



226987

Carlos Manuel Ferreira da Rocha **Aplicações dos Métodos e Algoritmos de Programação Semi-Infinita**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Matemática Aplicada, realizada sob a orientação científica do Dra. Tatiana Tchemisova Cordeiro, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro

o júri

presidente

Professor Doutor Domingos Moreira Cardoso
Professor Catedrático do Departamento de Matemática da Universidade de Aveiro

Professor Doutor Ismael Vaz
Professor Auxiliar da Escola de Engenharia da Universidade do Minho

Professora Doutora Tatiana Tchemisova Cordeiro
Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro

agradecimentos

À minha orientadora, Professora Doutora Tatiana Tchemisova Cordeiro pela excelente orientação, pela disponibilidade e compreensão demonstrada ao longo da elaboração desta dissertação.

À Catarina, por todo o seu apoio, carinho e paciência e a toda a minha família.

palavras-chave

Programação Semi-Infinita, Condição de Slater, Critério Implícito de Optimalidade, Pontos Imóveis e Ordens de Imobilidade.

resumo

O trabalho é dedicado ao estudo dos problemas de Programação Semi-Infinita (PSI) convexa, que consistem na minimização das funções de variáveis finitas em conjuntos definidos por um número infinito de restrições.

O crescente interesse dos investigadores pela teoria e métodos de PSI pode ser explicado não, só pelo extenso conjunto de aplicações de modelos de PSI na economia, engenharia, robótica, entre outros, mas também pela sua importância no desenvolvimento da teoria de Optimização em geral.

Nos últimos anos confere-se especial relevo às condições de optimalidade e aos métodos de PSI que não requerem condições adicionais para as restrições do problema, chamadas Qualificações de Restrições. Na dissertação estudam-se ambos os tipos de condições de optimalidade para problemas de PSI convexa: com e sem Qualificações de Restrições, e os métodos baseados nestas condições.

Uma das novas abordagens aos problemas de PSI convexa estudadas na dissertação está baseada numa característica qualitativa dos elementos do conjunto de índices de restrições activas, designada índice de imobilidade. O algoritmo de determinação dos índices de imobilidade dum problema de PSI convexo foi estudado e implementado em MATLAB. Os testes numéricos dos problemas das bases de dados existentes para PSI foram realizados.

keywords

Semi-infinite programming, Slater Condition, Optimality Implicity Criterion, Immobile Point and Immobility Orders.

abstract

The work is dedicated to a study of problems of Semi-Infinite Programming (SIP) that consist in minimization of finite variable functions in the sets defined by an infinite number of constraints.

The growing interest to theory and methods of SIP can be explained not only by a large number of applications of its models in economy, engineering, robotics, etc., but also by its influence on theory and methods of Optimization in general.

In the last years a special attention has been paid in SIP to the optimality conditions and methods that do not demand any special conditions on the constraints of the problem, so called Constraint Qualifications. In the dissertation we study both types of optimality conditions for the problems of convex PSI: with and without Constraint Qualifications.

One of the new approaches to problems of convex SIP considered in the dissertation is based on a qualitative characterization of the elements of the index set of active constraints, a so-called immobility order of the active index. The algorithm that permits to determine the immobility orders of the active indices of the convex SIP problem was studied and its program implementation realized in MATLAB. The existent SIP data bases were tested using the presented program.

Conteúdo

Introdução	1
1 Condições de Optimalidade de PSI	5
1.1 Notações e Definições Básicas	5
1.2 Condições de Optimalidade de Primeira Ordem (Clássicas)	7
1.3 Redução Local a um Problema Finito	8
1.4 Condições de Optimalidade de Segunda Ordem	12
2 Condições de Optimalidade para PSI Convexa sem CQ	15
2.1 Condições de Optimalidade para Problemas PSI Convexos com a Propriedade do Valor Médio Uniforme	16
2.2 Condições de Optimalidade Baseadas em Sucessões de Multiplicadores de Lagrange	19
2.3 Condições de Optimalidade Baseadas no Critério Implícito de Optimalidade	21
2.3.1 Critério Implícito de Optimalidade	21
3 Métodos Gerais de Programação Semi-Infinita	27
3.1 Métodos de Trocas	29
3.2 Métodos de Discretização	33
3.3 Métodos Baseados em Redução Local	34
3.4 Outros Métodos	35
3.4.1 Métodos de duas Fases	35
3.4.2 Métodos de direcções admissíveis (descendentes)	35
4 Algoritmo de Determinação das Ordens de Imobilidade	37
4.1 Descrição do Algoritmo	37

4.2	Propriedades do Algoritmo DIO	38
4.3	Exemplos de Aplicação do Algoritmo DIO	39
4.4	Resultados Numéricos	43
	Conclusões e Trabalho Futuro	47
	Bibliografia	49
	Apêndice A: Programa	55
	Apêndice B: Aplicações de PSI	61
	Investigação Operacional e Optimização Combinatória	61
	Aplicações à Robotica	64
	Aproximação de Chebyshev	66
	Economia e Finanças	67
	Programação Semi-definida (SDP)	69

Introdução

Um problema de Programação Semi-Infinita (PSI) (na língua inglesa usa-se a notação SIP- Semi-Infinite Programming), pode formular-se, de um modo geral, na forma seguinte:

$$(P) \quad \begin{array}{ll} \min & f(x) \\ \text{s.a} & g(x, t) \leq 0, t \in M, \end{array}$$

onde $x \in \mathbb{R}^n$ e o conjunto M é um subconjunto compacto infinito de \mathbb{R}^m . A f chama-se função objectivo, as desigualdades $g(x, t) \leq 0, t \in M$ representam as restrições do problema (P) . Designamos o problema (P) por Programação Semi-Infinita-PSI pelo facto de existirem infinitas restrições, mas as variáveis pertencerem a um espaço de dimensão finita.

Quando o conjunto M depende da variável x , ou seja, $M = M(x)$, o problema (P) pertence à classe da PSI generalizada (GSIP na literatura inglesa) [21], [25], [46]. Caso contrário, estamos perante um problema de PSI ordinária (standard).

No caso de f e $g(\cdot, t), t \in M$ serem funções lineares, dizemos que (P) é um problema semi-infinito linear (PSIL) [13], se as funções forem convexas, então o problema é convexo e nos restantes casos dizemos que (P) é um problema não linear de PSI.

A maior dificuldade na resolução dos problemas semi-infinitos prende-se com a existência de um número infinito de restrições. A ideia mais comumente utilizada na resolução dos problemas de PSI consiste na substituição destes problemas por problemas com um número finito de restrições (problemas finitos, que também são designados por problemas de Programação Não Linear (PNL)).

Na base da maioria dos métodos de PSI está o estudo local do conjunto admissível numa vizinhança dos pontos de extremos e a formulação das condições de optimalidade que sejam eficazes e fáceis de verificar. Por isso, o estudo da teoria de Programação Semi-infinita, incluindo a caracterização do conjunto admissível e as

condições de optimalidade para PSI, é de grande importância.

Nos nossos dias a construção de novos métodos numéricos para os problemas de PSI e comparação desses métodos tornaram-se numa área activa de investigação. Por norma os estudos de métodos numéricos são acompanhados por experiências numéricas.

O problema de criação de software eficaz que permite resolver problemas de PSI é muito importante. Recentemente foi incluído no servidor NEOS o programa NSIPS (*Nonlinear Semi-infinite Programming Solver*), codificado em AMPL (*A modeling Programming Language*), que permite através do uso da interface SIPAMPL (nome dado pelo autor do software), resolver problemas de Programação Semi-infinita codificados em linguagem C, ver [53] e [52].

No servidor NEOS é possível testar os vários tipos de problemas de optimização, em particular os de programação semi-infinita, sem o uso de um software específico e sem a necessidade de ter qualquer software instalado no computador. Por exemplo em [10] podemos encontrar como funciona o servidor NEOS.

Note-se que durante muito tempo não existia uma base de dados para problemas de PSI. Uma base de dados foi construída em [53] com a análise de bibliografia existente, sendo constituída por 144 problemas de PSI, permitindo a qualquer pessoa que pretende testar um novo método, um acesso rápido a uma boa quantidade de problemas de PSI.

O interesse para a teoria e métodos de PSI pode ser explicado pelo extenso conjunto de aplicações de PSI, por exemplo, na economia incluindo finanças, teoria dos jogos, geometria, probabilidades e estatística (teoria de risco e robustez estatística Bayesiana, entre outros), aproximação de funções (Aproximação de Chebyshev), espectrometria, controlo óptimo, optimização combinatória, robótica, problemas de transporte, citando apenas os mais comuns. Alguns exemplos recolhidos de [18],[13],[35] e [47] serão apresentadas numa secção posterior.

O objectivo principal desta dissertação consiste no estudo dos métodos de PSI convexa baseados em redução local, especialmente nos novos métodos que não exigem as condições especiais designadas de qualificações de restrições, "Constraint Qualification"(CQ). Para o estudo destes métodos é de grande interesse testa-los em problemas específicos onde os métodos "clássicos" de PSI não podem ser aplicados. Assim surgiu a ideia de estudar uma classe de problemas de PSI ordinária que não satisfaz as qualificações de restrições. Uma vez que a condição de Slater é

uma das mais fortes CQ, concentramo-nos nos problemas que não satisfazem esta condição.

Em [32] foi recentemente proposto um algoritmo construtivo (chamado de Algoritmo DIO) que permite verificar se um dado problema de PSI ordinário, na forma (P) com $M \subset \mathbb{R}$ satisfaz a Condição de Slater. Caso a Condição de Slater não seja satisfeita, o algoritmo determina os chamados "pontos imóveis" do conjunto de índices do problema que por si só são importantes características do problema.

Foi então efectuado o estudo de bibliografia, em geral e a base de dados NSIPS, em particular, para a procura destes problemas "não regulares" de PSI convexa ordinária. De seguida foi usado o algoritmo DIO para os testes.

Esta experiência revelou que a maioria dos problemas de base de dados existentes satisfazem a condição de Slater, que se pode explicar pelas seguintes razões:

- Os problemas são especialmente escolhidos de forma que o problema seja regular e a maioria dos métodos numéricos possam ser aplicados;
- Na escolha dos problemas para testar os seus métodos, os autores não estavam especialmente interessados em verificar se as condições CQ em geral eram satisfeitas;
- Quando os métodos baseados em discretização são aplicados então a regularidade do problema é, em geral, irrelevante.

Entretanto, muitos autores (por exemplo, ver [7], [23], [50] e suas referências) consideram que as situações em que as CQ não são satisfeitas são frequentes na prática, mas que é difícil saber a priori se o problema em estudo satisfaz estas condições.

O algoritmo DIO proposto em [31] foi implementado em MATLAB e uma série de vinte e nove problemas foram testados usando este algoritmo.

O trabalho encontra-se organizado da seguinte forma: no capítulo 1 introduzimos noções básicas de PSI e apresentamos condições de optimalidade mais gerais para problemas de PSI. Começa-se por apresentar condições denominadas de clássicas, que recorrem ao uso de CQ. De seguida, no capítulo 2 apresentam-se condições de optimalidade sem CQ propostas em [3] e [24]. Ainda no capítulo 2, apresentam-se condições de optimalidade baseadas no Critério Implícito de Optimalidade que estão na base do algoritmo a ser implementado.

Os métodos gerais de Programação Semi-infinita são desenvolvidos no capítulo 3, onde se descrevem os algoritmos dos três principais grupos de métodos de resolução.

O Algoritmo DIO - Determinação das Ordens de Imobilidade, está considerado no capítulo 4, bem como os resultados da sua implementação no computador.

O último capítulo é dedicado a algumas conclusões retiradas durante a realização desta dissertação.

Por último, são apresentados dois apêndices, o primeiro dedicado à explicação da implementação do algoritmo e o segundo à apresentação de algumas aplicações de PSI no mundo real.

Capítulo 1

Condições de Optimalidade de PSI

1.1 Notações e Definições Básicas

Designamos por $\|x\|$ a norma *Euclideana* do vector $x \in \mathbb{R}^n$. Dada uma função $g : \mathbb{R}^n \times \mathbb{R}^m \longrightarrow \mathbb{R}$ e $(\bar{x}, \bar{t}) \in \mathbb{R}^n \times \mathbb{R}^m$, denotaremos por $\nabla_t g(\bar{x}, \bar{t})$ a derivada parcial de g em ordem a t no ponto (\bar{x}, \bar{t}) , e por $\nabla_x g(\bar{x}, \bar{t})$ a derivada de g em ordem a x em (\bar{x}, \bar{t}) e $\nabla_{xx}^2 g(\bar{x}, \bar{t}), \nabla_{xt}^2 g(\bar{x}, \bar{t}), \nabla_{tt}^2 g(\bar{x}, \bar{t})$, como sendo, respectivamente, as $n \times n, n \times m, m \times m$ matrizes Hessianas de g em (\bar{x}, \bar{t}) .

Definição 1.1.1. Um conjunto $S \subset \mathbb{R}^n$ diz-se *convexo* se,

$$\forall x, y \in S \text{ e } \forall \lambda \in [0, 1] \text{ se tem } \lambda x + (1 - \lambda)y \in S.$$

Definição 1.1.2. Seja $S \subset \mathbb{R}^n$ um conjunto convexo. A função $f : S \longrightarrow \mathbb{R}$ diz-se *convexa* se

$$\forall x, y \in S, \forall \lambda \in [0, 1] : f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

Definição 1.1.3. O conjunto $K \subset \mathbb{R}^n$ diz-se um *cone* se $\forall x \in K, \forall \lambda > 0 : \lambda x \in K$.

Definição 1.1.4. Seja $\{a_1, \dots, a_m\} \in \mathbb{R}^n$. Ao conjunto $K(a_1, \dots, a_m) = \left\{ \sum_{i=1}^m \lambda_i a_i, \lambda_i \geq 0 \right\}$ chama-se *cone convexo gerado pelo conjunto* $\{a_1, \dots, a_m\}$.

Seja

$$\mathcal{M} = \{x \in \mathbb{R} : \varphi_i(x) = 0, \psi_j(x) \leq 0, i \in I, j \in J\} \quad (1.1)$$

com conjuntos I e J finitos em \mathbb{R} .

Dado $\bar{x} \in \mathcal{M}$, seja $J_0(\bar{x}) = \{j \in J : \psi_j(\bar{x}) = 0\}$ conjunto de índices de restrições activas em \bar{x} .

Definição 1.1.5. *Ao conjunto*

$$T_{\bar{x}} = \{\xi \in \mathbb{R}^n : \nabla\varphi_i(\bar{x}) \cdot \xi = 0, \nabla\psi_j(\bar{x}) \cdot \xi = 0, i \in I, j \in J_0(\bar{x})\} \quad (1.2)$$

chama-se espaço tangente a \mathcal{M} em \bar{x} .

Seja o problema de PSI ordinária na forma:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.a} \quad & g(x, t), t \in M \subseteq \mathbb{R}^m \end{aligned} \quad (1.3)$$

onde M é um conjunto compacto de \mathbb{R}^m que não depende de x .

Assumamos que no problema (1.3), além de M ser um conjunto compacto, as funções f e g são funções continuamente diferenciáveis em ordem a x , para todo \mathbb{R}^n e $\mathbb{R}^n \times \mathbb{R}^m$, respectivamente. Consideremos ainda o conjunto

$$X = \{x \in \mathbb{R}^n : g(x, t) \leq 0, t \in M\},$$

chamado conjunto admissível e seja $\bar{x} \in X$ uma solução admissível.

Note-se que no caso de $m = 1$, o conjunto

$$M = \bigcup_{i=1}^k [t_{*i}, t_i^*]$$

sempre pode ser considerado na forma (1.1) se supusermos: $I = \emptyset$, $J = \{1, 2, \dots, 2k\}$, $\psi_i = \psi_i(t) = t - t_i^*$, $i = 1, \dots, k$, $\psi_i = \psi_i(t) = t_{*i} - t$, $i = k + 1, \dots, 2k$.

Definição 1.1.6. *(Condição de Slater)* Seja $X \in \mathbb{R}^n$ o conjunto de soluções admissíveis do problema (1.3): $X = \{x \in \mathbb{R}^n : g(x, t) \leq 0, \forall t \in M\}$. Dizemos que a condição de Slater é satisfeita se $\exists \bar{x} \in X : g(\bar{x}, t) < 0, \forall t \in M$.

Definição 1.1.7. *Constraint Qualification (CQ)* são condições que se impõe nas restrições do problema de otimização (finita ou infinita) para garantir que determinadas condições de optimalidade sejam satisfeitas.

Da literatura são conhecidas várias CQ para problemas PNL [38] e para problemas de PSI, [48]. A condição de Slater é um exemplo de CQ.

1.2 Condições de Optimalidade de Primeira Ordem (Clássicas)

Nesta secção reformulamos condições de optimalidade obtidas em [18] para o problema de PSI na forma (1.3) com M na forma (1.1). Uma vez que estas condições de optimalidade são do tipo Karush-Kuhn-Tucker (KKT) e que estão na base da maioria dos métodos de PSI, vamos designar estas condições por condições clássicas.

Definimos

$$\bar{E} = E(\bar{x}) = \{t \in M : g(\bar{x}, t) = 0\} = \{\bar{t}_l : l \in L\} \quad (1.4)$$

o conjunto das restrições activas em \bar{x} .

Notemos aqui que o conjunto \bar{E} pode ser finito ou infinito.

Designaremos uma direcção $\xi \in \mathbb{R}^n$ como admissível para problema (1.3) em \bar{x} se existe uma curva parametrizada suave (continuamente diferenciável), admissível partindo de \bar{x} tangente a ξ . Obviamente, se ξ é uma direcção admissível em \bar{x} , então temos

$$\xi^T \nabla_x g^l \leq 0, \quad l \in L \quad (1.5)$$

com $\nabla_x g^l = \nabla_x g^l(\bar{x}, \bar{t}_l)$.

A seguinte condição pode ser interpretada como uma condição necessária para ξ ser uma direcção descendente admissível.

Lema 1.2.1. *(Lema 3.1 em [18]) Seja $\bar{x} \in X$ uma solução óptima para (1.3) e suponha-se que cada ξ com (1.5) é uma direcção admissível em \bar{x} . Então não existe $\xi \in \mathbb{R}^n$ tal que*

$$\xi^T \nabla f < 0, \quad \xi^T \nabla_x g^l \leq 0, \quad l \in L. \quad (1.6)$$

Existem várias condições que garantem que ξ com (1.5) é uma direcção admissível em \bar{x} . Uma dessas condições (que pode ser considerada uma CQ) está apresentada no fim desta secção (condição 1.9).

Para obter para o problema (1.3) as condições de optimalidade do tipo de Karush-Kuhn-Tucker (KKT), usaremos a seguinte versão do lema de Farkas [19].

Lema 1.2.2. *(Lema 3.2 em [18]) Seja $S \subset \mathbb{R}^n$ arbitrário e $co(S)$ o cone convexo gerado por S . Então para qualquer $v \in \mathbb{R}^n$, apenas uma das seguintes afirmações é verdadeira:*

1. $v \in cl(co(S))$;
2. existe uma solução ξ de

$$\xi^T v > 0, \quad \xi^T s \leq 0, \quad s \in S. \quad (1.7)$$

Aqui $cl(co(S))$ designa o fecho do conjunto $co(S)$. Aplicando o Lema 1.2.2 ao Lema 1.2.1, obtemos que nas condições do Lema 1.2.1 se tem:

$$\nabla f \in cl(C_g) \quad (1.8)$$

onde C_g é o cone convexo gerado por $\{\nabla_x g^l : l \in L\}$: $C_g = co(\{\nabla_x g^l : l \in L\})$.

Se C_g é fechado, a inclusão em (1.8) pode ser interpretada como que, ∇f uma combinação linear de um número finito de gradientes $\nabla_x g^l$. É conhecido (ver, por exemplo [33]) que C_g é fechado se a seguinte condição é satisfeita:

$$\text{Existe um } \xi \text{ tal que } \xi^T \nabla_x g^l < 0, \quad l \in L. \quad (1.9)$$

Comentário 1.2.1. *Em [18] está demonstrado que se (1.9) é válida em \bar{x} então todo ξ satisfazendo (1.5) é uma direcção admissível em \bar{x} . A demonstração é muito extensa, no entanto, assumindo (1.9), a prova do Lema 1.2.1 torna-se directa (ver [19]). Assim podemos concluir que a condição (1.9) é uma (CQ). Outras (CQ) poderiam ser impostas no Lema 1.2.1, como é exemplo a Condição de Slater.*

O seguinte teorema formula as condições necessárias de optimalidade para (1.3).

Teorema 1.2.1. *(Teorema 3.3 em [18])(KKT) Seja $\bar{x} \in X$ uma solução óptima de (1.3) e (1.9) é satisfeita. Então existe um conjunto $L' \subset L$, $|L'| < \infty$ e $u_l > 0$, $l \in L'$, tal que*

$$\nabla_x f = \sum_{l \in L'} u_l \nabla_x g^l \quad (\text{i.e., } \nabla f \in C_g). \quad (1.10)$$

1.3 Redução Local a um Problema Finito

A solução do problema de PSI na forma (1.3) torna-se possível caso possamos descrever uma aproximação local do conjunto admissível X , impondo apenas um número finito de restrições do problema (1.3).

Na base do Teorema 1.2.1 podemos concluir que estas restrições estão entre as restrições activas do problema.

Um dos processos, no qual se insere o modelo que aplicaremos mais à frente, é a redução local do problema (1.3).

Consideremos o problema paramétrico seguinte:

$$(O(x)) \quad \max_t \{g(x, t) : t \in M\}. \quad (1.11)$$

Este problema é designado, por vezes, de problema de nível inferior (*lower level problem*).

É evidente que a inclusão $x \in X$ é equivalente à condição $v(O(x)) \leq 0$, onde v representa o valor óptimo da função f em (1.3). Além do mais, para um dado $\bar{x} \in X$, os pontos $\bar{t}_l \in \bar{E}$ (c.f.(1.4)), $l = 1, \dots, r$ são soluções óptimas de (1.11).

Consideremos a seguinte suposição:

Suposição 1.3.1. (*Suposição 4.7 em [18]*) Dado $\bar{x} \in X$, o conjunto $\bar{E} = \{\bar{t}_1, \dots, \bar{t}_r\}$ é finito. Além disso, existem vizinhanças $\mathcal{U}_{\bar{x}}$ de \bar{x} , e $\mathcal{U}_{\bar{t}_l}$ de $\bar{t}_l, l = 1, \dots, r$, e as aplicações contínuas

$$t_l : \mathcal{U}_{\bar{x}} \longrightarrow \mathcal{U}_{\bar{t}_l} \cap M \quad (1.12)$$

tais que:

1. $t_l(\bar{x}) = \bar{t}_l, l = 1, \dots, r;$
2. $t_l(\bar{x})$ é a única solução do problema (1.11) em $\mathcal{U}_{\bar{t}_l} \cap M$, para $l = 1, \dots, r$ e para todo $x \in \mathcal{U}_{\bar{x}}$.

Se $\bar{x} \in X$ verifica a Suposição 1.3.1, então o problema de PSI pode ser localmente aproximado, na vizinhança $\mathcal{U}_{\bar{x}}$, por um problema reduzido finito, como nos prova o seguinte lema:

Lema 1.3.1. (*Lema 4.8 em [18]*) Seja $\bar{x} \in X$ dado e a Suposição 1.3.1 válida. Então existe uma vizinhança $\mathcal{U}_{\bar{x}}$ de \bar{x} tal que para todo $x \in \mathcal{U}_{\bar{x}}$, x é admissível para o problema (1.3) se e só se

$$G^l(x) = g(x, t_l(x)) \leq 0, l = 1, \dots, r. \quad (1.13)$$

Definimos o problema local reduzido $P_{red}(\bar{x})$ na forma:

$$\begin{aligned} \min_{x \in \mathcal{U}_{\bar{x}}} \quad & f(x) \\ \text{s.a} \quad & G^l(x) \leq 0, l = 1, \dots, r. \end{aligned} \quad (1.14)$$

Teorema 1.3.1. (Teorema 4.10 em [18]) Seja $\mathcal{U}_{\bar{x}}$ uma vizinhança de $\bar{x} \in X$ como no Lema (1.3.1). Então um ponto $x^0 \in \mathcal{U}_{\bar{x}}$ é localmente óptimo para (1.3) se e só se o é para $P_{red}(\bar{x})$ (1.14).

Por [19], podemos ver, que no caso de problemas PSI convexos, as funções $G^l(x)$, $l = 1, \dots, r$ são convexas. Assim, sendo a Suposição 1.3.1 satisfeita, podemos, localmente, dar uma descrição do conjunto admissível X do problema infinito (1.3) em termos de um número finito de restrições convexas, as quais, no entanto, estão definidas apenas implicitamente através de soluções de problemas de otimização paramétricos na forma (1.11).

O Teorema 1.3.1 torna-se uma ferramenta para transmitir a teoria e os métodos dos problemas finitos para problemas semi-infinitos.

Consideremos a seguinte suposição:

Suposição 1.3.2. Suponha que o conjunto $M \subset \mathbb{R}^m$ é definido por

$$M = \{t : \psi_j(t) \leq 0, j \in P\}, \quad (1.15)$$

com $|P| < \infty$ e $\psi_j(t)$ são funções duas vezes continuamente diferenciáveis em \mathbb{R}^m . Além disso, em cada ponto de $t \in M$ a condição de independência linear (LICQ) é válida, isto é, os vectores

$$\nabla_t \psi_j(t), j \in P_t = \{i : \psi_i(t) = 0\}$$

são linearmente independentes.

O seguinte resultado segue então:

Lema 1.3.2. (Lema 4.12 em [18]) Consideremos o problema (1.3) com conjunto M na forma (1.15) que satisfaz a Suposição 1.3.2. Para $\bar{x} \in \mathbb{R}^n$ e para $E^l \in \bar{E}$ de (1.4) sejam

$$P^l = \{j : \psi_j(t_l) = 0\} \quad (1.16)$$

e a função Lagrangeana associada ao problema $O(x)$:

$$\mathcal{L}^l(t, \lambda^l) = g(\bar{x}, t) - \sum_{j \in P^l} \alpha_j^l \psi_j(t). \quad (1.17)$$

Então existe um único vector de multiplicadores de Lagrange $\bar{\alpha}^l = (\bar{\alpha}_j^l, j \in P^l, l = 1, \dots, r)$, $\bar{\alpha}^l \geq 0$ tal que $\nabla_t \mathcal{L}^l(\bar{t}_l, \bar{\alpha}^l) = 0$.

Suposição 1.3.3. *Seja g uma função duas vezes diferenciável em $\mathbb{R}^n \times \mathbb{R}^m$ e seja válida a Suposição 1.3.2. Supomos que está satisfeita a condição suficiente de segunda ordem para que \bar{t}_l seja um maximizante local estrito do problema $O(x)$:*

$$\nabla_{tt}^2 \mathcal{L}^l(\bar{t}, \bar{\alpha}^l) \text{ é definida negativa no espaço tangente } T_{\bar{t}_l}.$$

Aqui, por (1.2) temos

$$T_{\bar{t}_l} = \{\eta : \bar{\alpha}_j^l \eta^T \nabla_t h^j(\bar{t}_l) = 0, j \in P^l\}. \quad (1.18)$$

Suposição 1.3.4. *Seja a Suposição 1.3.3 válida, para todo $\bar{t}_l \in E$. Supomos ainda que a condição de complementariedade estrita é válida, ou seja, $\bar{\alpha}_j^l > 0$, para todo $j \in P^l$.*

Obtemos assim o teorema seguinte:([17])

Teorema 1.3.2. *1. Assuma-se que a Suposição 1.3.4 é válida. Então a Suposição 1.3.1 mantém-se com funções $t^l : \mathcal{U}_{\bar{x}} \rightarrow \mathcal{U}_{\bar{t}_l} \cap M$ continuamente diferenciáveis. Além disso, existem aplicações $\alpha^l : \mathcal{U}_{\bar{x}} \rightarrow \mathbb{R}$ continuamente diferenciáveis tal que a Suposição 1.3.4 é válida para todos os ternos $(x, t_l(x), \alpha^l(x))$, $x \in \mathcal{U}_{\bar{x}}$. Então as derivadas $\nabla_x t_l = \nabla_x t_l(x)$, $\nabla_x \alpha^l = \nabla_x \alpha^l(x)$ são determinadas univocamente por:*

$$\begin{pmatrix} \nabla_{tt}^2 \mathcal{L}^l(t_l, \alpha^l) & \nabla_t H^l(t_l) \\ (\nabla_t H^l(t_l))^T & 0 \end{pmatrix} \begin{pmatrix} t_l \\ \alpha^l \end{pmatrix} = - \begin{pmatrix} \nabla_{tx}^2 g^l(x, t_l) \\ 0 \end{pmatrix} \quad (1.19)$$

com

$$\nabla_t H^l(t_l) = (\nabla_t \psi_j(t_l))_{j \in P^l} \in \mathbb{R}^{m \times |P^l|}. \quad (1.20)$$

2. As funções restrição $G^l(x) = g(x, t_l(x))$ do problema reduzido (1.14) são duas vezes continuamente diferenciáveis em $\mathcal{U}_{\bar{x}}$ e são dadas por:

$$\nabla_x G^l(x) = \nabla_x g(x, t_l(x)), \quad (1.21)$$

$$\nabla_{xx}^2 G^l(x) = \nabla_{xx}^2 g^l(x, t_l(x)) - (\nabla_x t_l(x))^T \nabla_{tt}^2 \mathcal{L}^l(t_l(x), \alpha^l(x)) \nabla_x t^l(x). \quad (1.22)$$

A prova de 1 usa o Teorema da Função Implícita e argumentos de continuidade. As relações (1.21) e (1.22) obtém-se de (1.19).

1.4 Condições de Optimalidade de Segunda Ordem

Uma forma comum de obter condições de optimalidade de segunda ordem para problemas de PSI é usar a redução local e substituir estas condições (para problemas infinitos), pelas condições de optimalidade de otimização finita ao problema reduzido (1.14).

Supomos que as Suposições 1.3.2, 1.3.3 estão satisfeitas e que as funções $G^l(x)$ são duas vezes continuamente diferenciáveis.

Aplicando os resultados obtidos em [17] podemos formular o seguinte teorema.

Teorema 1.4.1. *Suponha que $\bar{x} \in X$ é ponto admissível do problema (1.3) e que satisfaz a Suposição 1.3.4. Tem-se então:*

1. (Condição Necessária) *Se \bar{x} é solução ótima de (1.3), então para todo ξ em \mathcal{K} ,*

$$\mathcal{K} = \{\xi : \xi^T \nabla f \leq 0, \xi^T \nabla_x g^l \leq 0, l \in L\}, \quad (1.23)$$

existem $\lambda_0(\xi) \geq 0$, $\lambda(\xi) \geq 0$, $\lambda(\xi) = (\lambda_l(\xi), l \in L)$, não todos nulos, tais que, para

$$L(x, \lambda_0, \lambda) = \lambda_0 f(x) + \sum_{l \in L} \lambda_l g(x, t_l) \quad (1.24)$$

tem-se

$$\nabla_x L(\bar{x}, \lambda_0(\xi), \lambda(\xi), \bar{t}) = 0 \quad (1.25)$$

e

$$y(\xi) := \xi^T \nabla_{xx}^2 L(\bar{x}, \lambda_0(\xi), \lambda(\xi), \bar{t}) \xi - \sum_{l \in L} \lambda_l(\xi) (\nabla_x \bar{t}^l \xi)^T \nabla_{tt} \mathcal{L}^l(\bar{t}_l, \alpha^l(\bar{x})) (\nabla_x \bar{t}^l \xi) \geq 0. \quad (1.26)$$

2. (Condição Suficiente) *Se para todo o $\xi \in \mathcal{K}$, existem $\lambda_0(\xi)$ e $\lambda(\xi)$ que verificam as Condições Necessárias 1 e se para $\xi \neq 0$,*

$$y(\xi) > 0,$$

então \bar{x} é uma solução ótima local estrita de (1.3) (ou seja, existe uma vizinhança \bar{U} de \bar{x} tal que para $x \in \bar{U} \cap X$ temos $f(x) < f(\bar{x})$).

3. (Condição Suficiente Forte) A Condição Necessária 2 é especialmente verificada se

- os vectores $\nabla_x g^l, l \in L$ são linearmente independentes;
- existem $\lambda_l \geq 0$ únicos tais que a condição (KKT) (1.25) é válida com $\lambda_0(\xi) = 1$ e $\lambda_l(\xi) = \lambda_l, l \in L$;
- $y(\xi) > 0$, com $\lambda_0(\xi) = 1$ e $\lambda_l(\xi) = \lambda_l$, para todo o $\xi \in \Sigma$, onde

$$\Sigma = \{\xi : \lambda_l (\nabla_x g^l)^T \xi = 0, l \in L\}.$$

Comentário 1.4.1. O primeiro termo em (1.26) é o termo que se obtém nas condições de optimalidade de segunda ordem para um problema reduzido (finito) e o segundo termo reflecte a estrutura semi-infinita do problema e é gerado pelo desvio das restrições activas $t_l(x)$ como funções de x .

Capítulo 2

Condições de Optimalidade para PSI Convexa sem CQ

No capítulo anterior as condições suficientes e necessárias de optimalidade foram formuladas para o problema geral de PSI que satisfaz algumas CQ (por exemplo LICQ, Condição de Slater).

No entanto, existe um interesse em procurar condições de optimalidade que não exijam CQ, mesmo que essas condições só possam ser aplicadas para uma classe de problemas de PSI. Por exemplo, quando os problemas são convexos, é possível, usando a propriedade da convexidade, obter condições de optimalidade sem exigir a existência de uma CQ.

A aplicação destas condições e construção dos métodos baseados nestas condições dependem então da regularidade do problema. Nem todas as CQ são fáceis de verificar e, como indicam os autores de [8], [26], [54] e suas referências, as situações em que as CQ não são satisfeitas, são frequentes na prática.

A seguir apresentamos algumas condições de optimalidade que não exigem que qualquer CQ seja satisfeita pelo problema, obtidas em [3], [24], [31].

2.1 Condições de Optimalidade para Problemas PSI Convexos com a Propriedade do Valor Médio Uniforme

Consideremos agora o problema de PSI na forma (1.3) com a propriedade da convexidade, ou seja, f é convexa e diferenciável, $g(x, t)$ é convexa e diferenciável em x para cada t e contínua em t para cada x e por fim seja M um conjunto compacto.

O conjunto admissível X do problema (1.3) é, neste caso convexo, dado que é a intersecção de conjuntos convexos.

Vamos considerar aqui um vector $d \in \mathbb{R}^n$ como uma direcção em \bar{x} no conjunto X , se $\bar{x} + d \in X$. Para uma dada função $g(\cdot, t)$ e para $t \in E(\bar{x})$ fixo, definimos

$$D(\bar{x}, t) = \{d \in \mathbb{R}^n : \exists \bar{\alpha} > 0, g(\bar{x} + \alpha d, t) = g(\bar{x}, t), \text{ para todo } 0 \leq \alpha \leq \bar{\alpha}\}.$$

Vamos considerar, primeiro, as condições de optimalidade que exigem que as funções restrição do problema (1.3) possuam a propriedade do valor médio uniforme.

Definição 2.1.1. *Seja M um conjunto compacto em \mathbb{R}^n . A função $g : \mathbb{R}^n \times M \rightarrow \mathbb{R}$ tem a propriedade do valor médio uniforme ("uniform mean value property" em inglês) em $x \in \mathbb{R}^n$ se, para qualquer $d \in \mathbb{R}^n$ não nulo e para todo $\alpha > 0$, existe $\hat{\alpha} = \hat{\alpha}(d, \alpha), 0 < \hat{\alpha} < \alpha$ tal que*

$$\frac{g(x + \alpha d, t) - g(x, t)}{\alpha} \geq d^T \nabla_x g(x + \hat{\alpha} d, t) \text{ para todo } t \in M.$$

O teorema seguinte dá-nos as condições de optimalidade para (1.3).

Teorema 2.1.1. *(Teorema 1 em [3]) Seja \bar{x} uma solução admissível do problema convexo de PSI na forma (1.3) tal que $E(\bar{x}) \neq \emptyset$. Supomos que g tem a propriedade do valor médio uniforme. Então \bar{x} é solução óptima de (1.3) se e só se, para todo $\alpha^* > 0$, o sistema*

1.

$$d^T \nabla f(\bar{x}) < 0, \quad d \in \mathbb{R}^n \tag{2.1}$$

2.

$$d^T \nabla g(\bar{x} + \alpha^* d, t) \leq 0, \text{ para todo } t \in E(\bar{x}), \tag{2.2}$$

3.

$$\frac{d^T \nabla g(\bar{x} + \alpha^* d, t)}{g(\bar{x}, t)} \geq -\frac{1}{\alpha^*} \text{ para todo } t \in M \setminus E(\bar{x}) \quad (2.3)$$

é inconsistente.

Demonstração. Mostraremos que \bar{x} não é solução óptima se e só se existe $\alpha^* > 0$ tal que o sistema (2.1), (2.2) e (2.3) é consistente. Uma solução admissível \bar{x} não é óptima se existe $\bar{\alpha} > 0$ e $d \in \mathbb{R}^n, d \neq 0$ tal que

$$f(\bar{x} + \bar{\alpha}d) < f(\bar{x}), \quad (2.4)$$

$$g(\bar{x} + \bar{\alpha}d, t) \leq 0, \text{ para todo } t \in M. \quad (2.5)$$

Pela convexidade de f e (2.2), a existência de $\bar{\alpha} > 0$ satisfazendo (2.4) é equivalente a $d^T \nabla f(\bar{x}) < 0$.

Considerando (2.5), discutimos separadamente dois casos:

$$g(\bar{x} + \bar{\alpha}d, t) \leq 0, \text{ para todo } t \in E(\bar{x}), \quad (2.6)$$

$$g(\bar{x} + \bar{\alpha}d, t) \leq 0, \text{ para todo } t \in M \setminus E(\bar{x}). \quad (2.7)$$

Consideremos primeiro (2.6). Pela convexidade e propriedade do valor médio uniforme de g , temos

$$g(\bar{x} + \bar{\alpha}d, t) \geq g(\bar{x}, t) + \bar{\alpha} d^T \nabla g(\bar{x} + \hat{\alpha}d, t) \text{ para todo } t \in E(\bar{x}) \text{ e para algum } 0 < \hat{\alpha} \leq \bar{\alpha}. \quad (2.8)$$

Dado que $t \in E(\bar{x})$ e $\bar{\alpha} > 0$, (2.6) implica

$$d^T \nabla g(\bar{x} + \hat{\alpha}d, t) \leq 0. \quad (2.9)$$

Claramente $\hat{\alpha}$ existe e é positivo.

Por outro lado, a existência de $\alpha^* > 0$ tal que, para algum $t \in E(\bar{x})$,

$$d^T \nabla g(\bar{x} + \alpha^* d, t) \leq 0$$

implica (2.6) com $0 < \bar{\alpha} \leq \alpha^*$.

Fica para mostrar que a existência de $\bar{\alpha} > 0$, tal que (2.7) é válida, é equivalente à existência de $\bar{\alpha} > 0$, tal que (2.3) é válida. Suponhamos que (2.7) se mantém para

algum $\bar{\alpha} > 0$. Então, pela convexidade e propriedade do valor médio uniforme da função g , tem-se:

$$0 \geq g(\bar{x} + \bar{\alpha}d, t) \geq g(\bar{x}, t) + \bar{\alpha}d^T \nabla g(\bar{x} + \tilde{\alpha}d, t), \quad \forall t \in M \setminus E(\bar{x}) \text{ e para algum } 0 < \tilde{\alpha} \leq \bar{\alpha}. \quad (2.10)$$

Então

$$\begin{aligned} \frac{d^T \nabla g(\bar{x} + \tilde{\alpha}d, t)}{g(\bar{x}, t)} &\geq -\frac{1}{\tilde{\alpha}}, \text{ desde que } t \in M \setminus E(\bar{x}) \\ &\geq -\frac{1}{\tilde{\alpha}}, \text{ por (2.10)}. \end{aligned} \quad (2.11)$$

Usando a monotonia do gradiente da função convexa $g(\cdot, t)$, obtemos que

$$\frac{d^T \nabla g(\bar{x} + \alpha d, t)}{g(\bar{x}, t)} \geq \frac{d^T \nabla g(\bar{x} + \tilde{\alpha}d, t)}{g(\bar{x}, t)} \text{ para todo } 0 \leq \alpha \leq \tilde{\alpha}. \quad (2.12)$$

Isto dá

$$\frac{d^T \nabla g(\bar{x} + \tilde{\alpha}d, t)}{g(\bar{x}, t)} \geq -\frac{1}{\tilde{\alpha}}, \text{ por (2.11) e (2.12)},$$

que é (2.3), com $\alpha^* = \tilde{\alpha}$.

Suponhamos agora que (2.3) é verdadeira para algum $\alpha^* > 0$. Usando mais uma vez a monotonia do gradiente da função convexa $g(\cdot, t)$, e o facto que $g(\bar{x}, t) < 0$ para $t \in M \setminus E(\bar{x})$, obtemos

$$g(\bar{x}, t) + \alpha^* d^T \nabla g(\bar{x} + \alpha d, t) \leq 0, \text{ para todo } 0 < \alpha < \alpha^*. \quad (2.13)$$

Mas

$$\begin{aligned} g(\bar{x} + \alpha^* d, t) &= g(\bar{x}, t) + \alpha^* d^T \nabla g(\bar{x} + \alpha d, t), \quad \text{para algum } 0 < \alpha < \alpha^*, \alpha = \alpha(t) \\ &\quad \text{pelo teorema do valor intermédio} \\ &\leq 0, \text{ por (2.13)} \end{aligned}$$

que é (2.7), com $\bar{\alpha} = \alpha^*$.

Resumindo os resultados anteriores, obtemos as seguintes conclusões: Se \bar{x} não é solução óptima, então existe $\alpha^* = \min\{\hat{\alpha}, \tilde{\alpha}\} > 0$ tal que o sistema (2.1)-(2.3) é consistente.

Se existe $\alpha^* > 0$ tal que o sistema (2.1)-(2.3) é consistente, então existe $\alpha_0 > 0$ e $\bar{\alpha} > 0$ tal que

$$\begin{cases} f(\bar{x} + \alpha_0 d) < f(\bar{x}), \\ g(\bar{x} + \bar{\alpha}d, t) \leq 0, \text{ para todo } t \in E(\bar{x}), \\ g(\bar{x} + \bar{\alpha}d, t) \leq 0, \text{ para todo } t \in M \setminus E(\bar{x}). \end{cases} \quad (2.14)$$

Se designarmos $\hat{\alpha} = \min\{\alpha_0, \bar{\alpha}\} > 0$, então pela convexidade de $g(\cdot, t)$, (2.14) pode ser escrita

$$\begin{cases} f(\bar{x} + \hat{\alpha}d) < f(\bar{x}) \\ g(\bar{x} + \hat{\alpha}d, t) \leq 0, \text{ para todo } t \in M, \end{cases}$$

implicando que \bar{x} não é solução óptima. \square

Notemos que o Teorema 2.1.1 é válido apenas para funções convexas que satisfazem a propriedade do valor médio uniforme. Para tal consideremos o exemplo seguinte que mostra que o teorema falha no caso de a função não possuir a propriedade do valor médio.

Exemplo 2.1.1.

$$\begin{aligned} \min \quad & -x_2 \\ \text{s.a.} \quad & g(x_1, x_2) \leq 0, \quad \forall t \in [0, 1], \end{aligned} \tag{2.15}$$

onde

$$g(x_1, x_2, t) = \begin{cases} x_1^2 + tx_2(x_2 - t), & \text{se } x_2 < \frac{1}{2}t; \\ x_1^2 + \frac{t^3}{(2-t)^2}(x_2 - t + 1)(x_2 - 1), & \text{se } x_2 \geq \frac{1}{2}t. \end{cases}$$

A função g não possui a propriedade do valor médio uniforme. O conjunto admissível é

$$X = \{(0, x_2)^T : 0 \leq x_2 \leq 1\}$$

e a solução óptima é $\bar{x} = (0, 1)^T$. Contudo, para todo $\alpha^* > 0$ o sistema (2.1)-(2.3) é inconsistente em $x^* = \bar{0}$ um ponto que não é solução óptima. Dado que $\bar{E} = [0, 1]$ a condição (2.3) é aqui redundante, enquanto (2.1), (2.2) ficam, respectivamente,

$$\begin{cases} -d_2 < 0, \\ 2\alpha^*d_1^2 + t(2\alpha^*d_2 - t)d_2 \leq 0, & \text{se } 2\alpha^*d_2 < t; \\ 2\alpha^*d_1^2 + \frac{t^3}{(2-t)^2}(2\alpha^*d_2 - t)d_2 \leq 0, & \text{se } 2\alpha^*d_2 \geq t. \end{cases}$$

O sistema anterior não pode ser consistente para algum $\alpha^* > 0$, porque se as duas primeiras desigualdades fossem satisfeitas, a última desigualdade seria absurda para um valor pequeno de $t \in [0, 1]$.

2.2 Condições de Optimalidade Baseadas em Sucessões de Multiplicadores de Lagrange

Em [24] podemos encontrar outra forma de obter condições de optimalidade para PSI sem recurso a CQ, que apresentaremos resumidamente.

Mais uma vez consideremos o problema (1.3) com f e g funções convexas em relação a x . Seja $Z = \prod_M \mathbb{R}$ o espaço produto com o produto topológico e seja Z' o espaço generalizado de seqüências finitas introduzido em [24]. Designamos:

$$\Lambda = \{\lambda = (\lambda_t) \in Z' : \lambda_t \geq 0 \forall t \in M, \lambda_t = 0 \text{ para todos menos um número finito de } t \in M\}.$$

As condições de optimalidade para problemas de PSI convexas estão formuladas no Teorema 5.2 em [24]. Com base nesse teorema podemos obter o seguinte critério de optimalidade para PSI convexa.

Teorema 2.2.1. *Para o problema PSI convexo na forma (1.3), seja $x \in X$. Então \bar{x} é solução óptima de (1.3) se e só se: Existem sucessões $\{x_k\} \subset X$, $\{\lambda^k\} \subset \Lambda$, e $\{v_k\} \subset \mathbb{R}^n$ tais que*

$$v_k = \sum_{t \in M} \lambda_t^k \nabla_x g(x_k, t), \nabla f(\bar{x}) + v_k \rightarrow 0, \|x_k - \bar{x}\| \rightarrow 0, \text{ e } \lambda_k g(x_k) \rightarrow 0, \text{ quando } k \rightarrow \infty. \quad (2.16)$$

Notemos que o Teorema 2.2.1 dá as condições necessárias e suficientes de optimalidade para todos os problemas convexas de PSI sem exigir nenhuma condição CQ para as restrições destes problemas. Mas para aplicar este teorema na qualidade de condições suficientes de optimalidade é necessário encontrar as sucessões $\{\lambda^k\}$, $\{x_k\}$ e $\{v_k\}$ que satisfazem as condições (2.16), o que nem sempre é fácil.

Consideremos o exemplo seguinte.

Exemplo 2.2.1.

$$\begin{aligned} \min \quad & x^2 + y \\ \text{s.a.} \quad & x \leq 0, y \leq 0, \frac{x}{i} - y \leq 0, i = 3, 4, 5, \dots \end{aligned}$$

Sejam $f(x, y) = x^2 + y$, $g_1(x, y) = x$, $g_2(x, y) = y$, $g_i(x, y) = \frac{x}{i} - y$ para todo $i = 3, 4, 5, \dots$, e seja $g = (g_i), i \in \mathbb{N}$. O conjunto admissível do problema é

$$X = \{(x, y) : g_i(x, y) \leq 0 \forall i \in \mathbb{N}\} = \{(x, y) \in \mathbb{R}^2 : x \leq 0, y = 0\},$$

e $\bar{x} = (0, 0)$ é um mínimo do problema. Note que $\nabla f(0, 0) = \{(0, 1)\}$ e que não existe nenhum $(x_0, y_0) \in \mathbb{R}^2$ satisfazendo $g_i(x_0, y_0) < 0, \forall i \in \mathbb{N}$. Então para cada $\{x_k\} \in X$,

$$\lambda_t^k \nabla_x g(x_k; t) = \lambda_1^n(1, 0) + \lambda_2^n(0, 1) + \sum_{i \geq 3} \lambda_i^n \left(\frac{1}{i}, -1\right).$$

Consequentemente,

$$v_n = (v_1^n, v_2^n) = \sum_{t \in M} \lambda_t^k \nabla_x g(x, t) \Leftrightarrow \begin{cases} v_1^n = \lambda_1^n + \sum_{i \geq 3} \lambda_i^n \frac{1}{i}, \\ v_2^n = \lambda_2^n - \sum_{i \geq 3} \lambda_i^n. \end{cases}$$

Tomemos $\bar{\lambda}_1 = \bar{\lambda}_2 = (0, 0, \dots, 0, \dots)$, $\bar{\lambda}_n = (0, 0, \dots, 0, 1 + \frac{1}{n}, 0, \dots)$, $n \geq 3$ onde a única componente não nula se encontra na n -ésima posição.

Então, pela construção,

$$v_n = \left(\frac{1 + \frac{1}{n}}{n}, -1 - \frac{1}{n} \right).$$

Evidente que $-v_n \rightarrow (0, 1) = \nabla f(0, 0)$, e $\lambda_k g(x_k) = (\frac{1}{k}, 0) \rightarrow 0$ quando $k \rightarrow \infty$.

Note-se que, neste exemplo, o conjunto M não é compacto. Pretendia-se apenas ilustrar as condições de optimalidade enunciadas.

2.3 Condições de Optimalidade Baseadas no Critério Implícito de Optimalidade

2.3.1 Critério Implícito de Optimalidade

Critério Implícito de Optimalidade, apresentado em [32] também permite formular condições de optimalidade sem recurso a CQ. Seja o problema convexo de PSI na forma (1.3) com $M \subset \mathbb{R}$ compacto. Supomos que as funções f, g são duas vezes continuamente diferenciáveis em x e que g é suficientemente diferenciável em t . consideramos a seguinte função:

$$\begin{aligned} N(q) &= \emptyset, & \text{se } q < 0, \\ N(q) &= \{0, 1, \dots, q\}, & \text{se } q \geq 0, \quad q \in \mathbb{Z}. \end{aligned}$$

Relembremos que $\bar{E} = \{\bar{t}_l, l \in L\}$, $\bar{E} \subset M$ representa o conjunto de restrições activas em \bar{x} definida por (1.4).

Dado $t \in M, x \in X$ seja $\rho = \rho(x, t) \in \{-1, 0, 1, \dots\}$ um número tal que

$$g^{(s)}(x, t) = 0, \quad s \in N(\rho), \quad g^{(\rho+1)}(x, t) \neq 0, \quad (2.17)$$

onde $g^{(s)}(x, t) = \frac{\partial^s g}{\partial t^s}$.

Definição 2.3.1. (Definição 2.1 em [32]) Seja $t \in M$. Um número $q(t) \in \{-1, 0, 1, \dots\}$ designa-se por ordem de imobilidade de t no problema SIP (1.3) se

1. para cada $x \in X$ é satisfeita

$$g^{(r)}(x, t) = 0, r \in N(q(t)); \quad (2.18)$$

2. existe $x(t) \in X$ tal que

$$g^{(q(t)+1)}(x(t), t) \neq 0. \quad (2.19)$$

Da definição anterior e das restrições do problema (1.3), segue que:

Se $t \in \text{int}(M)$, então $q(t) + 1$ é par e $g^{(q(t)+1)}(x(t), t) < 0$.

Definição 2.3.2. Um ponto $t \in M$ é designado de ponto imóvel do problema (1.3) se $q(t) > -1$.

O seguinte teorema está demonstrado em [32].

Teorema 2.3.1. Seja $\bar{x} \in X$ uma solução admissível, e $|\bar{E}(\bar{x})| < \infty$. Então \bar{x} é solução ótima do problema convexo PSI na forma (1.3) se e só se é solução ótima do seguinte problema não linear:

$$\begin{aligned} \min \quad & f(x), \\ \text{s.a} \quad & g^{(s)}(x, \bar{t}_l) = 0, s \in N(q_l), \\ & g^{(q_l+1)}(x, \bar{t}_l) \leq 0, l \in L. \end{aligned} \quad (2.20)$$

Onde q_l representa a ordem de imobilidade de \bar{t}_l . Note-se que o Teorema 2.3.1 é o critério que permite substituir as condições de optimalidade para o problema PSI convexo (problema infinito) pelas tais condições para o problema (2.20) - problema finito. Uma vez que as condições de optimalidade para PSI convexa estão formuladas no Teorema 2.3.1 na forma implícita, este Teorema é designado em [32] por Critério Implícito de Optimalidade. Designemos por:

$$\begin{aligned} b_{ls}(x) &= g^{(s)}(x, \bar{t}_l), s \in N(q_l), \\ c_l(x) &= g^{(q_l+1)}(x, \bar{t}_l), l \in L. \end{aligned}$$

Então o problema (2.20) pode ser escrito na forma:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.a} \quad & b_{ls}(x) = 0, s \in N(q_l), \quad l \in L \\ & c_l(x) \leq 0. \end{aligned} \quad (2.21)$$

O Lagrangeano para o problema (2.21) é dado por

$$\mathcal{L}(x, \lambda) = \lambda_0 f(x) + \sum_{l \in L} \left(\sum_{s \in N(q_l)} \lambda_{ls} b_{ls}(x) + \mu_l c_l(x) \right) \quad (2.22)$$

com o vector dos multiplicadores de Lagrange

$$\lambda = (\lambda_0, \lambda_{ls}, s \in N(q_l), \mu_l, l \in L). \quad (2.23)$$

Designamos o conjunto de restrições - desigualdades activas em \bar{x} por

$$J_A(\bar{x}) = \{l \in L : c_l(\bar{x}) = 0\} \quad (2.24)$$

e consideramos o conjunto

$$\mathcal{K}(\bar{x}) = \{\xi \in \mathbb{R}^n : \xi^T \nabla f(\bar{x}) \leq 0, \xi^T \nabla b_{is}(\bar{x}) = 0, s \in N(q_i), i \in L, \xi^T \nabla c_l(\bar{x}) \leq 0, l \in J_A(\bar{x})\} \quad (2.25)$$

que vamos designar por cone de direcções críticas para o problema (2.21) no ponto \bar{x}

Aplicandos os resultados conhecidos de optimalidade para o problema NLP na forma (2.21) e o Critério Implícito de Optimalidade podemos obter as condições explícitas de optimalidade para o problema PSI na forma (1.3).

Condições Necessárias para PSI Convexa

Teorema 2.3.2. *(Teorema 3.3 em [31]) Seja \bar{x} um mínimo local do problema de PSI na forma (1.3) com $|\bar{E}(\bar{x})| < \infty$. Então as seguintes condições são satisfeitas:*

1.

$$\Lambda_r \neq 0, \quad (2.26)$$

onde Λ_r representa o conjunto dos multiplicadores de Lagrange;

2.

$$\max_{\lambda \in \Lambda_r, \|\lambda\|=1} \xi^T \nabla_{xx}^2 \mathcal{L}(\bar{x}, \lambda) \xi \geq 0, \forall \xi \in \mathcal{K}(\bar{x}). \quad (2.27)$$

O Teorema 2.3.2 é obtido pela aplicação de condições conhecidas (KKT e de segunda ordem) de optimalidade de programação não linear para o problema. (2.20)

Comentário 2.3.1. *Note-se que as condições necessárias de optimalidade de primeira ordem formuladas no Teorema 2.3.2 não são informativas [31], uma vez que elas estão satisfeitas para cada $x \in X$. Isto acontece porque o problema (2.20) é degenerado (no sentido em que as condições clássicas de optimalidade de primeira ordem para o problema de PSI são sempre satisfeitas com o multiplicador de Lagrange da função objectivo a desaparecer, e como tal essas condições não são eficientes e não podem produzir condições de optimalidade eficientes para o problema de PSI, [32]). Para obter as condições necessárias eficazes de optimalidade usando o Critério Implícito de Optimalidade é preciso aplicar ao problema (2.20) as condições necessárias válidas para problemas degenerados (ver [1], [6], [8]). As condições de segunda ordem são difíceis de verificar.*

Condições Suficientes

Agora vamos estudar novas Condições Suficientes para PSI convexa obtidas em [32].

Dada uma solução admissível x do problema (2.21), construa-se o correspondente conjunto de restrições activas $J_A(x)$, o cone de direcções críticas $\mathcal{K}(x)$ e o conjunto dos multiplicadores de Lagrange $\Lambda(x)$ que satisfazem as condições necessárias de primeira ordem (2.24), (2.25) e (2.26) respectivamente.

A seguinte condição suficiente de optimalidade de segunda ordem pode ser formulada para o problema (2.20) na base de [1], [20].

Afirmção 2.3.1. *Seja \bar{x} uma solução admissível de (2.21), suponha-se que $\Lambda(\bar{x}) \neq \emptyset$ e*

$$\max_{\lambda \in \Lambda(\bar{x}), \|\lambda\|=1} \xi^T \nabla_{xx}^2 \mathcal{L}(\bar{x}, \lambda) \xi > 0, \forall \xi \in \mathcal{K}(\bar{x}). \quad (2.28)$$

Então \bar{x} é um mínimo local estrito de (2.21).

Com base no Critério Implícito de Optimalidade e na Afirmção 2.3.1 o seguinte teorema foi obtido em [31].

Teorema 2.3.3. *Seja \bar{x} uma solução admissível para o problema convexo PSI (1.3). Suponha-se que $p = |\bar{E}| < \infty$ e que uma das condições seguintes é válida:*

1. *(Condição Suficiente de Primeira Ordem) Existe um vector $\lambda^0 = (\lambda_0^0, \lambda_l^0, l = 1, \dots, p)$ satisfazendo $\lambda_0^0 > 0, \lambda_l^0 \geq 0, l = 1, \dots, p$ e $\nabla_x L(\bar{x}, \lambda^0) = 0$.*

2. (Condição Suficiente de Segunda Ordem) $\Lambda(\bar{x}) \neq \emptyset$ e a condição (2.28) é satisfeita.

Então \bar{x} é solução óptima do problema (1.3).

Notemos que quando o problema (1.3) satisfaz a Condição de Slater, a condição 1 do Teorema 2.3.3 é também condição necessária de optimalidade de primeira ordem.

Na prática a aplicação do Teorema 2.3.3 não exige a verificação da Condição de Slater, nem outra qualquer CQ e pode ser aplicada para qualquer problema de PSI convexa na forma (1.3).

No entanto, é evidente que para aplicar o Teorema 2.3.3 é preciso conhecer todos os pontos imóveis do problema (1.3) e as suas ordens de imobilidade.

Caso os pontos imóveis do problema (1.3) e as correspondentes ordens de imobilidade sejam conhecidos, as condições suficientes de optimalidade formuladas no Teorema 2.3.3 são fáceis de verificar e podem ser usadas com sucesso para a construção de métodos numéricos de PSI.

Capítulo 3

Métodos Gerais de Programação Semi-Infinita

Nos nossos dias as aproximações numéricas para os problemas de PSI tem-se imposto como uma área activa de investigação. Os novos métodos de resolução de diferentes problemas de PSI estão propostos em, por exemplo, [13].

As experiências numéricas com diferentes métodos de resolução de problemas de PSI são de grande importância. Em [53] pode-se encontrar problemas de teste para PSI e os resultados de experiências usando métodos diferentes.

O problema de criação de software eficaz que permite resolver problemas de PSI é muito importante. Recentemente foi incluído no servidor NEOS o programa NSIPS (*Nonlinear Semi-infinite Programming Solver*), codificado em C, que permite através do uso da interface SIPAMPL (nome dado pelo autor do software), resolver problemas de Programação Semi-infinita codificados em AMPL (*A Modeling Programming Language*), ver [53] e [52].

Como primeira observação, realçamos que, do ponto de vista numérico, problemas de SIP são muito mais difíceis do que os problemas finitos. O conjunto admissível de problemas infinitos tem uma estrutura muito mais complexa e irregular do que o conjunto admissível do problema finito.

Por exemplo, se num problema finito

$$\begin{aligned} \min \quad & f(x) \\ \text{s.a} \quad & g_j(x) \leq 0, \forall j \in J = \{1, \dots, m\}, \end{aligned} \tag{3.1}$$

para verificar a admissibilidade de $\bar{x} \in \mathbb{R}$ apenas é necessário calcular m valores das funções g_j em \bar{x} e verificar quais desses valores não são negativos, no problema de

PSI na forma (P), verificar a admissibilidade de \bar{x} é equivalente a resolver o problema $O(\bar{x})$ na forma (1.11) e apurar se para uma solução global \bar{t} , a condição $g(\bar{x}, \bar{t}) \leq 0$ é satisfeita.

Os métodos numéricos de resolução de problemas de PSI são baseados na substituição do problema infinito por um problema finito ou sucessão de problemas finitos. A principal distinção entre métodos de PSI está na escolha dos problemas finitos e em métodos de resolução destes problemas. De seguida apresentamos os três principais grupos de métodos numéricos de resolução dos problemas em PSI: os métodos de discretização, os métodos de troca e os métodos de redução local, os quais serão alvo mais à frente. Contudo, ao longo dos últimos tem-se vindo a desenvolver novos métodos de resolução, de que são exemplo, métodos de duas fases, métodos híbridos, entre outros.

Relativamente aos métodos de discretização, a ideia base consiste na substituição do conjunto infinito M por um conjunto finito de pontos $M_k \subset M$ que é usualmente uma grelha de pontos. A implementação de um método de discretização traduz-se na resolução de uma sucessão de subproblemas finitos baseados nos conjuntos $M_k (M_0 \subset M_1 \subset \dots \subset M_k \subset M)$, em que $M_k \rightarrow M$ com $k \rightarrow \infty$. A escolha da sucessão de conjuntos M_k deve garantir a convergência da sucessão de soluções dos subproblemas finitos para a solução do problema semi-infinito.

Os métodos de troca são métodos iterativos que também substituem as restrições infinitas por um conjunto de restrições finitas. Em cada iteração de um método de troca procede-se à remoção de algumas restrições e à inserção de outras restrições no conjunto das restrições finitas.

Os métodos de redução local, consistem na substituição das restrições infinitas do problema de PSI por um número finito de restrições considerando as soluções globais do problema de nível inferior, e na construção na base destas soluções do problema reduzido. As condições de optimalidade consideradas no Capítulo 1 garantem que para problemas semi-infinitos que satisfazem certas CQ, as condições de optimalidade coincidem com as condições de optimalidade para o problema reduzido.

Os três métodos seguintes requerem a condição adicional que as funções g sejam continuamente diferenciáveis relativamente a t .

Todos estes métodos estão baseados na ideia de substituição do problema (1.3) por um problema ou uma sequência de problemas finitos, ou seja, por problemas com um número finito de restrições. A partir daí podem-se aplicar algoritmos de

programação finita linear ou não-linear, conforme o caso, para a resolução do problema.

Definimos então um problema finito com restrições-desigualdade.

Definição 3.0.3. *Um problema finito (P_F) para problema (1.3) é um problema da forma:*

$$\begin{aligned} \min \quad & f(x) \\ x \in \mathbb{R}^n \quad & \\ \text{s.a} \quad & g(x, t) \leq 0, \forall t \in \hat{M}, |\hat{M}| < \infty, \end{aligned} \tag{3.2}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}, g : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}, \hat{M} \subset M$.

3.1 Métodos de Trocas

Chamamos método de troca a qualquer método que tenha por base o seguinte algoritmo ([18]):

Um Algoritmo de Troca [18]:

Passo (i) Dado $M_{i-1} \subset M, |M_{i-1}| < \infty$.

1. Determinar a solução x^i de $(P_F(M_{i-1}))$ e soluções t^1, \dots, t^{r_i} (aproximadas) do problema $O(x^i)$ na forma (1.11).
2. Se $g(x^i, t^l) \leq 0, l = 1, \dots, r_i$, PARAR.
3. Caso contrário, escolher M_i tal que

$$M_i \subseteq M_{i-1} \cup \{t^1, \dots, t^{r_i}\}. \tag{3.3}$$

4. Executar o passo (i+1).

Note-se que a expressão "algoritmo de troca" se refere ao facto de que, em cada passo, novas restrições (correspondentes a t^1, \dots, t^{r_i}) são acrescentadas e outras retiradas (correspondentes a $t \in M_{i-1}$), ou seja, existe uma troca de restrições.

Para a regra simples:

$$M_i := M_{i-1} \cup \{t^1, \dots, t^{r_i}\} \tag{3.4}$$

não temos, na realidade, troca de restrições, pois nenhuma é retirada.

Denotando por $v(x)$ o valor da função (1.11), temos o seguinte teorema que garante que o Algoritmo de Troca pára depois de um número finito de iterações.

Teorema 3.1.1. (Teorema 7.2 em [18]) Assumimos que existe $\Gamma > 0$ com $X \subset \{x : \|x\|_\infty \leq \Gamma\}$ e em cada passo a condição $v(x^i) = g(x^i, t^j)$ é satisfeita para pelo menos um $j \in \{1, \dots, r_i\}$. Então, o algoritmo de Troca, com a regra de troca (3.4) pára após um número finito de passos com a solução de (1.3) ou a sucessão $\{x^i\}$ tem pelo menos um ponto de acumulação e cada um deles é solução óptima de (1.3).

Se se usar a regra (3.4), o número de restrições a ser incluídas em $(P_F(M_i))$ aumenta consideravelmente e a determinação de uma solução global torna-se muito dispendiosa para os casos em que $M \subset \mathbb{R}^m, m \geq 2$, mas no caso de $m = 1$ este método é bastante eficaz.

Um exemplo de um método de troca é o método de planos de corte. Em [36] encontramos uma explicação desta forma de resolução em problemas lineares semi-infinitos e em problemas convexos semi-infinitos. Aqui ilustraremos apenas, de forma sucinta, o seu modo de funcionamento.

Método de Planos de Corte para problemas Lineares Semi-infinitos

Um problema linear (P_L) de programação semi-infinita pode ser formulado por:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.a} \quad & \sum_{r=1}^n a_r^i(t)x_r \leq b_i(t); i = 1, \dots, p; t \in M; |M| = \infty, \\ & x_r \geq 0; r = 1, \dots, n, \end{aligned} \quad (3.5)$$

onde $c \in \mathbb{R}^n$, S um compacto de \mathbb{R}^k e $b_i(t), a_r^i(t), i = 1, \dots, p, r = 1, \dots, n$; são elementos de $C^2(S)$.

Seja

$$G = \{x \in \mathbb{R}^n : \sum_{r=1}^n a_r^i(t)x_r \leq b_i(t); i = 1, \dots, p, t \in M\},$$

e

$$H = \{x \in \mathbb{R}^n : Ax \leq b; x \geq 0\},$$

onde A e b são matrizes respectivamente $m \times n$ e $m \times 1$, escolhidas de forma a que a condição $G \in H$ seja cumprida.

Seguidamente apresentamos o algoritmo de resolução:

Algoritmo 1. (i) Encontrar uma solução óptima do programa linear:

$$\begin{aligned} (P^0) \quad & \min \quad c^T x \\ & \text{s.a} \quad x \in H. \end{aligned} \quad (3.6)$$

1. Seja $x^k = (x_r^k, r = 1, \dots, n)$ uma solução óptima de (P^k) . Fazer

$$L^k = \max_{i=1, \dots, p} [\max_{t \in M} \sum_{r=1}^n (a_r^i(t)x_r^k - b_i(t))] = a_r^{i_k}(t^k)x_r^k - b_{i_k}(t^k).$$

2. Se $L^k \geq 0$, então escrever: " x^k é solução óptima de (3.5)." PARAR

Caso contrário, avançar para o passo 4.

3. Adicionar as restrições:

$$\sum_{r=1}^n (a_r^{i_k}(t^k)x_r) - b_{i_k}(t^k) \leq 0 \quad (3.7)$$

ao problema (P^k) . Seja (P^{k+1}) o problema resultante.

4. Resolver o problema (P^{k+1}) . Seja x^{k+1} a sua solução.

5. Fazer $k = k + 1$ e voltar ao passo 2.

A validação da regra de paragem do algoritmo anterior, acenta no teorema presente em [36]:

Teorema 3.1.2. *Seja x^k uma solução óptima do problema (P^k) , e assumimos que*

$$L^k = \max_{i=1, \dots, p} [\max_{t \in M} \sum (a_r^i(t)x_r^k - b_i(t))] \leq 0.$$

Então x^k é uma solução óptima de (3.5).

Método de Planos de Corte para problemas Convexos Semi-infinitos

Consideremos o problema de PSI convexa na forma equivalente a (1.3):

$$(P_C) \quad \begin{array}{ll} \min & f(x) \\ \text{s.a} & g_i(x, t) \leq 0; i = 1, \dots, p; t \in M \subset \mathbb{R}^n, |M| = \infty, \\ & x \in \mathbb{R}^n \end{array} \quad (3.8)$$

Assumimos também que $f, g_i, b_i, i = 1, \dots, p$ são duas vezes diferenciáveis.

O problema (3.8) pode ser escrito na forma:

$$\begin{array}{ll} \min & \lambda \\ & f(x) - \lambda \leq 0 \\ \text{s.a} & g_i(x, t) \leq 0; i = 1, \dots, p; t \in M \subset \mathbb{R}^n, |M| = \infty, \\ & x \in \mathbb{R}^n \end{array} \quad (3.9)$$

que é equivalente ao problema:

$$(P'_C) \quad \begin{array}{ll} \min & c^T y \\ \text{s.a} & g_i(y, t) \leq 0; i = 0, 1, \dots, p; t \in M \subset \mathbb{R}^k, |M| = \infty, \\ & y \in \mathbb{R}^{n+1} \end{array} \quad (3.10)$$

onde $c = (1, 0, \dots, 0) \in \mathbb{R}^{n+1}$, $y = (\lambda, x_1, \dots, x_n)$, $g_0(y, t) = f(x) - \lambda$, e $g_i(y, t) = g_i(x, t), i > 0$.

Seja $H = \{y \in \mathbb{R}^{n+1} : Ay \leq b\}$ onde A é uma matriz $(p+1) \times (n+1)$, $b \in \mathbb{R}^{p+1}$ e H é definida tal que a condição seguinte é satisfeita:

$$G = \{y \in \mathbb{R}^{n+1} : g_i(y, t) \leq 0; i = 0, 1, \dots, p; t \in M\} \subset H$$

Apresentamos agora um algoritmo para a resolução do problema (P'_C) .

Algoritmo 2. 1. Encontrar uma solução ótima y^0 de :

$$\min_{y \in H} c^T y. \quad (3.11)$$

2. Fazer $k=0$. Seja y^k uma solução ótima, fazer:

$$L_c^k = \max_i [\max_{t \in M} (g_i(y^k, t))] = g_{i_k}(y^k, t^k). \quad (3.12)$$

3. Se $L_c^k \geq 0$, então escrever: " y^k é solução ótima de (P'_C) ." PARAR.

Caso contrário, avançar para o passo 4.

4. Adicionar a restrição

$$g_i(y^k, t^k) + [\nabla_y g_i(y^k, t^k)]^T (y - y^k) \leq 0, \quad (3.13)$$

a (P_C^k) . Seja (P_C^{k+1}) o problema resultante.

5. Resolver (P_C^{k+1}) . Seja y^{k+1} a sua solução. Fazer $k = k + 1$, e voltar ao passo 3.

Para justificar a regra de paragem do algoritmo, usaremos o seguinte resultado [36]:

Teorema 3.1.3. Se uma solução ótima y^k de (P_C^k) é tal que:

$$L_c^k = \max_{i=0,1,\dots,p} [\max_{t \in M} (g_i(y^k, t))] \leq 0. \quad (3.14)$$

Então y^k é solução ótima de (P_C) .

Ao adicionar umas restrições e retirar outras em cada iteração estamos a realizar cortes na região admissível, daí a designação planos de corte.

Os algoritmos apresentados representam apenas um exemplo do modo como se podem criar planos de corte para a resolução dos problemas semi-infinitos.

É possível encontrar, na literatura sobre PSI, (por exemplo [14], [5], [43] e [44]), outras formas de obter planos de corte para a resolução dos nossos problemas.

3.2 Métodos de Discretização

Um método de discretização, resume-se, de uma forma geral, à introdução em M de uma grelha de pontos, seguindo-se a resolução do problema finito nessa grelha. Sendo descritos para $M \subset \mathbb{R}$, estes métodos têm a forma seguinte. Consideremos, para $h \in \mathbb{R}$ de "passos" $h > 0$ e um $t^0 \in \mathbb{R}$ fixo, a grelha definida por:

$$G_h = \{t : t = t^0 + \alpha_j h, \alpha_j \in \mathbb{Z}\} \quad (3.15)$$

e

$$M_h = M \cap G_h. \quad (3.16)$$

Temos então o seguinte algoritmo que traduz a ideia do método de discretização:

Algoritmo do Método de Discretização

(i) Dado h^i , uma selecção de pontos \widetilde{M}_{h^i} da grelha M_{h^i} e uma solução \widetilde{x}^i do problema seguinte

$$(P_F(\widetilde{M}_{h^i})) \quad \begin{array}{l} \min_{x \in \mathbb{R}^n} f(x) \\ \text{s.a. } g(x, t) \leq 0, t \in \widetilde{M}_{h^i}. \end{array}$$

1. Seja $h^{i+1} = (1/n_i)h^i$, ($n_i \in \mathbb{N}$, $n_i \geq 2$).
2. Escolher um conjunto de pontos da grelha $\widetilde{M}_{h^{i+1}} \subset M_{h^{i+1}}$ (com base em \widetilde{x}^i , \widetilde{M}_{h^i} e, eventualmente nos valores testados anteriormente para, \widetilde{x}^{i+1} , $\widetilde{M}_{h^{i+1}}$).
3. Calcular a solução \widetilde{x}^{i+1} de $(P_F(\widetilde{M}_{h^{i+1}}))$. Se \widetilde{x}^{i+1} é solução de $(P_F(M_{h^{i+1}}))$ continuar em (4), caso contrário, repetir (2).
4. Se $i > i_0$ (onde i_0 representa o número de passos escolhido inicialmente) PARAR.

Caso contrário avançar para o passo $i + 1$.

Relativamente à eficiência dos métodos de discretização, é fundamental fazer uso do máximo de informação possível das grelhas anteriores quando se resolve $(P_F(\widetilde{M}_{h^i}))$, dado que $M_{h^i} \subset M_{h^{i-1}}$, x^{i-1} é geralmente um bom ponto de partida para resolver $(P_F(\widetilde{M}_{h^i}))$.

A informação sobre as grelhas anteriores deve ser usada no passo (2) para a eliminação de restrições de $(P_F(M_{h^i}))$. A maneira mais fácil de seleccionar $\widetilde{M}_{h^i} \subset M_{h^i}$ é a escolha

$$\widetilde{M}_{h^i} \supset M_{h^i}^\gamma = \{t : t \in M_{h^i}, g(\widetilde{x}^{i-1}, t) \leq -\gamma\}, \quad (3.17)$$

com $\gamma > 0$ sendo uma constante qualquer escolhida inicialmente.

A escolha de γ em (3.17) é muito importante. Um γ muito grande implica a existência de muitas restrições em $(P_F(\widetilde{M}_{h^i}))$. No entanto, se γ for muito pequeno, temos também um efeito negativo nos problemas seguintes, uma vez que partes de Y podem ser esquecidas.

Para um conhecimento mais alargado sobre algoritmos de discretização, ver, por exemplo, o trabalho de Polak e He ([41]).

3.3 Métodos Baseados em Redução Local

Os métodos de redução local podem ser descritos pelo seguinte algoritmo.

Um Algoritmo de Métodos Baseados em Redução Local

(i) Dado x^i (não necessariamente admissível),

Fazer:

1. Determinar todos os máximos locais t^1, \dots, t^{r_i} do problema (1.11) (procura multi-local)
2. Aplicar k_i passos de um algoritmo de programação não linear ao problema reduzido:

$$(P_{red}(x^i)) \quad \min\{f(x) : g(x, t^l(x)) \leq 0, l = 1, \dots, r_i\} \quad (3.18)$$

Seja $x^{i,k}$ a solução de (3.18) no passo $1 \leq k \leq k_i$

3. Supôr $x^{i+1} = x^{i,k_i}$ e continuar no passo $(i + 1)$

O subpasso (1) é muito dispendioso, já que requer a procura global de máximos globais de $g(x^i, t)$, em $M \subset \mathbb{R}^m$. A estratégia passa por evitar a execução deste passo o máximo possível. Além disso, este subpasso assume a existência de um número finito de máximos locais, caso não seja essa a situação, devem usar-se outros métodos, por exemplo, o de discretização.

No subpasso (2), é necessário resolver o problema finito (3.18). No entanto, isto apenas requer procura local, o que normalmente pode ser feito de forma eficiente, usando, por exemplo, o método de Newton.

3.4 Outros Métodos

São variadas as tentativas de resolução dos problemas de SIP, dependendo da classe de funções em que o problema se insere, por exemplo, LSIP, problemas com funções convexas e muitos outros.

Aqui pretendemos apenas abordar de forma breve alguns dos métodos mais referenciados que não foram mencionados nas secções anteriores.

3.4.1 Métodos de duas Fases

Os métodos de duas fases combinam o método de discretização na sua fase inicial, denominada primeira fase, e o método de redução local numa fase posterior, designada por segunda fase. O processo de troca entre a discretização e a redução local é considerado por alguns autores uma terceira fase, como se pode comprovar em [16] e [12].

3.4.2 Métodos de direcções admissíveis (descendentes)

Estes métodos consistem em partir de um ponto dado admissível (não óptimo) x_k , para o ponto seguinte $x_{k+1} = x_k + \tau_k d_k$ de tal forma que x_{k+1} permaneça admissível e tenha um valor, na função objectivo, menor.

A escolha mais simples será mover ao longo de uma direcção descendente definida por vector d que satisfaz

$$\nabla f(\bar{x})d < 0, \nabla_x g(\bar{x}, t)d \leq 0, \forall t \in \bar{E}(\bar{x}). \quad (3.19)$$

Os fundamentos dos métodos descendentes podem ser encontrados em vários artigos da literatura como [40], [26],[51].

Mais métodos vão surgindo à medida que a investigação na área da otimização semi-infinita avança, sendo exemplo, métodos híbridos, que combinam mais do que uma forma no seu desenvolvimento, métodos de purificação, entre outros.

Capítulo 4

Algoritmo de Determinação das Ordens de Imobilidade

Neste capítulo apresentamos o algoritmo proposto em [32]. Dado um problema convexo de PSI, este algoritmo permite verificar se este problema satisfaz a condição de Slater e determina os pontos imóveis do problema e as suas ordens de imobilidade.

4.1 Descrição do Algoritmo

Consideremos o problema convexo PSI na forma (1.3) com conjunto admissível X .

Fazemos então a seguinte suposição:

Suposição 4.1.1. *Suponha-se que $X \neq \emptyset$ e que existe $\hat{x} \in X$ tal que $g(\hat{x}, t) \neq 0, t \in M$.*

Suponhamos que a suposição 4.1.1 é válida para este problema. Escolhemos qualquer $\bar{x} \in X$ com um conjunto finito de pontos activos $\bar{E}(\bar{x}) = \{t_l, l \in L\}$, $L = L(\bar{x}) = \{1, 2, \dots, \bar{p}\}$, $\bar{p} = p(\bar{x}) < \infty$.

O algoritmo seguinte é designado por DIO - Determination of Immobility Orders (em inglês).

Algoritmo de Determinação de Ordens de Imobilidade (DIO)

Suponha-se, $k = 0$ e $q_i^{(0)} = -1, \forall i \in L$.

A k -ésima iteração começa com um conjunto de números $q_i^{(k)}, i \in L$, construído na iteração anterior do algoritmo. Introduzimos os conjuntos

$$X_i^{(k)} = \{z \in \mathbb{R}^n : g^{(s)}(z, t_i) = 0, s \in N(q_i^{(k)}), g^{q_i^{(k)}+1}(z, t_i) \leq 0\}, i \in L; \quad (4.1)$$

$$X^{(k)} = \bigcap_{i \in L} X_i^{(k)}. \quad (4.2)$$

Para cada $i \in L$, resolve-se o problema de programação não linear

$$\begin{aligned} \min_z \quad & g_i^{(k)}(z) = g_i^{(q_i^{(k)+1})}(z, t_i), \\ \text{s.a.} \quad & z \in X^{(k)}. \end{aligned} \quad (4.3)$$

Como $\bar{x} \in X^{(k)}$ então $X^{(k)} \neq \emptyset$ e ou o problema (4.3) admite uma solução óptima ou a sua função objectivo $g_i^{(k)}(z)$ não é limitada inferiormente no conjunto admissível $X^{(k)}$. Designando por $x^{(i)}$ a solução óptima do problema (4.3), no caso dessa solução existir. Caso contrário designa-se por $x^{(i)}$ qualquer solução admissível do problema (4.3) que satisfaz a desigualdade $g_i^{(k)}(x^{(i)}) < 0$.

Considere-se o conjunto $L^k = \{i \in L : g_i^{(k)}(x^{(i)}) = 0\}$.

Se $L^{(k)} = \emptyset$, então o algoritmo pára obtendo os seguintes valores de $q(t), t \in M$:

$$q(t_i) = q_i^{(k)}, i \in L; \quad q(t) = -1, t \in M \setminus \bar{E}(\bar{x}). \quad (4.4)$$

Se $L^{(k)} \neq \emptyset$, então faz-se:

$$q_i^{(k+1)} = q_i^{(k)} + 2, i \in L^{(k)}; \quad q_i^{(k+1)} = q_i^{(k)}, i \in L \setminus L^{(k)}, \quad (4.5)$$

e passa-se à próxima iteração com $k = k + 1$.

Fica assim o algoritmo descrito.

4.2 Propriedades do Algoritmo DIO

O algoritmo DIO possui as propriedades que a seguir apresentamos.

Propriedade 1. *O problema (4.3) é convexo.*

Esta propriedade é importante do ponto de vista técnico, uma vez que a convexidade dos conjuntos X^k e das funções $g_i^{(k)}(z)$ é usada na demonstração do Teorema 2.3.1 (Critério Implícito de Optimalidade) e do ponto de vista prático, uma vez que na resolução do problema (4.3) podem ser aplicados os métodos específicos de programação convexa.

Propriedade 2. *O algoritmo DIO é finito. De facto, se designarmos por $k_* \in \mathbb{N}$, o número das suas iterações, verifica-se que a seguinte estimativa é verdadeira:*

$$k_* \leq \sum_{i \in L} \frac{\rho(\bar{x}, t_i) + 1}{2}.$$

O teorema seguinte de [32] prova que o Algoritmo DIO encontra de facto as ordens de imobilidade.

Teorema 4.2.1. *Dado $t \in M$, o valor $q(t)$ construído pelo Algoritmo DIO satisfaz a Definição 2.3.1.*

Comentário 4.2.1. *O Teorema 4.2.1 também estabelece que existe sempre um vector \tilde{x} que satisfaz (2.18) e (2.19) para todo $t \in M$ simultaneamente. Sendo assim, na Definição 2.3.1, podemos sempre supor que $x(t) \equiv \tilde{x}, \forall t \in M$.*

Propriedade 3. *Segue da Definição 2.3.1 e Comentário 4.2.1 que as restrições do problema (1.3) satisfazem a condição de Slater se e só se $q(t) = -1, \forall t \in M$. Logo, o algoritmo permite verificar se um dado problema satisfaz a Condição de Slater, uma vez que esta condição é uma das mais fortes CQ [37], podendo assim concluir que, se o problema satisfaz a Condição de Slater, então esse problema satisfaz outra qualquer CQ.*

4.3 Exemplos de Aplicação do Algoritmo DIO

Esta secção é dedicada à ilustração do Algoritmo DIO através da resolução de exemplos onde se realizam todos os passos do algoritmo.

Exemplo 4.3.1. *Seja $g(x, t) = -t^2x_1 + tx_2 + \sin(t)x_3 + x_4^2$, $t \in [-1, 2]$, $x \in \mathbb{R}^4$.*

Consideremos a solução admissível $\bar{x} = (4, 1, -1, 0)$. Então obtemos:

$$g(\bar{x}, t) = -4t^2 + t - \sin(t), \quad t \in [-1, 2].$$

De acordo com as notações $\bar{E}(\bar{x}) = \{0\} = \{t_i, i \in I\}$, $I = \{1\}$, $t_1 = 0$.

Calculando as derivadas de g em ordem a t :

$$\nabla_t g(\bar{x}, t) = -8t + 1 - \cos(t) \Rightarrow \nabla_t g(\bar{x}, 0) = 0;$$

$$\nabla_{tt} g(\bar{x}, t) = -8 + \sin(t) \Rightarrow \nabla_{tt} g(\bar{x}, 0) = -8.$$

Logo $\bar{\rho}(\bar{x}, 0) = 1$

Considerando $t_1 = 0$, obtemos $g(x, t_1) = x_4^2$.

- $k = 0$, $q_1^{(0)} = -1$.

$$X^{(0)} = \{x \in \mathbb{R}^4 : x_4^2 \leq 0\}.$$

Pretendemos então resolver o seguinte problema:

$$\begin{aligned} \min \quad & x_4^2 \\ \text{s.a} \quad & x \in X^{(0)}. \end{aligned}$$

Obtendo como solução $x^{(1)} = (4, 1, -1, 0)$ com valor $g(x^{(1)}) = 0$,

$$I^{(0)} = \{i \in I : g(x^{(1)}) = 0\} = \{1\}.$$

Como $I^{(0)} \neq \emptyset$ então passamos à próxima iteração.

- $k = 1, q_1^{(0)} = 1$

Como $\nabla_t g(x, t) = -2tx_1 + x_2 + \cos(t)x_3$ e $\nabla_{tt}g(x, t) = -2x_1 - \sin(t)x_3$ vem que:

$$X^{(1)} = \{x \in \mathbb{R} : x_4^2 = 0, x_2 + x_3 = 0, -2x_1 \leq 0\}.$$

Vamos então resolver o problema:

$$\begin{aligned} \min \quad & -2x_1 \\ \text{s.a} \quad & x \in X^{(1)}. \end{aligned}$$

A solução encontrada é $x^{(1)} = 10^{16} \times (4.8038, 0, 0, 0)$ com valor óptimo

$$g(x^{(1)}) = -9.6077 \times 10^{16}$$

$I^{(1)} = \emptyset$ logo o algoritmo PÁRA.

Podemos resumir os resultados na seguinte tabela:

k	$q_1^{(k)}$	$g^{(q_1^{(k)}+1)}(x^{(1)}, t_1)$	$I^{(k)}$
0	-1	0	{1}
1	1	-9.6077×10^{16}	\emptyset

Neste exemplo verificamos que apenas existe um ponto imóvel $t = 0$ com ordem de imobilidade 1. Sendo assim a função restrição deste exemplo satisfaz a Condição de Slater.

O gráfico seguinte mostra o ponto $t = 0$ como ponto imóvel da função restrição g .

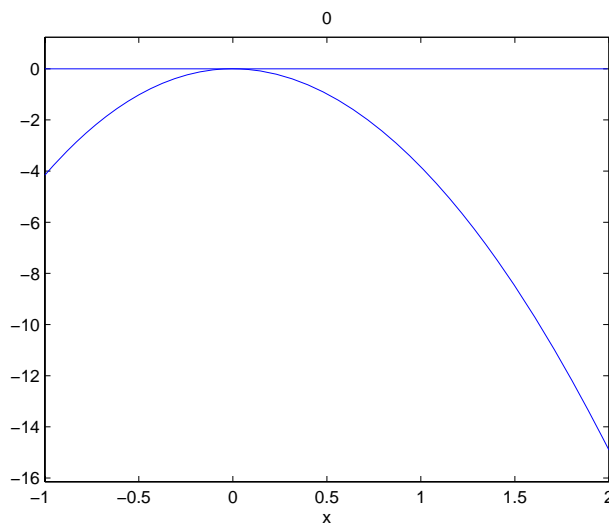


Figura 4.1: Gráfico de g para $\bar{x} = (4, 1, -1, 0)$

De seguida apresentamos outro exemplo.

Exemplo 4.3.2. Seja $g(x, t) = 2x_1^2t^2 - t^4 + x_1^2 - x_2$, $t \in [-1, 1]$, $x \in \mathbb{R}^2$.

Consideremos a solução admissível $\bar{x} = (\frac{\sqrt{2}}{2}, \frac{3}{4})$. Então obtemos:

$$g(\bar{x}, t) = t^2 - t^4 - \frac{1}{4}, \quad t \in [-1, 1].$$

De acordo com as notações $\bar{E}(\bar{x}) = \{-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\} = \{t_i, i \in I\}$, $I = \{1, 2\}$, $t_1 = -\frac{\sqrt{2}}{2}$, $t_2 = \frac{\sqrt{2}}{2}$.

Calculando as derivadas de g em ordem a t :

$$\begin{aligned} \nabla_t g(\bar{x}, t) &= 2t - 4t^3 \Rightarrow \nabla_t g(\bar{x}, \frac{\sqrt{2}}{2}) = 0 \text{ e } \nabla_t g(\bar{x}, -\frac{\sqrt{2}}{2}) = 0; \\ \nabla_{tt} g(\bar{x}, t) &= 2 - 8t^2 \Rightarrow \nabla_{tt} g(\bar{x}, \frac{\sqrt{2}}{2}) = -2 \text{ e } \nabla_{tt} g(\bar{x}, -\frac{\sqrt{2}}{2}) = -2. \end{aligned}$$

Logo $\bar{\rho}(\bar{x}, \frac{\sqrt{2}}{2}) = 1$ e $\bar{\rho}(\bar{x}, -\frac{\sqrt{2}}{2}) = 1$.

Sendo $t_1 = -\frac{\sqrt{2}}{2}$, e $t_2 = \frac{\sqrt{2}}{2}$ obtemos $g(x, t_1) = 2x_1^2 - x_2 - \frac{1}{4}$ e $g(x, t_2) = 2x_1^2 - x_2 - \frac{1}{4}$

- $k = 0$, $q_1^{(0)} = -1$ e $q_2^{(0)} = -1$

$$\begin{aligned} X_1^{(0)} &= \{x \in \mathbb{R}^2 : 2x_1^2 - x_2 - \frac{1}{4} \leq 0\}, \quad X_2^{(0)} = \{x \in \mathbb{R}^2 : 2x_1^2 - x_2 - \frac{1}{4} \leq 0\} \\ X^{(0)} &= X_1^{(0)} \cap X_2^{(0)} = \{x \in \mathbb{R}^2 : 2x_1^2 - x_2 - \frac{1}{4} \leq 0\} \end{aligned}$$

Pretendemos então resolver o seguinte problema:

$$\begin{aligned} \min \quad & 2x_1^2 - x_2 - \frac{1}{4} \\ \text{s.a.} \quad & x \in X^{(0)}. \end{aligned}$$

Obtendo como solução $x^{(1)} = 10^{15}(0, 1.1480)$ com valor $g(x^{(1)}) = -1.1480 \times 10^{15}$.

Como a função $g(\bar{x}, t_1) = g(\bar{x}, t_2)$ então as soluções são as mesmas para ambos $t_i, i \in I$, ou seja, $x^{(2)} = 10^{15}(0, 1.1480)$ com valor $g(x^{(2)}) = -1.1480 \times 10^{15}$.

$I^{(0)} = \{i \in I : g(x^{(i)}) = 0\} = \emptyset$, logo algoritmo PÁRA.

Resumimos os resultados na tabela:

k	$q_1^{(k)}$	$g^{(q_1^k+1)}(x^{(1)}, t_1)$	$q_2^{(k)}$	$g^{(q_2^k+1)}(x^{(2)}, t_1)$	$I^{(k)}$
0	-1	-1.1480×10^{15}	-1	-1.1480×10^{15}	\emptyset

Neste exemplo verificamos que não existem pontos imóveis, o que significa que a Condição de Slater é verificada neste problema, como é possível ver pelo gráfico seguinte.

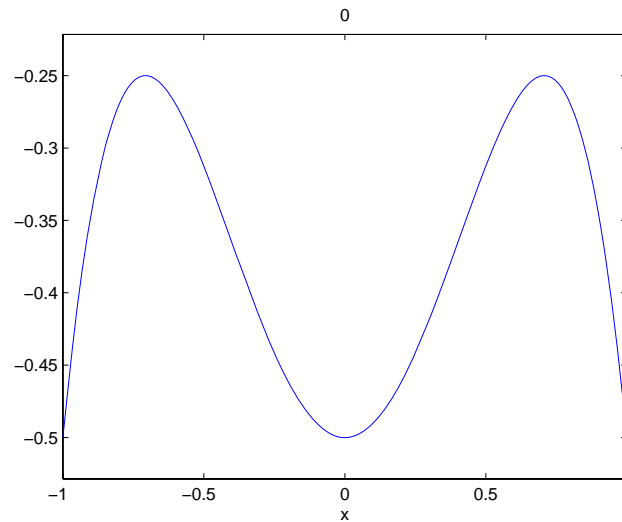


Figura 4.2: Gráfico de g para $\bar{x} = (\sqrt{2}/2, 1)$

Com \bar{x} considerado para a representação da função é possível observar que $g(\bar{x}, t) < 0, \forall t \in [-1, 1]$ e como tal, satisfaz a Condição de Slater.

4.4 Resultados Numéricos

A implementação do algoritmo DIO foi realizada em MATLAB 7.0, num computador Pentium M, 1.4GHz. Para testar o algoritmo foram utilizados os problemas da bibliografia [30], [35], [4], [18], [36], [42], [31] e por fim alguns problemas da base de problemas de PSI construída em [53].

Note-se que durante muito tempo não existia uma base de dados para problemas de PSI. Esta base de dados foi construída em [53] com a análise de bibliografia existente, sendo constituída por 144 problemas.

Notemos que o Algoritmo DIO, na sua versão apresentada na secção 4.1 é aplicável apenas aos problemas de PSI convexa na forma (1.3) onde M é um subconjunto compacto de \mathbb{R} . Daí surge um problema de limitação do número de problemas para testar.

Em cada iteração do algoritmo resolve-se o problema finito não linear (4.3). Para a resolução deste problema foi usado o pacote *Optimization Toolbox* do MATLAB.

Nos exemplos a seguir vamos considerar apenas restrições dos problemas de PSI convexa. Dada a função $g(x, t)$, o conjunto $M \subset \mathbb{R}$ compacto e o ponto admissível inicial \hat{x} , o algoritmo de pontos imóveis I_M de M e as suas ordens de imobilidade $q(t), t \in I_M$.

A seguir apresentamos as funções usadas nos testes. Mas antes começamos por fazer uma referência mais específica da origem dos problemas.

Os problemas P1 e P2 foram recolhidos de [30]. O problema P3 foi retirado de [49], P4 é um exemplo do apêndice de [4], o problema P5 foi retirado de [18], de [36] foram retirados os problemas P6 e P7. Os problemas P8 e P9 estão presentes em [42] e por fim os problemas P10 e P11 foram retirados de [31].

Todos os restantes problemas usados nos testes, que possuem um nome de referência, foram retirados da base de dados construída por [53].

Problemas P1-P11:

$$P1 : \quad g(x, t) = -tx_1 - t^3x_2 - 1, \quad t \in \left[-\frac{3}{2}, \frac{3}{2}\right], x \in \mathbb{R}^2$$

$$P2 : \quad g(x, t) = -x_1 - tx_2 - t^2x_3 + t^3 - t, \quad t \in [-2, 2], x \in \mathbb{R}^3$$

$$P3 : \quad g(x, t) = x_1(1 - t)(1 + t) + t^2(t^2 - x_2), \quad t \in [-1, 1], x \in \mathbb{R}^2$$

$$P4 : \quad g(x, t) = -(x_1 - t)^2 - x_2, \quad t \in [0, 1], x \in \mathbb{R}^2$$

$$P5 : \quad g(x, t) = 2tx_1 + x_2 - t^2, \quad t \in [-1, 1], x \in \mathbb{R}^2$$

$$P6 : \quad g(x, t) = -x - t - 1, \quad t \in [0, 1], x \in \mathbb{R}$$

$$P7 : \quad g(x, t) = x^2 - 1 - t^2 - t, \quad t \in [0, 1], x \in \mathbb{R}$$

$$P8 : \quad g(x, t) = t - e^{x_1+x_2}, \quad t \in [0, 1], x \in \mathbb{R}^2$$

$$P9 : \quad g(x, t) = x_1 + x_2e^{x_3t} + e^{2t} - 2\sin(4x), \quad t \in [0, 1], x \in \mathbb{R}^3$$

$$P10 : \quad g(x, t) = -t^2x_1 + tx_2 + \sin(t)x_3 + x_4^2 - t^2 - \frac{t^3}{6} - 5t^4, \quad t \in [-1, 2], x \in \mathbb{R}^4$$

$$P11 : \quad \begin{aligned} g(x, t) = & 18[(t - 0.14)^6(t - 0.6)^2(t - 0.4)^4(x_1^2 + (x_2 + \frac{1}{3})^2 + x_3^2 + (x_4 - 4)^2 - 1) \\ & + (t - 0.14)^4(1 - \cos(t - 0.6))\sin^4(t - 0.94)((x_1 + x_2 + x_3 + \frac{1}{2})^2 - 1) \\ & + \sin^4(t - 0.14)(t - 0.94)^2\sin^2(t - 0.94)((x_2 + x_4 - 3)^4 + 4x_1^2x_3^2 - 1)], \\ & t \in [0, 1], x \in \mathbb{R}^4 \end{aligned}$$

A tabela seguinte representa os resultados do teste realizados aos problemas referenciados anteriormente. Nela constam o ponto inicial, \hat{x} utilizado e o(s) respectivo(s)

valor(es) de t , os pontos imóveis e as suas ordens de imobilidade e por fim o número de iterações realizadas pelo algoritmo.

Resultados do teste dos conjuntos admissíveis pelo Algoritmo DIO

Problema	\hat{x}/t obtido	Pontos Imóveis	Ordem de Imobilidade	Nº iterações
Hettich 11	$(0, 0)/t = 0$	\emptyset	-1	1
leon 12	$(0, 1)/t = \pi/2$	\emptyset	-1	1
leon 13	$(1, 1)/t = 1$	\emptyset	-1	1
leon 14	$(2, 0)/t = 0$	\emptyset	-1	1
leon 15	$(-1, 1)/t = 0$	\emptyset	-1	1
leon 16	$(1, 2, 0)/t = 0$	\emptyset	-1	1
leon 17	$(1, 0, 0)/t = 1$	\emptyset	-1	1
leon 18	$(-1, 1)/t = 1$	\emptyset	-1	1
Pricek	$(-1, 0)/t = \pi$	\emptyset	-1	1
Powell	$(0, 1)/t = \frac{1}{4}$	\emptyset	-1	1
Watson 1	$(0, 0)/t = 0$	$\{0\}$	1	2
Watson 2	$(1, \frac{1+\sqrt{5}}{2})/t = 0$	\emptyset	-1	1
Watson 3	$(-\frac{3}{2}, -1, 2)/t = -\frac{1}{4} \arcsin(\frac{3}{4})$	\emptyset	-1	1
Watson 6	$(0, 0)/t = 0$	\emptyset	-1	1
Zhou 1	$(1, 1)/t = 1$	\emptyset	-1	1
Userman	$(-10, 0)/t = 0$	\emptyset	-1	1
P1	$(\frac{2}{3}, 0)/t = -\frac{3}{2}$	\emptyset	-1	1
P2	$(4, -1, 1)/t = 2$	\emptyset	-1	1
P3	$(0, 1)/t \in \{-1, 0, 1\}$	\emptyset	-1	1
P4	$(1, 0)/t = 1$	\emptyset	-1	1
P5	$(1, -1)/t = 1$	\emptyset	-1	1
P6	$1/t = 0$	\emptyset	-1	1
P7	$1/t = 0$	\emptyset	-1	1
P8	$(0, 0)/t = 1$	\emptyset	-1	1
P9	$(-e^2, 0, e^2)/t = 1$	\emptyset	-1	1
P10	$(-1, 1, -1, 0)/t = 0$	$\{0\}$	1	2
P11	$(0, 0.5, 0, 3.5)/t \in \{0.6, 0.14, 0.94\}$	$\{0.14, 0.95\}$	$\{3, -1, 3\}$	3

Pela análise da tabela de resultados é possível observar que a maior parte dos problemas testados não possui pontos imóveis, e como consequência, satisfazem a Condição de Slater.

Conclusões e Trabalho Futuro

O estudo da teoria e métodos de PSI convexa efectuados nesta dissertação revelou que a construção dos métodos baseados em redução local de problemas semi-infinitos a problemas finitos, continua a ser uma questão em aberto.

Grande parte dos métodos numéricos existentes, são baseados na discretização que nem sempre pode garantir a convergência. Entretanto, muitos destes métodos permitem obter boas aproximações da solução dos problemas SIP e são bastante eficazes.

Os métodos baseados na redução local envolvem as CQ que o conjunto admissível deve satisfazer. Estas condições nem sempre são satisfeitas e nem sempre são fáceis de verificar.

Por isso a nossa atenção foi concentrada nos métodos de PSI (caso mais simples de problemas convexos com continuum de restrições), que não exigem que o problema satisfaça nenhuma CQ.

O estudo de alguns destes métodos foi realizado e um dos métodos estudados tem por base a ideia de redução local usando o Critério Implícito de Optimalidade que permite transformar um problema de PSI convexo num problema de Programação Não Linear.

Uma das propriedades importante do Critério Implícito de Optimalidade é que permite o seu funcionamento sem o recurso a nenhuma CQ. Isso faz com que as condições de optimalidade para problemas convexos de SIP se reduzam às condições para problemas de programação não linear. A implementação deste método está baseada na noção de pontos imóveis e as suas ordens de imobilidade e envolve a aplicação do algoritmo DIO proposto em [31].

O algoritmo DIO determina os pontos imóveis e as suas ordens de imobilidade num número finito de iterações e permite verificar se a Condição de Slater é ou não satisfeita num problema convexo de PSI.

A aplicação do Critério Implícito de Optimalidade possibilita desenvolver novas condições mais eficientes para os problemas SIP.

O principal objectivo deste trabalho passava pelo estudo e implementação do algoritmo DIO, e realização de experiências numéricas para testar todos os problemas de PSI convexa com continuum de restrições da literatura e da base de dados, [53], do ponto de vista de existência nestes problemas, pontos imóveis.

Fica como possível trabalho futuro o alargamento da implementação do algoritmo DIO em problemas que possuam mais do que uma função restrição, caso de $m > 1$ e também construção de métodos numéricos de PSI convexa baseados no Critério Implícito de Optimalidade.

Bibliografia

- [1] Arutiunov, A.V, Optimality Conditions: Abnormal and Degenerate Problems. *Kluwer Academic Publishers*. 2000.
- [2] Ben-Tal, A., Ben-Israel, A., Zlobec, S., Characterization of Optimality in Convex Programming Without a Constraint Qualification. *Journal of Optimization Theory and Applications*. :vol. 20, 417-437, 1976.
- [3] Ben-Tal, A., Kerzner, L., Zlobec, S., Optimality Conditions for Convex Semi-infinite Programming Problems. *Naval Research Logistics Quarterly*. :vol. 27, 1980.
- [4] Bhattacharjer, B., Lemonidis, P., Green Jr., W.H., Barton, P.I., Global Solution of Semi-Infinite Programs. *Springer-Verlag*. : 2005.
- [5] Blankenship, J.W., Falk, J.E., Infinitely Constrained Optimization Problems. *Journal of Optimization Theory and Applications*.:vol. 19, 261-281, 1976.
- [6] Bonnans, J.F., Shapiro, A., Perturbation Analysis of Optimization Problems. *Springer-Verlag, New-York*. 2000.
- [7] Borwein, J.M., Wolkowicz, H., Characterizations of Optimality Without Constraint Qualifications for the Abstract Convex Program. *Math. Programming Stud.* : 19, 77-100, 1982.
- [8] Brezhneva, O.A., Tretyakov, A., Optimality Conditions for Degenerate Extremum Problems with Equality Constraints. *SIAM J: Control Optimization* . :vol. 42, 729-745, 2003.
- [9] Charnes, A., Cooper, W.W., Kortanek, K.O., On representations of semi-infinite programs which have no duality gaps. *Management Science*. :vol. 12, 113-121, 1965.

- [10] Czyzyk, J., Mesnier, M., Moré, J., The NEOS Server. *IEEE Journal on Computational Science and Engineering*. 5, 68-75, 1998.
- [11] Dambrine, M., Pierre, M., About Stability of Equilibrium Shapes. *Mathem. Modelling and Num. Analysis*. :vol. 34, 811-834, 2000.
- [12] Glashoff, K., Gustafson, S.A., Linear Optimization and Approximation. *Springer-Verlag, Berlin*. 1983.
- [13] Goberna, M. A., Linear Semi-infinite Programming in the 21th Century . 2006.
- [14] Gribik, P.R., A Central-cutting-plane Algorithm for Semi-infinite Programming Problems. . 66-82, 1979.
- [15] Gribik, P.R., Kortanek, K.O., Equivalence Theorems and Cutting-plane algorithms for a class of experimental design problems. *SIAM J. Appl. Math*,32. :vol. 32, 232-256, 1977.
- [16] Gustafson, S.A., A three-phase Algorithm for Semi-infinite Programs, Semi-infinite Programming and Applications. *Lecture Notes Econ. Math. Syst.* :vol. 215, 136-157, 1983.
- [17] Hettich, R.,Jungen, H.TH., Semi-infinite Programming: Conditions of Optimality and Applications. *Optimization Techniques, J. Stoer, Lectures Notes in Control and Inform.* , 1-11, 1978.
- [18] Hettich, H., Kortanek, K.O., Semi-Infinite Programming: Theory, Methods and Applications. *SIAM Review*. :vol. 35, 380-429, 1993.
- [19] Hettich, R., Zencke, P., Numerische Methoden der Approximation und Semi-infiniten Optimierung. *Teubner Studienbücher Mathematic, Stuttgart.* , 1982.
- [20] Ioffe, A.D., Necessary and Sufficient Conditions for a Local Minima.3. Second Order Conditions and Augmented Duality. *SIAM J. Control Optim.* :vol. 17, 266-288, 1979.
- [21] Jane, J. Ye, Soon-Yi Wu, First Order Optimality Conditions for Generalized Semi-Infinite Problems.

- [22] Jerez, B., A Dual Characterization of Incentive Efficiency. *Journal Econom. Theory.* :vol. 112, 1-34, 2003.
- [23] Jeyakumar, V., Asymptotic Dual Conditions Characterizing Optimality for Convex Programs. *Journal Optim. Theory Appli.:* 14, 153-165, 1997.
- [24] Jeyakumar, V., Lee, G.M.,Dinh, N., New Sequential Lagrange Multiplier Conditions Characterizing Optimality Without Constraint Qualification for Convex Programs. *SIAM Journal Optim.:*vol. 14, 534-547,2003
- [25] Jongen, H., Rückmann, J.J., Stein, O., Generalized Semi-Infinite Optimization: A First-Order Optimality Condition and Examples. *Mathematical Programming.* :vol. 83, 145-158, 1998.
- [26] Kaplan, A., Tichatschke, R., Iterative Methods for Solving Incorrect Convex Variational Problems. *Sov. Dokl.* :vol. 315, 275-278, 1990.
- [27] Kortanek, K.O., Medvedev, V.G., Building and Using Dynamic Interest Rate Models *J. Wiley, Chichester.* 2001.
- [28] Kortanek, K.O., Yamasaki, M., Semi-Infinite Transportation Problems. *Academic Press, Inc.* 1982.
- [29] Kortanek, K.O., Zhang, Q., Perfect Duality in Semi-infinite and semidefinite programming. *Math. Programming.* :vol. 91A, 127-144, 2001.
- [30] Kortanek, K.O.,Semi-infinite Programming. .
- [31] Kostyukova, O.I., Tchemisova, T.V., Convex Semi-Infinite Programming: Explicit Optimality Conditions, *Cadernos de Matemática- Série de Investigação, University of Aveiro,* 1-20, 2006.
- [32] Kostyukova, O.I., Tchemisova, T.V., Yermalinskaya, S.A., On the Algorithm of Determination of Immobile Indices for Convex SIP Problems. *Accepted to publication by IJAMAS- International Journal of Applied Mathematics and Statistics,* 2007.
- [33] Krabs, W., Optimization and Approximation. *John Wiley and Sons, Chichester, New York, Brisbane, Toronto.,* 1979.

- [34] Krishnan, K., Mitchell, J.E., A Unifying Framework for Several Cutting Plane Methods for semidefinite programming. *Optimization Methods and Software*. :vol. 21, 57-74, 2006.
- [35] Lopez, M., Still, G., Semi-infinite Programming. . 2005.
- [36] Luhandjula, M.K., Ouanes, M., A Cutting-Plane Approach for Semi-infinite Mathematical Programming. *African Journal of Science and Technology, Science and Engi. S.* :vol. 2, 1-10, 2001.
- [37] Mangasarian, O. L., Fromovitz, S., The Fritz John Necessary Optimality Conditions in the Presence of Equality and Inequality Constraints. *Journal of Mathematical Analysis and Applications*. :vol. 17, 37-47, 1967.
- [38] Mangasarian, O. L., Nonlinear Programming. *Siam, Philadelphia*. 1994.
- [39] Marin, S.P., Optimal Parametrization of Curves for Robot Trajectory design. *IEEE Trans. On Automatic Control*. :vol. AC-33, 209-214, 1988.
- [40] Polak, E., On the Mathematical Foundation of Nondifferentiable Optimization in Engineering design. *SIAM Review*. :vol. 29, 21-89, 1987.
- [41] Polak, E., He, L., Rate preserving discretization strategies for semi-infinite programming and optimal control. *Memo. No. UCB/ERL M89/112, Univ, of California, Berkeley, CA*. 1989.
- [42] Qi, L., Wu, S-Y, Zhou, G., Semismooth Newton Methods for Solving Semi-Infinite Programming Problems. *Applied Mathematics Report*. , 1-20, 2001.
- [43] Reemtsen, R., A Cutting-plane Method for Solving Minimax Problems in the Complex Plane. *Numerical Algorithms*. . :vol. 2, 409-436, 1992.
- [44] Reemtsen, R., Some Outer Approximation Methods for Semi-infinite Optimization Problems. *Journal of Computational and Applied Mathematics*. :vol. 53, 87-108, 1994.
- [45] Rogosinsky, W.W, Moments of Non-negative Mass. *Proc. Roy. Soc. London Ser. A* :vol. 245, 1-27, 1958.

- [46] Rückmann, J.J., Shapiro, A., First-order Optimality Conditions Generalized Semi-infinite Programming, *Journal of Optimization Theory and Applications*, vol. 10, 677-691, 1999.
- [47] Soriano, J., Llorca, N., Tijs, S., Timmer, J., Semi.infinite Assignment and Transportation Games *Journal of Economic Literature Classification*. 1-11.
- [48] Stein, O., On Constraint Qualification in Nonsmooth Optimization, *Journal of Optimization Theory and Applications*, 647-671, 2004.
- [49] Still, G.J., Discretization in Semi-infinite Programming: the rate of approximation. , 2001.
- [50] Thibault, L., Sequential Convex Subdifferential Calculus and sequential Lagrange Multipliers. *SIAM Journal Control Optim...*:35, 1434-1444, 1997.
- [51] Tichatschke, R., Kirsten, H., About the Efficiency of a method of feasible directions for solving variational inequalities, in Recent Advances and Historical Development of vector Optimization, J. Jahn and W. Knabs,eds. *Lecture Notes in Econom. and Math. Systems, Springer-Verlag, Berlin, Heidelberg, New York*. :vol. 194, 379-400, 1987.
- [52] Vaz, A., Fernandes, E., Gomes, M., A sequential quadratic programming with a dual parametrization approach to nonlinear semi-infinite programming. *Top 11 (1)*. 109-130, 2003.
- [53] Vaz, A., Fernandes, E., Gomes, M., SIPAMPL: Semi.Infinite Programming with AMPL, *ACM trans. Math. Software 30 (1)*. 47-61, 2004.
- [54] Yang, Q., A New Proof of the Strong Duality Theorem for Semidefinite programming. *J. Math. Anal. Appl.* :vol. 303, 622-626, 2005.

Apêndice A: Programa

Um dos objectivos deste trabalho era a implementação do Algoritmo DIO, para a determinação, caso existam, dos pontos imóveis das funções restrição e suas respectivas ordens de imobilidade.

A seguir explica-se como essa implementação foi realizada, e apresenta-se o código.

A primeira parte do algoritmo consiste em definir a função $g(x, t)$ e encontrar um ponto admissível. Para tal é necessário começar por definir as variáveis x_1, x_2, \dots, x_n . Depois de introduzida a função $g(x, t)$ e um ponto \hat{x} , é produzido o gráfico da função $g(\hat{x}, t)$, onde se pode verificar se a solução é ou não admissível.

```
clear all
syms x1 x2 x3 x4 t x
func_inicial=input('Escrever a função inicial:'); % Inserir a
função inicial g(x,t)
%f_principal=input('Função principal:')
x0=input('Ponto inicial:'); % Inserir um ponto para ver se é admissível
f_aux=subs(func_inicial,[x1,x2,x3,x4],x0) % Calculo de g(x0,t)
a=input('Limite Inferior:'); % Limite inferior do intervalo T
b=input('Limite Superior:'); % Limite inferior do intervalo T
ezplot(f_aux,[a,b]) % Representação da função g(x0,t)
hold on
ezplot(0,[a,b])
hold off
```

Se ponto escolhido é não é admissível, fazemos a escolha correspondente e o programa pára. Se o ponto é admissível então optamos pela outra opção e o programa

continua.

Sub-Programa: admissi

```
function wordVer = admissi
% Para escolher a admissibilidade do ponto inicial.
% Continuar a perguntar até obter uma resposta válida.

repeatQuery = 1;
while(repeatQuery)
    repeatQuery = 0;
    disp('[1] Se a solução não é admissível');
    disp('[2] Se a solução é admissível');
    wordVerNum = input(sprintf('Decisão de admissibilidade: '));
    switch(wordVerNum)
        case(1)
            wordVer = 'Não é admissível';
            error([sprintf('Escolher novo ponto admissivel')]);
        case(2)
            wordVer = 'É admissível';
            return;
        otherwise
            repeatQuery = 1;
            disp([10 sprintf('Invalid selection') 10]);
    end
end
```

Após a decisão de admissibilidade, é necessário determinar os candidatos a pontos imóveis, para tal o sub-programa "solution", resolve esse problema.

Sub-programa: Solution

```
function s=solution(f,x0)
syms x1 x2 x3 x4 t
f_aux=subs(f,[x1 x2 x3 x4],x0)
t=solve(f_aux);
fprintf('As soluções são'); t
```

Depois de se obter os pontos imóveis, é preciso calcular a expressão da função $g(x, t_i)$ nesses pontos e verificar se $g \equiv 0$. Caso não seja, resolve-se o problema de minimização, que corresponde ao problema (4.3), usando o pacote *Optimization Toolbox*.

Caso o valor da função objectivo seja diferente de zero, o programa pára, caso contrário, continua.

Se $g \equiv 0$, então o mínimo da função objectivo é obviamente 0, e o programa calcula as próximas derivadas, até que pelo menos umas das derivadas dos respectivos t_i seja diferente de zero.

Após excluirmos todos os valores t_i para os quais o mínimo é zero o algoritmo pára e obtemos como resultado as ordens de imobilidade e os pontos imóveis.

Apresenta-se a seguir o código completo, mas antes apenas um comentário.

Inicialmente é necessário abrir os ficheiros necessários para a introdução das funções objectivo e das restrições que serão utilizados na resolução dos vários problemas de minimização.

Programa Algoritmo DIO

```
clear all
syms x1 x2 x3 x4 t x
% Inserir a função inicial
func_inicial=input('Escrever a função inicial:'); g(x,t)
% Inserir um ponto para ver se é admissível
x0=input('Ponto inicial:');

f_aux=subs(func_inicial, [x1,x2,x3,x4],x0) % Calculo de g(x0,t)
a=input('Limite Inferior:'); % Limite inferior do intervalo T
b=input('Limite Superior:'); % Limite inferior do intervalo T
ezplot(f_aux, [a,b]) % Representação da função g(x0,t)
hold on
ezplot(0, [a,b])
hold off

%-----
```



```

admissi                                % Escolha da admissibilidade

sprintf('Press any key to continue...')
pause

solution(func_inicial,x0)              % Função para calcular os zeros de g(x0,t)

sprintf('Press any key to continue...')
pause
%-----

ti=input('0 valor de ti:')              % Zeros da função, t_is
df_i=func_inicial;
i=-1;
dft_i=0;
% Cálculo das derivadas da função f(x,ti), e quando elas valem zero

    while dft_i==0
        df_i=diff(df_i,t,i+1);
        dft_i=subs(df_i,t,ti)
        i=i+1;
    end
i=i-1;
%-----
if i==-1
    fprintf('Introduzir no ficheiro objfun.m a função objectivo:\n')
    pause
    x0=input('Ponto inicial:')
    options = optimset('LargeScale','off');
    [x,fval,exitflag,output] = fminunc(@objfun,x0,options)
else
    fprintf('Introduzir no ficheiro confunexp3.m as restrições:\n')
    fprintf('Introduzir no ficheiro objfunexp3.m a função objectivo:\n')

```

```

    pause
% Parte para obter o mínimo das funções df(x,ti)
    x0=input('Ponto inicial:')
    options=optimset('LargeScale','off');
    [x, fval]=...
fmincon(@objfunexp3,x0,[],[],[],[],[],[],@confunexp3,options)
end
%fprintf('Voltar a realizar o mesmo processo mas sem o valor de
ti\n para o qual a derivada é diferente de zero\n')

%-----
% Cálculo das derivadas da função f(x,ti), e quando elas valem zero

if fval==0
    df_i=diff(df_i,t);
    dft_i=subs(df_i,t,ti)
    fprintf('Introduzir no ficheiro confunpp.m as restrições:\n')
    fprintf('Introduzir no ficheiro objfunexpp.m a função objectivo:\n')
    x0=input('Ponto inicial:')
    options=optimset('LargeScale','off');
    [x, fval]=...
fmincon(@objfunpp,x0,[],[],[],[],[],[],@confunexpp,options)
fprintf('A ordem de imobilidade do ponto ti é:');i+2
else
    fprintf('A ordem de imobilidade do ponto ti é:');i+1
end

```

Fica assim descrito o programa.

Apêndice B: Aplicações de PSI

É extenso o conjunto de aplicações de PSI. Os modelos de PSI podem ser encontrados na economia (incluindo finanças), teoria dos jogos, geometria, probabilidades e estatística (e.g., teoria de risco e robustez da estatística Bayesiana), desenho óptimo (engenharia), aproximação de funções (Chebychev), espectrometria, controlo óptimo, optimização combinatória, robótica, problemas de transporte, citando apenas os mais comuns.

Seguidamente apresentamos descrições de alguns dos exemplos supracitados e recolhidos de, entre outros, [18], [13], [35].

Investigação Operacional e Optimização Combinatória

Na área da Investigação Operacional encontram-se os exemplos de problemas semi-infinitos de transportes e afectação.

Problema Semi-infinito de Transportes

Num problema de transportes, pretende-se maximizar o lucro de transporte de um produto qualquer, de um determinado número de fornecedores para um determinado número de consumidores.

Os problemas semi-infinitos de transporte, são problemas onde o número de um tipo de agentes (fornecedores ou consumidores, por exemplo) ou ambos, é infinito. Por exemplo, este tipo de situação aparece em modelos de mercado onde o número de potenciais consumidores pode ser visto como infinito. Quando uma empresa introduz um produto novo no mercado, cada consumidor que faz parte do conjunto infinito de potenciais compradores tem uma procura finita para esse produto.

Um problema de transportes pode ser descrito por (P, Q, T, s, d) onde o vector $s = \{s_i\}_{i \in P}$ representa a oferta e o vector $d = \{d_j\}_{j \in Q}$ representa a procura.

Assumamos que pelo menos um dos conjuntos $Q \subset \mathbb{R}$ e $P \subset \mathbb{R}$ é infinito e que os custos $C = [c_{ij}]$ de uma unidade de produto a ser transportada são limitados, isto é, $\|C\|_\infty < \infty$.

Um *plano de transporte* $X = [x_{ij}]_{i \in P, j \in Q}$ é uma matriz com entradas inteiras onde x_{ij} é o número de unidades do bem que será transportado do ponto de oferta i para o ponto de procura j . No problema de transportes x é uma incógnita a determinar e obviamente, cada fornecedor $i \in P$ não pode oferecer mais do que s_i unidades do produto, ou seja,

$$\sum_{j \in Q} x_{ij} \leq s_i, i \in P.$$

Da mesma forma, cada consumidor $j \in Q$ pretende receber, no máximo, d_j unidades, o que implica

$$\sum_{i \in P} x_{ij} \leq d_j, j \in Q.$$

O custo mínimo que os pontos de oferta e procura podem alcançar é:

$$\begin{aligned} \min \quad & \sum Cx_{ij} \\ & \sum_{j \in Q} x_{ij} \leq s_i, i \in P \\ \text{s.a.} \quad & \sum_{i \in P} x_{ij} \leq d_j, j \in Q \\ & x_{ij} \geq 0. \end{aligned} \tag{6}$$

Evidentemente que o problema de transporte é um problema de PSI Linear e pode ser resolvido por um dos métodos de PSI conhecidos. Caso as variáveis x_{ij} sejam inteiras, temos problema semi-infinito de transportes com variáveis inteiras e nesse caso os métodos específicos devem ser aplicados (ver [28]).

Problemas semi-infinitos de afectação

Um problema de afectação, é um tipo especial de problema de transporte, onde tanto a procura como a oferta são iguais a um.

Neste caso estamos interessados em emparelhar, por exemplo, um conjunto finito de máquinas com um conjunto de tarefas, de forma a obtermos o maior lucro possível. Pensemos numa empresa com um número finito de máquinas de corte de vidro, que podem ser programadas para produzir um vaso. A empresa pode escolher entre um número infinito de formas e as máquinas podem produzir todas essas formas, mas com lucros diferentes dependendo da forma. Cada máquina produz apenas um tipo de vasos. Mostra-se então necessário resolver o problema de afectar um número

finito de máquinas a um número infinito de possíveis desenhos, uma vez que o seu objectivo é maximizar o lucro ao emparelhar as máquinas às formas.

Um problema semi-infinito de afectação, neste caso, pode-se denotar pelo terno (P, Q, A) , onde $P = \{1, 2, \dots, m\}$ é um conjunto finito de máquinas, $Q = \mathbb{N}$ o conjunto infinito, $A = [a_{ij}], i = 1, \dots, m, j \in Q$ uma matriz com um número infinito de colunas e cujos elementos a_{ij} representam os lucros que são limitados, para todo $i \in P, j \in W$. Faremos uso de A para representar o problema de afectação (M, Q, A) .

Um *plano de afectação* $Y = [y_{ij}]_{i \in P, j \in Q}$ é uma matriz de 0s e 1s onde $y_{ij} = 1$, se i é afectado a j (máquina i produz vasos da forma j) e $y_{ij} = 0$ caso contrário. Cada elemento de P é afectado por um elemento de Q e cada elemento de Q é afectado, no máximo, por elemento de P de forma que temos:

$$\sum_{j \in Q} y_{ij} \leq 1 \text{ e } \sum_{i \in P} y_{ij} \leq 1.$$

Então

$$v_p(A) = \sup \left\{ \sum_{(i,j) \in P \times Q} a_{ij} y_{ij} : Y \text{ é um plano de afectação} \right\} \quad (7)$$

é o menor limite superior dos lucros que a empresa pode obter. Um plano de afectação Y é também chamado de solução para A . Tal solução é óptima se

$$\sum_{(i,j) \in P \times Q} a_{ij} y_{ij} = v_p(A). \quad (8)$$

Optimização Combinatória

Relativamente à optimização combinatória, temos como exemplo o problema de corte máximo de um grafo, que consiste em maximizar o peso total dos cortes nos grafos e surge, entre outros campos, na electrónica e na estatística ligada à física. Por outro lado, os algoritmos dos problemas de corte máximo são boas ferramentas para a partição de grafos e outros problemas combinatórios.

Problema de Optimização de Forma

Consideremos um problema de minimização de aproximação por defeito proposto em [35] em que a sua resolução nos leva a um modelo de um problema de PSI.

A título de exemplo vejamos o "problema de optimização da forma" (shape optimization problem), considerado em [11]:

Encontrar uma região (conexa) $G \in \mathbf{R}^m$, com volume normalizado $\mu(G) = 1$, e uma função $u \in C^2(\text{cl}G, \mathbf{R})$ que resolve o seguinte problema de otimização com função objectivo $F(G, u)$:

$$\begin{aligned} \min_{G,u} \quad & F(G, u) \\ \text{s.a.} \quad & \Delta u(t) = K, \forall t \in G, \\ & u(t) = 0, \forall t \in \text{fr}(G), \\ & \mu(G) = 1. \end{aligned} \tag{9}$$

onde Δu representa o Laplaciano ($\Delta u = \frac{\partial^2 u}{\partial t_1^2} + \frac{\partial^2 u}{\partial t_2^2} + \dots + \frac{\partial^2 u}{\partial t_m^2}$) e K uma constante positiva.

Este problema tem uma região G variável e pode ser resolvido, aproximadamente, pelo seguinte problema GSIP:

Escolher uma família apropriada de regiões $G(z)$, dependendo do parâmetro $z \in \mathbf{R}^p$ e satisfazendo $\mu(G(z)) = 1$, para todo z . Fixar um erro (pequeno) $\varepsilon > 0$. Por fim resolver o problema:

$$\begin{aligned} \min_{z,x} \quad & F(G(z), u(x, \cdot)) \\ \text{s.a.} \quad & \pm(\Delta u(x, t) - K)\varepsilon, \forall t \in G(z), \\ & \pm u(x, t) \leq \varepsilon, \forall t \in \text{fr}(G(z)). \end{aligned} \tag{10}$$

onde funções $u(x, t)$ são obtidas por experimentação do conjunto

$$S = \{u(x, t) = \sum_{i=1}^n x_i u_i(t)\}. \tag{11}$$

Aplicações à Robotica

Controlo da Trajectoria de Robots

O controlo da trajectoria de robôs fornece muitos exemplos de PSI. Por exemplo, para evitar obstáculos, cada ponto do braço de um robot, tem que estar fora de uma determinada região, em todo o instante de tempo e a aceleração fornecida por um motor que faz o braço funcionar é limitada.

Consideremos o seguinte modelo, dado em [39]. Suponhamos que damos a um robô a posição de controlo descrita pelas coordenadas $\Theta_1, \dots, \Theta_r$ (que representam, por exemplo, os ângulos das juntas, o comprimento das partes do braço). Assumamos um caminho $\Theta(\tau), \tau \in [0, 1]$. Então a reparametrização $t = h(\tau), \tau = h^{-1}(t)$

para o tempo é feita tal que $\tilde{\Theta}(t) = \Theta(h^{-1}(t))$ representa um movimento executável em que o tempo realizado $T = h(1)$ é minimizado.

Existem diferentes formas de definir as funções h . Em [39], por exemplo, estas são restritas às funções da forma $h(\tau) = \int_0^\tau \phi(s)ds$ onde $\phi(s) = \sum_{j=1}^n z_j B_j(s)$, representa uma spline cúbica num conjunto equidistante de $n - 4$ nós em $(0, 1)$.

No caso mais simples, $\tilde{\Theta}(t)$ é considerado um movimento executável se para constantes $c_{ij}, i = 1, \dots, r, j = 1, 2, 3$, dadas, temos

$$\left| \frac{d^j \tilde{\Theta}_i(t)}{dt^j} \right| \leq c_{ij}, j = 1, 2, 3, \quad (12)$$

ou seja, se as velocidades, acelerações e movimentos bruscos são limitados.

Desta forma, o problema de reparametrização é do tipo:

$$\begin{aligned} \min \quad & c^T z \\ \text{s.a.} \quad & g_l^{i,j}(z, \tau) \leq 0, \tau \in [0, 1], i = 1, \dots, r, j = 1, 2, 3, l = 1, 2, \end{aligned} \quad (13)$$

onde $g_l^{i,j}$ dependem não-linearmente de z para $j = 2, 3$. Obtemos então um problema de PSI não linear.

Problema da Manobrabilidade

Outro problema ligado à robótica prende-se com a manobrabilidade, que formulamos como foi proposto em [35]

Representemos por $\Theta = \Theta(\tau) \in \mathbf{R}^m$ a posição do ponto central do robot, ou seja, as coordenadas da sua posição, para o tempo τ . Sejam $\dot{\Theta}$ e $\ddot{\Theta}$ as correspondentes velocidades e acelerações respectivamente. As equações da dinâmica tomam, muitas vezes, a forma $g(\Theta, \dot{\Theta}, \ddot{\Theta}) = A(\Theta)\ddot{\Theta} + F(\Theta, \dot{\Theta}) = K$, com forças externas $K \in \mathbf{R}^m$. $A(\Theta)$ representa a matriz da inercia e F descreve a fricção, gravidade, força centrífuga e restantes forças que possam existir. As forças K são limitadas: $K^- \leq K \leq K^+$. Para $\Theta, \dot{\Theta}$ fixos, o conjunto admissível de aceleração é dado por:

$$Z(\Theta, \dot{\Theta}) = \{\ddot{\Theta} : K^- \leq g(\Theta, \dot{\Theta}, \ddot{\Theta}) \leq K^+\}.$$

Uma vez que g é linear em $\ddot{\Theta}$, para $\Theta, \dot{\Theta}$ fixos, o conjunto $Z(\Theta, \dot{\Theta})$ é convexo. Considerando, por exemplo, a "região de operação", dada por:

$$Q = \{(\Theta, \dot{\Theta}) \in \mathbf{R}^{2m} : (\Theta^-, \dot{\Theta}^-) \leq (\Theta, \dot{\Theta}) \leq (\Theta^+, \dot{\Theta}^+)\},$$

então, o conjunto de acelerações admissíveis $\ddot{\Theta}$, ou seja, o conjunto de acelerações que são permitidas em cada ponto $(\Theta, \dot{\Theta}) \in Q$, pode ser descrito na forma:

$$Z_0 = \bigcap_{(\Theta, \dot{\Theta}) \in Q} Z(\Theta, \dot{\Theta}) = \{\ddot{\Theta} : K^- \leq g(\Theta, \dot{\Theta}, \ddot{\Theta}) \leq K^+, \forall (\Theta, \dot{\Theta}) \in Q\}. \quad (14)$$

O conjunto Z_0 é convexo, pois é a intersecção de conjuntos convexos.

Para controlar o robô é necessário verificar se uma determinada aceleração $\ddot{\Theta}$ é possível, isto é, se está satisfeita a inclusão: $\ddot{\Theta} \in Z_0$.

Na prática este teste leva muito tempo, devido à descrição complicada de Z_0 , daí utilizarmos um teste mais rápido: Primeiro, procuramos um conjunto T , tão grande quanto possível de forma a que esteja contido em Z_0 . De seguida verificamos se $\ddot{\Theta} \in T$.

A construção de $T \subset Z_0$ traduz-se num problema generalizado de PSI (GSIP), que formulamos a seguir.

Admitamos então que o conjunto $T(d)$ depende do parâmetro $d \in \mathbf{R}^q$ e que $V(d)$ é o volume de $T(d)$. O que pretendemos maximizar é o volume sujeito à condição $T \subset Z_0$, obtendo assim o problema de manobrabilidade:

$$\begin{aligned} \max \quad & V(d) \\ \text{s.a.} \quad & K^- \leq g(\Theta, \dot{\Theta}, \ddot{\Theta}) \leq K^+, \forall (\Theta, \dot{\Theta}) \in Q, \ddot{\Theta} \in T(d). \end{aligned} \quad (15)$$

Aproximação de Chebyshev

Dada uma função $f \in C(\mathbb{R}^n)$ e um conjunto de funções aproximadas $p(x, \cdot) \in C(\mathbb{R}^n, \mathbb{R}^m)$ parametrizadas por $t \in \mathbb{R}^m$, pretendemos aproximar f pelas funções $p(x, \cdot)$ usando a norma Chebyshev (ou norma *max*), $\|f\|_\infty = \max_{t \in T} |f(t)|$ definida num conjunto compacto $T \subset \mathbf{R}^m$. Minimizar o erro de aproximação dado por $\varepsilon = \|f - p\|_\infty$ é um problema que pode ser expresso na forma do problema de PSI:

$$\begin{aligned} \min_{x, \varepsilon} \quad & \varepsilon \\ \text{s.a.} \quad & g^\pm(x, t) = \pm(f(t) - p(x, t)) \leq \varepsilon, \forall t \in T. \end{aligned} \quad (16)$$

No problema inverso de Chebyshev fixamos o erro de aproximação ε e fazemos a região T o maior possível. Suponhamos que $T = T(d)$ é parametrizada por $d \in \mathbf{R}^k$ e que $V(d)$ representa o volume de $T(d)$. Este problema leva-nos ao problema de

GSIP (com $\varepsilon > 0$ pequeno e fixo)

$$\begin{aligned} \max_{d,x} \quad & V(d) \\ \text{s.a.} \quad & \pm(f(t) - p(x, t)) \leq \varepsilon, \forall t \in T(d), \end{aligned} \tag{17}$$

onde $T(d)$ depende da variável d .

Economia e Finanças

Na área da economia e finanças, certos "problemas de decisão de risco" foram inicialmente formulados, por alguns autores, como problemas de LSIP sem, no entanto, usarem essa designação. Da mesma forma, em vez de recorrerem aos processos clássicos de teoria estocástica à matemática financeira, [27] formularam e resolveram, modelos de taxas de interesse dinâmicas como problemas específicos de LSIP, onde o sistema de restrições era analítico e satisfazia as CQ de Farkas-Minkowsky [9].

Um outro exemplo prende-se com a informação assimétrica que gera selecção adversa e problemas morais de risco.

A caracterização da economia sob informação simétrica (por exemplo, mercados competitivos), é um problema desafiante. Em [22] foram caracterizadas localizações eficientes neste ambiente servindo-se da teoria da dualidade dos problemas de LSIP.

Apresentamos de seguida exemplos de aplicações nesta área.

Portfolio Problem

Imaginemos que pretendemos investir K euros em n tipos de acções de várias companhias, digamos por um período de um ano. Investimos x_i euros numa acção i e esperamos, no final do período, um retorno de t_i euros por cada euro investido na acção i .

O nosso objectivo é maximizar o valor do portfolio $V = t^T x$ após um ano onde $x = (x_1, \dots, x_n)$ e (t_1, \dots, t_n) . O problema é que os valores t_i não são conhecidos inicialmente (se fossem investiríamos tudo na acção com valor t_i máximo). Não obstante, o conhecimento do passado e de modelos económicos permite-nos prever que os coeficientes de ganho t_i variam entre certos valores: $a_i \leq t_i \leq b_i$, $i = 1, 2, \dots, n$. Então podemos assumir, que o vector t estará contido num conjunto compacto $T \subset \mathbb{R}^n$: $T = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n]$.

Assim sendo, podemos formular o nosso problema, onde pretendemos maximizar o ganho de V , para o pior caso do vector $t \in T$, na forma:

$$\begin{aligned} \max_{v,x} \quad & V = t^T x \\ \text{s.a.} \quad & t^T x - V \geq 0, \forall t \in T \\ & \sum_i x_i = K, x \geq 0 \end{aligned} \tag{18}$$

Optimização sob Medidas de Probabilidades

Sejam $f_j, j = 1, \dots, n$ funções contínuas num conjunto compacto B . Consideremos o vector de n^2 funções definidas por $a(t) = (f_1^2(t), f_1(t)f_2(t), \dots, f_i(t)f_j(t), \dots, f_n^2(t))$.

Seja $M_{n^2} = \text{co}\{a(t) : t \in B\}$ um caso especial de cone convexo, designado por "moment cone".

Nas aplicações em probabilidade e estatística interessa-nos o invólucro convexo de $\{a(t) : t \in B\}$ que pode ser escrito na forma:

$$\Omega = \left\{ \omega = \sum_{t \in B} a(t) \mu(t) : \mu \in \mathbb{R}_+^B, \sum_{t \in B} \mu(t) = 1 \right\}$$

que representa um conjunto convexo.

Análogo a um resultado de Rogosinsky [45], Ω pode ser escrito de forma equivalente com medidas regulares não negativas de Borel em Ω .

Utilizando o Lema 6.1[18], a classe dos problemas de regressão estatística de desenho experimental pode ser formulado por $\inf\{\sigma(\omega) : \omega \in \Omega\}$, onde σ é uma função convexa em Ω , que satisfaz certas condições de diferenciabilidade, mas σ pode ser infinita sobre a fronteira de Ω .

Para ilustração, assumimos que σ é continuamente diferenciável num conjunto aberto de \mathbb{R}^{n^2} que contém o subconjunto de Ω sobre o qual σ é finito. Apresentamos um procedimento geral (ver [15]) para obter um sistema linear semi-infinito de desigualdades que caracteriza o valor óptimo ω^* .

O procedimento aplica-se ao caso geral onde σ pode não estar definido em ω^* . Mencionamos anteriormente que uma forma intuitiva de caracterizar um $\omega^* \in \Omega$ óptimo é formular um sistema linear de desigualdades de acordo com a observação seguinte: *Nenhum $\omega \in \Omega$ piorará sendo movido ligeiramente na direcção de ω^** (ver [15]). O bom desta ideia é que $\nabla\sigma(\omega^*)$ não precisa ser calculado. Em termos de

derivadas direccionais de σ , esta condiçao de optimalidade tem a forma:

$$\nabla_t \sigma((1-t)\omega + t\omega^*)|_{t=0} \leq 0,$$

para todo ω , para o qual $\sigma(\omega)$ existe, isto e,

$$\sum_{j=1}^n \sum_{i=1}^n \nabla_{\omega_{ij}}(\omega)(\omega_{ij}^* - \omega_{ij}) \leq 0,$$

para todo ω , que representa um sistema linear em ω^* .

Programaao Semi-definida

Existe uma ligaao directa entre problemas de PSI e de Programaao Semi-definida (SDP). O problema de programaao Semi-definida pode ser formulado por:

$$\begin{aligned} \min \quad & C \cdot X \\ \text{s.a.} \quad & A_i \cdot X = b_i, i = 1, \dots, k \\ & X \geq 0, X \in S_n, \end{aligned} \tag{19}$$

onde S_n representa o espao linear de matrizes simetricas $n \times n$,

$$C := \{A_i, i = 1, \dots, k\} \subset S_n.$$

Aqui $A_i, i = 1, \dots, k$ sao colunas da matriz C . O produto interno em S_n e definido por $Y \cdot X = \text{tr}(YX)$, onde tr representa o trao da matriz e a restriao $X \geq 0$ expressa a positividade semi-definida da matriz $X \in S_n$.

O problema dual ao (19)(ver [13]) pode ser formulado segundo:

$$\begin{aligned} \max \quad & b'y \\ \text{s.a.} \quad & \sum_{i=1}^k y_i A_i + S = C \\ & S \geq 0, y \in \mathbf{R}^k. \end{aligned} \tag{20}$$

Claramente, $S \geq 0$ se e so se $t'St \geq 0, \forall t \in T$, onde T representa a esfera unitaria em \mathbf{R}^n , de tal forma que (20) possa ser formulado como um problema continuo de LSIP (ver [13]).

Como e referido em [29], a teoria da dualidade dos problemas LSIP tem sido utilizada para obter teoremas de dualidade para os problemas de SDP. A prova do teorema de dualidade forte para os problemas de SDP tais que $\{A_i, i = 1, \dots, k\}$

são linearmente independentes, dada em [54] é baseada na aplicação, para a LSIP reformulação de (20), do teorema da convergência para o método de discretização. A mesma reformulação de (20) tem sido explorada por vários autores para a resolução numérica dos problemas de SDP.

Em [34] vários métodos de plano de corte para os problemas SDP são colocados de forma unificada, ilustrando o modo segundo o qual cada aproximação aparece como um realçar natural de um esquema baseado em formulação LSIP dos problemas de SDP.