das restrições MFQC implica os seguintes inconvenientes:

- o conjunto dos multiplicadores é ilimitado;
- não existe caminho central;
- os gradientes das restrições activas são linearmente dependentes;
- a linearização das restrições pode ser arbitrariamente inconsistente perto da solução.

Como consequência, a maioria da teoria desenvolvida para a programação não linear não pode ser aplicada directamente ao problema MPCC. É mostrado que os algoritmos SQP (*Sequential Quadratic Programming*) convergem quadraticamente para um multiplicador

básico, desde que todos os subproblemas QP se mantenham consistentes [38]; porém, o facto de não satisfazer a MFCQ faz com que qualquer subproblema QP se possa tornar arbitrariamente inconsistente perto de um ponto estacionário.

Finalmente, mesmo que um algoritmo NLP, aplicado a um problema MPCC, produza um ponto admissível, neste momento nada se sabe acerca das suas propriedades de estacionaridade, uma vez que as análises de convergência dos algoritmos pressupõem que a condição de Mangasarian Fromovitz seja satisfeita nos pontos limite [36]. Esta característica funciona como motivação para o desenvolvimento de algoritmos específicos para resolver problemas MPCC.

No capítulo seguinte é apresentada a abordagem utilizada para resolver o problema MPCC, bem como os fundamentos teóricos que lhe estão subjacentes.

Capítulo 3

Técnicas de resolução

Criar um *solver* para resolver com fiabilidade um MPCC é, hoje em dia, uma área de intensa investigação. Neste capítulo faz-se uma síntese dos algoritmos específicos que têm surgido na literatura para resolução do problema MPCC. Contudo, a maioria dos trabalhos encontra-se ainda num estado prematuro de desenvolvimento, necessitando de condições fortes para assegurar a convergência dos métodos, comprometendo a sua robustez. Reformulou-se o problema MPCC num problema NLP equivalente. Estudos recentes estabeleceram uma relação entre estacionaridade forte do problema MPCC e as condições KKT do problema NLP equivalente.

3.1 Estado da arte

Na última década surgiram novas formulações das relações de complementaridade com vista ao desenvolvimento de *solvers* específicos para resolver problemas MPCC. De entre os algoritmos propostos recentemente, destacam-se três abordagens: penalidade, suavização e relaxação. Todas estas técnicas podem conjugar-se com vários métodos de optimização existentes, dando origem a novos algoritmos.

Em meados da década de 90, Luo, Pang e Ralph [46] apresentaram um novo código: PIPA - *Penalty Interior Point Algorithm*. O objectivo era substituir a restrição de complementaridade pelo produto de Hadamard e traçar o caminho interior do problema agora parametrizado. Mas, os resultados não foram muito animadores: foi recentemente descoberto por Leyffer [44] que o PIPA pode convergir, em certas circunstâncias, para pontos não estacionários.

Liu e Sun [45] sugerem um outro método de pontos interiores onde as restrições de complementaridade são relaxadas para desigualdades através de um parâmetro de relaxação, enquanto que Demiguel *et al* [9] propõem um esquema de relaxação que preserva a admissibilidade estrita não vazia de cada subproblema. Todos estes *solvers* estão ainda numa fase embrionária a realizar testes com problemas académicos.

Facchinei, Jiang e Qi [19] sugerem que a condição de complementaridade seja reformulada para uma condição semi-suave, sendo esta de seguida suavizada com um parâmetro positivo μ . Nesta abordagem, apenas se obtêm pontos KKT com C-estacionaridade.

Por seu turno Fukushima, Luo e Pang [28] sugerem um algoritmo cuja suavização é feita com a função de Fisher-Burmeister, combinada com o clássico método da função de penalidade. Este trabalho, além de obrigar a extensos cálculos devido à falta de informação das condições de segunda ordem, ainda está apenas a ser testado para problemas de optimização com restrições de complementaridade lineares. No seguimento deste trabalho, Jiang e Ralph [37] introduzem uma classe de funções, parametrizadas por escalares reais, para transformar os problemas não suaves em problemas não lineares suaves. Este procedimento de suavização produz benefícios uma vez que melhora a perspectiva de admissibilidade e estabilidade das restrições dos problemas não lineares associados.

Os algoritmos SQP aliados à técnica de relaxação, têm sido aplicados com algum sucesso à resolução do problema MPCC - estudos concluíram que os *solvers* SQP têm convergência superlinear perto de um ponto estacionário forte em determinadas condições. Anitescu [3] mostrou a importância do modo elástico nos códigos SQP, que consiste em modificar o NLP por relaxamento das restrições e adicionar um termo de penalidade à função objectivo.

Uma heurística para lidar com a inconsistência QP próxima dos pontos admissíveis é relaxar a linearização das restrições de complementaridade, como foi adoptada no FilterMPEC proposto por Fletcher e Leyffer [25]. O método SQP usado é um algoritmo que envolve regiões de confiança e usa uma técnica de filtros para promover a convergência global.

Apesar dos primeiros testes numéricos mostrarem que estes *solvers* poderão ser promissores no futuro, ainda se está longe de os tornar capazes de resolver problemas de grande di-

mensão provenientes de aplicações reais. Todas estas novas técnicas requerem um acréscimo significativo de esforço quando comparadas com a abordagem NLP. Dado que os *solvers* acima referidos se encontram em fase de desenvolvimento e ainda não estão comercializáveis urge repensar a forma de resolver um problema MPCC. Uma série de estudos recentes apresentaram uma abordagem interessante: reformular o MPCC um problema NLP equivalente.

3.2 Reformulação NLP

Um abordagem interessante para resolver um problema MPCC, sugerida por Leyffer [43], é considerar a sua formulação não linear equivalente. Por um lado, esta abordagem permitiria tratar o problema MPCC como um problema NLP, aproveitando a vantagem de certos algoritmos NLP, robustos e sofisticados, permitindo obter convergência local rápida. Por outro lado, funcionaria como um desafio ao próprio algoritmo NLP usado, pois permitiria testar a sua eficiência e robustez e ainda avaliar a sua capacidade de lidar com as irregularidades inerentes aos problemas MPCC.

Definição 3.1 *Formulação NLP do problema MPCC*

$$\begin{aligned}
 \min \quad & f(x) \\
 \text{s.a.} \quad & c_i(x) = 0, i \in E \\
 & c_i(x) \geq 0, i \in I \\
 & x_1 \geq 0 \\
 & x_2 \geq 0 \\
 & x_1^T x_2 \leq 0
 \end{aligned} \tag{MPCC-NLP}$$

Todas as entidades envolvidas em (MPCC-NLP) foram definidas em (MPCC). Em (MPCC-NLP) a restrição de complementaridade de (MPCC) é substituída pela desigualdade $x_1^T x_2 \leq 0$. As condições KKT deste problema dão origem ao seguinte sistema de equações:

$$\begin{aligned}
& \nabla f(x) - \nabla c(x)^T \lambda - \begin{pmatrix} 0 \\ v_1 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ v_2 \end{pmatrix} + \xi \begin{pmatrix} 0 \\ x_2 \\ x_1 \end{pmatrix} = 0 \\
& \lambda_i \geq 0, \quad i \in E \cup I, \quad v_1, \quad v_2, \quad \xi \geq 0 \\
& c_i(x) = 0, \quad i \in E \\
& c_i(x) \geq 0, \quad i \in I \\
& x_1 \geq 0, \quad x_2 \geq 0, \quad x_1^T x_2 \leq 0 \\
& c(x)^T \lambda = 0, \quad x_1^T v_1 = 0, \quad x_2^T v_2 = 0, \quad \xi x_1^T x_2 = 0
\end{aligned} \tag{3.1}$$

em que λ , v_1 , v_2 e ξ são os multiplicadores das restrições não lineares, de $x_1 \geq 0$, $x_2 \geq 0$ e $x_1^T x_2 \leq 0$, respectivamente.

O problema definido desta forma, (MPCC-NLP), continua a violar a qualificação das restrições de Mangasarian-Fromovitz em qualquer ponto admissível, originando os inconvenientes apresentados no Capítulo 2. Esta razão, provocou uma desmotivação na comunidade científica, originando um abandono da ideia de resolver o problema MPCC numa perspectiva NLP.

3.3 Estacionaridade forte do problema MPCC

Recentemente surgiram estudos que demonstram alguns conceitos de estacionaridade para o problema MPCC, bem como a sua relação com as condições KKT do problema (MPCC-NLP). Scheel e Scholtes [56] observaram que a estacionaridade forte é equivalente às condições KKT (3.1) de (MPCC-NLP). Considere-se a seguinte definição de estacionaridade forte:

Definição 3.2 *Ponto fortemente estacionário*

Um ponto x^ é designado ponto fortemente estacionário se e só se existem multipli-*

adores λ , \widehat{v}_1 e \widehat{v}_2 tais que:

$$\begin{aligned} \nabla f(x^*) - \nabla c^T(x^*)\lambda - \begin{pmatrix} 0 \\ \widehat{v}_1 \\ \widehat{v}_2 \end{pmatrix} &= 0 \\ c_i(x^*) &= 0, \quad i \in E \\ c_i(x^*) &\geq 0, \quad i \in I \\ x_1^* &\geq 0 \quad x_2^* \geq 0 \\ x_{1j}^* &= 0 \quad \text{ou} \quad x_{2j}^* = 0 \\ \lambda &\geq 0 \\ c_i(x^*)\lambda_i &= 0, \quad i \in E \cup I \\ x_{1j}^*\widehat{v}_{1j} &= x_{2j}^*\widehat{v}_{2j} = 0 \\ \text{se } x_{1j}^* &= x_{2j}^* = 0 \quad \text{então } \widehat{v}_{1j} \geq 0 \quad \text{e} \quad \widehat{v}_{2j} \geq 0 \end{aligned} \tag{3.2}$$

A estacionaridade forte pode ser interpretada como as condições KKT do seguinte problema NLP relaxado:

Definição 3.3 NLP relaxado

Dados dois conjuntos de índices $\chi_1, \chi_2 \subseteq \{1, \dots, p\}$ com $\chi_1 \cup \chi_2 = \{1, \dots, p\}$ denota-se os respectivos conjuntos complementares em $\{1, \dots, p\}$ por χ_1^\perp e χ_2^\perp . Para cada par de conjuntos de índices, define-se o NLP relaxado correspondente ao MPCC como:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.a.} \quad & c_i(x) = 0, \quad i \in E \\ & c_i(x) \geq 0, \quad i \in I \\ & x_{1j} = 0 \quad \forall j \in \chi_2^\perp \quad \text{e} \quad x_{1j} \geq 0 \quad \forall j \in \chi_2 \\ & x_{2j} = 0 \quad \forall j \in \chi_1^\perp \quad \text{e} \quad x_{2j} \geq 0 \quad \forall j \in \chi_1 \end{aligned} \tag{NLP}_{rel}$$

Para um determinado conjunto de multiplicadores de Lagrange, v_1 , v_2 e ξ em (3.1), relativos às restrições de complementaridade de (MPCC-NLP), definem-se $\pi_1 = v_1 - \xi x_2$, e $\pi_2 = v_2 - \xi x_1$. Então qualquer outros multiplicadores \widehat{v}_1 , \widehat{v}_2 , $\widehat{\xi} \geq 0$, que mantenham

(π_1, π_2) constantes também satisfazem as condições (3.1). Os multiplicadores π_1 e π_2 são os multiplicadores de NLP_{rel} relativos às restrições $x_1 \geq 0$ e $x_2 \geq 0$, respectivamente. É mostrado em [24] que existe um vector de multiplicadores definido por

$$\xi = \max \left\{ 0, \max_{i \in \mathcal{I}_2^+} \frac{-\widehat{v}_1 i}{x_{2i}^*}, \max_{i \in \mathcal{I}_1^+} \frac{-\widehat{v}_2 i}{x_{1i}^*} \right\} \quad (3.3)$$

relativo à restrição $x_1^T x_2 \geq 0$. As componentes não nulas deste multiplicador correspondem ao conjunto linearmente independente dos gradientes das restrições, originando a verificação de uma qualificação das restrições designada por MPCC-LICQ.

A estacionaridade forte é uma caracterização computacional conveniente, uma vez que é relativamente fácil encontrar um ponto estacionário, sob razoáveis suposições, num *solver* NLP.

Este resultado animador, serviu para muitos investigadores, por exemplo Leyffer [26, 43], estabelecerem propriedades de convergência nos algoritmos SQP, para resolver problemas MPCC na versão (MPCC-NLP).

Foram também estes resultados uma forte motivação para que no capítulo seguinte, uma colecção de problemas MPCC fosse testada na sua versão NLP, utilizando-se *solvers* NLP, tirando-se partido das suas conhecidas propriedades de convergência e da sua robustez.

Capítulo 4

Solvers NLP

Com o intuito de resolver um conjunto de problemas MPCC e testar a fiabilidade dos *solvers* NLP na resolução desta classe de problemas, foram seleccionados quatro pacotes de *software* distintos: LOQO (um método de pontos interiores), MATLAB e NPSOL (métodos baseados na programação quadrática sequencial) e Filter SQP (método baseado na filosofia dos filtros). Neste capítulo é feita uma descrição sumária destes *solvers* apresentando as suas principais características.

Investigações recentes têm mostrado que os *solvers* NLP estão cada vez mais estáveis e robustos. Assim, uma abordagem interessante é testar até que ponto estes algoritmos são fiáveis com problemas de difícil resolução, como é o caso do problema MPCC.

Para este estudo foram seleccionados quatro *solvers*: LOQO, MATLAB, NPSOL e Filter SQP. Esta selecção baseou-se no facto de haver acessibilidade a qualquer um deles - existiam licenças para o MATLAB e NPSOL, a versão do LOQO é livre e o Filter SQP está a ser desenvolvido pelo grupo de investigação orientado pela supervisora desta dissertação. Todos os *solvers* considerados estão vocacionados para a resolução de problemas de optimização não linear, e existe para todos uma interface que permite a resolução de problemas codificados em AMPL [27]. Esta linguagem de modelação para programação matemática foi criada por Fourer, Gay e Kernighan. As linguagens de modelação ajudam a organizar e automatizar as tarefas de modelação, podendo lidar com um grande volume de dados; além disso, podem ser utilizadas em máquinas e *solvers* independentes, permitindo ao utilizador concentrar-se no modelo em vez da metodologia para chegar à solução. A

versão disponível do AMPL é a *student edition*, que permite a resolução de problemas até um máximo de 300 variáveis ou 300 restrições.

O LOQO [67], criado por Robert Vanderbei, trata-se de um programa para resolver problemas de optimização suave com restrições. Surgiu em 1991, e desde então tem vindo a sofrer bastantes modificações com o objectivo de melhorar o seu desempenho e simultaneamente alargar o leque de problemas a resolver. Para cada variável livre uma nova restrição é adicionada, implicitamente, fazendo com que as variáveis livres sejam escritas como a diferença entre dois números não negativos. As restrições de desigualdade são transformadas em igualdade, através da inclusão de variáveis de folga.

O LOQO é baseado num método primal-dual de pontos interiores aplicando uma sequência de aproximações quadráticas ao problema. Na fase primal, a admissibilidade não é requerida no início do processo, apenas vai ser forçada ao longo dele. Os pontos interiores referem-se ao facto das variáveis de folga serem estritamente positivas durante todo o processo. Neste *solver*, a matriz Hessiana é exacta e, para induzir a convergência global usa a técnica de procura unidimensional.

A programação quadrática sequencial (SQP) é uma das técnicas mais eficientes e divulgadas na optimização não linear, gerando direcções de procura através da resolução de uma sequência de subproblemas quadráticos. MATLAB e NPSOL são dois pacotes de *software* que se baseiam nesta técnica.

O uso do programa MATLAB tem vindo a aumentar, quer nos meios académicos quer nos meios industriais. Isto deve-se à facilidade com que lida com as matrizes e cálculos numéricos extensos. Este programa possui também um conjunto de *toolboxes* específicas com distintas aplicações.

A resolução de problemas de optimização no MATLAB implica a utilização da *toolbox Optimization*, nomeadamente a função *fmincon* e requer que a função objectivo e restrições sejam contínuas. O procedimento para encontrar a solução envolve duas fases: a primeira implica o cálculo de um ponto admissível, quando este existe; a segunda envolve a geração de uma sequência iterativa de pontos admissíveis que convergem para a solução.

O método usado na *toolbox* envolve uma estratégia de conjunto activo semelhante à do NPSOL. É utilizada uma função mérito na procura unidimensional e, em cada iteração, é calculada uma aproximação quasi-Newton da matriz Hessiana da função Lagrangeana através da actualização BFGS.

O NPSOL foi criado por Gill *et al.* [29]. É um *package* em *Fortran 77*, desenhado para resolver problemas de optimização não linear, onde as funções devem ser suaves mas não necessariamente convexas. Este algoritmo SQP apenas necessita das primeiras derivadas - quando estas não são fornecidas podem ser estimadas por diferenças finitas. A matriz Hessiana da função Lagrangeana é actualizada com a aproximação quasi-Newton BFGS.

O NPSOL utiliza um método baseado num esquema de procura unidimensional e envolve dois processos iterativos - um interno e outro externo. No processo iterativo interno resolve-se um subproblema quadrático para determinar a direcção de procura. Uma vez encontrada, o processo iterativo externo calcula uma nova aproximação à solução exigindo um decréscimo na função mérito (Lagrangeana aumentada). A sequência de aproximações geradas pelo processo iterativo externo converge para um ponto que verifica as condições de primeira ordem ou KKT. Como as restrições dos subproblemas são linearizadas e a função objectivo é uma aproximação quadrática à Lagrangeana, nem a função nem os valores do gradiente são necessários durante a solução de cada QP. Assim é especialmente eficaz para problemas não lineares cujas funções e gradientes envolvam extensos cálculos.

É de salientar que o subproblema QP interno é resolvido pela subrotina LSSOL. Este pacote interno resolve problemas lineares, quadráticos e de mínimos quadrados com restrições lineares. As restrições lineares e as de limites simples são tratadas separadamente por um método de conjunto activo. Possui um algoritmo numericamente estável e é especialmente recomendado para programação quadrática e programação linear densa. O LSSOL envolve duas fases para a resolução de um problema quadrático: na primeira, a fase de admissibilidade, é encontrado um ponto admissível através da minimização da violação das restrições; na segunda fase, a fase da optimalidade, a função objectivo quadrática é minimizada na região admissível.

O método dos filtros é uma técnica recente e o Filter SQP é um novo algoritmo para resolver problemas NLP proposto por Antunes e Monteiro [4] e que ainda está numa fase de aperfeiçoamento. Tem por base a ideia de Fletcher e Leyffer [24] apresentada em 2000. Mas, enquanto que estes autores recorrem à técnica de regiões de confiança, o Filter SQP usa uma estratégia de procura unidimensional para promover a convergência global.

Não deixando de ser um método SQP, a sua característica inovadora consiste na substituição da função penalidade por um esquema baseado no conceito de filtro, apresentando uma filosofia semelhante à da optimização multi-objectivo. Enquanto que na função de penalidade, os dois objectivos competitivos - minimização da função objectivo e satisfação das restrições - são combinados num único problema de minimização, na estratégia do filtro, são vistos como objectivos separados. Desta forma, são evitados os inconvenientes associados à função de penalidade, tais como a escolha do parâmetro e a não diferenciabilidade de algumas funções.

O filtro é visto como um conjunto de pares (função objectivo, violação das restrições), e em cada iteração funciona como um oráculo para aceitar ou rejeitar uma aproximação. Um novo par é aceite no filtro se reduz a função objectivo ou a violação das restrições. Assim, o critério de descida da função penalidade é substituído pela exigência do novo par ser aceite no filtro.

O método iterativo básico tem subjacente a programação quadrática sequencial, onde cada subproblema QP é resolvido pela rotina LSSOL do NPSOL. O algoritmo termina quando encontra um ponto KKT.

De salientar, que apesar do método dos filtros ser ainda uma técnica de optimização bastante recente, existe já um número significativo de trabalhos produzidos nesta área, apresentado-se como uma ferramenta promissora.

Capítulo 5

Experiências computacionais

Neste capítulo descrevem-se as experiências computacionais desenvolvidas. São feitas algumas considerações acerca do conjunto de problemas teste seleccionado, assim como do ambiente de programação onde estes problemas foram executados. Por fim, a análise dos resultados computacionais obtidos proporcionou a apresentação de tabelas, estatísticas e gráficos de modo a sintetizar o trabalho experimental efectuado.

5.1 Detalhes de implementação

Com o intuito de analisar a resolução do problema MPCC na sua versão NLP, através de *solvers* NLP, foi seleccionado um conjunto de problemas teste, codificados em AMPL, provenientes da MacMPEC [42]. Esta, é uma biblioteca recentemente compilada por Sven Leyffer e que contém uma vasta colecção de problemas MPCC de diferentes dimensões e origens distintas. O objectivo deste tipo de base de dados é possibilitar ao investigador o desenvolvimento de *software* numérico, quer para demonstrar a sua capacidade de resolução quer para testar a sua eficiência.

No Apêndice A estão listados os 92 problemas utilizados nesta experiência computacional, assim como a sua classificação. Nem todos os problemas da colecção MacMPEC foram incluídos pois, por uma questão de limitação da versão do AMPL disponível - versão *student* - não foi possível resolver problemas de grande dimensão. Para todo o conjunto

utilizado neste estudo, a condição de complementaridade usual na formulação (MPCC) foi substituída pela desigualdade não linear equivalente em (MPCC-NLP).

Os *solvers* NLP utilizados foram descritos no capítulo anterior. Para todos eles, o critério de paragem utilizado baseou-se na seguinte condição: número máximo de iterações (1000) ou erro relativo da aproximação ($\leq 10^{-6}$).

Para todos eles, houve a necessidade de recorrer a uma *interface* AMPL como elo de ligação entre o problema e o *solver*, tal como mostra a Figura 5.1. Como se pode verificar, isto permite uma total transparência no processo de experimentação, pois o problema inicial não sofre modificações na sua utilização pelo *solver*.

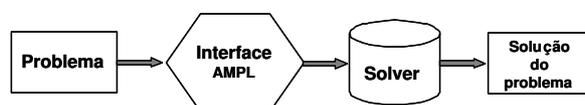


Figura 5.1: Esquema de implementação dos *solvers*

Para o LOQO, NPSOL e Filter SQP apenas foi necessário executar no Ms-Dos e, dentro da directoria que contém o *solver*, a seguinte linha de comandos:

```
C:\solver>ampl problema.mod
```

em que *problema.mod* representa o ficheiro em AMPL do problema a testar.

O uso do MATLAB para resolver um problema codificado em AMPL é um pouco mais complexo. A função MEX *amplfunc* permite ao MATLAB aceder aos dados do problema e, por forma a uniformizar os parâmetros da função *amplfunc* foram desenvolvidas duas M-files (*funcao.m* e *restricao.m*). O esquema de ligação é o apresentado na Figura 5.2:

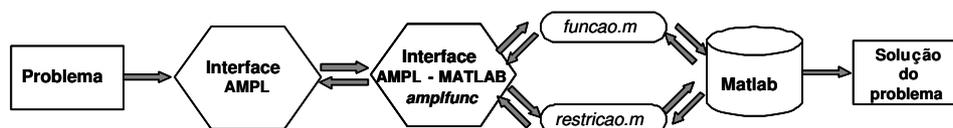


Figura 5.2: Esquema de implementação do MATLAB

As duas *M-files* são apresentadas de seguida:

```
funcao.m
```

```
function [f]=funcao(x,cl,cu) [f,c]=amplfunc(x,0);
```

```
restricao.m
```

```
function [c,ceq]=restricao(x,cl,cu) c=[]; ceq=[]; [f,cesp] =
amplfunc(x,0); aux1=size(cl); aux2=aux1(1); for i = 1:aux2
    if (cl(i)==cu(i))
        ceq=[ceq;cesp(i)-cl(i)];
    end
    if (cl(i)==-Inf)
        c=[c;cesp(i)-cu(i)];
    end
    if (cu(i)==Inf)
        c=[c;-cesp(i)+cl(i)];
    end
    if ((cl(i)~-Inf) & cu(i)~=Inf)
        c=[c;cesp(i)-cu(i);-cesp(i)+cl(i)];
    end
end
end
```

Para testar o problema foi necessário, na janela de comandos do MATLAB, colocar as seguintes linhas de comando:

```
>>[x0,lb,ub,v,cl,cu]=amplfunc('problema');
>>options=optimset;
>>ti=cputime;[x,fval,flag,output]=fmincon('funcao',x0,[],[],[],[],lb,ub,
'restricao',options,cl,cu);tf=cputime;tempo=tf-ti
```

Os testes foram realizados num Pentium IV, com um processador de 2600 Mhz e uma memória de 512 Mb de RAM no sistema operativo Windows XP.

5.2 Análise de resultados

Uma vez que a leitura, das tabelas com todos os resultados obtidos durante a fase experimental tornar-se-ia uma tarefa fastidiosa e a respectiva análise e interpretação pouco

claras, optou-se pela apresentação de tabelas e gráficos, que incluem a síntese dos resultados. Mais detalhes podem ser encontrados no Apêndice B - onde constam o nome do problema com o respectivo tempo CPU dispendido, número de iterações, cálculos de função e de gradiente usados para a determinação da solução, para cada *solver*.

Do conjunto dos testes, nem todos os problemas foram resolvidos com sucesso. A Tabela 5.1 mostra dois tipos de falhas que ocorreram: a primeira, onde houve um bloqueio do sistema, relacionada com questões internas do *software* (bloq.); a segunda refere-se ao facto do *solver* chegar ao limite de iterações sem confirmar a optimalidade (lim. iter.). À excepção do MATLAB, os resultados em termos de robustez foram muito animadores.

Solver	bloq.	lim. iter.
LOQO	1	11
MATLAB	10	8
NPSOL	0	4
Filter SQP	0	10

Tabela 5.1: Número de falhas ocorridas por *solver*

Os *solvers* NLP foram também analisados usando uma aplicação que permite comparar o desempenho de vários códigos relativamente a um parâmetro, desenvolvida por Dolan e Moré [13]. Esta comparação é tanto mais interessante quanto maior for o número de problemas teste considerado. Com esta ferramenta foram analisados os parâmetros tempo de CPU e o número de iterações gastas em cada problema.

Assim, seja p um problema teste considerado nesta experiência e PT o conjunto dos problemas teste. Para cada *solver* s , tem-se:

$$X = \frac{\text{tempo}(s,p)}{\text{melhortempo}(p)}, \forall p \in PT \text{ e } Y = \log_2(X) \quad (5.1)$$

onde $\text{tempo}(s,p)$ é o tempo de CPU usado pelo *solver* s a resolver o problema p e $\text{melhortempo}(p)$ é o valor mínimo do conjunto dos tempos de CPU dispendidos por todos os *solvers* no problema p . Facilmente se constata que $X \geq 1$ e quando X toma o valor 1 corresponde ao caso em que o *solver* teve o melhor tempo para a resolução do problema p .

A função distribuição (ou de probabilidade acumulada), $F(x) : [1, +\infty[\rightarrow [0, 1]$ define-se por:

$$F(x) = P(X \leq x).$$

É possível visualizar um conjunto de funções contínuas crescentes e cumulativas, onde um *solver* apresenta melhor desempenho relativamente a um parâmetro quando a respectiva função tende mais rapidamente para um. No entanto, também na opinião de Dolan e Moré, a interpretação dos gráficos não é intuitiva. O gráfico fornece a percentagem de problemas em que o *solver* é, na pior das hipóteses, x vezes mais lento do que o melhor *solver*.

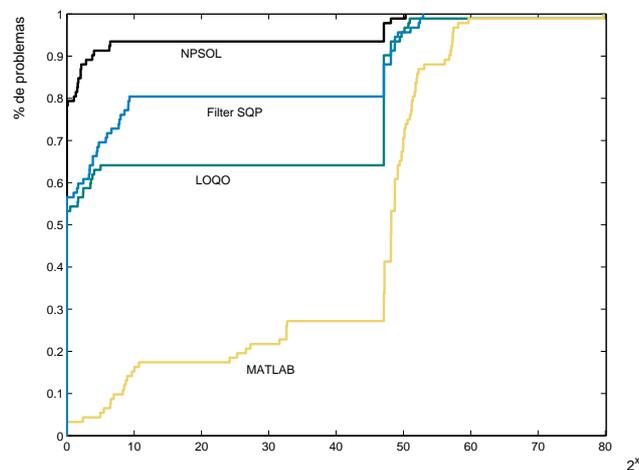


Figura 5.3: Perfis de desempenho relativo ao tempo de CPU

Note-se que as falhas ocorridas no conjunto dos problemas teste também foram consideradas nesta comparação: nos problemas onde os *solvers* bloquearam é sugerido por Dolan e Moré a atribuição de um tempo penalizador de 10^8 segundos.

A Figura 5.3 representa os perfis de desempenho relativamente ao tempo. Podemos verificar que o NPSOL foi o solver mais eficiente; em contrapartida, o MATLAB foi o que apresentou piores resultados devido à sua menor robustez.

Analogamente, foram traçados os perfis de desempenho relativamente ao número de iterações, como mostra a Figura 5.4. Note-se que para uma falha grave ocorrida num

solver para um dado problema foi-lhe atribuída uma penalização de 10^6 iterações. É possível verificar que, mais uma vez, foram o NPSOL e o Filter SQP, que obtiveram os melhores resultados.

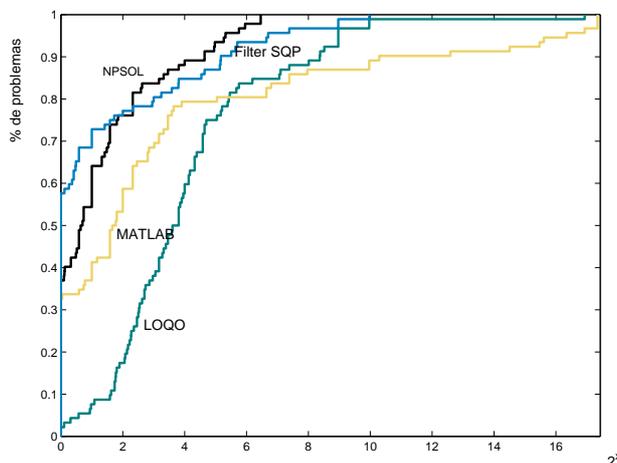


Figura 5.4: Perfis de desempenho relativo ao número de iterações

Fez-se também uma outra análise: dos 92 problemas teste considerados inicialmente, foram retirados todos aqueles que bloquearam em pelo menos um *solver*, ficando apenas com um universo de 81 problemas. Na Tabela 5.2 é apresentado um resumo estatístico dos resultados obtidos.

De uma forma geral, verifica-se que os quatro *solvers* apresentam bons resultados em termos de dispêndio de recursos. Curiosamente, o Filter SQP, que no gráfico de Dolan e Moré apresentava um bom desempenho em termos do número de iterações utilizadas, aqui em termos médios, não apresenta resultados tão satisfatórios. Isto deve-se ao facto, da estatística média ser muito influenciada por valores extremos, ou seja, como nalguns problemas teste o Filter SQP atingiu as 1000 iterações, originou uma média bastante elevada. Se for tido em consideração o valor mediano, constata-se que apresenta melhores valores.

A Figura 5.5 fornece informação acerca do número de iterações requeridas para cada problema para cada *solver*. O MATLAB, NPSOL e Filter SQP, que têm subjacente a

Parâmetros	Estatísticas	Loqo	Matlab	Npsol	Filter SQP
Tempo CPU	Mínimo	0	0	0	0
	Máximo	46.968	2725.1	23.156	151.297
	Soma	58.273	4183.788	51.501	471.801
	Média	0.719	51.652	0.636	5.825
	Moda	0	0.16	0	0
	Mediana	0.015	0.094	0	0
	Desvio Padrão	5.224	308.641	3.565	24.137
Núm. iterações	Mínimo	9	1	1	1
	Máximo	1000	1000	1000	1000
	Soma	12727	8442	3070	8139
	Média	157.123	104.222	37.901	100.481
	Moda	1000	1	1	1
	Mediana	28	5	6	3
	Desvio Padrão	314.05	292.13	155.41	277.39
Cálc. Função	Mínimo	17	7	1	2
	Máximo	11521	100045	2953	1001
	Soma	66243	342151	7854	9102
	Média	817.814	4224.086	96.963	112.370
	Moda	47	13	2	2
	Mediana	55	79	8	4
	Desvio Padrão	2321.933	13666.561	459.610	294.917
Cálc. Gradiente	Mínimo	17	7	1	2
	Máximo	11521	100045	2953	20001
	Soma	66243	342151	7854	113216
	Média	817.815	4224.086	96.963	1397.728
	Moda	47	13	2	2
	Mediana	55	79	8	4
	Desvio Padrão	2321.933	13666.561	459.610	4732.744

Tabela 5.2: Estatísticas relativas ao desempenho dos *solvers*

programação quadrática sequencial, resolveram um número elevado de problemas em menos de 10 iterações.

O mesmo tipo de comparação mas em termos de cálculos de função e de gradiente é feita nas Figuras 5.6 e 5.7, respectivamente. O LOQO e o MATLAB foram os pacotes de *software* que apresentaram um desempenho menos favorável.

Os resultados obtidos neste trabalho vão ao encontro de outros estudos realizados, com objectivos mais ou menos semelhantes. Gu *et al* [31] observaram que algumas das funções utilizadas na *toolbox Optimization* do MATLAB eram um pouco instáveis e a sua *performance* pobre na tomada de decisão. Benson *et al* [6] analisaram a situação relativamente ao LOQO, observando algumas das limitações aquando da sua implementação

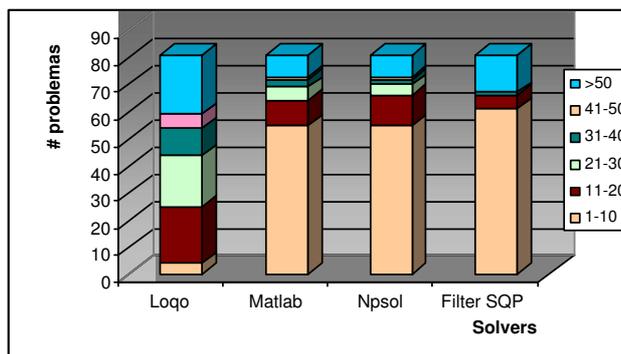
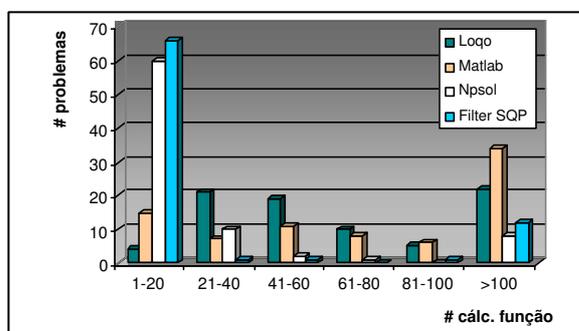
Figura 5.5: Número de iterações usadas pelos *solvers*

Figura 5.6: Cálculo de funções

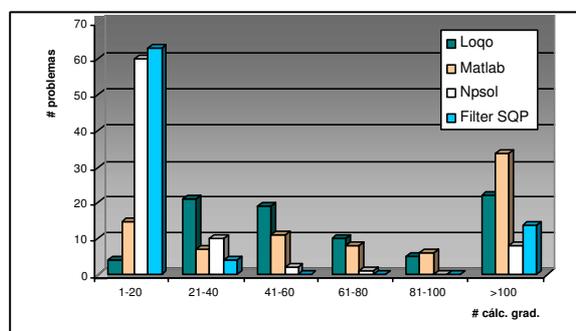


Figura 5.7: Cálculo de gradientes

em problemas não lineares, propondo algumas abordagens para ultrapassar os aspectos menos positivos. Manuseamento cuidadoso das variáveis de folga, alteração de parâmetros de convergência e tratamento especial dos multiplicadores ilimitados são sugestões para melhorar o desempenho do método de pontos interiores.

Tendo sido testado um número considerável de problemas MPCC na sua versão NLP usando quatro *solvers* vocacionados para a programação não linear, constatou-se que é possível resolvê-los tirando partido das propriedades dos algoritmos que estão subjacentes aos vários *softwares*.

Capítulo 6

Conclusões e trabalho futuro

Para finalizar a primeira parte desta dissertação, foi realizada uma reflexão sobre o trabalho realizado inferindo algumas conclusões. São também definidas linhas de orientação relacionadas com o trabalho desenvolvido que vão merecer num futuro próximo dedicação, abrangendo a possibilidade de estender a investigação a áreas afins.

6.1 Conclusões

O problema MPCC foi o objecto de estudo desta investigação. A utilização deste problema na forma original, ou seja, com as restrições de complementaridade ainda não é, na maioria dos casos, viável uma vez que as condições de optimalidade usuais não se verificam e os *solvers* específicos para a sua resolução ainda estão em fase de aperfeiçoamento.

Houve a necessidade de recorrer a uma abordagem diferente para a sua resolução: a reformulação do problema MPCC num problema NLP. A aproximação NLP, no conjunto dos problemas teste resolvidos, apresentou-se como uma forma prática e eficaz para contornar as dificuldades do problema MPCC, pois os pacotes de *software* vocacionados para problemas NLP apresentaram índices de robustez elevados.

Relativamente aos *solvers* seleccionados para este estudo, foram os métodos que têm subjacente os algoritmos SQP que apresentaram melhores resultados. O NPSOL foi o que se destacou mais em termos de tempo de CPU dispendido. O *solver* que tem por base a

filosofia dos filtros apresentou resultados surpreendentes em termos do número de iterações, cálculos de função e de gradiente.

O problema MPCC apresenta-se como uma forma recente de modelar questões económicas e de engenharia, bem como uma nova abordagem de alguns problemas de optimização.

Este projecto, a nível pessoal, para além de desenvolver os conhecimentos na área da optimização, fomentou os meus conhecimentos informáticos, através da utilização de vários pacotes de *software* numérico, de uma linguagem de modelação de problemas matemáticos e de um programa específico para a escrita desta dissertação. Permitiu ainda o contacto com uma plataforma na internet para resolver os problemas. Além disso, possibilitou o contacto com a comunidade científica - quer através da correspondência por *mail*, quer nas conferências - possibilitando um desenvolvimento sustentável deste trabalho.

6.2 Trabalho Futuro

É possível destacar algumas linhas de acção futuras, que possam complementar este trabalho. Numa primeira fase, pretende-se observar o comportamento de problemas mais complexos e de maior dimensão quando resolvidos pelos *solvers* NLP.

Ainda relativamente aos *solvers*, seria interessante analisar as modificações iniciais sugeridas por alguns autores nos problemas teste onde o LOQO apresentou resultados menos satisfatórios e comentar o seu desempenho relativamente aos outros *solvers*. Além disso, poder-se-ia com esta mesma colecção de problemas, testar outros *solvers* de pontos interiores, com o intuito de observar se os resultados obtidos se devem à especificidade do LOQO ou à filosofia subjacente ao método de pontos interiores.

Trabalhar mais estritamente com os autores do Filter SQP, para observar e tentar colmatar possíveis falhas deste código relativamente à resolução do problema MPCC é uma proposta para um futuro próximo. Os seus autores estão actualmente a desenvolver um algoritmo específico para problemas MPCC: no filtro em vez de ser considerado um conjunto de pares (função objectivo, restrições) passa a ser analisado um conjunto de ternos (função objectivo, restrições não lineares, restrições de complementaridade). Para gestão

deste filtro, um conjunto de heurísticas é utilizado onde as restrições de complementaridade têm um tratamento especial.

O acompanhamento da evolução dos *solvers* específicos para MPCC, é também um projecto aliciante, pois neste momento não existe nenhum comercialmente disponível para ser testado. Como objectivo mais ambicioso, apresenta-se o desenvolvimento de um algoritmo específico para a resolução de problemas MPCC.

Por último, seria interessante poder codificar e trabalhar ainda mais de perto com problemas MPCC reais, pois como diz Polya, um matemático apoiante fervoroso da resolução de problemas, "resolver um problema é encontrar um caminho onde nenhum caminho é conhecido de imediato, é encontrar um caminho para sair de uma dificuldade, é encontrar um caminho em torno de um obstáculo, é atingir um objectivo desejado que não é imediatamente acessível, e fazê-lo com os meios apropriados".

Parte II

Aplicação ao mercado de energia eléctrica

Capítulo 1

Introdução

Desde sempre o Homem gostou de ver a aplicabilidade física dos conceitos. Por isso, este projecto não ficaria completo, se utilizasse apenas problemas académicos. Neste capítulo, é elaborada uma contextualização do problema real a ser estudado - mercado de energia eléctrica - mostrando simultaneamente a actualidade e pertinência do tema.

1.1 Motivação

No início deste projecto, quando se definiram os objectivos a atingir, ficou acordado que um dos mais motivadores seria resolver um problema real. Teria que ser um problema que ainda não estivesse codificado em AMPL e ainda não tivesse sido explorado na sua relação MPCC-NLP. A meta final seria enviar este problema para divulgação numa biblioteca específica para problemas MPCC, de modo a poder ser testado e usado noutras investigações.

Após uma pesquisa exaustiva e muitos contactos realizados com a comunidade científica através da internet, a esperança de atingir o objectivo ia-se desvanecendo. As dificuldades encontradas eram muitas: desde problemas de grande dimensão (pouco viáveis para este projecto) a problemas cuja informação fornecida era insuficiente para a sua codificação.

Todavia, e já na parte final desta dissertação, surgiu um problema, de média dimensão, e que aliava a engenharia à economia: um mercado de energia eléctrica.

1.2 Mercados de energia eléctrica

O mercado de energia eléctrica está numa fase de transição. Até agora, o negócio era controlado pelos governos de cada país e era regulado pelo poder central. Não havia competição entre companhias: a cada empresa era atribuído um conjunto de consumidores e, ao consumidor não lhe era facultada a opção de escolher qual a empresa que lhe fornecia a energia.

Nos mercados desregularizados, a produção de electricidade torna-se uma actividade liberal; o planeamento de expansão e a calendarização de operacionalização já não dependem de procedimentos centrais e administrativos, mas sim de decisões das companhias produtoras. Enquanto que num mercado regularizado, o objectivo das empresas era minimizar os seus custos (uma vez que o preço estava tabelado), num mercado desregularizado o seu objectivo máximo é aumentar o lucro.

Depois da desregularização, as empresas poderão vender a energia a qualquer consumidor desde que cheguem a um acordo com este. A esperança é que num mercado competitivo entre empresas o preço não seja inflacionado, saindo, deste modo, o consumidor beneficiado. Neste mercado, as empresas de electricidade assumem um risco maior à medida que vão sendo cada vez mais responsáveis pelas suas próprias decisões. Assim, as empresas necessitam de modelos de suporte à decisão que não só considerem as limitações técnicas operacionais como também a interacção de todos os intervenientes no mercado.

Esta desregularização tem vindo a efectuar-se um pouco por toda a parte. Em 1998 os EUA iniciaram um processo de desregularização: Califórnia, Massachusetts e Rhode Island foram os pioneiros, logo seguidos por outros estados americanos [53].

Na Europa, o processo iniciou-se mais cedo: a Inglaterra e o País de Gales começaram o seu processo na década de 80. Fusões e aquisições tem sido o cenário no sector eléctrico europeu nos últimos anos, resultando num elevado grau de concentração de mercado por parte de um pequeno número de companhias. Uma vez que a electricidade é difícil de armazenar, é fácil para as empresas dominantes do mercado manipularem o preço, alterando-o para valores muito superiores ao razoável especialmente em tempos de muita

procura [7].

Além disso, as directivas da União Europeia para a liberalização do mercado energético, fizeram com que as ligações físicas e institucionais entre mercados de diferentes países se expandissem nos últimos cinco anos [14, 15]. Têm surgido, recentemente, artigos relativos a estudos ainda em curso, sobre os mercados alemão, francês e dos Países Baixos [34, 35, 61].

Existem duas razões pelas quais os mercados de electricidade são significativamente diferentes dos mercados dos outros bens. Por um lado, a electricidade é um bem difícil de armazenar e toda a energia é transportada pelo mesmo conjunto de linhas de transmissão, qualquer que seja a empresa. Por outro, a electricidade possui propriedades físicas muito especiais, designadamente as leis de Kirchhoff. Tudo isto levou a que a comunidade científica tentasse encontrar modelos para tentar prever como vão reagir os preços nesta nova estrutura de mercado.

Capítulo 2

O problema

Para se compreender o problema do mercado de energia eléctrica com uma empresa líder, apresenta-se a interligação de dois conceitos: optimização de dois níveis e o jogo de Stackelberg. De seguida, o problema é enunciado de forma detalhada, mostrando a sua formulação matemática, a rede utilizada e os dados para a sua implementação.

2.1 Optimização de dois níveis e o jogo de Stackelberg

No Capítulo 1 da Parte I já se tinha referenciado que, quer o jogo de Stackelberg quer a optimização de dois níveis, sob certas condições, podem formular-se como problemas MPCC.

No modelo de Stackelberg, existe uma situação de não competitividade, na qual o líder tem o poder de manipular os preços e a capacidade de produção e de expansão por forma a maximizar o seu lucro [68]. Neste modelo, o líder toma as suas decisões tendo em conta a antevisão das reacções das firmas seguidoras, e estas não têm a percepção de como as suas decisões afectam as decisões do líder.

A teoria de jogo de Stackelberg motivou muitos autores para o estudo dos problemas de optimização de dois níveis, pois existem muitas semelhanças entre ambos.

Apresenta-se, de seguida, a definição matemática do problema de optimização de dois níveis.

Definição 2.1 *Problema de otimização de dois níveis*

Um problema de otimização de dois níveis é constituído por um problema de nível superior:

$$\begin{aligned} \min_{x,y} \quad & F(x,y) \\ \text{s.a} \quad & g(x,y) \leq 0 \end{aligned} \quad (2.1)$$

em que y , para cada valor de x , é a solução do problema do nível inferior:

$$\begin{aligned} \min_y \quad & f(x,y) \\ \text{s.a} \quad & h(x,y) \leq 0 \end{aligned} \quad (2.2)$$

com $x \in \mathbb{R}^{nx}$, $y \in \mathbb{R}^{ny}$, $F, f : \mathbb{R}^{nx+ny} \rightarrow \mathbb{R}$, $g : \mathbb{R}^{nx+ny} \rightarrow \mathbb{R}^{nu}$ e $h : \mathbb{R}^{nx+ny} \rightarrow \mathbb{R}^{nl}$

As variáveis x/y são chamadas as variáveis de nível superior/inferior, $g(x,y) / h(x,y)$ as restrições de nível superior/inferior e $F(x,y) / f(x,y)$ a função objectivo de nível superior/inferior.

O problema de dois níveis é convexo se $f(x,y)$ e $h(x,y)$ forem funções convexas em y para todos os valores de x , ou seja, se o problema de nível inferior for convexo [69]. A vantagem do problema de dois níveis convexo é a de que, sob certas condições, o problema de nível inferior pode ser substituído pelas suas condições KKT obtendo-se um problema equivalente a um nível, isto é, transformando-se num problema MPCC.

2.2 Formulação do problema

2.2.1 Cenário de mercado com uma empresa líder

O modelo apresentado de seguida vem documentado no artigo [33] e na tese da mesma autora [47].

Neste modelo, existem empresas produtoras, cada uma com um conjunto de unidades geradoras. Cada unidade faz uma proposta/oferta de preço, por hora, a um operador de sistema independente - ISO, *Independent System Operator*. Cada proposta é da forma

linear não decrescente relativa à função do preço marginal (\$/MWh). O ISO decide então quanta energia compra a cada unidade, quanta energia entrega a cada consumidor e a que preços, baseado no fluxo de energia óptimo [59].

Cada unidade geradora determina a sua proposta, sendo esta a peça fundamental do modelo. Mas a proposta raramente é a verdadeira função custo: através de um desvio da sua proposta do custo marginal, sem o conhecimento do mundo exterior, a empresa pode aumentar o seu lucro, como mostra a Figura 2.1.

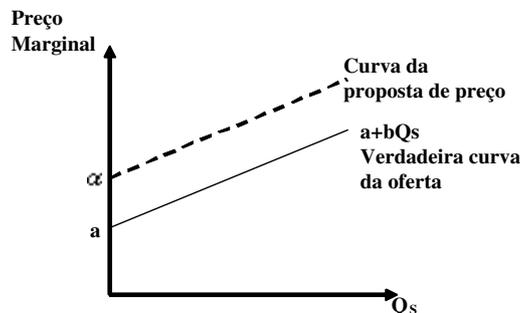


Figura 2.1: Curva da oferta (custo marginal) e curva da proposta

Neste modelo, assume-se que as empresas apenas manipulam α (ordenada na origem da curva da proposta) e não o declive b . Existem razões para que isto aconteça: primeiro, os declives do custo marginal para produtores individuais são geralmente muito baixos, e portanto, uma alteração acentuada do declive resultante da sua manipulação não seria credível; em segundo lugar, as manipulações que satisfazem a empresa podem ser feitas através de alterações unicamente em α ; por último, se a e b fossem ambos alterados, foi mostrado em [30, 32] que a solução única do problema raramente existe. Este modelo destaca-se de outros apresentados, pois aqui o equilíbrio do preço no mercado oligopolista é calculado numa rede onde há a variação conjectural da função de oferta, enquanto que se considera uma estratégia de propostas contínuas para cada fornecedor.

A empresa líder toma como *input* a sua percepção das condições de mercado (incluindo as propostas das firmas concorrentes, funções de procura e oferta) e maximiza o seu lucro dentro de um conjunto de restrições do equilíbrio de preço espacial (EPE, já referenciado no primeiro capítulo desta dissertação) juntamente com as restrições relativas às leis de Kirchhoff.

2.2.2 Formulação matemática

Na essência, o problema da empresa líder é um problema de otimização de dois níveis. No nível superior - o nível do líder - o parâmetro da curva de proposta corresponde à variável de primeiro nível. No nível inferior - o nível dos seguidores - há a simulação das conjecturas de mercado promovidas pelo ISO, onde as quantidades de energia gerada e consumida e a transmissão entre nós são as variáveis de segundo nível. O modelo tenta determinar as propostas óptimas de cada empresa.

De seguida é apresentada a notação usada para a formulação do problema:

Índices:

- i nó da rede
- ij arco de i para j
- m ciclos independentes da rede

Conjuntos:

- \mathcal{N} conjunto dos nós
- \mathcal{A} conjunto dos arcos
- \mathcal{S}_f conjunto dos nós produtores sob o controlo da firma f dominante
- \mathcal{P} conjunto dos nós produtores
- \mathcal{D} conjunto dos nós de procura
- \mathcal{L} conjunto dos ciclos independentes orientados na rede
- \mathcal{L}_m conjunto dos arcos orientados no ciclo m

Na prática, os conjuntos \mathcal{P} e \mathcal{D} não são necessariamente disjuntos e a sua união pode ser um subconjunto próprio de \mathcal{N} . Nos modelos linearizados DC [59, 70] as leis de Kirchhoff

das tensões e das correntes asseguram a unicidade do fluxo na rede em cada arco na solução do problema. Além disso, o número de ciclos independentes necessários são $\#\mathcal{A} - \#\mathcal{N} + 1$.

Parâmetros:

- a_i, b_i ordenada na origem e declive da curva de oferta (custo marginal)
 $Q_{S_i} \longrightarrow a_i + b_i Q_{S_i}$ para o produtor no nó $i \in \mathcal{P}$
- c_i, d_i ordenada na origem e declive da função de procura $Q_{D_i} \longrightarrow c_i - d_i Q_{D_i}$
 para os nós consumidores $i \in \mathcal{D}$
- $\bar{\alpha}_i$ limite superior para a proposta do nó produtor $i \in \mathcal{S}_f$
- \bar{Q}_{S_i} limite superior da capacidade de produção para o nó produtor $i \in \mathcal{P}$
- \bar{T}_{ij} capacidade máxima de transmissão no arco $ij \in \mathcal{A}$
- r_{ij} reactância no arco $ij \in \mathcal{A}$
- s_{ijm} ± 1 correspondendo à orientação do arco $ij \in \mathcal{A}$ no ciclo $m \in \mathcal{L}$
 (+1 se ij tem a mesma orientação que o ciclo m)

Variável de primeiro nível:

- α proposta de preço da unidade

Variáveis primais no problema EPE de segundo nível:

- Q_{S_i} quantidade de energia gerada no nó produtor i ($Q_{S_i} = 0$ se $i \notin \mathcal{P}$)
- Q_{D_i} quantidade de energia consumida no nó i ($Q_{D_i} = 0$ se $i \notin \mathcal{D}$)
- T_{ij} MW transmitidos de i para j

Variáveis duais no problema EPE de segundo nível:

- λ_i custo marginal no nó i
- μ_i valor marginal da capacidade de produção da fábrica no nó i
- θ_{ij} valor marginal da capacidade de transmissão no arco ij
- γ_m preço sombra devido à lei de voltagem de Kirchhoff para o ciclo m

Consideremos α_i para todos os nós $i \in \mathcal{P} \setminus \mathcal{S}_f$ fixos em ambos os níveis, definidos como as propostas das empresas concorrentes, apenas variando α_i para $i \in \mathcal{S}_f$. O problema de segundo nível é formalmente estabelecido como um programa quadrático convexo nas variáveis Q_S, Q_D e T , parametrizada pelas propostas α_i para $i \in \mathcal{S}_f$ [57].

Define-se, em seguida, o problema de segundo nível.

Função objectivo do problema de segundo nível

Maximização do bem estar social:

$$\max \sum_{i \in \mathcal{D}} \left(c_i Q_{D_i} - \frac{1}{2} d_i Q_{D_i}^2 \right) - \sum_{i \in \mathcal{P}} \left(\alpha_i Q_{S_i} + \frac{1}{2} b_i Q_{S_i}^2 \right) \quad (2.3)$$

Restrições

- Variáveis de procura não negativas:

$$Q_{D_i} \geq 0, \quad i \in \mathcal{D} \quad (2.4)$$

- Variáveis de transmissão não negativas (limite inferior) e com capacidade máxima de transmissão (limite superior):

$$0 \leq T_{ij} \leq \bar{T}_{ij}, \quad ij \in \mathcal{A} \quad (2.5)$$

- Capacidades mínima e máxima de produção para as variáveis de oferta (neste modelo a capacidade mínima é zero):

$$0 \leq Q_{S_i} \leq \bar{Q}_{S_i}, \quad i \in \mathcal{P} \quad (2.6)$$

Se a capacidade mínima não for zero, então uma substituição pode ser feita nas variáveis, de modo que seja novamente limitada inferiormente por zero; isto aplica-se em casos cujo custos iniciais de abertura de uma unidade geradora sejam muito elevado, não fazendo sentido abri-la para uma produção pequena de energia.

- Restrições de conservação:

$$Q_{D_i} - Q_{S_i} + \sum_{j:ij \in \mathcal{A}} T_{ij} - \sum_{j:ji \in \mathcal{A}} \bar{T}_{ij} = 0 \quad (2.7)$$

- Lei de Kirchhoff das tensões:

$$\sum_{ij \in \mathcal{L}_m} s_{ijm} r_{ij} T_{ij} = 0 \quad (2.8)$$

Consideremos, ainda, Δ a matriz de incidência do (nó, arco) da rede eléctrica:

$$\Delta_{il} = \begin{cases} 1 & \text{se } l = ij \in \mathcal{A} \text{ para algum } j \in \mathcal{N} \\ -1 & \text{se } l = ji \in \mathcal{A} \text{ para algum } j \in \mathcal{N} \\ 0 & \text{caso contrário} \end{cases} \quad (2.9)$$

Seja R a matriz relativa aos coeficientes das reactâncias, precedidos do sinal de orientação do arco em relação ao ciclo (arco, ciclo):

$$R_{ij,m} = \begin{cases} s_{ijm}r_{ij} & \text{se } ij \in \mathcal{L}_m \\ 0 & \text{caso contrário} \end{cases} \quad (2.10)$$

As condições de optimalidade KKT do problema primal são:

$$\begin{aligned} 0 \leq \bar{Q}_S - Q_S & \perp \mu \geq 0 \\ 0 \leq Q_S & \perp -\lambda + \mu + \alpha + \text{diag}(b)Q_S \geq 0 \\ 0 \leq Q_D & \perp \lambda - c + \text{diag}(d)Q_D \geq 0 \\ 0 \leq \theta & \perp \bar{T} - T \geq 0 \\ 0 \leq T & \perp \Delta^T \lambda + \theta + R\gamma \geq 0 \\ \lambda \text{ livre} & \quad Q_D - Q_S + \Delta T = 0 \\ \gamma \text{ livre} & \quad R^T T = 0 \end{aligned} \quad (2.11)$$

em que μ , θ , λ e γ são as variáveis duais. A notação $\text{diag}(w)$ representa a matriz diagonal cuja as entradas na diagonal principal são as componentes do vector w .

Com o problema de segundo nível definido, pode-se enunciar o problema de primeiro nível, relacionado com a firma dominante f . Especificamente, tomando $\alpha_i \notin \mathcal{S}_f$ como dado, a firma f determina um vector de propostas $\alpha^f \equiv (\alpha_i : i \in \mathcal{S}_f)$, um vector de oferta Q_S , um vector de procura Q_D e um vector de fluxos T com o objectivo de maximizar o lucro:

$$\begin{aligned} \text{maximizar} \quad & \pi_f(\lambda, Q_S) \equiv \sum_{i \in \mathcal{S}_f} (\lambda_i Q_{S_i} - a_i Q_{S_i} - \frac{b_i}{2} Q_{S_i}^2) \\ \text{sujeito a} \quad & 0 \leq \alpha_i \leq \bar{\alpha}_i, \quad \forall i \in \mathcal{S}_f \end{aligned} \quad (2.12)$$

em que Q_S , Q_D e T , para cada valor de α , são a solução do problema do nível inferior (2.3)-(2.8).

Prova-se que, para cada vector α , existe uma única solução óptima global para o programa quadrático (2.3)-(2.8), denotado por $(Q_D(\alpha), Q_S(\alpha), T(\alpha))$ [39], podendo o problema de nível inferior (2.3)-(2.8) ser substituído pelas suas condições de optimalidade KKT obtendo-se o seguinte problema, agora a um só nível:

$$\begin{aligned}
\text{maximizar} \quad & \pi_f(\lambda, Q_S) \equiv \sum_{i \in \mathcal{S}_f} (\lambda_i Q_{S_i} - a_i Q_{S_i} - \frac{b_i}{2} Q_{S_i}^2) \\
\text{sujeito a} \quad & 0 \leq \alpha_i \leq \bar{\alpha}_i, \quad \forall i \in \mathcal{S}_f \\
& 0 \leq \bar{Q}_S - Q_S \quad \perp \quad \mu \geq 0 \\
& 0 \leq Q_S \quad \perp \quad -\lambda + \mu + \alpha + \text{diag}(b)Q_S \geq 0 \\
& 0 \leq Q_D \quad \perp \quad \lambda - c + \text{diag}(d)Q_D \geq 0 \\
& 0 \leq \theta \quad \perp \quad \bar{T} - T \geq 0 \\
& 0 \leq T \quad \perp \quad \Delta^T \lambda + \theta + R\gamma \geq 0 \\
& \lambda \text{ livre} \quad \quad \quad Q_D - Q_S + \Delta T = 0 \\
& \gamma \text{ livre} \quad \quad \quad R^T T = 0
\end{aligned} \tag{2.13}$$

O problema (2.13) apresenta-se como um MPCC. Nesta forma, a função objectivo nem é convexa nem é côncava nos seus argumentos, por causa do termo bilinear $\lambda_i Q_{S_i}$. Por várias razões, será útil reformular a função objectivo para uma função côncava. Esta reformulação é na realidade possível ao explorar as restrições do sistema (2.11). A função objectivo resultante, a qual nos vamos referir por $P_f(\lambda, Q_D, Q_S, T, \theta, \lambda, \mu)$, é usada no procedimento computacional para resolver o problema de maximização do lucro da firma dominante f (2.12). Assim, se λ , Q_D , Q_S , T , θ , λ e μ satisfazem (2.11), então:

$$\begin{aligned}
\pi_f(\lambda, Q_S) & \equiv P_f(\lambda, Q_D, Q_S, T, \theta, \lambda, \mu) \\
& = \sum_{i \in D} (c_i Q_{D_i} - d_i Q_{D_i}^2) - \sum_{i \in \mathcal{S}_f} (a_i Q_{S_i} + \frac{b_i}{2} Q_{S_i}^2) \\
& \quad - \sum_{ij \in A} \theta_{ij} \bar{T}_{ij} - \sum_{i \in P \setminus \mathcal{S}_f} (\mu_i \bar{Q}_{S_i} + a_i Q_{S_i} + b_i Q_{S_i}^2)
\end{aligned} \tag{2.14}$$

2.2.3 Dados do problema

Os dados do problema, relativos à produção, procura e transmissão de energia foram baseados em [2]. A rede eléctrica considerada inclui um circuito de 30 nós, 41 arcos, 12 ciclos, 6 nós de produção e 21 nós de procura. Dos nós de produção propostos assume-se que a empresa A - firma dominante - possui três geradores e, os restantes três pertencem à empresa B. A Figura 2.2 concentra toda esta informação.

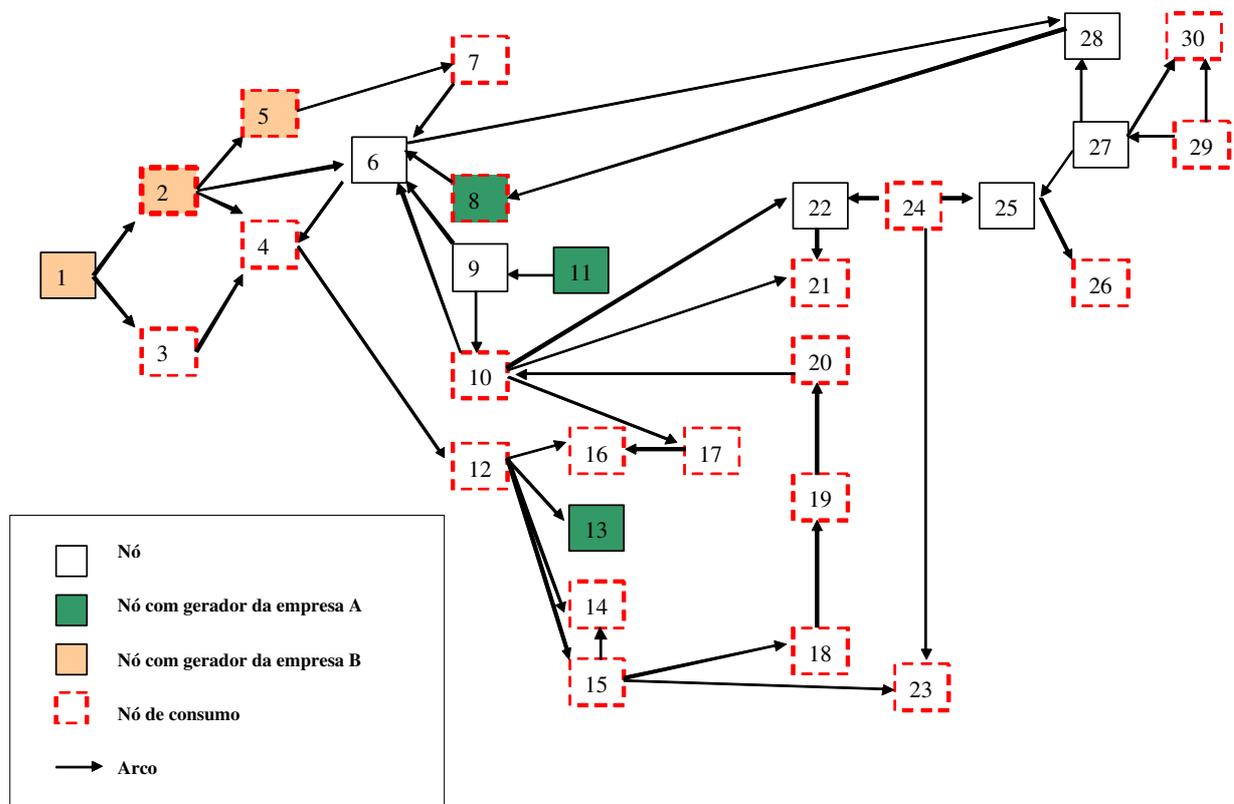


Figura 2.2: Esquema da rede de energia eléctrica

Também é dado, nesse artigo, as funções de custo de produção, reactâncias, limites superiores das quantidades de procura e da capacidade de transmissão dos arcos. Note-se que os limites de transmissão considerados para este modelo são 60% dos valores assumidos nesse artigo. Os valores encontram-se tabelados no Apêndice C.

Para resolver o problema da firma dominante A, assume-se que as propostas da outra empresa são iguais aos custos marginais para todas os seus geradores, isto é, $\alpha = a$. A curva da procura em cada um dos nós consumidores é determinada por $P_i = 40 - d_i Q_{D_i}$ onde d_i é escolhido de tal forma que $P_i = \$30/MWh$ quando Q_{D_i} iguala o valor assumido em [2].

Segue-se no próximo capítulo a resolução do problema.

Capítulo 3

Análise de resultados

Para a resolução do problema do mercado de energia eléctrica, foi necessário recorrer a uma plataforma na internet - NEOS - que disponibiliza uma série de *solvers* de optimização. Neste capítulo, são apresentados e analisados os resultados obtidos. São tecidas algumas considerações finais.

Dado que o problema de energia eléctrica é de média dimensão e o número de restrições do problema é elevado, a versão AMPL disponível não permitiu a sua resolução através dos *solvers* usados na Parte I. Assim, houve a necessidade de recorrer a uma plataforma informática para a sua resolução - NEOS [1].

NEOS - *Network-Enabled Optimization System* é um serviço de optimização que utiliza a internet como seu distribuidor. Possui um vasto conjunto de *solvers* de optimização, intitulado-se como "o estado da arte na optimização".

Para submeter o problema, o utilizador apenas necessita de seleccionar o *software* mais apropriado para a sua resolução, verificando se existe uma *interface* para a linguagem de modelação em que o problema está codificado. De seguida, o NEOS verifica a consistência do *input*, executa o *solver* seleccionado e gera os resultados apropriados, devolvendo a solução ao utilizador. Esta, pode apenas ficar visível no monitor ou também pode ser encaminhada para o mail do utilizador, caso este o tenha fornecido no início do processo.

Foram seleccionados quatro *solvers* relativos à optimização não linear com restrições. Na Tabela 3.1 apresentam-se as características principais de cada um.

Solver	Características principais
KNITRO	Método de pontos interiores; Regras especiais para o ajustamento da região de confiança.
LANCELOT	Método de região de confiança adaptado; Técnica da Lagrangeana aumentada; Especialmente eficaz em problemas de grande escala.
LOQO	Método primal-dual de pontos interiores; Procura unidimensional para promover convergência global; Hessiana Exacta.
SNOPT	Método SQP de procura unidimensional; Função mérito Lagrangeana aumentada; Hessiana quasi-exacta.

Tabela 3.1: Principais características dos *solvers*

Apresentam-se, de seguida, os resultados relativamente ao problema do mercado de energia eléctrica. A Tabela 3.2, mostra os valores obtidos para a função objectivo e a variável de primeiro nível (proposta de preço).

Solver	Função lucro (π)	α^f
KNITRO	37.53	(35.83, 21.14, 20)
LANCELOT	37.53	(35.83, 40, 29.80)
LOQO	37.53	(35.83, 36.09, 20)
SNOPT	37.53	(35.83, 39.99, 0)

Tabela 3.2: Resultados da função objectivo e variável do primeiro nível

Curiosamente, apesar de ter sido alcançado um valor idêntico para todos os *solvers* em relação à função lucro, o mesmo não aconteceu relativamente à proposta de preço, induzindo a existência de vários pontos maximizantes locais.

Para não tornar a exposição exaustiva, são apresentados os resultados para as variáveis de segundo nível apenas relativas ao KNITRO. É de salientar que os resultados obtidos pelos outros *solvers* são muito semelhantes e da mesma ordem de grandeza.

alfa [*] :=

```

1  20
2  17.5
5  10
8  35.83
11 21.14
13 20;
```

QD [*] :=

```

2 44.9753      10 7.22345e-08   18 8.85027e-09   26 0
3  2.54964     12 3.43051e-09   19 8.40564e-09   29 0
4  6.87363     14 9.79465e-09   20 1.95494e-08   30 0;
5 41.0397      15 6.89599e-09   21 0
7  5.06664e-09 16 2.27821e-08   23 8.75919e-09
8 10.008       17 2.89901e-13   24 0
```

QS [*] :=

```

1 44.305
2 10.0936
5 41.0397
8 10.008
11 1.08121e-07
13 0;
```

T [*,*] (tr)

```

:      1      2      3      4      5      6      :=
2  34.8818      .      .      .      .      .
3   9.42328      .      .      .      .      .
4      .      4.33368e-08  6.87363      .      .      6.17154e-08
5      .      3.03646e-08      .      .      .      .
6      .      2.90953e-08      .      .      .      .
7      .      .      .      .      .      2.0116e-11      .
12     .      .      .      .      -2.72996e-16      .      .
28     .      .      .      .      .      .      0

:      7      8      9      10     11     12     :=
6  -1.37955e-10  0  3.9744e-08  0      .      .
9      .      .      .      .      1.08125e-07      .
10     .      .      7.20154e-08      .      .      .
13     .      .      .      .      .      .      0
14     .      .      .      .      .      .      3.28246e-09
15     .      .      .      .      .      .      -3.21718e-11
16     .      .      .      .      .      .      6.42464e-09
17     .      .      .      5.52149e-09      .      .
```

```

:      15      17      18      19      20      21 :=
10     .          .          .          .          1.66811e-10  0
14    5.70652e-09 .          .          .          .          .
16     .          1.26546e-08 .          .          .          .
18    2.38619e-09 .          .          .          .          .
19     .          .          6.09102e-09 .          .          .
20     .          .          .          4.98669e-09 .          .
23   -1.14278e-10 .          .          .          .          .

```

```

:  22  24  25  27  28  29   :=
8   .   .   .   .   0   .
10  0   .   .   .   .   .
21  0   .   .   .   .   .
22  .   0   .   .   .   .
23  .   0   .   .   .   .
25  .   0   .   0   .   .
26  .   .   0   .   .   .
27  .   .   .   .   .   0
28  .   .   .   0   .   .
30  .   .   .   0   .   0;

```

Existem determinados nós consumidores que praticamente não recebem energia elétrica. Isto poderá explicar-se pelo facto de não ser viável economicamente o transporte de energia para esses locais e também pela existência de grandes nós consumidores próximos dos geradores que absorvem toda a energia produzida.

No seguimento deste problema, e como trabalho futuro, seria interessante investigar, por um lado, o que aconteceria se a empresa B fosse a líder de mercado; por outro, como funcionaria este mesmo mercado, caso este problema tivesse sido codificado como um jogo de Nash, ou seja, tentar encontrar um equilíbrio, onde as duas empresas competissem ao mesmo nível. Por último, irá ser disponibilizado código do problema numa biblioteca da especialidade, para toda a comunidade científica.

A prossecução da pesquisa nesta área é de grande interesse, pois a formulação MPCC tem-se apresentado como uma importante resposta para problemas de outros ramos científicos.

Bibliografía

- [1] NEOS, webpage: www-neos.mcs.ang.gov/neos, 2005.
- [2] O. Alsac and B. Scott. Optimal load flow with steady-state security. *IEEE Transactions on Power Systems*, 93(3):745–751, 1973.
- [3] M. Anitescu. On solving mathematical programs with complementarity constraints as nonlinear programs. *SIAM Journal on Optimization*, 15(4):1203–1236, 2004.
- [4] A. S. Antunes and M. T. Monteiro. A SQP - filter algorithm with line search in nonlinear programming. In Departamento de Estadística e Investigación Operativa de la Universidad de Cádiz, editor, *XXVIII Congreso Nacional de Estadística e Investigación Operativa*, Cádiz, October 2004. ISBN 84-609-0438-5.
- [5] C. Ballard, D. Fullerton, J. B. Shoven, and J. Whalley. *A General Equilibrium Model for Tax Policy Evaluation*. National Bureau of Economic Research Monograph, Chicago, 1985.
- [6] H. Benson, A. Sen, D. Shano, and R. Vanderbei. Interior-point algorithms penalty methods and equilibrium problems. Technical Report Technical Report ORFE-03-02, Operations Research and Financial Engineering Princeton University, October 2003.
- [7] M. Boots and M. Schweepers. Modelling concentration on the electricity market. In *6th IAEE European Conference 2004: Modelling in Energy Economics and Policy*, 2004.
- [8] J. C. Carbone, C. Helm, and T. F. Rutherford. Coalition formation and international trade in greenhouse gas emission rights. October 2003.
- [9] A.-V. Demiguel, M. P. Friedlander, F. J. Nogales, and S. Scholtes. An interior-point method for MPECs based on strictly feasible relaxations. Technical Report Contract W-31-109-Eng-38, Argonne National Laboratory, 2004.
- [10] S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52(3):333–359, June 2003.
- [11] S. P. Dirkse. *Robust Solution of Mixed Complementarity Problems*. PhD thesis, Computer Sciences Department, University of Wisconsin, 1994.
- [12] S. P. Dirkse and M. C. Ferris. MCPLIB: A collection of nonlinear mixed complementarity problems. *Optimization Methods and Software*, 5:319–345, 1995.
- [13] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, Ser. A 91:201–213, 2002.
- [14] EC-2002. Second benchmarking report on the implementation of the internal electricity and gas markets. Technical Report 2, Commission Staff Working paper SEC, October 2002.
- [15] EC-2003. Third benchmarking report on the implementation of the internal electricity and gas markets. Technical Report 1, Commission draft Staff paper, March 2004.

- [16] A. Ehrenmann and K. Neuhoff. A comparison of electricity market designs in networks. Technical Report CMI Working Paper 31, Department of Applied Economics, University of Cambridge, August 2003.
- [17] A. Ehrenmann and Y. Smeers. The investment game in a meshed power system. February, 20005.
- [18] H. Ehtamo and T. Raivio. On applied nonlinear and bilevel programming for pursuit-evasion games. *Journal of Optimization Theory and Applications*, 108:65–96, 2001.
- [19] F. Facchinei, H. Jiang, and L. Qi. A smoothing method for mathematical programs with equilibrium constraints. Technical Report AMR 96/15, Applied Mathematics Report, University of New South Wales, 1996.
- [20] M. C. Ferris, S. P. Dirkse, and A. Meeraus. Mathematical programs with equilibrium constraints: Automatic reformulation and solution via constrained optimization. Technical Report 02/11, Oxford University Computing Laboratory, July 2002.
- [21] M. C. Ferris and J. S. Pang. Engineering and economic applications of complementarity problems. *SIAM Rev.*, 39(4):669–713, December 1997.
- [22] M. C. Ferris and F. Tin-Loi. On the solution of a minimum weight elastoplastic problem involving displacement and complementarity constraints. *Computer Methods in Applied Mechanics and Engineering*, 174:107–120, 1999.
- [23] M. F. Ferris and F. Tin-Loi. Limit analysis of frictional block assemblies as a mathematical program with complementarity constraints. *International Journal of Mechanical Sciences*, 43:209–224, 2001.
- [24] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Math. Programming*, 91:239–270, 2002.
- [25] R. Fletcher and S. Leyffer. Solving mathematical programs with complementarity constraints as nonlinear programs. *Optimization Methods and Software*, 19(1):15–40, February 2004.
- [26] R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes. Local convergence of SQP methods for Mathematical Programs with Equilibrium Constraints. Technical Report Numerical Analysis Report NA/209, Department of Mathematics, University of Dundee, 2002.
- [27] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modelling Language for Mathematical Programming*. Duxburg Press, Massachusetts, 1993.
- [28] M. Fukushima, Z. Q. Luo, and J. S. Pang. A globally convergent sequential quadratic programming for mathematical programs with linear complementarity constraints. *Computational Optimization and Applications*, Kluwer Academic Publishers, 10:5–34, 1998.
- [29] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. User’s guide for npsol 5.0: A fortran package for nonlinear programming. Technical Report SOL 86-1, July 1998.
- [30] R. Green and D. Newbery. Competition in the british electric spot market. *J. Poli. Econ.*, 100:929–953, 1992.
- [31] X. Gu, J. Renaud, L. Ashe, S. Batill, A. Budhiraja, and L. Krajewski. Decision-based collaborative optimization under uncertainty. Technical Report DETC2000/DAC-14297, 2000.
- [32] B. F. Hobbs and K. A. Kelly. Using game theory to analyse electric transmission pricing policies in the US. *Euro. J. Oper. Res.*, 56(2):154–171, 1992.
- [33] B. F. Hobbs, C. B. Metzler, and J.-S. Pang. Strategic gaming analysis for electric power systems: An MPEC approach. *IEEE Transactions on Power Systems*, 15(2):638–645, May 2000.

- [34] B. F. Hobbs and F. A. M. Rijkers. Strategic generation with conjectured transmission price responses in a mixed transmission pricing system. *IEEE Transactions on Power Systems*, 19(2):707–879, May 2004.
- [35] B. F. Hobbs and F. A. M. Rijkers. The more cooperation, the more competition? a cournot analysis of the benefits of electric market coupling. Technical Report CMI Working Paper 73, University of Cambridge, January 2005.
- [36] X. M. Hu and D. Ralph. Convergence of a Penalty Method for Mathematical Programming with Complementarity Constraints. *Journal of Optimization Theory and Applications*, 123(2):365–390, 2004.
- [37] H. Jiang and D. Ralph. Smooth SQP methods for mathematical programs with nonlinear complementarity constraints. *SIAM J. Optimization*, 10(3):779–808, 2000.
- [38] M. Kocvara and J. V. Outrata. Optimization problems with equilibrium constraints and their numerical solution. *Math. Program., Ser. B* 101:119–149, 2004.
- [39] V. Krishna and V. C. Ramesh. Intelligent agents in negotiations in market games, part 2: Application. *IEEE Transactions on Power Systems*, 13(3):1109–1114, Aug 1998.
- [40] K. W. Kuhn and A. W. Tucker. Nonlinear programming. In University of California Press, editor, *Proc. 2nd Berkeley Symposium on Mathematical Statistics and Probabilistics*, pages 481–492, Berkeley, 1951.
- [41] D. Lavigne, R. Loulou, and G. Savard. Pure competition, regulated and stackelberg equilibria: application to the energy system of quebec. *European Journal of Operational Research*, 125:1–17, 2000.
- [42] S. Leyffer. MacMPEC, webpage: www.mcs.nl.gov/leyffer/MacMPEC, 2000.
- [43] S. Leyffer. Complementarity constraints as nonlinear equations: Theory and numerical experience. Technical Report Preprint ANL / MCS-P1054-0603, Mathematics and Computer Science Division, Argonne National Laboratory, June 2003.
- [44] S. Leyffer. The penalty interior point method fails to converge. *Optimization Methods and Software*, 20(4-5):559–568, 2005.
- [45] X. Liu and J. Sun. Generalized stationary points and an interior-point method for mathematical programs with equilibrium constraints. *Mathematical Programming*, 101:231–261, 2004.
- [46] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, Cambridge, 1996.
- [47] C. B. Metzler. Complementarity models of competitive oligopolistic electric power generation markets. Master’s thesis, The Johns Hopkins University, 2000.
- [48] O. Drissi-Kaitouni and J. T. Lundgren. Bilivel origin-destination matrix estimation using a descent approach. Technical Report LITH-MAT-R-1992-49, Linköping Institute of Technology, Department of Mathematics, Sweden, 1992.
- [49] K. P. Oh and P. K. Goenka. The elastohydrodynamic solution of journal bearings under dynamic loading. *Trans. ASME*, 107:389–395, 1985. *J. Tribology*.
- [50] IEEE International Conference on Robotics and Automation, editors. *Optimal Control of Multiple Robot Systems with Friction Using MPC*, New Orleans - LA, April 2004.
- [51] J. V. Outrata. On necessary optimal conditions for stackelberg problems. *Journal Optim. Theory Appl.*, 76:305–320, 1993.

- [52] J. S. Pang. Complementarity problems. In Kluwer Academic Publishers, editor, *Handbook in Global Optimization*. R. Horst and P. Pardalos, Boston, 1994.
- [53] H. Pieper. *Algorithms for Mathematical Programs with Equilibrium Constraints with Applications to Deregulated Electricity Markets*. PhD thesis, Stanford University, 2001.
- [54] A. U. Raghunathan and L. T. Biegler. Mathematical programs with equilibrium constraints (MPECs) in process engineering. Technical report, CMU Chemical Engineering, November 2002.
- [55] J. F. Rodrigues. *Obstacle Problems in Mathematics Physics*. Elsevier Publishing Company, Amsterdam, 1987.
- [56] H. Scheel and S. Scholtes. Mathematical programs with complementarity constraints: Stationarity, optimality and sensitivity. *Mathematics of Operations Research*, 25:1–22, 2000.
- [57] F. M. Scherer and D. Ross. *Industrial Markets Structure and Economic Performance*. Houghton-Mifflin, Boston, 3rd edition, 1990.
- [58] S. Scholtes. Convergence properties of a regularization scheme for mathematical programs with complementarity constraints. *SIAM Journal Optimization*, 11(4):918–936, 2001.
- [59] F. C. Schweppe, M. C. Caramanis, R. E. Tabors, and R. E. Bohn. *Spot Pricing of Electricity*. Kluwer, Norwell, 1998.
- [60] H. D. Sherali, A. L. Soyster, and F. H. Murphy. Stackelberg-nash-cournot equilibria: Characterizations and computations. *Operations Research*, 31:253–276, 1983.
- [61] Y. Smeers. How well can one measure market power in restructured electricity systems? 2004.
- [62] T. Srinivasav and J. Whalley. *General Equilibrium Trade Policy Modelling*. MIT Press, 1986.
- [63] H. Van Stackelberg. *The Theory of Market Economy*. Oxford University Press, Oxford, 1952.
- [64] M. Stöhr. *Nonsmooth Trust Region Methods and their Applications to Mathematical Programs with Equilibrium Constraints*. Shaker Verlag, Aachen, 2000.
- [65] R. L. Tobin. A variable dimension solution approach for the general spatial equilibrium problem. *Math. Programming*, 40:33–51, 1988.
- [66] R. L. Tobin. Uniqueness results and algorithm for Stackelberg-Cournot-Nash equilibria. *Ann. Oper. Research*, 34:21–36, 1992.
- [67] R. J. Vanderbei. LOQO user’s manual, version 4.05. Technical Report ORFE-99, October Operations Research and Financial Engineering, Princeton University.
- [68] M. Ventosa, R. Denis, and C. Redondo. Expansion planning in electricity markets. two different approaches. In *14th PSCC*, Sevilla, 2002. 14th PSCC, Sevilla, 24-28 June.
- [69] L. N. Vicente and P. H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5:291–306, 1994.
- [70] C. D. Wood and B. F. Wollenberg. *Power Generation, Operational and Control*. J. Wiley, New York, 2nd edition, 1995.

Apêndices

Apêndice A

Classificação dos problemas teste

Problema	Classificação	Problema	Classificação	Problema	Classificação
bar-truss-3	LQR-MY-NCP-29-22-6	ex9.2.6	LLR-AY-NLP-16-12-6	nash1	QLR-AY-LCP-6-2-2
bard1	QLR-AY-NLP-5-1-3	ex9.2.7	LLR-AY-NLP-10-9-4	outrata31	QUR-AN-NCP-5-0-4
bard2	QLR-AY-NLP-12-5-4	ex9.2.8	LLR-AY-NLP-6-5-2	outrata32	QUR-AN-NCP-5-0-4
bard3	QLR-AY-NLP-6-3-2	ex9.2.9	LLR-AY-NLP-9-8-3	outrata33	QUR-AN-NCP-5-0-4
bard1m	QLR-AY-NLP-5-1-4	fp2	QLR-AN-LCP-4-2-2	outrata34	QUR-AN-NCP-5-0-4
bard3m	QLR-AY-NLP-6-1-4	fp4-1	QLR-AN-LCP-80-60-30	portfl-i-1	QLR-AY-NLP-87-25-12
bilevel1	LLR-AY-LCP-9-3-6	fp4-2	QLR-AN-LCP-110-110-60	portfl-i-2	QLR-AY-NLP-87-25-12
bilevel2	QLR-AY-LCP-20-5-12	fp4-3	QLR-AN-LCP-140-170-70	portfl-i-3	QLR-AY-NLP-87-25-12
bilevel3	QLR-AY-LCP-12-7-4	gauvin	QOR-AY-LCP-13-4-8	portfl-i-4	QLR-AY-NLP-87-25-12
bilin	LLR-AY-NLP-8-1-6	gnash10	QOR-AY-LCP-13-4-8	portfl-i-6	QLR-AY-NLP-87-25-12
dempe	QQR-AY-LCP-4-2-2	gnash11	QOR-AY-LCP-13-4-8	qpec-100-1	QLR-ANLCP-105-102-100
design-cent-1	QOR-AY-NLP-12-9-3	gnash12	QOR-AY-LCP-13-4-8	qpec-100-2	QLR-ANLCP-110-102-100
design-cent-2	QOR-MY-NLP-13-13-3	gnash13	QOR-AY-LCP-13-4-8	qpec-100-3	QLR-ANLCP-110-104-100
design-cent-4	QOR-AY-NLP-22-9-12	gnash14	QOR-AY-LCP-13-4-8	qpec-100-4	QLR-ANLCP-120-104-100
desilva	QQR-AY-NCP-6-2-2	gnash15	QOR-AY-LCP-13-4-8	qpec1	QUR-AN-LCP-30-0-20
df1	QQR-AY-NCP-2-2-1	gnash16	QOR-AY-LCP-13-4-8	qpec2	QUR-AN-LCP-30-0-20
ex9.1.1	LLR-AY-NLP-13-12-5	gnash17	QOR-AY-LCP-13-4-8	ralph1	LUR-AN-LCP-2-0-1
ex9.1.2	LLR-AY-NLP-10-9-4	gnash18	QOR-AY-LCP-13-4-8	ralph2	LUR-AN-LCP-2-0-1
ex9.1.3	LLR-AY-NLP-23-21-6	gnash19	QOR-AY-LCP-13-4-8	ralphmod	QUR-AN-LCP-104-0-100
ex9.1.4	LLR-AY-NLP-10-9-4	hakonsen	QOR-MY-NCP-9-8-4	scholtes1	QLR-AY-NCP-3-1-1
ex9.1.5	LLR-AY-NLP-13-12-5	hs044-i	QLR-MY-NLP-20-14-10	scholtes2	QLR-AY-NCP-3-1-1
ex9.1.6	LLR-AY-NLP-14-13-6	incid-set1c-8	LOR-MY-LCP-117-119-49	scholtes3	QUR-AN-LCP-2-0-2
ex9.1.7	LLR-AY-NLP-17-15-6	incid-set1c-8	LOR-MY-LCP-117-126-49	scholtes4	LLR-AY-LCP-3-2-2
ex9.1.8	LLR-AY-NLP-14-12-5	incid-set2-8	LOR-MY-LCP-117-119-49	scholtes5	QUR-AN-LCP-3-2-2
ex9.1.9	LLR-AY-NLP-12-11-5	incid-set2c-8	LOR-MY-LCP-117-126-49	sil	QLR-AY-LCP-8-2-3
ex9.1.10	LLR-AY-NLP-14-12-5	jr1	QLR-AN-LCP-2-1-1	stackelberg1	QLR-AY-LCP-3-1-1
ex9.2.1	LLR-AY-NLP-10-9-4	jr2	QLR-AN-LCP-2-1-1	tap-09	QOR-AY-NCP-86-68-32
ex9.2.2	LLR-AY-NLP-10-11-4	kth1	LLR-AN-LCP-2-0-1	tap-15	QOR-AY-NCP-194-167-83
ex9.2.3	LLR-AY-NLP-16-16-6	kth2	QLR-AN-LCP-2-0-1	water-net	QOR-MY-LCP-52-22-13
ex9.2.4	LLR-AY-NLP-8-7-2	kth3	QLR-AN-LCP-2-0-1	water-FL	QOR-MY-LCP-169-72-33
ex9.2.5	LLR-AY-NLP-8-7-3	liswet1-050	QLR-AN-NLP-152-103-50		

Os problemas teste estão classificados segundo onze parâmetros:

XXX-XX-XXX-n-m-p

O primeiro grupo de três caracteres descreve o aspecto das funções objectivo e das restrições:

Car.	Funç. Objectivo	Car.	Funç. restrição	Car.	Regularidade
N	Não definida	U	Sem restrições	R	Regular
C	Constante	X	As restrições são variáveis fixas	I	Irregular
L	Linear	B	As restrições são variáveis limitadas		
Q	Quadrática	N	Restrições de rede		
S	Soma dos quadrados	L	Linear		
O	Nenhuma das anteriores	O	Nenhuma das anteriores		

O segundo grupo diz respeito à origem do modelo e pronuncia-se se existe relação entre as variáveis de estado e as de controlo:

Car.	Origem do problema	Car.	Ligações das variáveis
A	Académico	Y	Sim
M	Modelado	N	Não
R	Real		

O grupo seguinte de caracteres refere-se ao tipo de restrições de complementaridade existentes:

Car.	Restrição de complementaridade
MCP	Problema de Complementaridade Misto
LMCP	Problema de Complementaridade Misto Linear
NCP	Problema de Complementaridade Não Linear
LCP	Problema de Complementaridade Linear
NLP	Condições de optimalidade de um programa não linear

O grupo final é composto por um conjunto de números inteiros, que representam a dimensão do problema:

Car.	Dimensão
n	Número de variáveis (de controlo e de estado)
m	Número de restrições (excluindo as de complementaridade)
p	Número de restrições de complementaridade

Apêndice B

Análise detalhada dos resultados

Problema	Tempo CPU (segundos)				N. iterações			
	Loqo	Matlab	Npsol	Filter SQP	Loqo	Matlab	Npsol	Filter SQP
bar-truss-3	0	0.532	0.031	0.015	24	12	7	9
bard1	0	0.438	0	0.031	17	3	6	3
bard2	0.187	0.093	0	0	1000	3	5	2
bard3	0.015	0.046	0	0	54	7	1	2
bard1m	0	92.14	0	0	17	6	6	3
bard3m	0	0.953	0	0.015	24	9	1	52
bilevel1	0	0.11	0	0	35	4	3	2
bilevel2	0.015	31.235	0.015	0	23	1000	5	4
bilevel3	0	0.125	0	0.031	23	12	13	93
bilin	0	0.062	0	0.031	25	4	1	104
dempe	0.015	0.015	0	0.015	267	2	50	2
design-cent-1	0.031	0.141	0	0.234	122	10	29	1000
design-cent-2	0	0.219	0	0	20	12	11	1
design-cent-4	0.218	0.047	0	0	1000	2	6	2
desilva	0	0.032	0	0	32	2	2	19
df1	0	0	0	0	20	1	2	2
ex9.1.1	0.015	0.047	0	0	24	3	3	1
ex9.1.2	0	0.031	0	0.015	16	3	2	1
ex9.1.3	0.015	0.359	0	0.015	43	11	5	1
ex9.1.4	0.015	0.11	0	0	73	11	1	1
ex9.1.5	0	0.031	0	0	15	3	1	1
ex9.1.6	0	0.031	0	0	20	5	1	1
ex9.1.7	0.015	0.25	0	0	33	11	1	1
ex9.1.8	0	0.063	0	0	24	5	1	1
ex9.1.9	0.015	0.047	0.015	0	18	3	2	1
ex9.1.10	0	0.109	0	0	24	5	1	1
ex9.2.1	0	0.032	0	0	14	4	1	1
ex9.2.2	0.093	0.266	0	0.047	1000	33	25	1
ex9.2.3	0.015	0.172	0	0	42	7	1	1
ex9.2.4	0	0.032	0	0	11	5	5	1
ex9.2.5	0	0.125	0.015	0	14	15	5	1
ex9.2.6	0.015	0.078	0	0	21	4	1	1
ex9.2.7	0	0.047	0	0	14	4	1	1
ex9.2.8	0	0.016	0	0	12	3	1	1
ex9.2.9	0	0.031	0	0	14	3	1	1

Problema	Tempo CPU (segundos)				N. iterações			
	Loqo	Matlab	Npsol	Filter SQP	Loqo	Matlab	Npsol	Filter SQP
flp2	0	0.046	0	0	13	10	5	2
flp4-1	0.25	0.046	0.05	0.093	36	1	1	4
flp4-2	0.265	1.39	0.016	0.156	330	1	1	3
flp4-3	1.812	2.516	0.056	1.313	51	1	1	5
gauvin	0	0.031	0	0	14	5	3	8
gnash10	0.015	18.141	0.015	0	30	1000	10	10
gnash11	0.015	0.391	0	0	25	19	10	6
gnash12	0	18.547	0	0	20	1000	9	6
gnash13	0.015	bloq	0	0	21	bloq	10	6
gnash14	0	0.391	0	0	26	24	10	7
gnash15	0.015	15.781	0	0	83	1000	10	12
gnash16	bloq	19.531	0	0	bloq	1000	10	8
gnash17	0.015	17.156	0	0	47	1000	10	6
gnash18	0.015	7.906	0	0	38	1000	14	9
gnash19	0.015	14.562	0	0	45	1000	10	14
hakonsen	0	0.469	0	0.064	25	10	54	1
hs044-i	0.015	0.265	0	0.015	29	6	9	9
incid-set1-8	0.203	64.86	0.593	0.578	24	45	18	5
incid-set1c-8	0.171	62.156	0.562	0.547	22	36	14	5
incid-set2-8	0.265	132.312	23.156	151.297	31	25	1000	880
incid-set2c-8	0.281	469.562	22.375	107.094	28	26	1000	362
jr1	0	0.015	0	0	9	1	2	2
jr2	0.015	0.016	0	0	11	1	9	2
kth1	0	0.015	0	0	9	2	1	1
kth2	0	0	0	0	11	2	3	3
kth3	0.015	0.016	0	0	10	4	5	1
liswet1-050	0.031	12.609	0	0	35	7	6	2
nash1	0	0.063	0.14	0.546	31	7	12	2
outrata31	0.078	0.016	0	0	1000	1	88	14
outrata32	0.015	0.016	0	0.015	137	1	16	8
outrata33	0	0.016	0	0	44	1	88	12
outrata34	0	0.032	0	0	37	1	62	36
portfl-i-1	0.031	bloq	0.078	0.171	40	bloq	16	6
portfl-i-2	0.015	1.859	0.046	0.218	33	6	12	8
portfl-i-3	0.015	bloq	0.109	0.156	35	bloq	18	6
portfl-i-4	0.015	bloq	0.062	0.328	37	bloq	12	13
portfl-i-6	0.015	bloq	0.062	0.218	41	bloq	13	8
qpec-100-1	2.265	39.063	0.424	107.547	55	28	30	1000
qpec-100-2	0.859	bloq	0.609	127.062	22	bloq	24	1000
qpec-100-3	3.14	bloq	0.984	219.782	64	bloq	43	1000
qpec-100-4	2.703	384.37	0.89	24.125	70	34	33	109
qpec1	0.015	0.078	0.031	0.015	13	2	63	3
qpec2	0.171	0.469	0.015	0.828	1000	25	2	1000
ralph1	0.015	0.297	0	0.62	258	1	31	1000
ralph2	0	0.015	0	0	21	11	20	2
ralphmod	46.968	2725.1	3.375	35.985	1000	834	103	143
scholtes1	0.015	0.094	0	0	1000	3	6	4
scholtes2	0	0.047	0	0	104	9	14	4
scholtes3	0.015	0.015	0	0.046	29	4	20	100
scholtes4	0.031	0.016	0	0.859	731	2	28	1000
scholtes5	0	0.015	0	0	10	6	3	3
sll	0.125	0.032	0	0.047	1000	2	4	2
stackelberg1	0	0.031	0	0	17	5	7	5
tap-09	1.75	64.833	0.062	39.344	1000	6	6	1000
tap-15	5.218	bloq	71.641	330.953	1000	bloq	1000	162
water-net	0.014	bloq	0.062	7.828	289	bloq	20	1000
water-FL	2.421	bloq	39.953	239.812	1000	bloq	1000	1000

Problema	Cálculos da Função				Cálculos do Gradiente			
	Loço	Matlab	Npsol	Filter SQP	Loço	Matlab	Npsol	Filter SQP
bar-truss-3	47	491	25	10	47	491	25	24
bard1	34	27	8	4	34	27	8	4
bard2	11512	55	6	3	11521	55	6	3
bard3	108	63	2	3	108	63	2	3
bard1m	33	56	8	4	33	56	8	4
bard3m	47	79	2	212	47	79	2	56
bilevel1	69	59	7	3	69	59	7	3
bilevel2	45	35821	8	5	45	35821	8	5
bilevel3	45	181	13	673	45	181	13	94
bilin	49	49	2	685	49	49	2	105
dempe	533	14	173	22	533	14	173	3
design-cent-1	940	153	61	19806	940	153	61	1001
design-cent-2	39	268	14	4	39	268	14	2
design-cent-4	2946	71	7	3	2946	71	7	3
desilva	63	23	4	27	63	23	4	20
dfl	39	8	4	3	39	8	4	3
ex9.1.1	47	58	4	2	47	58	4	2
ex9.1.2	31	47	3	2	31	47	3	2
ex9.1.3	85	291	6	2	85	291	6	2
ex9.1.4	148	140	5	2	148	140	5	2
ex9.1.5	29	59	2	2	29	59	2	2
ex9.1.6	39	91	1	2	39	91	1	2
ex9.1.7	65	219	2	2	65	219	2	2
ex9.1.8	47	94	4	2	47	94	4	2
ex9.1.9	35	55	3	2	35	55	3	2
ex9.1.10	47	94	4	2	47	94	4	2
ex9.2.1	27	72	1	2	27	72	1	2
ex9.2.2	11321	396	39	2	11321	396	39	2
ex9.2.3	83	184	4	2	83	184	4	2
ex9.2.4	21	59	7	2	21	59	7	2
ex9.2.5	27	153	9	2	27	153	9	2
ex9.2.6	41	89	1	2	41	89	1	2
ex9.2.7	27	72	1	2	27	72	1	2
ex9.2.8	23	31	1	2	23	31	1	2
ex9.2.9	27	42	1	2	27	42	1	2
fp2	25	65	7	3	25	65	7	3
fp4-1	71	163	2	19	71	163	2	5
fp4-2	3292	223	2	4	3292	223	2	4
fp4-3	101	283	2	26	101	283	2	6
gauvin	27	29	6	9	27	29	6	9
gnash10	59	29024	13	17	59	29024	13	11
gnash11	49	365	13	7	49	365	13	7
gnash12	39	28889	11	7	39	28889	11	7
gnash13	41	bloq	13	7	41	bloq	13	7
gnash14	51	366	13	8	51	366	13	8
gnash15	165	28799	16	16	165	28799	16	13

Problema	Cálculos da Função				Cálculos do Gradiente			
	Loqo	Matlab	Npsol	Filter SQP	Loqo	Matlab	Npsol	Filter SQP
gnash16	bloq	28811	13	12	bloq	28811	13	9
gnash17	95	28949	13	7	95	28949	13	7
gnash18	75	28825	21	14	75	28825	21	10
gnash19	89	28576	13	15	89	28576	13	15
hakonsen	52	97	143	3	52	97	143	2
hs044-i	57	155	16	11	57	155	16	10
incid-set1-8	48	5493	25	6	48	5493	25	6
incid-set1c-8	43	4485	21	6	43	4485	21	6
incid-set2-8	62	3107	2948	16643	62	3107	2948	881
incid-set2c-8	55	3229	2953	1539	55	3229	2953	363
jr1	17	7	4	3	17	7	4	3
jr2	21	7	14	3	21	7	14	3
kth1	17	11	2	2	17	11	2	2
kth2	21	11	2	4	21	11	2	4
kth3	19	19	5	2	19	19	5	2
liswet1-050	69	1252	8	3	69	1252	8	3
nash1	72	63	8	3	72	63	9	3
outrata31	2011	13	16	17	2011	13	16	15
outrata32	296	13	272	16	296	13	272	9
outrata33	87	13	19	13	87	13	19	13
outrata34	74	13	270	39	74	13	270	37
portfl-i-1	79	bloq	17	25	79	bloq	17	7
portfl-i-2	65	622	13	38	65	622	13	9
portfl-i-3	69	bloq	21	24	69	bloq	21	7
portfl-i-4	75	bloq	14	48	75	bloq	14	14
portfl-i-6	81	bloq	14	22	81	bloq	14	9
qpec-100-1	109	3102	31	8918	109	3102	31	1001
qpec-100-2	43	bloq	26	8163	43	bloq	26	1001
qpec-100-3	127	bloq	44	17355	127	bloq	44	1001
qpec-100-4	241	4347	34	367	241	4347	34	110
qpec1	25	95	102	4	25	95	102	4
qpec2	4041	851	4	19830	4041	851	4	1001
ralph1	525	7	48	1001	525	7	48	1001
ralph2	52	47	21	3	52	47	21	3
ralphmod	2001	100045	148	1451	2001	10045	148	144
scholtes1	2001	19	11	5	2001	19	11	5
scholtes2	227	68	34	5	227	68	34	5
scholtes3	57	19	32	20001	57	19	32	1001
scholtes4	4659	14	41	1680	4659	14	41	1001
scholtes5	19	34	4	4	19	34	4	4
sll	11332	29	14	3	11332	29	14	3
stackelberg1	33	29	8	6	33	29	8	6
tap-09	4969	615	8	19967	4969	615	8	1001
tap-15	2001	bloq	3008	3225	2001	bloq	3008	163
water-net	580	bloq	30	19841	580	bloq	30	1001
water-FL	7502	bloq	2882	19701	7502	bloq	2882	1001

Apêndice C

Dados do problema

Firma	Nó	a_i	b_i	Capacidade máxima
Firma B	1	20	0.0375	200
	2	17.5	0.1750	80
	5	10	0.6250	50
Firma A	8	32.5	0.0834	35
	11	30	0.25	30
	13	30	0.25	40

Tabela C.1: Dados relativos aos nós de oferta (produtores)

Nó	c_i	d_i
2	40	0.461
3	40	4.167
4	40	1.316
5	40	0.106
7	40	0.439
8	40	0.333
10	40	1.724
12	40	0.893
14	40	1.613
15	40	1.220
16	40	2.857
17	40	1.111
18	40	3.125
19	40	1.053
20	40	4.545
21	40	0.571
23	40	3.125
24	40	1.149
26	40	2.857
29	40	4.167
30	40	0.943

Tabela C.2: Dados relativos aos nós de procura (consumidores)

Arco	Trans. Máxima	Reactância	Arco	Trans. Máxima	Reactância
1-3	78	0.1852	22-21	19.2	0.0236
3-4	78	0.0379	21-10	19.2	0.0749
1-2	78	0.0575	22-10	19.2	0.1499
2-5	78	0.1983	11-9	39	0.2080
2-4	39	0.1737	9-6	39	0.2080
2-6	39	0.1763	9-10	39	0.1100
4-12	39	0.2560	10-6	39	0.5560
12-13	39	0.1400	10-17	19.2	0.0845
12-14	19.2	0.2559	20-10	19.2	0.2090
15-14	9.6	0.1997	8-6	19.2	0.0420
12-15	19.2	0.1304	6-28	19.2	0.0599
12-16	19.2	0.1987	7-6	78	0.0820
15-23	9.6	0.2020	5-7	42	0.1160
16-18	9.6	0.2185	28-8	19.2	0.2000
18-19	9.6	0.1292	27-28	39.2	0.3960
19-20	19.2	0.0680	27-30	9.6	0.6027
24-23	9.6	0.2700	29-30	9.6	0.4533
25-26	9.6	0.3800	29-27	9.6	0.4153
24-25	9.6	0.3292	17-16	9.6	0.1932
27-25	9.6	0.2087	6-4	54	0.0414
24-22	9.6	0.1790			

Tabela C.3: Dados relativos aos arcos

Ciclo	Nós associados
1	2 - 6 - 4
2	6 - 28 - 8
3	6 - 9 - 10
4	10 - 22 - 21
5	12 - 14 - 15
6	27 - 30 - 29
7	1 - 2 - 4 - 3
8	2 - 5 - 7 - 6
9	4 - 6 - 10 - 17 - 16 - 12
10	6 - 28 - 27 - 25 - 24 - 22 - 10
11	10 - 20 - 19 - 18 - 15 - 12 - 16 - 17
12	10 - 22 - 24 - 23 - 15 - 18 - 19 - 20

Tabela C.4: Dados relativos aos ciclos considerados

Apêndice D

Código do problema

```
#####
# electricpower.mod
# October/2005
#
# This problem is described in "Strategic Gaming Analysis
# for Electric Power Systems: an MPEC Approach",
# by Benjamin F. Hobbs, Carolyn Metzler and Jong Shi-Pang
#
# coded by Helena Sofia Rodrigues
#####

# SETS -----
set N; # set of all nodes
set A in N cross N; # set of all arcs
set Sf; # set of nodes with generators under control firm A
set P; # set of all generator nodes
set D; # set of all demand nodes
set L; # set of Kirchhoff's voltage loops

# PARAMETERS -----
param a{P}; # intercept of supply function
param b{P}; # slope of supply function
param c{D}; # intercept of demand function
param d{D}; # slope of demand function
param alfa_s{P}; # upper bound of the bid
param alfa_i{P}; # lower bound of the bid
param QS_s{P}; # upper bound of production capacity
param Ts{A}; # maximum transmission capacity on arc ij
param RR{L, A} default 0; # matrix of signed reactance coefficients
param RRT{A, L} default 0;
param delta{N, A} default 0; # electric network
param deltaT{A, N} default 0;

# VARIABLES -----
# First-level decision variables of firm f .....
var alfa{P}; # bid for the unit at node in P

# Primal variables in 2nd-level SPE/OPF .....
var QS{P}; # quantity of power generated by the unit
var QD{D}; # quantity of power demanded
var T{A}; # MW transmitted from i to j

# Dual variables in 2nd-level SPE/OPF .....
var lambda{N}; # marginal cost at node i
var miu{P}; # marginal value of generation capacity for the unit at node i
var teta{A}; # marginal value of transmission capacity
var gamma{L}; # shadow price for Kirchhoff voltage law

# FUNCTION TO MAXIMIZE -----
maximize profit:
sum{n in D} (c[n]*QD[n] - d[n]*QD[n]^2) - sum{k in Sf} (a[k]*QS[k] + (b[k]/2)*(QS[k]^2)) -
sum{(i,j) in A} (teta[i,j]*Ts[i,j]) -
sum{l in P} (if 1 in Sf then 0 else (miu[1]*QS_s[1] + a[1]*QS[1] + b[1]*QS[1]^2));

# CONSTRAINTS -----
subject to
```

```

r1 {n in P}:   alfa_i[n] <= alfa[n] <= alfa_s[n];

#r2{n in P}:   0<= QS_s[n]-QS[n] complements miu[n] >=0;
r2a{n in P}:   QS_s[n]-QS[n] >=0;
r2b{n in P}:   miu[n] >=0;
r2c{n in P}:   (QS_s[n]-QS[n])*(miu[n]) <=0;

#r3{n in P}:   0<= QS[n] complements alfa[n] - lambda[n]+miu[n]+b[n]*QS[n] >=0;
r3a{n in P}:   QS[n] >= 0;
r3b{n in P}:   alfa[n] - lambda[n]+miu[n]+b[n]*QS[n] >=0;
r3c{n in P}:   (QS[n])*(alfa[n] - lambda[n]+miu[n]+b[n]*QS[n]) <= 0;

#r4{n in D}:   0<= QD[n] complements lambda[n] - c[n] + d[n]*QD[n] >= 0;
r4a{n in D}:   QD[n] >= 0;
r4b{n in D}:   lambda[n] - c[n] + d[n]*QD[n] >= 0;
r4c{n in D}:   QD[n]*(lambda[n] - c[n] + d[n]*QD[n]) <= 0;

#r5{(i,j) in A}: 0<= teta[i,j] complements Ts[i,j] - T[i,j] >= 0;
r5a{(i,j) in A}: teta[i,j] >= 0;
r5b{(i,j) in A}: Ts[i,j]-T[i,j] >= 0;
r5c{(i,j) in A}: (teta[i,j])*( Ts[i,j] - T[i,j]) <= 0;

#r6{(i,j) in A}: 0<= T[i,j] complements sum{n in N}(deltaT[i,j,n]*lambda[n])+teta[i,j]+sum {k in L} (RRT[i,j,k]*gamma[k]) >= 0;
r6a{(i,j) in A}: T[i,j] >= 0;
r6b{(i,j) in A}: sum{n in N}(deltaT[i,j,n]*lambda[n])+teta[i,j]+sum {k in L} (RRT[i,j,k]*gamma[k]) >=0;
r6c{(i,j) in A}: T[i,j]*(sum{n in N}(deltaT[i,j,n]*lambda[n])+teta[i,j]+sum {k in L} (RRT[i,j,k]*gamma[k])) <= 0;

r7 {n in N}:
  (if n in D then QD[n] else 0)-(if n in P then QS[n] else 0)+sum {(i,j) in A} (delta[n,i,j]*T[i,j])=0;

r8 {k in L}:
  sum{(i,j) in A} (RR[k,i,j]*T[i,j])=0;

data;
-----
set N := 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30;
set A := (1,2) (1,3)
          (2,4) (2,5) (2,6)
          (3,4)
          (6,4) (4,12)
          (5,7)
          (7,6) (8,6) (9,6) (10,6) (6,28)
          (28,8)
          (9,10)
          (10,17) (20,10) (21,10) (22,10)
          (11,9)
          (12,13) (12,14) (12,15) (12,16)
          (15,14)
          (15,18) (15,23)
          (17,16)
          (18,19)
          (19,20)
          (22,21)
          (24,22)
          (24,23)
          (24,25)
          (25,26) (27,25)
          (29,27) (27,30)
          (27,28)
          (29,30);

set Sf := 8 11 13;
set P := 1 2 5 8 11 13;
set D := 2 3 4 5 7 8 10 12 14 15 16 17 18 19 20 21 23 24 26 29 30;
set L := 1 2 3 4 5 6 7 8 9 10 11 12;

param a :=
1 20
2 17.5
5 10
8 32.5
11 30
13 30;

param b :=
1 0.0375
2 0.1750
5 0.6250
8 0.0834
11 0.2500
13 0.2500;

param c :=

```

```
2 40
3 40
4 40
5 40
7 40
8 40
10 40
12 40
14 40
15 40
16 40
17 40
18 40
19 40
20 40
21 40
23 40
24 40
26 40
29 40
30 40;
```

```
param d :=
2 0.461
3 4.167
4 1.316
5 0.106
7 0.439
8 0.333
10 1.724
12 0.893
14 1.613
15 1.220
16 2.857
17 1.111
18 3.125
19 1.053
20 4.545
21 0.571
23 3.125
24 1.149
26 2.857
29 4.167
30 0.943;
```

```
param alfa_s :=
1 20
2 17.5
5 10
8 40
11 40
13 40;
```

```
param alfa_i :=
1 20
2 17.5
5 10
8 0
11 0
13 0;
```

```
param QS_s :=
1 200
2 80
5 50
8 35
11 30
13 40;
```

```
param: Ts :=
1 2 78
1 3 78
2 4 39
2 5 78
2 6 39
3 4 78
4 12 39
5 7 42
6 4 54
6 28 19.2
7 6 78
8 6 19.2
9 6 39
```

```

9 10 39
10 6 19.2
10 17 19.2
11 9 39
12 13 39
12 14 19.2
12 15 19.2
12 16 19.2
15 14 9.6
15 18 9.6
15 23 9.6
17 16 9.6
18 19 9.6
19 20 19.2
20 10 19.2
21 10 19.2
22 10 19.2
22 21 19.2
24 22 9.6
24 23 9.6
24 25 9.6
25 26 9.6
27 25 9.6
27 28 39
29 27 39
27 30 39
28 8 19.2
29 30 39;
```

```

param:      RR :=
1 2 6 0.1763
1 6 4 0.0414
1 2 4 -0.1737
2 6 28 0.0599
2 28 8 0.2000
2 8 6 0.0420
3 9 6 -0.2080
3 9 10 0.1100
3 10 6 0.5560
4 22 10 -0.1499
4 22 21 0.0236
4 21 10 0.0749
5 12 14 0.2559
5 15 14 -0.1997
5 12 15 -0.1304
6 27 30 0.6027
6 29 30 -0.4533
6 29 27 0.4153
7 1 2 0.0575
7 2 4 0.1737
7 3 4 -0.0379
7 1 3 -0.1852
8 2 5 0.1983
8 5 7 0.1160
8 7 6 0.0820
8 2 6 -0.1763
9 6 4 -0.0414
9 10 6 -0.5560
9 10 17 0.0845
9 17 16 0.1932
9 12 16 -0.1987
9 4 12 -0.2560
10 6 28 0.0599
10 27 28 -0.3960
10 27 25 0.2087
10 24 25 -0.3292
10 24 22 0.1790
10 22 10 0.1499
10 10 6 0.5560
11 20 10 -0.2090
11 19 20 -0.0680
11 18 19 -0.1292
11 15 18 -0.2185
11 12 15 -0.1304
11 12 16 0.1987
11 17 16 -0.1932
11 10 17 -0.0845
12 22 10 -0.1499
12 24 22 -0.1790
12 24 23 0.2700
12 15 23 -0.2020
12 15 18 0.2185
12 18 19 0.1292
12 19 20 0.0680
12 20 10 0.2090;
```

```

param:      RRT :=
2  6  1  0.1763
6  4  1  0.0414
2  4  1  -0.1737
6  28 2  0.0599
28  8  2  0.2000
8  6  2  0.0420
9  6  3  -0.2080
9  10 3  0.1100
10  6  3  0.5560
22  10 4  -0.1499
22  21 4  0.0236
21  10 4  0.0749
12  14 5  0.2559
15  14 5  -0.1997
12  15 5  -0.1304
27  30 6  0.6027
29  30 6  -0.4533
29  27 6  0.4153
1  2  7  0.0575
2  4  7  0.1737
3  4  7  -0.0379
1  3  7  -0.1852
2  5  8  0.1983
5  7  8  0.1160
7  6  8  0.0820
2  6  8  -0.1763
6  4  9  -0.0414
10  6  9  -0.5560
10  17 9  0.0845
17  16 9  0.1932
12  16 9  -0.1987
4  12 9  -0.2560
6  28 10 0.0599
27  28 10 -0.3960
27  25 10 0.2087
24  25 10 -0.3292
24  22 10 0.1790
22  10 10 0.1499
10  6  10 0.5560
20  10 11 -0.2090
19  20 11 -0.0680
18  19 11 -0.1292
15  18 11 -0.2185
12  15 11 -0.1304
12  16 11 0.1987
17  16 11 -0.1932
10  17 11 -0.0845
22  10 12 -0.1499
24  22 12 -0.1790
24  23 12 0.2700
15  23 12 -0.2020
15  18 12 0.2185
18  19 12 0.1292
19  20 12 0.0680
20  10 12 0.2090;

```

```

param:      delta :=
1  1  2  1
1  1  3  1
2  1  2  -1
2  2  4  1
2  2  5  1
2  2  6  1
3  1  3  -1
3  3  4  1
4  2  4  -1
4  3  4  -1
4  6  4  -1
4  4  12  1
5  2  5  -1
5  5  7  1
6  2  6  -1
6  7  6  -1
6  8  6  -1
6  9  6  -1
6  10 6  -1
6  6  4  1
6  6  28 1
7  5  7  -1
7  7  6  1
8  8  6  1
8  28 8  -1
9  9  6  1
9  9  10 1
9  11 9  -1
10  9  10 -1
10  20 10 -1

```

```

10 21 10 -1
10 22 10 -1
10 10 6 1
10 10 17 1
11 11 9 1
12 4 12 -1
12 12 13 1
12 12 14 1
12 12 15 1
12 12 16 1
13 12 13 -1
14 12 14 -1
14 15 14 -1
15 12 15 -1
15 15 14 1
15 15 18 1
15 15 23 1
16 12 16 -1
16 17 16 -1
17 17 16 1
17 10 17 -1
18 15 18 -1
18 18 19 1
19 18 19 -1
19 19 20 1
20 19 20 -1
20 20 10 1
21 21 10 1
21 22 21 -1
22 22 10 1
22 22 21 1
22 24 22 -1
23 15 23 -1
23 24 23 -1
24 24 22 1
24 24 23 1
24 24 25 1
25 24 25 -1
25 27 25 -1
25 25 26 1
26 25 26 -1
27 29 27 -1
27 27 25 1
27 27 28 1
27 27 30 1
28 6 28 -1
28 27 28 -1
28 28 8 1
29 29 27 1
29 29 30 1
30 27 30 -1
30 29 30 -1;

```

```

param:      deltaT :=
1 2 1 1
1 3 1 1
1 2 2 -1
2 4 2 1
2 5 2 1
2 6 2 1
1 3 3 -1
3 4 3 1
2 4 4 -1
3 4 4 -1
6 4 4 -1
4 12 4 1
2 5 5 -1
5 7 5 1
2 6 6 -1
7 6 6 -1
8 6 6 -1
9 6 6 -1
10 6 6 -1
6 4 6 1
6 28 6 1
5 7 7 -1
7 6 7 1
8 6 8 1
28 8 8 -1
9 6 9 1
9 10 9 1
11 9 9 -1
9 10 10 -1
20 10 10 -1
21 10 10 -1
22 10 10 -1
10 6 10 1

```

```
10 17 10 1
11 9 11 1
4 12 12 -1
12 13 12 1
12 14 12 1
12 15 12 1
12 16 12 1
12 13 13 -1
12 14 14 -1
15 14 14 -1
12 15 15 -1
15 14 15 1
15 18 15 1
15 23 15 1
12 16 16 -1
17 16 16 -1
17 16 17 1
10 17 17 -1
15 18 18 -1
18 19 18 1
18 19 19 -1
19 20 19 1
19 20 20 -1
20 10 20 1
21 10 21 1
22 21 21 -1
22 10 22 1
22 21 22 1
24 22 22 -1
15 23 23 -1
24 23 23 -1
24 22 24 1
24 23 24 1
24 25 24 1
24 25 25 -1
27 25 25 -1
25 26 25 1
25 26 26 -1
29 27 27 -1
27 25 27 1
27 28 27 1
27 30 27 1
6 28 28 -1
27 28 28 -1
28 8 28 1
29 27 29 1
29 30 29 1
27 30 30 -1
29 30 30 -1;
```

```
solve;
```

```
display profit;
display alfa;
display QD;
display QS;
display T;
```

